

Oracle® Application Express

Installation Guide



Release 19.2
F18077-02
November 2019



Oracle Application Express Installation Guide, Release 19.2

F18077-02

Copyright © 2017, 2019, Oracle and/or its affiliates. All rights reserved.

Primary Author: Terri Jennings

Contributors: Harish Konakondla, Christina Cho, Joel Kallman, David Peake, Marc Sewtz, Jason Straub, Vlad Uvarov, Patrick Wolf, Christian Neumueller

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Audience	xii
Documentation Accessibility	xii
Related Documents	xii
Conventions	xiii
Third-Party License Information	xiii

Changes in This Release

Changes in Oracle Application Express Release 19.2	xiv
--	-----

1 Oracle Application Express Installation Requirements

1.1 Oracle Database Requirements	1-1
1.2 Checking the MEMORY_TARGET of the Target Database	1-2
1.3 Browser Requirements	1-3
1.4 Web Listener Requirements	1-3
1.5 Disk Space Requirement	1-4
1.6 Oracle XML DB Requirement	1-4

2 Oracle Application Express Installation Overview

2.1 About Accessing Oracle Application Express in Oracle Database Cloud	2-1
2.2 Understanding the Installation Process	2-2
2.2.1 About Planning Your Installation	2-2
2.2.2 About Patch Sets and Point Releases	2-3
2.2.3 About the Installation Scripts	2-3
2.2.4 About Accessing Oracle Application Express	2-4
2.2.5 Requesting a Workspace from the Sign In Dialog	2-5
2.2.6 Resetting Your Password from the Sign In Page	2-6
2.2.7 Recovering Your Workspace Name	2-7
2.3 Upgrading from a Previous Oracle Application Express Release	2-7
2.3.1 About Release Numbering Conventions	2-8

2.3.2	Sample Upgrade Scenarios	2-9
2.3.3	Viewing the Oracle Application Express Release Number	2-9
2.3.4	Viewing the Oracle REST Data Services Release Number	2-9
2.3.5	About Determining the Version of Oracle REST Data Services	2-10
2.3.6	About Installing the Oracle Application Express Release Included with the Oracle Database	2-10
2.3.7	About Upgrading Existing Applications	2-11
2.3.8	About Specifying an Application Compatibility Mode	2-11
2.3.9	About Testing Requirements	2-12
2.3.10	About Cleaning Up Your Environment	2-12
2.3.11	About Reverting to a Previous Release	2-12
2.4	Choosing a Web Listener	2-13
2.4.1	About Choosing a Web Listener	2-13
2.4.2	Comparing Web Listener Options	2-14
2.4.3	Security Considerations When Using the Embedded PL/SQL Gateway	2-15
2.4.4	About Oracle REST Data Services	2-15
2.4.5	About the Embedded PL/SQL Gateway	2-16
2.4.6	About Oracle HTTP Server with mod_plsql	2-17
2.4.7	Selecting a Web Listener in an Oracle RAC Environment	2-18
2.5	About the Oracle Application Express Runtime Environment	2-18

3 Utilizing the Multitenant Architecture in Oracle Database 12c or Later

3.1	Understanding the Installation Choices	3-1
3.2	Installing Application Express into an Application Container	3-2
3.2.1	About Application Containers	3-3
3.2.2	Creating Application Container	3-3
3.2.3	Installing or Upgrading Application Express in an Application Container	3-3
3.2.4	Verifying the Application Container Installation	3-4
3.2.5	Creating an Application Seed	3-5
3.2.6	Creating an Application PDB from the Application Root Seed	3-6
3.2.7	Configure HTTP Access to the Application PDB	3-6
3.3	Installing Application Express into Different PDBs	3-7
3.3.1	Uninstalling Application Express from a CDB	3-7
3.3.2	Installing Application Express Locally in a PDB	3-8
3.3.3	Installing Application Express into a CDB	3-10
3.4	Plugging in a PDB When Application Express is Installed in the Root Container	3-13
3.4.1	Scenario 1: Plug-in Non-CDB with Application Express	3-13
3.4.2	Scenario 2: Plug-in PDB with a Common Application Express from Another CDB	3-14

3.4.3	Scenario 3: Plug-in PDB with a Local Application Express from Another CDB	3-15
3.4.4	Scenario 4: Plug-in Non-CDB or PDB with No Application Express	3-16
3.4.5	Working with Incompatible Application Express Versions	3-16
3.4.5.1	Patching or Upgrading Application Express in the CDB	3-16
3.4.5.2	Patching or Upgrading Application Express in a PDB	3-17
3.5	Plugging in a PDB When Application Express Is Not in the Root Container of the Target CDB	3-19
3.5.1	Scenario 1: Plug-in a Non-CDB with Application Express	3-20
3.5.2	Scenario 2: Plug-in PDB with Common Application Express from Another CDB	3-20
3.5.3	Scenario 3: Plug-in PDB with a Local Application Express from Another CDB	3-21
3.5.4	Scenario 4: Plug-in a Non-CDB or PDB with No Application Express	3-21

4 Installing Application Express and Configuring Oracle REST Data Services

4.1	Oracle Database Requirements	4-2
4.2	Oracle REST Data Services Requirement	4-2
4.3	Performing Pre-installation Tasks for Oracle Application Express	4-2
4.4	Downloading and Installing Oracle Application Express	4-3
4.4.1	Installing Application Express	4-4
4.4.2	Creating or Updating Your Instance Administration Account	4-7
4.4.2.1	What Is an Instance Administrator?	4-7
4.4.2.2	About apxchpwd.sql	4-8
4.4.2.3	Running apxchpwd.sql	4-8
4.4.3	Restarting Processes	4-9
4.4.4	Configuring the APEX_PUBLIC_USER Account	4-9
4.4.4.1	About the APEX_PUBLIC_USER Account	4-9
4.4.4.2	Unlocking the APEX_PUBLIC_USER Account	4-10
4.4.4.3	Changing the Password for the APEX_PUBLIC_USER Account	4-10
4.4.4.4	About Password Expiration in Oracle Database 11g and Later	4-11
4.4.5	Configuring RESTful Services	4-11
4.5	Downloading and Installing Oracle REST Data Services	4-12
4.5.1	Downloading Oracle REST Data Services	4-13
4.5.2	About Configuring Oracle REST Data Services Behind a Reverse Proxy or Load Balancer	4-13
4.5.3	Web Server HTTP POST Request Limits	4-14
4.5.4	Disabling Oracle XML DB Protocol Server	4-14
4.6	Configuring Oracle REST Data Services	4-15
4.6.1	Copying the Images Directory	4-15

4.6.2	Validating the Oracle REST Data Services Installation	4-15
4.6.3	Configuring Static File Support	4-16
4.6.4	Securing Oracle REST Data Service	4-16
4.7	Enabling Network Services in Oracle Database 11g or Later	4-17
4.7.1	When and Why Network Services Must be Enabled	4-17
4.7.2	Granting Connect Privileges Prior to Oracle Database 12c	4-18
4.7.3	Granting Connect Privileges in Oracle Database 12c or Later	4-19
4.7.4	Troubleshooting an Invalid ACL Error	4-20
4.8	Performing Security Tasks	4-21
4.9	Controlling the Number of Concurrent Jobs	4-21
4.9.1	About Managing the Number of Concurrent Jobs	4-22
4.9.2	Viewing the Number of JOB_QUEUE_PROCESSES	4-22
4.9.2.1	Viewing JOB_QUEUE_PROCESSES in the Installation Log File	4-22
4.9.2.2	Viewing JOB_QUEUE_PROCESSES in Oracle Application Express	4-22
4.9.2.3	Viewing JOB_QUEUE_PROCESSES from SQL*Plus	4-23
4.9.3	Changing the Number of JOB_QUEUE_PROCESSES	4-23
4.10	About Running Oracle Application Express in Other Languages	4-23
4.11	Installing Translated Versions of Oracle Application Express	4-24
4.11.1	About Installing Translated Versions of Oracle Application Express	4-24
4.11.2	Installing a Translated Version of Oracle Application Express	4-25
4.12	Creating a Workspace and Adding Oracle Application Express Users	4-26
4.12.1	About Workspaces and Users	4-26
4.12.2	Accessing Administration Services Using Oracle REST Data Services	4-27
4.12.3	Creating a Workspace Manually	4-28
4.12.4	Creating Oracle Application Express Users	4-29
4.12.5	Signing into a Workspace Using Oracle REST Data Services	4-30
4.13	Performing Post Installation Tasks for Upgrade Installations	4-31
4.13.1	About Removing Prior Oracle Application Express Installations	4-32
4.13.2	Verifying if a Prior Installation Exists	4-32
4.13.3	Removing Schemas and SYS Objects from Prior Installations	4-33
4.13.4	Removing Schemas from Prior Installations in a CDB	4-33
4.13.5	Fixing Invalid ACL in Oracle Database 11g or Later	4-33
4.14	About Performance Optimization Tasks	4-34
4.15	Converting Between Runtime and Full Development Environments	4-34
4.15.1	About Runtime and Full Development Environments	4-35
4.15.2	Converting a Runtime Environment to a Full Development Environment	4-35
4.15.3	Converting a Full Development Environment to a Runtime Environment	4-36

A Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway

A.1	Oracle Database Requirements	A-2
A.2	Performing Pre-installation Tasks for Oracle Application Express	A-2
A.3	Downloading and Installing Oracle Application Express	A-3
A.3.1	Installing Application Express	A-4
A.3.2	Creating or Updating Your Instance Administration Account	A-7
A.3.2.1	What Is an Instance Administrator?	A-7
A.3.2.2	About apxchpwd.sql	A-8
A.3.2.3	Running apxchpwd.sql	A-8
A.3.3	Restart Processes	A-9
A.4	Configuring the Embedded PL/SQL Gateway	A-9
A.4.1	Running the apex_epg_config.sql Configuration Script	A-10
A.4.2	Updating the Images Directory When Upgrading from a Previous Release	A-11
A.4.3	Verifying the Port Assigned to HTTP on the Oracle XML DB HTTP Server	A-12
A.4.4	Verifying the Oracle XML DB Protocol Server Port	A-13
A.4.5	Enabling Oracle XML DB Protocol Server	A-14
A.4.6	Configuring Static File Support	A-15
A.4.7	About Configuring Embedded PL/SQL Gateway Behind a Reverse Proxy or Load Balancer	A-15
A.5	Enabling Network Services in Oracle Database 11g or Later	A-16
A.5.1	When and Why Network Services Must be Enabled	A-16
A.5.2	Granting Connect Privileges Prior to Oracle Database 12c	A-17
A.5.3	Granting Connect Privileges in Oracle Database 12c or Later	A-18
A.5.4	Troubleshooting an Invalid ACL Error	A-19
A.6	Performing Security Tasks	A-20
A.7	Controlling the Number of Concurrent Jobs	A-20
A.7.1	About Managing the Number of Concurrent Jobs	A-21
A.7.2	Viewing the Number of JOB_QUEUE_PROCESSES	A-21
A.7.2.1	Viewing JOB_QUEUE_PROCESSES in the Installation Log File	A-21
A.7.2.2	Viewing JOB_QUEUE_PROCESSES in Oracle Application Express	A-21
A.7.2.3	Viewing JOB_QUEUE_PROCESSES from SQL*Plus	A-22
A.7.3	Changing the Number of JOB_QUEUE_PROCESSES	A-22
A.8	About Running Oracle Application Express in Other Languages	A-22
A.9	Installing Translated Versions of Oracle Application Express	A-23
A.9.1	About Installing Translated Versions of Oracle Application Express	A-23
A.9.2	Installing a Translated Version of Oracle Application Express	A-24
A.10	About Configuring the SHARED_SERVERS Parameter	A-25

A.11	Creating a Workspace and Adding Oracle Application Express Users	A-25
A.11.1	About Workspaces and Users	A-26
A.11.2	Accessing Administration Services Using Embedded PL/SQL Gateway	A-26
A.11.3	Creating a Workspace Manually	A-27
A.11.4	Creating Oracle Application Express Users	A-28
A.11.5	Signing into a Workspace Using the Embedded PL/SQL Gateway	A-30
A.12	Performing Post Installation Tasks for Upgrade Installations	A-31
A.12.1	About Removing Prior Oracle Application Express Installations	A-31
A.12.2	Verifying if a Prior Installation Exists	A-31
A.12.3	Removing Schemas and SYS Objects from Prior Installations	A-32
A.12.4	Fixing Invalid ACL in Oracle Database 11g or Later	A-32
A.13	About Performance Optimization Tasks	A-33
A.14	Converting Between Runtime and Full Development Environments	A-33
A.14.1	About Runtime and Full Development Environments	A-34
A.14.2	Converting a Runtime Environment to a Full Development Environment in a non-CDB or locally in a PDB	A-34
A.14.3	Converting a Full Development Environment to a Runtime Environment in a non-CDB or locally in a PDB	A-35
A.14.4	Converting a Runtime Environment to a Full Development Environment in a CDB	A-36
A.14.5	Converting a Full Development Environment to a Runtime Environment in a CDB	A-37

B Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server

B.1	Oracle Database Requirements	B-2
B.2	Performing Pre-installation Tasks for Oracle Application Express	B-2
B.3	Downloading and Installing Oracle Application Express	B-3
B.3.1	Installing Application Express	B-4
B.3.2	Creating or Updating Your Instance Administration Account	B-7
B.3.2.1	What Is an Instance Administrator?	B-7
B.3.2.2	About apxchpwd.sql	B-7
B.3.2.3	Running apxchpwd.sql	B-8
B.3.3	Restarting Processes	B-9
B.3.4	Configuring the APEX_PUBLIC_USER Account	B-9
B.3.4.1	About the APEX_PUBLIC_USER Account	B-9
B.3.4.2	Unlocking the APEX_PUBLIC_USER Account	B-9
B.3.4.3	Changing the Password for the APEX_PUBLIC_USER Account	B-10
B.3.4.4	About Password Expiration in Oracle Database 11g	B-11
B.4	Configuring the Oracle HTTP Server	B-11

B.4.1	About Copying the Images Directory	B-12
B.4.2	Copying the Images Directory	B-12
B.4.3	Editing the dads.conf File	B-13
B.4.4	Configuring Static File Support	B-15
B.4.5	Stopping and Restarting Oracle HTTP Server	B-15
B.4.5.1	Stopping and Restarting Oracle HTTP Server with Oracle Database 10g and Lower	B-15
B.4.5.2	Stopping and Restarting Oracle HTTP Server 12c	B-16
B.4.6	Disabling Oracle XML DB Protocol Server	B-17
B.4.7	About Configuring Oracle HTTP Server Behind a Reverse Proxy or Load Balancer	B-17
B.5	Enabling Network Services in Oracle Database 11g or Later	B-18
B.5.1	When and Why Network Services Must be Enabled	B-18
B.5.2	Granting Connect Privileges Prior to Oracle Database 12c	B-19
B.5.3	Granting Connect Privileges in Oracle Database 12c or Later	B-20
B.5.4	Troubleshooting an Invalid ACL Error	B-21
B.6	Performing Security Tasks	B-22
B.7	Controlling the Number of Concurrent Jobs	B-22
B.7.1	About Managing the Number of Concurrent Jobs	B-23
B.7.2	Viewing the Number of JOB_QUEUE_PROCESSES	B-23
B.7.2.1	Viewing JOB_QUEUE_PROCESSES in the Installation Log File	B-23
B.7.2.2	Viewing JOB_QUEUE_PROCESSES in Oracle Application Express	B-23
B.7.2.3	Viewing JOB_QUEUE_PROCESSES from SQL*Plus	B-24
B.7.3	Changing the Number of JOB_QUEUE_PROCESSES	B-24
B.8	About Running Oracle Application Express in Other Languages	B-24
B.9	Installing Translated Versions of Oracle Application Express	B-25
B.9.1	About Installing Translated Versions of Oracle Application Express	B-25
B.9.2	Installing a Translated Version of Oracle Application Express	B-26
B.10	Creating a Workspace and Adding Oracle Application Express Users	B-27
B.10.1	About Workspaces and Users	B-27
B.10.2	Accessing Administration Services Using Oracle HTTP Server	B-27
B.10.2.1	Creating a Workspace Manually	B-28
B.10.3	Creating Oracle Application Express Users	B-29
B.10.4	Signing into a Workspace Using the Oracle HTTP Server	B-31
B.11	Performing Post Installation Tasks for Upgrade Installations	B-32
B.11.1	About Removing Prior Oracle Application Express Installations	B-32
B.11.2	Verifying if a Prior Installation Exists	B-33
B.11.3	Removing Schemas and SYS Objects from Prior Installations	B-33
B.11.4	Fixing Invalid ACL in Oracle Database 11g or Later	B-34
B.12	About Performance Optimization Tasks	B-34
B.13	Converting Between Runtime and Full Development Environments	B-35

B.13.1	About Runtime and Full Development Environments	B-35
B.13.2	Converting a Runtime Environment to a Full Development Environment in a non-CDB or locally in a PDB	B-36
B.13.3	Converting a Full Development Environment to a Runtime Environment in a non-CDB or locally in a PDB	B-37
B.13.4	Converting a Runtime Environment to a Full Development Environment in a CDB	B-38
B.13.5	Converting a Full Development Environment to a Runtime Environment in a CDB	B-39

C Automating the Oracle Application Express Installation Process

C.1	About apxsilentins.sql	C-1
C.2	Running apxsilentins.sql	C-1

D Maximizing Uptime During an Application Express Upgrade

E Oracle Application Express Installation Troubleshooting

E.1	Reviewing a Log of an Installation Session	E-1
E.2	Verifying the Validity of an Oracle Application Express Installation	E-2
E.3	Cleaning Up After a Failed Installation	E-2
E.3.1	Reverting to a Previous Release After a Failed Upgrade Installation	E-2
E.3.1.1	Verifying If You Have a Previous Release of Oracle Application Express	E-3
E.3.1.2	Reverting the Images Directory	E-3
E.3.1.3	Reverting to a Previous Release	E-3
E.3.1.4	Removing the Oracle Application Express Release Schema	E-35
E.3.2	Removing Oracle Application Express from the Database	E-36
E.4	About Images Displaying Incorrectly in Oracle Application Express	E-37
E.5	About Page Protection Violation	E-38

F Upgrading Oracle Application Express within Oracle Database Express Edition

F.1	Upgrading to the Latest Oracle Application Express Release	F-1
F.2	Understanding the Differences Between Later Oracle Application Express Releases	F-3
F.2.1	User Interface Differences	F-4

Index

Preface

This guide explains how to install and configure Oracle Application Express.

- [Audience](#)
- [Documentation Accessibility](#)
- [Related Documents](#)
- [Conventions](#)
- [Third-Party License Information](#)

Audience

Oracle Application Express Installation Guide is intended for anyone responsible for installing Oracle Application Express.

To use this manual, you must have administrative privileges on the computer where you installed your Oracle database and familiarity with object-relational database management concepts.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

For more information, see these Oracle resources:

- *Oracle Application Express Release Notes*
- *Oracle Application Express App Builder User's Guide*
- *Oracle Application Express End User's Guide*
- *Oracle Application Express Administration Guide*
- *Oracle Application Express SQL Workshop Guide*

- *Oracle Application Express API Reference*
- *Oracle Application Express Application Migration Guide*
- *Oracle Database Concepts*
- *Oracle Database Administrator's Guide*
- *Oracle Database SQL Language Reference*
- *SQL*Plus User's Guide and Reference*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Third-Party License Information

Oracle Application Express contains third-party code. Please see the *Oracle Database Licensing Information User Manual* book for notices Oracle is required to provide.

Note, however, that the Oracle program license that accompanied this product determines your right to use the Oracle program, including the third-party software, and the terms contained in the following notices do not change those rights.

Changes in This Release

This preface contains:

- [Changes in Oracle Application Express Release 19.2](#)

Changes in Oracle Application Express Release 19.2

All content in *Oracle Application Express Installation Guide* has been updated to reflect release 19.2 functionality.

Deprecated and Desupported Features

See "[Deprecated Features](#)" and "[Desupported Features](#)" *Oracle Application Express Release Notes*.

1

Oracle Application Express Installation Requirements

Before installing Oracle Application Express you must verify your configuration meets the minimum installation requirements.

- [Oracle Database Requirements](#)
Oracle Application Express release 19.2 requires an Oracle Database release 11.2.0.4 or later, including Enterprise Edition and Express Edition (Oracle Database 18c XE).
- [Checking the MEMORY_TARGET of the Target Database](#)
Oracle Application Express requires the system global area (SGA) and program global area (PGA) to be at least 300 MB.
- [Browser Requirements](#)
Oracle Application Express requires a JavaScript-enabled browser and supports the current and prior major release of Google Chrome, Mozilla Firefox, Apple Safari, Microsoft Internet Explorer, and Microsoft Edge.
- [Web Listener Requirements](#)
Oracle Application Express requires access to a Web listener.
- [Disk Space Requirement](#)
Oracle Application Express disk space requirements are described in this section.
- [Oracle XML DB Requirement](#)
Oracle XML DB must be installed in the Oracle database that you want to use if you are installing a full development environment. Oracle XML DB is not required for runtime only installations. If you are using a preconfigured database created either during an installation or by Database Configuration Assistant (DBCA), Oracle XML DB is already installed and configured.

1.1 Oracle Database Requirements

Oracle Application Express release 19.2 requires an Oracle Database release 11.2.0.4 or later, including Enterprise Edition and Express Edition (Oracle Database 18c XE).

If you are upgrading an Oracle Database version 12.1 CDB, you must download from My Oracle Support the one off patch for bug 20618595. Search for **20618595** on the Patches tab.

1.2 Checking the MEMORY_TARGET of the Target Database

Oracle Application Express requires the system global area (SGA) and program global area (PGA) to be at least 300 MB.

Databases typically use automatic memory management, where the memory can be controlled by the server parameter `MEMORY_TARGET`. If your database does not use automatic memory management, consult the *Oracle Database Administrator's Guide* to find out how to configure manual memory parameters (for example, `SGA_TARGET`, `PGA_AGGREGATE_TARGET`, `SHARED_POOL_SIZE`) instead, for a similar result.

To check the `MEMORY_TARGET` of the target database:

1. Start the database:

```
SQL> STARTUP
```

2. If necessary, enter the following command to determine whether the system uses an initialization parameter file (`initSID.ora`) or a server parameter file (`spfileDBNAME.ora`):

```
SQL> SHOW PARAMETER PFILE;
```

This command displays the name and location of the server parameter file or the initialization parameter file.

3. Determine the current values of the `MEMORY_TARGET` parameter:

```
SQL> SHOW PARAMETER MEMORY_TARGET
```

4. If the value is 0, your database is using manual memory management. Consult the *Oracle Database Administrator's Guide* to learn how to configure an equivalent memory size using manual memory management, instead of continuing with the steps that follow.

If the system is using a server parameter file, set the value of the `MEMORY_TARGET` initialization parameter to at least 300 MB:

```
SQL> ALTER SYSTEM SET MEMORY_TARGET='300M' SCOPE=spfile;
```

5. If the system uses an initialization parameter file, change the value of the `MEMORY_TARGET` parameter to at least 300 MB in the initialization parameter file (`initSID.ora`).

6. Shut down the database:

```
SQL> SHUTDOWN
```

7. Restart the database:

```
SQL> STARTUP
```



See Also:

Oracle Database Administrator's Guide

1.3 Browser Requirements

Oracle Application Express requires a JavaScript-enabled browser and supports the current and prior major release of Google Chrome, Mozilla Firefox, Apple Safari, Microsoft Internet Explorer, and Microsoft Edge.

Note:

Microsoft Internet Explorer 11 is the prior major release, with Microsoft Edge being the current Microsoft browser.

1.4 Web Listener Requirements

Oracle Application Express requires access to a Web listener.

Supported Web listeners include:

- **Oracle REST Data Services (ORDS)**

Oracle Application Express requires Oracle REST Data Services (ORDS) 19.1 or 19.2.

Oracle REST Data Services (ORDS) is Java-based Web Server and is the preferred option for use with Oracle Application Express. ORDS features file system caching, offers improved file upload capability, and is certified with Oracle WebLogic Server and Apache Tomcat

- **Embedded PL/SQL Gateway (Legacy)**

Oracle XML DB Protocol Server with the embedded PL/SQL gateway installs with Oracle Database 11g Release 1 (11.1) or later. It provides the database with a Web server and the necessary infrastructure to create dynamic applications.

- **Oracle HTTP Server and mod_plsql (Deprecated)**

Oracle HTTP Server uses the `mod_plsql` plug-in to communicate with the Oracle Application Express engine within the Oracle database.

See Also:

- ["Choosing a Web Listener"](#)
- ["Installing Application Express and Configuring Oracle REST Data Services"](#)
- ["Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway"](#)
- ["Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server"](#)

1.5 Disk Space Requirement

Oracle Application Express disk space requirements are described in this section.

Oracle Application Express disk space requirements are as follows:

- Free space for Oracle Application Express software files on the file system: 310 MB if using English only download (`apex_19.2_en.zip`) and 705 MB if using full download (`apex_19.2.zip`).
- Free space in Oracle Application Express tablespace: 220 MB
- Free space in `SYSTEM` tablespace: 100 MB
- Free space in Oracle Application Express tablespace for each additional language (other than English) installed: 60 MB

1.6 Oracle XML DB Requirement

Oracle XML DB must be installed in the Oracle database that you want to use if you are installing a full development environment. Oracle XML DB is not required for runtime only installations. If you are using a preconfigured database created either during an installation or by Database Configuration Assistant (DBCA), Oracle XML DB is already installed and configured.

 **Tip:**

The installer does a prerequisite check for Oracle XML DB and will exit if it is not installed.

 **Tip:**

The installation of Oracle XML DB creates the user `ANONYMOUS`. In order for Oracle Application Express workspace provisioning to work properly, the `ANONYMOUS` user must not be dropped from the database.

 **Tip:**

For more information about manually adding Oracle XML DB to an existing database, see "Administration of Oracle XML DB" in *Oracle XML DB Developer's Guide*

2

Oracle Application Express Installation Overview

Oracle Application Express Installation Guide describes how to install Oracle Application Express in a on-premises (or local) installation.

How you sign in and access Oracle Application Express depends upon your user role and where Oracle Application Express resides. Oracle Application Express may reside in a local on-premises Oracle Database or in an Oracle Cloud Service.

- [About Accessing Oracle Application Express in Oracle Database Cloud](#)
Learn about accessing Oracle Application Express in Oracle Database Cloud.
- [Understanding the Installation Process](#)
Installing Oracle Application Express is a multiple step process. The Web listener you choose determines the actual installation procedure.
- [Upgrading from a Previous Oracle Application Express Release](#)
Upgrading Oracle Application Express creates new database objects in a new schema and migrates the application metadata to the new release.
- [Choosing a Web Listener](#)
In order to run, Oracle Application Express must have access to a Web listener.
- [About the Oracle Application Express Runtime Environment](#)
Learn about the Oracle Application Express runtime environment.

2.1 About Accessing Oracle Application Express in Oracle Database Cloud

Learn about accessing Oracle Application Express in Oracle Database Cloud.

Oracle Application Express is installed and enabled in Oracle Autonomous Database services such as Oracle Autonomous Transaction Processing and Oracle Autonomous Data Warehouse.

Oracle Application Express is available in Exadata Cloud Service and Database Cloud Service in Oracle Cloud Infrastructure. However, you need to manually customize your databases to install and enable Oracle Application Express by following on-premises installation process or using cloud tooling such as Terraform.

 **See Also:**

- "Creating Applications with Oracle Application Express in Autonomous Database" in *Using Oracle Autonomous Transaction Processing*
- "Creating Applications with Oracle Application Express on Autonomous Database" in *Using Oracle Autonomous Data Warehouse*
- "Oracle Application Express (APEX) on Oracle Cloud Infrastructure Database" white paper on the [Oracle Cloud Infrastructure Technical Resources](#) page

2.2 Understanding the Installation Process

Installing Oracle Application Express is a multiple step process. The Web listener you choose determines the actual installation procedure.

- [About Planning Your Installation](#)
Learn about the steps needed to install Oracle Application Express.
- [About Patch Sets and Point Releases](#)
Patch sets provide bug fixes only. A point release includes bug fixes and incorporates all current patch sets.
- [About the Installation Scripts](#)
You can install Oracle Application Express or update from previous release using the same installation procedure and the installation scripts.
- [About Accessing Oracle Application Express](#)
You access the Oracle Application Express development environment, by signing in to a shared work area called a workspace.
- [Requesting a Workspace from the Sign In Dialog](#)
Request a workspace from the Sign In dialog.
- [Resetting Your Password from the Sign In Page](#)
Reset your password by clicking a link on Oracle Application Express Sign In page.
- [Recovering Your Workspace Name](#)
Recover your workspace name from the Oracle Application Express Sign In page.

 **See Also:**

["Upgrading from a Previous Oracle Application Express Release"](#)

2.2.1 About Planning Your Installation

Learn about the steps needed to install Oracle Application Express.

Oracle recommends you take the time to carefully plan your installation.

Installing Oracle Application Express involves the following steps:

1. **Decide on a Full or Runtime Environment** - Determine whether to install a full development environment or runtime environment. A **full development** environment provides complete access to the App Builder development environment to develop applications. A **runtime environment** is the appropriate choice for production implementations in which you want to run applications that cannot be modified.
See "[About the Oracle Application Express Runtime Environment.](#)"
2. **Choose a Web listener** - The Web listener you choose determines the installation process. Available options include the Oracle REST Data Services, Embedded PL/SQL Gateway, or Oracle HTTP Server and `mod_plsql`.
See "[Choosing a Web Listener.](#)"
3. **Verify installation requirements**- Before installing, verify your system meets the minimum requirements.
See "[Oracle Application Express Installation Requirements .](#)"
4. **Install the software** - Once you decide on a Web listener, follow the procedures in one of the following sections to install the software and complete the configuration tasks:
 - "[Installing Application Express and Configuring Oracle REST Data Services](#)"
 - "[Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway](#)"
 - "[Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server](#)"

2.2.2 About Patch Sets and Point Releases

Patch sets provide bug fixes only. A point release includes bug fixes and incorporates all current patch sets.

Patch sets are a mechanism for delivering fully tested and integrated product fixes. Patch sets provide bug fixes only. Patch sets typically do not include new functionality and they do not require certification on the target system. Patch sets include all of the libraries that have been rebuilt to implement the bug fixes in the set. All of the fixes in the patch set have been tested and are certified to work with each other.

In between major product releases, Oracle may offer a point release. A point release (for example Oracle Application Express release 5.0.3) includes bug fixes and incorporates all current patch sets. Typically, point releases do not introduce new functionality.



See Also:

["Upgrading from a Previous Oracle Application Express Release"](#)

2.2.3 About the Installation Scripts

You can install Oracle Application Express or update from previous release using the same installation procedure and the installation scripts.

The installation script checks for the latest existing Oracle Application Express schema and automatically copies the instance metadata, workspaces, and applications from the previous schema into the current schema. The original schema associated with the previous release is left completely unaltered. Following best practices, Oracle recommends that you create new tablespaces for a new release of Oracle Application Express and follow the appropriate installation instructions as outlined in this document.

2.2.4 About Accessing Oracle Application Express

You access the Oracle Application Express development environment, by signing in to a shared work area called a workspace.

How you sign in and access Oracle Application Express depends upon your user role.

A workspace enables multiple users to work within the same Oracle Application Express installation while keeping their objects, data, and applications private. Each workspace has a unique ID and name. An instance administrator can create a workspace manually within Oracle Application Express Administration Services or have users submit requests. Oracle Application Express Administration Services is a separate application for managing an entire Oracle Application Express instance.

Users are divided into four primary roles:

- **Instance administrators** are *superusers* that manage an entire hosted instance using a separate application called Oracle Application Express Administration Services. Instance administrators manage workspace provisioning, configure features and instance settings, and manage security.
- **Workspace administrators** can perform administrator tasks specific to a workspace such as configuring workspace preferences, managing user accounts, monitoring workspace activity, and viewing log files.
- **Developers** are users who sign in to a workspace and create and edit applications.
- **End users** can only run existing database or Websheet applications.

If you are a developer, an administrator must grant you access to shared work area called a workspace. If you are an Instance administrator, you must sign in to Oracle Application Express Administration Services, determine whether to specify a provisioning mode, create a workspace, and then sign in to that workspace.

About Specifying a Provisioning Mode

The Instance administrator determines how the process of provisioning (or creating) a workspace works for a specific Oracle Application Express instance. To determine how provisioning works, an Instance Administrator selects a Provisioning Methods on the Instance Settings page:

- **Manual** - An Instance administrator creates new workspaces and notifies the Workspace administrator regarding the Sign In credentials.
- **Request** - Users request a workspace. Once an administrator approves the request, the user receives an email containing an email verification link. After the user clicks the email verification link, the workspace is created.
- **Automatic** - Works similar to **Request** except requests are automatically approved with no administrator review required

 **See Also:**

"About Specifying How Workspaces Are Created" and "Selecting a Workspace Provisioning Mode" in *Oracle Application Express Administration Guide*

About Creating Workspaces and Users

Before you can develop or install applications, an administrator must create a workspace and add Oracle Application Express users. To learn more contact your administrator, or see the "Creating a Workspace and Adding Oracle Application Express Users" section for your Web listener in this document.

 **See Also:**

- "Creating Workspaces" in *Oracle Application Express Administration Guide*
- "Making a Service Request" in *Oracle Application Express Administration Guide*
- "Managing Workspace and Change Requests in *Oracle Application Express Administration Guide*

2.2.5 Requesting a Workspace from the Sign In Dialog

Request a workspace from the Sign In dialog.

 **Note:**

This information does not apply to Oracle Cloud products such as Oracle Database Cloud Service (Database Schema).

Your administrator determines how you request a new workspace. If your administrator has set Provisioning Method to either **Request** or **Automatic** and has configured email, you can request a workspace on the Sign In dialog.

1. To request a workspace from the Sign In dialog:
 1. Navigate to the Oracle Application Express Sign in dialog.
 2. Under Sign In, click **Request a Workspace**.
The Request a Workspace Wizard appears.
 3. For Identification:
 - a. First Name - Enter your first name.
 - b. Last Name - Enter your last name.

- c. Email - Enter the email address. A link to activate your workspace will be sent to this email address.
 - d. Workspace - Enter a workspace name that name uniquely identifies your development environment.
 - e. Click **Next**.
4. If defined, review and accept the service agreement and click **Next**.
 5. Verify your request and click **Submit Request**.
- Once you complete the Identification form, the following events occur:
- a. You will receive an email containing a verification link.
 - b. When you click the verification link, the workspace is created.
 - c. You will receive another email containing Sign In credentials (that is, the workspace name, username, and password).

Once you complete the Identification form, the following events occur:

1. You will receive an email containing a verification link.
2. When you click the verification link, the workspace is created.
3. You will then receive another email containing Sign In credentials (that is, the workspace name, username, and password).



See Also:

"About Specifying How Workspaces Are Created" in *Oracle Application Express Administration Guide*

2.2.6 Resetting Your Password from the Sign In Page

Reset your password by clicking a link on Oracle Application Express Sign In page.



Tip:

To reset your password from the Sign In page, you must provide your email address and the workspace name.

1. To reset your password from the Sign In Page:
 1. In a web browser, navigate to the Oracle Application Express Sign In page.
The Sign In page appears.
 2. Under **Sign In**, click **Reset Password**.
 3. In the Reset Password form, enter your email address, workspace name, and click **Reset Password**.

You will receive an email confirming your workspace name and username and containing a **Reset Password URL** link.

4. In the email, click the **Reset Password URL** link.
5. In the Change Password form:
 - a. New Password - Enter your new password.

 **Tip:**

Passwords are case sensitive.

- b. Confirm Password - Enter your new password again.
- c. Click **Apply Changes**.

 **Tip:**

You can also reset your password within Oracle Application Express. See "Changing Your Profile or Password" in *Oracle Application Express App Builder User's Guide*

2.2.7 Recovering Your Workspace Name

Recover your workspace name from the Oracle Application Express Sign In page.

If you cannot remember your workspace name, you can request a list of all workspace names associated with your email address.

To find your workspace name:

1. In a web browser, navigate to the Oracle Application Express Sign In page.
2. On the Sign In page, click **Reset Password**.
3. Click **Find My Workspace**.
4. Enter your email address and click **Find Workspace**.

You will receive an email listing all workspaces associated with the email address you provided.

2.3 Upgrading from a Previous Oracle Application Express Release

Upgrading Oracle Application Express creates new database objects in a new schema and migrates the application metadata to the new release.

If you have Oracle Application Express release 19.1x or earlier, following any of the installation scenarios in this guide upgrades your Oracle Application Express instance to release 19.2 and creates Oracle Application Express 19.2 database objects in a new schema and migrates the application metadata to the new release.

- [About Release Numbering Conventions](#)
New releases of Oracle Application Express correlate to the calendar year.

- [Sample Upgrade Scenarios](#)
Common upgrade scenarios include upgrading from prior release and installing an Oracle Database release which include Oracle Application Express.
- [Viewing the Oracle Application Express Release Number](#)
View your Oracle Application Express release number on the Workspace home page or on the About Application Express page.
- [Viewing the Oracle REST Data Services Release Number](#)
View the Oracle REST Data Services release number on the About Application Express page.
- [About Determining the Version of Oracle REST Data Services](#)
You can view the release number of Oracle REST Data Services (ORDS) on the About Application Express page.
- [About Installing the Oracle Application Express Release Included with the Oracle Database](#)
Learn about the Oracle Application Express release included with Oracle Database releases.
- [About Upgrading Existing Applications](#)
If you install a new release of Oracle Application Express, the installation process updates existing applications to the latest release.
- [About Specifying an Application Compatibility Mode](#)
You can specify the compatibility mode in the application definition within each application.
- [About Testing Requirements](#)
Determining the appropriate amount of regression testing when upgrading Oracle Application Express depends upon the complexity, size, and number of applications you are upgrading.
- [About Cleaning Up Your Environment](#)
Following the successful upgrade of all of the environments to the latest release of Oracle Application Express, you should clean-up the environments.
- [About Reverting to a Previous Release](#)
You can revert to a previous release of Oracle Application Express.



See Also:

- ["Understanding the Installation Process"](#)
- ["Maximizing Uptime During an Application Express Upgrade"](#)

2.3.1 About Release Numbering Conventions

New releases of Oracle Application Express correlate to the calendar year.

In 2018 and starting with release 18.1 and 18.2, Oracle Application Express introduced correlating the release number to the calendar year.

In addition, Application Express now only offers full releases and no longer provides patch set releases (such as 5.1.1). Eliminating patch set releases reduces downtime

when updating existing installations. Application Express architecture also enables developers to revert releases if necessary.

Patch set exceptions (PSEs) may still be delivered for major defects. To learn more about PSEs, visit the [Oracle Application Express Known Issues page](#).

2.3.2 Sample Upgrade Scenarios

Common upgrade scenarios include upgrading from prior release and installing an Oracle Database release which include Oracle Application Express.

[Table 2-1](#) lists common upgrade scenarios.

Table 2-1 Sample Upgrade Scenarios

Upgrade Scenarios	Action
Upgrade from a prior Oracle Application Express release	Download the latest files from the Downloads page.
You install Oracle Database which include Oracle Application Express.	Download the latest files from the Downloads page.

2.3.3 Viewing the Oracle Application Express Release Number

View your Oracle Application Express release number on the Workspace home page or on the About Application Express page.

You can view the Oracle Application Express release number on the Workspace home page or on the About Application Express page:

- Workspace home page:
 - Sign in to Oracle Application Express.
 - On the Workspace home page, the current release number displays in the bottom right corner.
- About Application Express page:
 - Sign in to Oracle Application Express.
 - Click the **Help** menu in the upper right and select **About**.
 - On the About Application Express page, the release number appears next to **Product Build**.

2.3.4 Viewing the Oracle REST Data Services Release Number

View the Oracle REST Data Services release number on the About Application Express page.

To view the Oracle REST Data Services release number:

1. Sign in to Oracle Application Express.
2. Click the **Help** menu in the upper right and select **About**.
3. Under the **CGI Environment** section, find `APEX_LISTENER_VERSION`.



Tip:

If `APEX_LISTENER_VERSION` does not appear, then your Oracle Application Express instance is not using Oracle REST Data Services as its Web listener.

2.3.5 About Determining the Version of Oracle REST Data Services

You can view the release number of Oracle REST Data Services (ORDS) on the About Application Express page.

To access the About Application Express page, go to the **Help** menu and then select **About**. Under the **CGI Environment** section, find `APEX_LISTENER_VERSION`. If `APEX_LISTENER_VERSION` does not appear, then your Oracle Application Express instance is not using Oracle REST Data Services as its web listener.

2.3.6 About Installing the Oracle Application Express Release Included with the Oracle Database

Learn about the Oracle Application Express release included with Oracle Database releases.



Note:

Starting with Oracle Database 12c Release 2 (12.2), Oracle Application Express is included in the Oracle Home on disk and is no longer installed by default in the database.

Oracle Application Express is included with the following Oracle Database releases:

- Oracle Database 11g Release 1 (11.1) - Oracle Application Express Release 3.0.
- Oracle Database 11g Release 2 (11.2) - Oracle Application Express Release 3.2.
- Oracle Database 12c Release 1 (12.1) - Oracle Application Express Release 4.2.
- Oracle Database 12c Release 2 (12.2)- Oracle Application Express Release 5.0.
- Oracle Database 18c - Oracle Application Express Release 5.1.
- Oracle Database 19c - Oracle Application Express Release 18.1.

The Oracle Database releases less frequently than Oracle Application Express. Therefore, Oracle recommends updating to the latest Oracle Application Express release available on Oracle Technology Network. To learn more, see the installation instructions for the appropriate Web Listener in your environment.

 **Note:**

If upgrading Oracle Application Express from a release that ships with the database, do not alter any Oracle Application Express files in the Oracle home directory (for example, `/u01/app/oracle/product/18.0.0/dbhome_1/apex`).

2.3.7 About Upgrading Existing Applications

If you install a new release of Oracle Application Express, the installation process updates existing applications to the latest release.

The upgrade process is designed to not alter the application user interface or application components. As a result, application components that may impact the user interface are left untouched by the upgrade process. It is the developer's responsibility to manually review these components and update them, or leave them as is. Examples of components which may impact existing functionality include new features added to interactive reports, charts, certain validations, and so on.

 **Tip:**

Charts that use custom XML must be updated manually otherwise the custom XML will be lost.

 **See Also:**

"Running Advisor to Check Application Integrity" in *Oracle Application Express App Builder User's Guide*

2.3.8 About Specifying an Application Compatibility Mode

You can specify the compatibility mode in the application definition within each application.

Within each application, you can also specify a **Compatibility Mode** in the Application Definition. The **Compatibility Mode** attribute controls the compatibility mode of the Application Express runtime engine. Certain runtime behaviors change from release to release. You can use this attribute to obtain specific application behavior.

Compatibility Mode options include Pre 4.1, 4.1, 4.2, 5.0, 5.1/18.1, 18.2, 19.1, and 19.2.

 **See Also:**

- "Compatibility Mode" in *Oracle Application Express Release Notes*
- "Editing Application Attributes" and "Properties" in *Oracle Application Express App Builder User's Guide*

2.3.9 About Testing Requirements

Determining the appropriate amount of regression testing when upgrading Oracle Application Express depends upon the complexity, size, and number of applications you are upgrading.

You should include the majority of complex pages, particularly those that incorporate significant JavaScript or extensive PL/SQL computations or processes. Developers should ensure pages which they manually update based on the Upgrade Application or Advisor are also included in regression tests. Not all remaining pages have to be included in regression testing. Oracle recommends you include a good representation of different page types includes reports, charts, and forms. An application should always be included in regression testing if its compatibility mode was modified post-upgrade.

While regression testing of upgraded applications is imperative to minimize risk of disrupting the end users, it is important that testing is not drawn out for an extended period. As a general rule:

- Step 1: Upgrade your development environment first. Allow developers to review the applications and make initial updates as needed.
- Step 2: Upgrade your QA/Test environment.
- Step 3: Upgrade applications from development are built into this environment.
- Step 4: Upgrade your production environment.
- Step 5: Build upgraded applications into this environment.

2.3.10 About Cleaning Up Your Environment

Following the successful upgrade of all of the environments to the latest release of Oracle Application Express, you should clean-up the environments.

Once you start developing with the newer release, the Oracle Application Express schema associated with the prior release can be deleted. If you installed the prior release into a separate tablespace, you can simply drop the specific tablespace. Oracle recommends leaving the older Oracle Application Express schema(s) for a few weeks and then remove them from the development, test, and production environments. This cleanup process releases disk space and ensures that no one accesses an outdated schema using tools such as SQL Developer or SQL*Plus.

2.3.11 About Reverting to a Previous Release

You can revert to a previous release of Oracle Application Express.

Because Oracle Application Express creates a new schema for each major release, reverting back to a prior release is a relatively simple process. If you revert to a prior release, any modifications made in the current Oracle Application Express instance are lost. The main task is to switch the public synonyms and grants to point at the previous schema instead of the new schema.



See Also:

["Reverting to a Previous Release After a Failed Upgrade Installation"](#)

2.4 Choosing a Web Listener

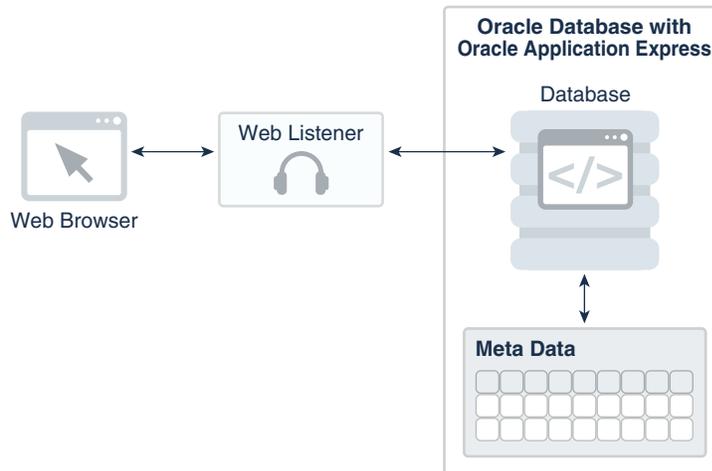
In order to run, Oracle Application Express must have access to a Web listener.

- [About Choosing a Web Listener](#)
You can choose any one of the three Web listeners - Oracle REST Data Services, Oracle HTTP Server, and Embedded PL/SQL gateway.
- [Comparing Web Listener Options](#)
Learn about comparing Web listener options for Oracle Application Express.
- [Security Considerations When Using the Embedded PL/SQL Gateway](#)
Oracle does not recommend the embedded PL/SQL gateway for applications running on the Internet
- [About Oracle REST Data Services](#)
Oracle REST Data Services (formerly Oracle Application Express Listener) is Java-based, can be installed into any supported J2EE compliant Web Server, and is the preferred option for use with Oracle Application Express.
- [About the Embedded PL/SQL Gateway](#)
Oracle XML DB Protocol Server with the embedded PL/SQL gateway installs with Oracle Database.
- [About Oracle HTTP Server with mod_plsql](#)
The Oracle HTTP Server (Apache) with `mod_plsql` can be placed on the same physical machine as the database, or on a separate physical machine.
- [Selecting a Web Listener in an Oracle RAC Environment](#)
Oracle recommends using Oracle REST Data Services when running Oracle Application Express in an Oracle Real Application Clusters (Oracle RAC) environment.

2.4.1 About Choosing a Web Listener

You can choose any one of the three Web listeners - Oracle REST Data Services, Oracle HTTP Server, and Embedded PL/SQL gateway.

Oracle Application Express uses a simple architecture where pages are dynamically generated using metadata stored within the Oracle Database. There is no code generation or file based compilation. Once fully installed, a Uniform Resource Locator (URL) is defined for both developers and end users to access Oracle Application Express. Users require only a Web browser and the required URL. No additional client software is required.



The Web listener functions as a communications broker between the Web browser and the Oracle Application Express objects in the Oracle database by mapping browser requests into database stored procedure calls. Oracle Application Express supports the following Web listeners:

- Oracle REST Data Services
- Embedded PL/SQL gateway (legacy web listener)
- Oracle HTTP Server (deprecated)

2.4.2 Comparing Web Listener Options

Learn about comparing Web listener options for Oracle Application Express.

[Table 2-2](#) compares Web Listener options for Oracle Application Express.

Table 2-2 Web Listener Comparisons

Capabilities	Oracle REST Data Services	Embedded PL/SQL Gateway	Oracle HTTP Server with <code>mod_plsql</code>
Image location	File system	Within the database	File system
Configuration options	GUI interface (ORDS 3.x); Administration pages	Database initialization parameters	Database Access Descriptor (DAD)
Connection Pool settings	JDBC parameters	SHARED_SERVERS; MAX_SHARED_SERVERS	Min/MaxSpareServers; MaxClients
Emit RESTful Web Services	Yes (ORDS 3.x Release)	No	No
Support multi-databases	Yes, including Oracle RAC	No	Yes, including Oracle RAC
Virus scan files	Yes, with integration of ICAP server	No	No
PDF printing	Yes, included FOP support	No	No

Table 2-2 (Cont.) Web Listener Comparisons

Capabilities	Oracle REST Data Services	Embedded PL/SQL Gateway	Oracle HTTP Server with <code>mod_plsql</code>
Environment recommendations	All	Development only	All Note: Deprecated as of Oracle HTTP Server 12c (12.1.3).

2.4.3 Security Considerations When Using the Embedded PL/SQL Gateway

Oracle does not recommend the embedded PL/SQL gateway for applications running on the Internet.

The embedded PL/SQL gateway runs in the database as part of the Oracle XML DB HTTP listener. The Oracle XML DB HTTP listener and embedded PL/SQL gateway provides the equivalent core features of Oracle HTTP Server and `mod_plsql`. Because the HTTP Listener runs in the same database where Oracle Application Express is installed, it is not possible to separate the HTTP listener from the database. For this reason, Oracle does not recommend the embedded PL/SQL gateway for applications that run on the Internet. Additionally, the embedded PL/SQL gateway does not provide the same flexibility of configuration and detailed logging as Oracle REST Data Services or Oracle HTTP Server.

2.4.4 About Oracle REST Data Services

Oracle REST Data Services (formerly Oracle Application Express Listener) is Java-based, can be installed into any supported J2EE compliant Web Server, and is the preferred option for use with Oracle Application Express.

It is a free tool that is fully supported against Oracle WebLogic Server and Apache Tomcat. Oracle REST Data Services is part of the reference architecture used to operate the Oracle Database Cloud Service.

Note:

There are licensing costs associated with Oracle WebLogic Server.

Using Oracle REST Data Services creates an architecture which consists of:

- A web browser
- Oracle REST Data Services
- Oracle Database containing Oracle Application Express

The major advantage of this architecture is the separation of the mid-tier and the database tier.



Tip:

This architecture is appropriate for Oracle Real Application Clusters (Oracle RAC) environments.



See Also:

- ["Web Listener Requirements"](#)
- ["Installing Application Express and Configuring Oracle REST Data Services"](#)
- ["Installing Oracle REST Data Services" in *Oracle REST Data Services Installation, Configuration, and Development Guide*](#)
- ["Selecting a Web Listener in an Oracle RAC Environment"](#)

2.4.5 About the Embedded PL/SQL Gateway

Oracle XML DB Protocol Server with the embedded PL/SQL gateway installs with Oracle Database.

Embedded PL/SQL gateway provides the Oracle database with a Web server and also the necessary infrastructure to create dynamic applications. The embedded PL/SQL gateway runs in the Oracle XML DB Protocol server in the Oracle database and includes the core features of `mod_plsql`.

The embedded PL/SQL gateway consists of a simple two tier architecture: a Web browser and an Oracle database containing the embedded PL/SQL gateway and Oracle Application Express.

Advantages of the embedded PL/SQL gateway include:

- Ease of configuration
- Included in the database
- No separate server installation

Where Images Are Stored When Using the Embedded PL/SQL Gateway

When running Oracle Application Express with the Embedded PL/SQL Gateway, images are stored directly in the database within the Oracle XML DB repository. You can access images by using the WebDAV feature of Oracle XML DB or by using FTP.

 **See Also:**

- ["Security Considerations When Using the Embedded PL/SQL Gateway"](#)
- ["Web Listener Requirements"](#)
- ["Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway"](#)
- "Repository Access Using Protocols" in *Oracle XML DB Developer's Guide*

2.4.6 About Oracle HTTP Server with mod_plsql

The Oracle HTTP Server (Apache) with `mod_plsql` can be placed on the same physical machine as the database, or on a separate physical machine.

 **Note:**

`mod_plsql` is deprecated as of Oracle HTTP Server 12c (12.1.3). For more information, see My Oracle Support Note 1576588.1. Oracle recommends using Oracle REST Data Services. See "[Installing Application Express and Configuring Oracle REST Data Services](#)."

If Oracle HTTP Server is installed on the same physical machine as the database then the Oracle HTTP Server is included as part of the limited-use license included with the Oracle Database license. Otherwise, you must obtain a separate Oracle HTTP Server license.

Using Oracle HTTP Server (Apache) with `mod_plsql` creates an architecture which consists of:

- a Web browser
- Oracle HTTP Server with `mod_plsql`
- Oracle Database containing Oracle Application Express

This architecture separates the mid-tier and the database tier.

 **Tip:**

This architecture is appropriate for Oracle Real Application Clusters (Oracle RAC) environments.

Where Images Are Stored When Using Oracle HTTP Server

In an Oracle HTTP Server or Oracle Application Server configuration, images are stored on the file system in the location referenced by the alias `/i/`. If you are upgrading Oracle Application Express from a prior release, you can locate the images

directory on the file system, by reviewing the following files and searching for the text alias `/i/`:

- Oracle HTTP Server distributed Oracle9i Release 2—see the `httpd.conf` file.
- Oracle Application Server 10g—see the `marvel.conf` or `dads.conf` files.
- Oracle HTTP Server distributed with Oracle Database 11g or 12c —see the `marvel.conf` or `dads.conf` files.

Specific examples for locating the text alias `/i/` can be found in Copying the Images Directory.

See Also:

- ["Web Listener Requirements"](#)
- ["Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server"](#)
- ["Selecting a Web Listener in an Oracle RAC Environment"](#)
- ["About Configuring Oracle HTTP Server with `mod_plsql` with Oracle Application Express" in *Oracle Application Express App Builder User's Guide*](#)
- ["Copying the Images Directory ."](#)

2.4.7 Selecting a Web Listener in an Oracle RAC Environment

Oracle recommends using Oracle REST Data Services when running Oracle Application Express in an Oracle Real Application Clusters (Oracle RAC) environment.

Oracle REST Data Services permits you to specify a connection in the service name format, so that one HTTP Server can access all nodes.

2.5 About the Oracle Application Express Runtime Environment

Learn about the Oracle Application Express runtime environment.

As with any software development life cycle, Oracle strongly recommends that you have different environments for development, testing/QA, and production. For testing and production instances, Oracle Application Express supports the ability to install just a runtime version of Oracle Application Express. This runtime environment minimizes the installed footprint and privileges and improves application security since in a runtime instance developers cannot inadvertently update a production application.

An Oracle Application Express runtime environment enables you to run production applications, but it does not provide a Web interface for administration. A runtime environment only includes the packages necessary to run your application, making it a more hardened environment. You administer the Oracle Application Express runtime environment using SQL*Plus or SQL Developer and the `APEX_INSTANCE_ADMIN` API.

 **See Also:**

"About the Advantages of the Application Express Runtime Environment" in *Oracle Application Express App Builder User's Guide*

Scripts are provided to remove or add the developer interface from an existing instance. To learn more, see one of the following for the corresponding type of installation:

 **See also:**

- For [Installing Application Express and Configuring Oracle REST Data Services](#), see "Converting Between Runtime and Full Development Environments"
- For [Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway](#), see "Converting Between Runtime and Full Development Environments"
- For [Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server](#), see "Converting Between Runtime and Full Development Environments"

3

Utilizing the Multitenant Architecture in Oracle Database 12c or Later

Learn about installation choices and different scenarios associated with copying and moving pluggable databases introduced by the Oracle Database 12c or later multitenant architecture with respect to Oracle Application Express (Application Express).

- [Understanding the Installation Choices](#)
Learn about the installation choices in Oracle Application Express.
- [Installing Application Express into an Application Container](#)
Learn about the application container that stores data and metadata for application back ends.
- [Installing Application Express into Different PDBs](#)
You can install different versions of Oracle Application Express into different PDBs.
- [Plugging in a PDB When Application Express is Installed in the Root Container](#)
Learn about scenarios in which the target database has Application Express installed into the root container, `CDB$ROOT` - the default installation option.
- [Plugging in a PDB When Application Express Is Not in the Root Container of the Target CDB](#)
The scenarios in this section describe when Application Express is not installed in the root container, `CDB$ROOT`, by explicitly removing it as described in "Uninstalling Application Express from a CDB".

3.1 Understanding the Installation Choices

Learn about the installation choices in Oracle Application Express.

Oracle Database 12c Release 1 (12.1) introduces the multitenant architecture. This database architecture has a multitenant container database (CDB) that includes a root container, `CDB$ROOT`, a seed database, `PDB$SEED`, and multiple pluggable databases (PDBs). Each pluggable database is equivalent to a separate database instance in Oracle Database release 11g. The root container, `CDB$ROOT`, holds common objects that are accessible to every PDB utilizing metadata links or object links. The seed database, `PDB$SEED`, is used when creating a new PDB to seed the new database. The key benefit of the Oracle Database 12c or later multitenant architecture is that the database resources, such as CPU and memory, can be shared across all of the PDBs. This architecture also enables many databases to be treated as one for tasks such as upgrades or patches, and backups.

When configuring multitenant architecture, Oracle Application Express is installed in the root container database by default in Oracle Database 12c Release 1 (12.1). In the default installation the root container, `CDB$ROOT`, includes the `APEX_040200` schema to store the common database objects for the Application Express engine such as packages, functions, procedures and views. The seed database, `PDB$SEED`, also

includes the `APEX_040200` schema to store the tables that are part of the Application Express engine.

You can create a new PDB by copying `PDB$SEED`, which includes the `APEX_190200` schema with the Application Express tables, and creating metadata links back to the common database objects held in the `APEX_190200` schema within the `CDB$ROOT`. As such there are multiple copies of the Application Express engine tables and only single copies of the Application Express engine packages, functions, procedures and views. Each PDB will have the `APEX_190200` schema and have its own copy of the Application Express engine's tables so that it can hold the metadata for the Application Express applications defined within that PDB.

 **Tip:**

Oracle recommends removing Oracle Application Express from the root container database for the majority of use cases, except for hosting companies or installations where all pluggable databases (PDBs) utilize Oracle Application Express and they all need to run the exact same release and patch set of Oracle Application Express.

 **See Also:**

["Installing Application Express into Different PDBs"](#)

3.2 Installing Application Express into an Application Container

Learn about the application container that stores data and metadata for application back ends.

- [About Application Containers](#)
An application container is a CDB component that stores data and metadata for application backends.
- [Creating Application Container](#)
To create a PDB within a CDB as an application container, you use the `AS APPLICATION CONTAINER` clause of the `create PDB` command.
- [Installing or Upgrading Application Express in an Application Container](#)
- [Verifying the Application Container Installation](#)
- [Creating an Application Seed](#)
An application seed is used to provision application PDBs with the application root's applications pre-installed.
- [Creating an Application PDB from the Application Root Seed](#)
An application PDB is created by issuing the `CREATE PLUGGABLE DATABASE` statement from the application root.

- [Configure HTTP Access to the Application PDB](#)
Configure a new application PDB for HTTP access.

3.2.1 About Application Containers

An application container is a CDB component that stores data and metadata for application backends.

Oracle Application Express can be installed into an application container using the `apxappcon.sql` script. An application container consists of an application root where the application is defined and one or more PDBs that share data and metadata about the application from the application root. You can have multiple application containers within a CDB and each container can have a different version of Oracle Application Express.

Patching or upgrading Oracle Application Express in an application container is simplified, because these actions are done against the application root. When an application PDB wishes to uptake the patch or upgraded version, it simply syncs with the application root. Oracle Application Express continues to run in the application PDB at the existing version until the application PDB syncs with the application root.

3.2.2 Creating Application Container

To create a PDB within a CDB as an application container, you use the `AS APPLICATION CONTAINER` clause of the `create PDB` command.

To create Application Container:

1. Use the `AS APPLICATION CONTAINER` clause of the `CREATE PLUGGABLE DATABASE` statement to create an application container.
2. Open the application container.

For Example:

```
CREATE PLUGGABLE DATABASE apex_aproot1 AS APPLICATION CONTAINER admin
user admin IDENTIFIED
    BY <admin_password> FILE_NAME_CONVERT=('pdbseed','apex_aproot1');
ALTER PLUGGABLE DATABASE apex_aproot1 open;
```

Note:

`apex_aproot1` and the `admin` user in the previous example can be any valid ORACLE identifier.

3.2.3 Installing or Upgrading Application Express in an Application Container

To install or upgrade Application Express in an Application Container:

1. Connect to Application Container.
2. Run `apxappcon.sql`.

`apxappcon.sql` installs Application Express as an application named `APEX` into the application root.

The script takes the exact same first four arguments as the `apexins.sql` script, with the addition of a fifth parameter which is the password to use for the `APEX_PUBLIC_USER` password. In an upgrade installation, the fifth argument is ignored because the `APEX_PUBLIC_USER` database user will already exist.

For example:

```
ALTER SESSION SET CONTAINER = apex_aproot1;

@apxappcon.sql SYS AUX SYS AUX TEMP /i/ P@ssw0rd!
```

3.2.4 Verifying the Application Container Installation

To verify the Application Container installation:

1. Manually inspect the log file of the installation for ORA- or PLS- errors.
2. Compile invalid objects by running the following command:

For example:

```
ALTER SESSION SET CONTAINER=apex_aproot1;

begin
  sys.dbms_utility.compile_schema( 'APEX_190200', false );
  sys.dbms_utility.compile_schema( 'FLOWS_FILES', false );
end;
/
```

3. Query `dba_applications` and `dba_app_errors`.

```
SQL> select app_name, app_version, app_status from dba_applications
where app_name = 'APEX';
```

APP_NAME	APP_VERSION	APP_STATUS

APEX	19.2	NORMAL

```
SQL> select app_name, app_statement, errornum, errormsg from
dba_app_errors where app_name = 'APEX';
```

```
no rows selected
```

3.2.5 Creating an Application Seed

An application seed is used to provision application PDBs with the application root's applications pre-installed.

To create an Application Seed:

1. Connect to CDB\$ROOT as sysdba.
2. Alter session and set container to the application root.
3. Use the AS SEED clause of the CREATE PLUGGABLE DATABASE statement to create an application seed.
4. Sync the APEX application with the application seed.
5. Compile invalid objects.
6. Open the application seed in read only mode.

 **Note:**

apex_approot1 and the admin user in the following example can be any valid Oracle identifier.

For example:

```
ALTER SESSION SET CONTAINER=apex_approot1;

CREATE PLUGGABLE DATABASE as seed admin user admin identified by
<admin_password> file_name_convert=('pdbseed','apex_approot1_seed');

ALTER PLUGGABLE DATABASE apex_approot1$seed open;

ALTER SESSION SET CONTAINER=apex_approot1$seed;

ALTER PLUGGABLE DATABASE application APEX sync;

begin
  sys.dbms_utility.compile_schema( 'APEX_190200', false );
  sys.dbms_utility.compile_schema( 'FLOWS_FILES', false );
end;
/

ALTER PLUGGABLE DATABASE close immediate;

ALTER PLUGGABLE DATABASE open read only;
```

3.2.6 Creating an Application PDB from the Application Root Seed

An application PDB is created by issuing the `CREATE PLUGGABLE DATABASE` statement from the application root.

The `PLUGGABLE DATABASE` is created from the application container seed so the APEX application is already installed and ready for configuration.

To create an Application PDB from the Application Root Seed:

1. Connect to `CDB$ROOT` as `sysdba`.
2. Alter session and set container to the application root.
3. Use the `CREATE PLUGGABLE DATABASE` command to create a PDB from the application seed.

Note:

`apex_approot1` and the `admin` user in the following example can be any valid Oracle identifier.

For example:

```
ALTER SESSION SET CONTAINER=apex_approot1;

CREATE PLUGGABLE DATABASE apex_pdb1 admin user admin identified by
<admin password> file_name_convert=('apex_approot1_seed', 'apex_pdb1');

ALTER PLUGGABLE DATABASE apex_pdb1 open;

ALTER SESSION SET CONTAINER=apex_pdb1;

SQL> select app_name, app_version, app_status from dba_applications
where app_name = 'APEX';
```

APP_NAME	APP_VERSION	APP_STATUS
APEX	19.2	NORMAL

3.2.7 Configure HTTP Access to the Application PDB

Configure a new application PDB for HTTP access.

Configure the new application PDB for HTTP access by following the instructions starting with the section "[Downloading and Installing Oracle REST Data Services](#)." If you are using legacy web listeners, follow the corresponding sections in "[Installing](#)

[Application Express and Configuring Web Listener - Embedded PL/SQL Gateway](#)" and ["Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server"](#).

3.3 Installing Application Express into Different PDBs

You can install different versions of Oracle Application Express into different PDBs.

Providing Application Express is not installed in the container database, you can install a local Application Express within each PDB as required. When Application Express is installed locally there are no Application Express metadata linked objects and all packages, views, and tables are created within the `APEX_190200` schema, within each PDB where Application Express is installed.

- [Uninstalling Application Express from a CDB](#)
Learn how to uninstall Application Express from a CDB.
- [Installing Application Express Locally in a PDB](#)
Learn how to install Application Express locally in a PDB.
- [Installing Application Express into a CDB](#)

3.3.1 Uninstalling Application Express from a CDB

Learn how to uninstall Application Express from a CDB.

To uninstall Oracle Application Express from a CDB:

Note:

Installing or removing Application Express from a CDB requires a local connection to the database.

This section describes removing Application Express from a CDB. If you wish to remove Oracle Application Express from the CDB that shipped with Oracle Database 12.1, you should use `apxremov_con.sql` from either `$ORACLE_HOME/apex`, or from a 4.2.6 APEX distribution.

1. Change to the `apex` directory in the location where you unzipped the distribution.
2. Start SQL*Plus and connect to the database where Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxremov.sql`.

For example:

```
@apxremov.sql
```

 **Note:**

If you run `apexremov.sql` after PDBs have been added to the CDB, then Application Express uninstalls from all of the PDBs. Therefore, any applications defined in any of the PDBs will be removed.

3.3.2 Installing Application Express Locally in a PDB

Learn how to install Application Express locally in a PDB.

Once you have removed Application Express from the container database by following the instructions in "[Uninstalling Application Express from a CDB](#)," you can install Application Express locally in a PDB.

To install Application Express locally in a PDB:

1. Change the `apex` directory in the location where you unzipped the distribution.
2. Start SQL*Plus and connect to the database where Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Set the container to the PDB you want to install Application Express locally:

```
ALTER SESSION SET CONTAINER = <PDB_name>;
```

4. Select the appropriate installation option.

Full development environment provides complete access to the App Builder environment to develop applications. A **Runtime environment** enables users to run applications that cannot be modified.

Available installation options include:

- **Full development environment** - Run `apexins.sql` passing the following four arguments in the order shown:

```
@apexins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- `tablespace_apex` is the name of the tablespace for the Application Express application user.

- *tablespace_files* is the name of the tablespace for the Application Express files user.
- *tablespace_temp* is the name of the temporary tablespace or tablespace group.
- *images* is the virtual directory for Application Express images. To support future Application Express upgrades, define the virtual image directory as */i/*.

For example:

```
@apexins.sql SYSaux SYSaux TEMP /i/
```

- **Runtime environment** - Run `apxrtins.sql` passing the following four arguments in the order shown:

```
@apxrtins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- *tablespace_apex* is the name of the tablespace for the Application Express application user.
- *tablespace_files* is the name of the tablespace for the Application Express files user.
- *tablespace_temp* is the name of the temporary tablespace or tablespace group.
- *images* is the virtual directory for Application Express images. To support future Application Express upgrades, define the virtual image directory as */i/*.

For example:

```
@apxrtins.sql SYSaux SYSaux TEMP /i/
```

5. If you are using Oracle REST Data Services, complete the appropriate steps in *Installing Application Express and Configuring Oracle REST Data Services*.
6. If you are using Oracle HTTP Server, complete the appropriate steps in *Installing Application Express and Configuring Embedded PL/SQL Gateway*.
7. If you are using Embedded PL/SQL Gateway, run `apex_epg_config.sql` passing the path to the Oracle home and then complete the appropriate steps in *Installing Application Express and Configuring Oracle HTTP Server*.

 **Note:**

Oracle Database SQL Language Reference for more information about SQL*Plus

When Application Express installs, it creates the following database accounts:

- `APEX_190200` - This account owns the Application Express schema and metadata.
- `FLows_FILES` - This account owns the Application Express uploaded files.

- `APEX_PUBLIC_USER` - This minimally privileged account is used for Application Express configuration with Oracle REST Data Services or Oracle HTTP Server and `mod_plsql`.
- `APEX_INSTANCE_ADMIN_USER` - A minimally privileged account used for the REST Administration Interface.

 **Tip:**

To learn more about the REST Administration Interface, see "About the REST Administration Interface" in *Oracle Application Express Administration Guide*

If you configured RESTful Web services, then these additional accounts are created:

- `APEX_REST_PUBLIC_USER` - The account used when invoking RESTful Services definitions stored in Application Express.
- `APEX_LISTENER` - The account used to query RESTful Services definitions stored in Application Express.

 **See Also:**

- ["Installing Application Express and Configuring Oracle REST Data Services."](#)
- ["Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server."](#)
- ["Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway."](#)

3.3.3 Installing Application Express into a CDB

To install Application Express into a CDB:

 **Note:**

Installing or removing Application Express from a CDB requires a local connection to the database.

1. Change your working directory to the `apex` directory in the location where you unzipped the distribution.
2. Start SQL*Plus and connect to `CDB$ROOT` of the database where Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Select the appropriate installation option.

Full development environment provides complete access to the App Builder environment to develop applications. A **Runtime environment** enables users to run applications that cannot be modified.

Available installation options include:

- **Full development environment.** Run `apexins.sql` passing the following four arguments in the order shown:

```
@apexins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- `tablespace_apex` is the name of the tablespace for the Oracle Application Express application user.
- `tablespace_files` is the name of the tablespace for the Application Express files user.
- `tablespace_temp` is the name of the temporary tablespace or tablespace group.
- `images` is the virtual directory for Application Express images. To support future Application Express upgrades, define the virtual image directory as `/i/`.

Example:

```
@apexins.sql SYSAUX SYSAUX TEMP /i/
```

- **Runtime environment.** Run `apxrtins.sql` passing the following arguments in the order shown:

```
@apxrtins.sql tablespace_apex tablespace_files tablespace_temp
images
```

Where:

- `tablespace_apex` is the name of the tablespace for the Application Express application user.
- `tablespace_files` is the name of the tablespace for the Application Express files user.

- `tablespace_temp` is the name of the temporary tablespace or tablespace group.
- `images` is the virtual directory for Application Express images. To support future Application Express upgrades, define the virtual image directory as `/i/`.

Example:

```
@apxrtins.sql SYSAUX SYSAUX TEMP /i/
```

4. If using Oracle REST Data Services, complete appropriate steps in Installing Application Express and Configuring Oracle REST Data Services.
5. If using the Embedded PL/SQL Gateway, complete appropriate steps in Installing Application Express and Configuring Embedded PL/SQL Gateway.
6. If using the Oracle HTTP Server, complete appropriate steps in Installing Application Express and Configuring Oracle HTTP Server.

When Application Express installs, it creates the following database accounts:

- `APEX_190200` - This account owns the Application Express schema and metadata.
- `FLows_FILES` - This account owns the Application Express uploaded files.
- `APEX_PUBLIC_USER` - This minimally privileged account is used for Application Express configuration with Oracle REST Data Services or Oracle HTTP Server and `mod_plsql`.
- `APEX_INSTANCE_ADMIN_USER` - A minimally privileged account used for the REST Administration Interface.

 **Tip:**

To learn more about the REST Administration Interface, see "About the REST Administration Interface" in *Oracle Application Express Administration Guide*

If you configured RESTful Web services, then these additional accounts are created:

- `APEX_REST_PUBLIC_USER` - The account used when invoking RESTful Services definitions stored in Application Express.
- `APEX_LISTENER` - The account used to query RESTful Services definitions stored in Application Express.

 **See Also:**

- ["Using SQL*Plus" in *SQL*Plus User's Guide and Reference*](#)
- ["Patching or Upgrading Application Express in the CDB"](#)
- ["About the Oracle Application Express Runtime Environment"](#)
- ["Installing Application Express and Configuring Oracle REST Data Services"](#)
- ["Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway"](#)
- ["Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server"](#)

3.4 Plugging in a PDB When Application Express is Installed in the Root Container

Learn about scenarios in which the target database has Application Express installed into the root container, CDB\$ROOT - the default installation option.

This section describes scenarios in which the target database has Application Express installed into the root container, CDB\$ROOT - the default installation option. Note there are multiple scenarios related to where the database being plugged in originated from and how Application Express was configured in the originating database.

- [Scenario 1: Plug-in Non-CDB with Application Express](#)
Plug-in a Non-CDB with Application Express.
- [Scenario 2: Plug-in PDB with a Common Application Express from Another CDB](#)
Plug-in a PDB with Application Express from another CDB.
- [Scenario 3: Plug-in PDB with a Local Application Express from Another CDB](#)
Plug-in a PDB with a local Application Express from another CDB.
- [Scenario 4: Plug-in Non-CDB or PDB with No Application Express](#)
Plug-in a Non-CDB or PDB if the Application Express is not installed.
- [Working with Incompatible Application Express Versions](#)
Learn how to work with the incompatible versions of Application Express.

3.4.1 Scenario 1: Plug-in Non-CDB with Application Express

Plug-in a Non-CDB with Application Express.

If you are upgrading from a previous Oracle Database release, then you first need to upgrade to a Oracle Database 12c non-CDB (or standalone database) or later and then plug the database into your CDB. Alternatively, if you have configured a non-CDB Oracle Database 12c or later, you may now want to plug this database into a CDB. In both cases, the originating database has Application Express installed and was not formerly a PDB.

As described in the *Oracle Database Installation Guide* for your operating system, when plugging in a standalone database you need to run the `$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql` script. This script creates the necessary metadata linked objects, instead of local objects and recompiles the database objects for all common database options, including Application Express.

After installing Application Express, you need to configure the Web listener for the PDB.

If the version of Application Express installed in the originating database (which is now a PDB) is different from what is installed into the root container of the target, an error will be raised when trying to open the PDB.



See Also:

- ["Installing Application Express and Configuring Oracle REST Data Services"](#)
- ["Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway"](#)
- ["Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server"](#)
- ["Working with Incompatible Application Express Versions"](#)

3.4.2 Scenario 2: Plug-in PDB with a Common Application Express from Another CDB

Plug-in a PDB with Application Express from another CDB.

If you are copying or moving a PDB from an existing Oracle Database 12c or later where the originating CDB had Application Express installed in the root container, you will not need to perform any additional steps, other than configuring the Web listener for the PDB.

This scenario assumes Oracle Application Express release 19.2 is installed and the `APEX_190200` schema within the PDB being plugged in already has the metadata linked objects defined and will compile without error against the metadata linked objects within the target CDB.

If the version of Application Express installed in the originating database is different from what is installed in the root container of the target an error is raised when trying to open the PDB.

 **See Also:**

- ["Installing Application Express and Configuring Oracle REST Data Services"](#)
- ["Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway"](#)
- ["Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server"](#)
- ["Working with Incompatible Application Express Versions"](#)

3.4.3 Scenario 3: Plug-in PDB with a Local Application Express from Another CDB

Plug-in a PDB with a local Application Express from another CDB.

If you are copying or moving a PDB from an existing Oracle Database 12c or later where Application Express was not installed in the root container but is installed locally, then you need to perform additional steps before the PDB can be opened without errors.

This scenario assumes Oracle Application Express release 19.2 is installed and the APEX_190200 schema within the PDB being plugged in contains all of the Application Express database objects and has no metadata linked objects. Therefore, you need to run `$ORACLE_HOME/rdbms/admin/apex_to_common.sql` to remove the common objects and create the metadata links for the packages, views and so forth.

To replace local objects with metadata links in the PDB:

1. Change your working directory to `$ORACLE_HOME/rdbms/admin`.
2. Start SQL*Plus and connect to the database where Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Set the container to the PDB to be configured:

```
ALTER SESSION SET CONTAINER = <PDB_name>;
```

4. Run `apex_to_common.sql`. For example:

```
@apex_to_common.sql
```

If the version of Application Express installed in the originating database is different from what is installed in the root container of the target an error is raised when trying to open the PDB.



See Also:

["Working with Incompatible Application Express Versions."](#)

3.4.4 Scenario 4: Plug-in Non-CDB or PDB with No Application Express

Plug-in a Non-CDB or PDB if the Application Express is not installed.

If you are plugging in a non-CDB, or copying or moving a PDB from another CDB, and Application Express is not installed in the originating database or PDB then a warning is raised in the alert log when opening the PDB in the target database.

In this scenario, the Application Express schema, such as `APEX_190200` for Oracle Application Express release 19.2, will not be present in the originating database or the PDB being plugged in. In order to open the PDB without issue and be able to run Application Express within the new PDB, you must install Application Express into the originating database or PDB before attempting to plug in to the target database. You should install the same version of Application Express into the originating database or PDB as the version installed into the target database.

3.4.5 Working with Incompatible Application Express Versions

Learn how to work with the incompatible versions of Application Express.

If the version of Application Express in the root container, `CDB$ROOT`, is not the same as the Application Express version in the PDB then an error is raised every time the PDB is opened preventing normal database operations within the PDB. The PDB can only be opened in restricted mode by users with `RESTRICTED SESSION` privilege, until the versions are compatible.

- [Patching or Upgrading Application Express in the CDB](#)
Learn how to patch or upgrade Application Express in the root container.
- [Patching or Upgrading Application Express in a PDB](#)
Learn how to patch or upgrade Application Express in a PDB.

3.4.5.1 Patching or Upgrading Application Express in the CDB

Learn how to patch or upgrade Application Express in the root container.

If the version of Application Express in the PDB is a later minor release version than the version of Application Express in the root container (for example, the PDB contains Application Express release 5.1.4 and the CDB contains Application Express release 5.1.3) then you must patch the version of Application Express in the root container to be able to open the PDB without error. If the major version of Application Express in the PDB is higher than the version in the CDB (for example the PDB has Application Express release 19.2 and the CDB has Application Express release 18.1) then you

must upgrade the version of Application Express in the CDB to be able to open the PDB without error.

To patch Application Express in the root container:

1. Download the appropriate patch from My Oracle Support.
2. Unzip and extract the installation files.
3. Change your working directory to where the installation files were extracted
4. Start SQL*Plus and connect to the database where Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

5. Run `apxpatch_con.sql` for example:

```
@apxpatch_con.sql
```

6. Follow the instructions outlined in the Patch Set Notes for updating the images directory based on the Web Server you are using. If you are using the Embedded PL/SQL Gateway, run `apxldimg_con.sql` for example:

```
@apxldimg_con.sql
```

Tip:

To upgrade Application Express in the root container, see Reinstalling Application Express into a CDB.

See Also:

["Installing Application Express into a CDB"](#)

3.4.5.2 Patching or Upgrading Application Express in a PDB

Learn how to patch or upgrade Application Express in a PDB.

If the minor version of Application Express in the PDB is lower than the version of Application Express in the root container (for example the PDB has Application Express release 4.2.0 and the CDB has Application Express release 4.2.6) then it will be necessary to patch the version of Application Express in the PDB. If the major version of Application Express in the PDB is lower than the version in the root container (for example, the PDB has Application Express release 4.2 and the CDB has

Application Express release 19.2) then the version of Application Express in the PDB will need to be upgraded.

- [Patching Application Express in a PDB](#)
Learn how to patch Application Express in a PDB.
- [Upgrading Application Express in a PDB](#)
Learn how to upgrade Application Express in a PDB.

3.4.5.2.1 Patching Application Express in a PDB

Learn how to patch Application Express in a PDB.

To patch Application Express in a PDB:

1. Download the appropriate patch from My Oracle Support.
2. Unzip and extract the installation files.
3. Change your working directory to where the installation files were extracted
4. Start SQL*Plus and connect to the database where Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

5. Run `apxpatch.sql` using `catcon.pl` like the following example:

```
host &OH_HOME/perl/bin/perl -I

&OH_HOME/rdbms/admin &OH_HOME/rdbms/admin/catcon.pl -b apxpatch -c '<PDB_name>'
apxpatch.sql
```

Where:

- `&OH_HOME` represents the full path to the Oracle home
 - `<PDB_name>` is the name of the PDB you are patching
6. Follow the instructions outlined in the patch set notes for updating the images directory based on the Web Server you are using. If you are using the Embedded PL/SQL Gateway, run `apex_epg_config.sql` for example:

```
ALTER SESSION SET CONTAINER = <PDB_name>;
@apex_epg_config.sql
```

3.4.5.2.2 Upgrading Application Express in a PDB

Learn how to upgrade Application Express in a PDB.

To upgrade Application Express in a PDB:

1. Unzip and extract the installation files.
2. Change your working directory to where the installation files were extracted

3. Start SQL*Plus and connect to the database where Application Express is installed as SYS specifying the SYSDBA role. For example:
 - On Windows:


```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```
 - On UNIX and Linux:


```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```
4. Run `apexins_nocdb.sql` or `apxrtins_nocdb.sql` using `catcon.pl` like the following example:


```
host &OH_HOME/perl/bin/perl -I

&OH_HOME/rdbms/admin &OH_HOME/rdbms/admin/catcon.pl -b apexins -c '<PDB_name>'
apexins_nocdb.sql --pSYSAUX --pSYSAUX --pTEMP --p/i/ --p1,2,3
```

Where:

 - `&OH_HOME` represents the full path to the Oracle home
 - `<PDB_name>` is the name of the PDB you are patching
5. Follow the instructions outlined in the patch set notes for updating the images directory based on the Web Server you are using.

3.5 Plugging in a PDB When Application Express Is Not in the Root Container of the Target CDB

The scenarios in this section describe when Application Express is not installed in the root container, `CDB$ROOT`, by explicitly removing it as described in "Uninstalling Application Express from a CDB".

In such cases, you can optionally install a local Application Express into each PDB independently. If Application Express is installed into a PDB it is considered to be installed locally and has no metadata linked objects. There are multiple scenarios related to where the database being plugged in originated from and how Application Express was configured in the originating database.

- [Scenario 1: Plug-in a Non-CDB with Application Express](#)
Plug-in a Non-CDB with Application Express.
- [Scenario 2: Plug-in PDB with Common Application Express from Another CDB](#)
Plug-in a PDB with Application Express from another CDB.
- [Scenario 3: Plug-in PDB with a Local Application Express from Another CDB](#)
Plug-in a PDB with local Application Express from another CDB.
- [Scenario 4: Plug-in a Non-CDB or PDB with No Application Express](#)
Plug-in a Non-CDB or PDB if Application Express is not installed.

**See Also:**

["Uninstalling Application Express from a CDB."](#)

3.5.1 Scenario 1: Plug-in a Non-CDB with Application Express

Plug-in a Non-CDB with Application Express.

If you are upgrading from a previous Oracle Database release then you need to upgrade to Oracle Database 12c or later non-CDB (or standalone database) and then plug the database into your CDB. Alternatively you may have configured a non-CDB Oracle Database 12c or later that you now want to plug into a CDB. In both cases, the originating database had Application Express installed and was not formerly a PDB.

As described in the *Oracle Database Installation Guide* for your operating system, when plugging in a standalone database you need to run the `$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql` script. This script creates the necessary metadata linked objects (instead of local objects) and recompiles the database objects for all common database options. However, because Application Express has been removed from the root container, the script will not create any metadata links for any of the Application Express objects. The script does not change the Application Express installation from the originating database and no additional steps are needed other than configuring the Web listener depending on the type of Web listener you are using.

**See Also:**

- ["Installing Application Express and Configuring Oracle REST Data Services"](#)
- ["Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway"](#)
- ["Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server"](#)

3.5.2 Scenario 2: Plug-in PDB with Common Application Express from Another CDB

Plug-in a PDB with Application Express from another CDB.

If you are copying or moving a PDB from an existing Oracle Database 12c where the originating CDB had Application Express installed in the root container, then an error is raised whenever you try to open the PDB. The error is due to the originating PDB included metadata links to objects in the originating root container which cannot be recompiled because the target root container does not include Application Express. You will not be able to open the PDB unless you remove Application Express from the PDB or if Application Express is already installed in the target root container. Oracle does not support installing Application Express in the root container if it contains PDBs with locally installed Application Express.

3.5.3 Scenario 3: Plug-in PDB with a Local Application Express from Another CDB

Plug-in a PDB with local Application Express from another CDB.

If you are copying or moving a PDB from an existing Oracle Database 12c or later where the originating PDB had a local Application Express installed (not in the CDB) then you do not need to perform any additional steps, other than configuring the Web listener in the PDB depending on the type of Web listener you are using.

This scenario assumes Oracle Application Express release 19.2 is installed and the `APEX_190200` schema within the PDB being plugged in, already has all of the Application Express objects defined locally and no metadata links.

See Also:

- ["Installing Application Express and Configuring Oracle REST Data Services"](#)
- ["Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway"](#)
- ["Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server"](#)

3.5.4 Scenario 4: Plug-in a Non-CDB or PDB with No Application Express

Plug-in a Non-CDB or PDB if Application Express is not installed.

If you are plugging in a non-CDB, or copying or moving a PDB from another CDB, where Application Express was not installed in the originating database or PDB then you do not need to perform any additional steps. There will be no Application Express engine schema, such as `APEX_190200`, within the PDB, and the PDB can be started without error.

4

Installing Application Express and Configuring Oracle REST Data Services

This section describes how to install Oracle Application Express by downloading a ZIP file from the Oracle Application Express download page and then downloading and installing Oracle REST Data Services.

These instructions apply to both new and upgrade installations.

- [Oracle Database Requirements](#)
Oracle Application Express release 19.2 requires an Oracle Database release 11.2.0.4 or later, including Enterprise Edition and Express Edition (Oracle Database 18c XE).
- [Oracle REST Data Services Requirement](#)
Oracle Application Express requires Oracle REST Data Services (ORDS) 19.1 or 19.2.
- [Performing Pre-installation Tasks for Oracle Application Express](#)
Review and perform pre-installation tasks before installing Oracle Application Express.
- [Downloading and Installing Oracle Application Express](#)
Learn about downloading and installing Oracle Application Express.
- [Downloading and Installing Oracle REST Data Services](#)
Learn about downloading and installing Oracle REST Data Services.
- [Configuring Oracle REST Data Services](#)
Configuring Oracle REST Data Services requires that you copy the images directory, run a configuration to configure RESTful Services, configure static files support, and secure Oracle REST data service.
- [Enabling Network Services in Oracle Database 11g or Later](#)
You must enable network services in Oracle Database 11g or later versions to send outbound mail, use Web services, or use PDF report printing in Oracle Application Express.
- [Performing Security Tasks](#)
Oracle recommends configuring and using Secure Sockets Layer (SSL) to ensure that passwords and other sensitive data are not transmitted in clear text in HTTP requests.
- [Controlling the Number of Concurrent Jobs](#)
Learn about specifying the number of concurrently running jobs.
- [About Running Oracle Application Express in Other Languages](#)
You can install a single instance of Oracle Application Express with one or more of translated versions.
- [Installing Translated Versions of Oracle Application Express](#)
Learn about installing translated versions of Oracle Application Express.

- [Creating a Workspace and Adding Oracle Application Express Users](#)
Before you can develop or install applications, you must create a workspace, add Oracle Application Express users, and sign in to your workspace.
- [Performing Post Installation Tasks for Upgrade Installations](#)
Once you have verified that your upgrade installation was successful and all upgraded applications function properly, you should remove schemas from prior Oracle Application Express installations.
- [About Performance Optimization Tasks](#)
Learn about performance optimization.
- [Converting Between Runtime and Full Development Environments](#)
Learn about converting between runtime and full development environments.



See Also:

["Web Listener Requirements"](#)

4.1 Oracle Database Requirements

Oracle Application Express release 19.2 requires an Oracle Database release 11.2.0.4 or later, including Enterprise Edition and Express Edition (Oracle Database 18c XE).

If you are upgrading an Oracle Database version 12.1 CDB, you must download from My Oracle Support the one off patch for bug 20618595. Search for **20618595** on the Patches tab.

4.2 Oracle REST Data Services Requirement

Oracle Application Express requires Oracle REST Data Services (ORDS) 19.1 or 19.2.

4.3 Performing Pre-installation Tasks for Oracle Application Express

Review and perform pre-installation tasks before installing Oracle Application Express.

Before installing Oracle Application Express, Oracle recommends that you complete the following steps:

1. Review and satisfy all Oracle Application Express installation requirements.
2. If you are actively using Oracle Application Express and upgrading the current installation, then shut down with normal or immediate priority the Oracle Database instances where you plan to install Oracle Application Express. On Oracle Real Application Clusters (Oracle RAC) systems, shut down all instances on each node.

An alternative to shutting down the database, you can prevent all users from accessing Oracle Application Express when upgrading your installation from a previous release of Oracle Application Express. Oracle only recommends this option in high availability production environments where planned outages are not available. For all other scenarios, the database should be shut down.

To disable access to Oracle Application Express when an existing installation is using Oracle REST Data Services, shut down the appropriate application server where Oracle REST Data Services is deployed.

Once you have prevented access from Oracle Application Express users, you should log in to SQL*Plus as `SYS`, connecting to the database where Oracle Application Express is installed, and query `V$SESSION` to ensure there are no long running sessions which would interfere with the upgrade process.

3. Back up the Oracle Database installation.

Oracle recommends that you create a backup of the current Oracle Database installation before you install Oracle Application Express. You can use Oracle Database Recovery Manager, which is included in the Oracle Database installation, to perform the backup.

4. Start the Oracle Database instance that contains the target database.

After backing up the system, you must start the Oracle instance that contains the target Oracle Database. Do not start other processes such as a Web listener. However, if you are performing a remote installation, make sure the Web listener for the remote database has started.

 **Note:**

If you are connecting to a remote database, then start the Web listener.

 **See Also:**

- "[Oracle Application Express Installation Requirements](#)"
- "[Oracle Database Backup and Recovery User's Guide](#)"

4.4 Downloading and Installing Oracle Application Express

Learn about downloading and installing Oracle Application Express.

How you install Oracle Application Express depends upon by the type of database you into which you are installing. This section describes how to download and install Oracle Application Express.

- [Installing Application Express](#)
Install Oracle Application Express by downloading a ZIP file from the Oracle Application Express download page.
- [Creating or Updating Your Instance Administration Account](#)
Learn how to create or update Instance Administrator account.
- [Restarting Processes](#)
Restart the processes that you stopped before you began the installation.

- [Configuring the APEX_PUBLIC_USER Account](#)
It is important to correctly configure the `APEX_PUBLIC_USER` account to enable proper operation of Oracle Application Express.
- [Configuring RESTful Services](#)
In a new installation or upgrade of Oracle Application Express, you must run the configuration script `apex_rest_config.sql` to configure RESTful Services.



See Also:

["Utilizing the Multitenant Architecture in Oracle Database 12c or Later."](#)

4.4.1 Installing Application Express

Install Oracle Application Express by downloading a ZIP file from the Oracle Application Express download page.

To install Oracle Application Express:

1. For installations where the development will be in English only, download the file `apex_19.2_en.zip` from the Oracle Application Express download page. If the development will include languages other than English, download `apex_19.2.zip` from the Oracle Application Express download page. See:

<http://www.oracle.com/technetwork/developer-tools/apex/downloads/index.html>

Note that the actual file name may differ if a more recent release has shipped since this document was published.

2. Unzip downloaded zip file:
 - If English only, unzip `apex_19.2_en.zip` as follows, preserving directory names:
 - UNIX and Linux: `$ unzip apex_19.2_en.zip`
 - Windows: Double click the file `apex_19.2_en.zip` in Windows Explorer
 - If multiple languages, unzip `apex_19.2.zip` as follows, preserving directory names:
 - UNIX and Linux: `$ unzip apex_19.2.zip`
 - Windows: Double click the file `apex_19.2.zip` in Windows Explorer



Note:

You should keep the directory tree where you unzip the files short and not under directories that contain spaces. For example, within Windows unzip to `C:\TEMP`.

3. Change your working directory to `apex`.

4. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

5. Disable any existing password complexity rules for the default profile.
6. Select the appropriate installation option.

Full development environment provides complete access to the App Builder environment to develop applications. A **Runtime environment** enables users to run applications that cannot be modified.

Available installation options include:

- **Full development environment.** Run `apexins.sql` passing the following four arguments in the order shown:

```
@apexins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- `tablespace_apex` is the name of the tablespace for the Oracle Application Express application user.
- `tablespace_files` is the name of the tablespace for the Oracle Application Express files user.
- `tablespace_temp` is the name of the temporary tablespace or tablespace group.
- `images` is the virtual directory for Oracle Application Express images. For installations using EPG, `/i/` is the required value for the images argument. To support future Oracle Application Express upgrades, define the virtual image directory as `/i/`.

Example:

```
@apexins.sql SYSAUX SYSAUX TEMP /i/
```

 **Note:**

If you receive the following error, exit SQL*Plus and change your working directory to where you unzipped the installation file, for example `C:\TEMP` in Windows, before starting SQL*Plus:

```
SP2-0310: unable to open file "apexins.sql"
```

- **Runtime environment.** Run `apxrtins.sql` passing the following arguments in the order shown:

```
@apxrtins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- *tablespace_apex* is the name of the tablespace for the Oracle Application Express application user.
- *tablespace_files* is the name of the tablespace for the Oracle Application Express files user.
- *tablespace_temp* is the name of the temporary tablespace or tablespace group.
- *images* is the virtual directory for Oracle Application Express images. To support future Oracle Application Express upgrades, define the virtual image directory as */i/*.

Example:

```
@apxrtins.sql SYSAUX SYSAUX TEMP /i/
```

When Application Express installs, it creates the following database accounts:

- **APEX_190200** - This account owns the Application Express schema and metadata.
- **FLows_FILES** - This account owns the Application Express uploaded files.
- **APEX_PUBLIC_USER** - This minimally privileged account is used for Application Express configuration with Oracle REST Data Services or Oracle HTTP Server and `mod_plsql`.
- **APEX_INSTANCE_ADMIN_USER** - A minimally privileged account used for the REST Administration Interface.

 **Tip:**

To learn more about the REST Administration Interface, see "About the REST Administration Interface" in *Oracle Application Express Administration Guide*

If you configured RESTful Web services, then these additional accounts will be created:

- **APEX_REST_PUBLIC_USER** - The account used when invoking RESTful Services definitions stored in Oracle Application Express.
- **APEX_LISTENER** - The account used to query RESTful Services definitions stored in Oracle Application Express.

If you are upgrading from a previous release, then **FLows_FILES** already exists and **APEX_PUBLIC_USER** is created if it does not already exist.

 **See Also:**

- "Configuring Password Protection" in *Oracle Database Security Guide*
- ["About the Oracle Application Express Runtime Environment"](#)
- *"SQL*Plus User's Guide and Reference"* for more information about SQL*Plus
- Oracle Application Express must be installed from a writable directory on the file system. See ["Reviewing a Log of an Installation Session"](#)

4.4.2 Creating or Updating Your Instance Administration Account

Learn how to create or update Instance Administrator account.

This section describes how to create or update your Instance Administrator account.

 **Tip:**

Skip this section if you are upgrading from a previous release of Oracle Application Express. In an upgrade scenario, the Instance Administrator account and password is preserved and carried over from the prior release.

- [What Is an Instance Administrator?](#)
Instance administrators are superusers that are responsible for managing an entire Oracle Application Express instance, including managing workspace provisioning, configuring features and instance settings, and managing security.
- [About apxchpwd.sql](#)
Running the `apxchpwd.sql` script enables you to create or update your Instance Administrator account.
- [Running apxchpwd.sql](#)
Run the `apxchpwd.sql` script to create and update your Instance Administrator account.

4.4.2.1 What Is an Instance Administrator?

Instance administrators are superusers that are responsible for managing an entire Oracle Application Express instance, including managing workspace provisioning, configuring features and instance settings, and managing security.

To perform these tasks, an Instance administrator signs in to the Oracle Application Express Administration Services application.

 **See Also:**

"Oracle Application Express Administration Services" in *Oracle Application Express Administration Guide*

4.4.2.2 About apxchpwd.sql

Running the `apxchpwd.sql` script enables you to create or update your Instance Administrator account.

You must run the `apxchpwd.sql` script in the following scenarios:

- **New Oracle Application Express installations** - Run `apxchpwd.sql` to create an Instance Administrator account and password.
- **Converting of a runtime environment to a development environment** - Run `apxchpwd.sql` to change the Instance Administrator account password.
- **Changing Your Instance Administrator Password** - Run `apxchpwd.sql` to change the password for an existing Instance Administrator account.
- **Unlocking Your Instance Administrator Account** - Run `apxchpwd.sql` to unlock an existing Instance Administrator account.

 **Tip:**

You do not need to run `apxchpwd.sql` when upgrading from a previous release of Oracle Application Express. In an upgrade scenario, the Instance Administrator account password is preserved and carried over from the prior release.

4.4.2.3 Running apxchpwd.sql

Run the `apxchpwd.sql` script to create and update your Instance Administrator account.

To create or update your Instance Administrator account:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxchpwd.sql`. For example:

```
@apxchpwd.sql
```

Follow the on-screen instructions. You will be prompted provide a username, password, and email address. If the account username does not exist, it will be created for you.

See Also:

*SQL*Plus User's Guide and Reference* for more information about SQL*Plus

4.4.3 Restarting Processes

Restart the processes that you stopped before you began the installation.

After you install Oracle Application Express, you must restart the processes that you stopped before you began the installation.

4.4.4 Configuring the APEX_PUBLIC_USER Account

It is important to correctly configure the `APEX_PUBLIC_USER` account to enable proper operation of Oracle Application Express.

- [About the APEX_PUBLIC_USER Account](#)
The `APEX_PUBLIC_USER` account is created with a random password in a new installation of Oracle Application Express.
- [Unlocking the APEX_PUBLIC_USER Account](#)
Unlock the `APEX_PUBLIC_USER` account by running a SQL statement.
- [Changing the Password for the APEX_PUBLIC_USER Account](#)
Change the password for the `APEX_PUBLIC_USER` account by running a SQL statement.
- [About Password Expiration in Oracle Database 11g and Later](#)
You can set `PASSWORD_LIFE_TIME` parameter to unlimited by altering `APEX_PUBLIC_USER` to prevent password expiration. To do this create another profile in which the `PASSWORD_LIFE_TIME` parameter is set to unlimited and alter the `APEX_PUBLIC_USER` account and assign it to the new profile.

4.4.4.1 About the APEX_PUBLIC_USER Account

The `APEX_PUBLIC_USER` account is created with a random password in a new installation of Oracle Application Express.

You must change the password for this account before configuring the database access descriptor (DAD) in a new installation.

4.4.4.2 Unlocking the APEX_PUBLIC_USER Account

Unlock the APEX_PUBLIC_USER account by running a SQL statement.

**Tip:**

If you are upgrading from a prior release of Oracle Application Express, this step is unnecessary.

To unlock the APEX_PUBLIC_USER account:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. If Oracle Application Express is installed in the CDB, ensure you connect to CDB\$ROOT. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following statement:

```
ALTER USER APEX_PUBLIC_USER ACCOUNT UNLOCK
```

4.4.4.3 Changing the Password for the APEX_PUBLIC_USER Account

Change the password for the APEX_PUBLIC_USER account by running a SQL statement.

**Tip:**

If you are upgrading from a prior release of Oracle Application Express, this step is unnecessary.

To change the password for the APEX_PUBLIC_USER account:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. If Oracle Application Express is installed in the CDB, ensure you connect to CDB\$ROOT. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following statement:

```
ALTER USER APEX_PUBLIC_USER IDENTIFIED BY new_password
```

Where `new_password` is the new password you are setting for `APEX_PUBLIC_USER`. You will use this password when creating the DAD in the sections that follow.

4.4.4.4 About Password Expiration in Oracle Database 11g and Later

You can set `PASSWORD_LIFE_TIME` parameter to unlimited by altering `APEX_PUBLIC_USER` to prevent password expiration. To do this create another profile in which the `PASSWORD_LIFE_TIME` parameter is set to unlimited and alter the `APEX_PUBLIC_USER` account and assign it to the new profile.

In the default profile in Oracle Database 11g or later, the parameter `PASSWORD_LIFE_TIME` is set to 180. If you are using Oracle Database 11g or later with Oracle Application Express, this causes the password for `APEX_PUBLIC_USER` to expire in 180 days. As a result, your Oracle Application Express instance will become unusable until you change the password.

 **See Also:**

Oracle Database Security Guide for information on creating profiles and assigning them to database users

4.4.5 Configuring RESTful Services

In a new installation or upgrade of Oracle Application Express, you must run the configuration script `apex_rest_config.sql` to configure RESTful Services.

Once configured, the instance administrator can control the availability of the feature. If the instance administrator has disabled RESTful Services for this Application Express instance, RESTful Services are not available for this instance and the RESTful Services icon does not display.

To configure RESTful Services in Oracle Application Express:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apex_rest_config.sql`. For example:

```
@apex_rest_config.sql
```

When Prompted, enter a password for the `APEX_LISTENER` and `APEX_REST_PUBLIC_USER` accounts.

When configuring RESTful Services in Oracle Application Express, it creates two new database accounts.

- `APEX_LISTENER` - The account used to query RESTful Services definitions stored in Oracle Application Express.
- `APEX_REST_PUBLIC_USER` - The account used when calling RESTful Services definitions stored in Oracle Application Express.



See Also:

"Enabling RESTful Services for an Instance" in *Oracle Application Express Administration Guide*

4.5 Downloading and Installing Oracle REST Data Services

Learn about downloading and installing Oracle REST Data Services.

- [Downloading Oracle REST Data Services](#)
Learn how to download Oracle REST Data Services.
- [About Configuring Oracle REST Data Services Behind a Reverse Proxy or Load Balancer](#)
When Oracle Application Express is running behind a reverse proxy or load balancer, it is important to communicate the original HTTP hostname and protocol as seen by the user's browser to the Application Express engine.
- [Web Server HTTP POST Request Limits](#)
Learn about Web Server HTTP POST request limits.
- [Disabling Oracle XML DB Protocol Server](#)
If you were previously using the embedded PL/SQL gateway and now want to use Oracle REST Data Services, you need to disable the Oracle XML DB Protocol Server.



See Also:

Oracle REST Data Services Installation, Configuration, and Development Guide

4.5.1 Downloading Oracle REST Data Services

Learn how to download Oracle REST Data Services.

Tip:

By default, the context root for accessing Oracle Application Express through Oracle REST Data Services is `/ords`. If you wish to have a context root of `/apex` for accessing Oracle Application Express, rename the `ords.war` file to `apex.war` before installing Oracle REST Data Services. See "Installing Oracle REST Data Services" in *Oracle REST Data Services Installation, Configuration, and Development Guide*.

To download Oracle REST Data Services:

1. Download the latest release of [Oracle REST Data services](#).
2. Unzip the downloaded zip file into a directory (or folder) of your choice:
 - UNIX and Linux: `unzip ords.version.number.zip`
 - Windows: Double-click the file `ords.version.number.zip` in Windows Explorer
3. Copy the images directory, `apex/images`, from the Oracle Application Express software ZIP to a location on the file system where Oracle REST Data Services is installed.
4. See *Oracle REST Data Services Installation, Configuration, and Development Guide* and follow and complete all installation and configuration steps.
5. For Oracle Database 12c or later multitenant architecture, ensure that you configure the connection using the service name of the specific pluggable database (PDB) you want to access. Do not use the service name of the `CDB$ROOT` unless you are configuring Oracle REST Data Services to address PDBs through the URL. See *Oracle REST Data Services Installation, Configuration, and Development Guide* for more information.

4.5.2 About Configuring Oracle REST Data Services Behind a Reverse Proxy or Load Balancer

When Oracle Application Express is running behind a reverse proxy or load balancer, it is important to communicate the original HTTP hostname and protocol as seen by the user's browser to the Application Express engine.

The Application Express engine uses this information to generate valid URLs in HTML responses and HTTP redirects that the user's browser can successfully follow. The exact configuration steps depend on your Java EE application server. For example, for Oracle WebLogic Server, this is accomplished using Oracle WebLogic Server Proxy Plug-Ins. To learn more, see your Java EE application server documentation.

4.5.3 Web Server HTTP POST Request Limits

Learn about Web Server HTTP POST request limits.

When running Oracle REST Data Services in standalone mode or within a Tomcat Java Container, size limits are being imposed on POST requests which are **not** file uploads. Application Express users will encounter these limits when uploading data in SQL Workshop using Copy and Paste or when using Copy and Paste while building an Application from Spreadsheet.

- When running Oracle REST Data Services in **Standalone Mode**, the default limit is 10MB for ORDS 3.0.9 and above. It is recommended to increase the limit as follows:

Set the Java System property

`org.eclipse.jetty.server.Request.maxFormContentSize` to a higher value in bytes. You can set this property upon startup of Oracle REST Data Services. For example: `java -`

```
Dorg.eclipse.jetty.server.Request.maxFormContentSize=3000000 -jar  
ords.war
```

- When running on Apache Tomcat, the default limit is 2 megabytes. Adjust Apache Tomcat's `maxPostSize` parameter to change that limit.

See Also:

["http://tomcat.apache.org/"](http://tomcat.apache.org/) for more information.

4.5.4 Disabling Oracle XML DB Protocol Server

If you were previously using the embedded PL/SQL gateway and now want to use Oracle REST Data Services, you need to disable the Oracle XML DB Protocol Server.

To disable Oracle XML DB Protocol Server:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

2. Run the following command:

```
EXEC DBMS_XDB.SETHTTPPORT(0);
```

4.6 Configuring Oracle REST Data Services

Configuring Oracle REST Data Services requires that you copy the images directory, run a configuration to configure RESTful Services, configure static files support, and secure Oracle REST data service.

How to configure RESTful Services in a new installation is discussed in the following section.

- [Copying the Images Directory](#)
Whether you are loading a new installation or upgrading from a previous release, you must copy the images directory from the top level of the `apex\images` directory, for example `C:\TEMP`, to the location used by your Oracle REST Data Services installation.
- [Validating the Oracle REST Data Services Installation](#)
In a new installation or upgrade of Oracle Application Express, you must validate the Oracle REST Data Services installation.
- [Configuring Static File Support](#)
For configuring static files, you must run `apex_rest_config.sql` after a new installation or upgrade of Oracle Application Express .
- [Securing Oracle REST Data Service](#)
In a configuration for Oracle Application Express, Oracle recommends setting the parameter `security.requestValidationFunction` to `wwv_flow_epg_include_modules.authorize`.

4.6.1 Copying the Images Directory

Whether you are loading a new installation or upgrading from a previous release, you must copy the images directory from the top level of the `apex\images` directory, for example `C:\TEMP`, to the location used by your Oracle REST Data Services installation.

During an upgrade, you overwrite your existing images directory. Before you begin the upgrade, to ensure that you can revert to the previous version, Oracle recommends that you create a copy of your existing images directory for Oracle Application Express, indicating the release number of the images (for example, `images_5_1`).

See Also:

Oracle REST Data Services Installation, Configuration, and Development Guide

4.6.2 Validating the Oracle REST Data Services Installation

In a new installation or upgrade of Oracle Application Express, you must validate the Oracle REST Data Services installation.

For validating the Oracle REST Data Services installation in a new installation or upgrade of Oracle Application Express, run the following :

```
java -jar ords.war validate [--database <dbname>]
```

**See Also:**

Oracle REST Data Services Installation, Configuration, and Development Guide

4.6.3 Configuring Static File Support

For configuring static files, you must run `apex_rest_config.sql` after a new installation or upgrade of Oracle Application Express .

Oracle Application Express enables application developers to include static files with their applications. Static files can be associated with a workspace, an application, a plug-in, or an application theme. When using Oracle REST Data Services as your web listener, static files are served using RESTful service module built into Oracle Application Express. Therefore, you must run `apex_rest_config.sql` after a new installation or upgrade of Oracle Application Express.

**See Also:**

["Configuring Oracle REST Data Services"](#)

4.6.4 Securing Oracle REST Data Service

In a configuration for Oracle Application Express, Oracle recommends setting the parameter `security.requestValidationFunction` to `wwv_flow_epg_include_modules.authorize`.

Set parameter `security.requestValidationFunction` to `wwv_flow_epg_include_modules.authorize` activates the white list of callable procedures which ships with Oracle Application Express and prohibits calls to other procedures.

**See Also:**

Oracle Application Express App Builder User's Guide

4.7 Enabling Network Services in Oracle Database 11g or Later

You must enable network services in Oracle Database 11g or later versions to send outbound mail, use Web services, or use PDF report printing in Oracle Application Express.

- [When and Why Network Services Must be Enabled](#)
Enabling network services enables support for sending outbound mail in Oracle Application Express, use of Web services in Oracle Application Express, and PDF report printing.
- [Granting Connect Privileges Prior to Oracle Database 12c](#)
Demonstrates how to grant connect privileges to any host for the APEX_190200 database user.
- [Granting Connect Privileges in Oracle Database 12c or Later](#)
Procedures CREATE_ACL, ASSIGN_ACL, ADD_PRIVILEGE and CHECK_PRIVILEGE in DBMS_NETWORK_ACL_ADMIN are deprecated in Oracle Database 12c. Oracle recommends to use APPEND_HOST_ACE.
- [Troubleshooting an Invalid ACL Error](#)
Learn how to identify any invalid ACL error by running the query.

4.7.1 When and Why Network Services Must be Enabled

Enabling network services enables support for sending outbound mail in Oracle Application Express, use of Web services in Oracle Application Express, and PDF report printing.

By default, the ability to interact with network services is disabled in Oracle Database 11g Release 2 or later. Therefore, if you are running Oracle Application Express with Oracle Database 11g Release 2 or later, you must use the new DBMS_NETWORK_ACL_ADMIN package to grant connect privileges to any host for the APEX_190200 database user. Failing to grant these privileges results in issues with:

- Sending outbound mail in Oracle Application Express.
Users can call methods from the APEX_MAIL package, but issues arise when sending outbound email.
- Using Web services in Oracle Application Express.
- PDF report printing.

 **Note:**

When upgrading Application Express on a database 12c or newer, based on the configuration of the old Application Express version the upgrade automatically configures Network Services.

 **Tip:**

To run the examples described in this section, the compatible initialization parameter of the database must be set to at least 11.1.0.0.0. By default an 11g or 12c database will already have the parameter set properly, but a database upgraded to 11g or 12c from a prior version may not. For information about changing database initialization parameters, see "Creating and Configuring an Oracle Database" in *Oracle Database Administrator's Guide*.

4.7.2 Granting Connect Privileges Prior to Oracle Database 12c

Demonstrates how to grant connect privileges to any host for the `APEX_190200` database user.

The following example demonstrates how to grant connect privileges to any host for the `APEX_190200` database user. This example assumes you connected to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role.

```
DECLARE
  ACL_PATH VARCHAR2(4000);
BEGIN
  -- Look for the ACL currently assigned to '*' and give APEX_190200
  -- the "connect" privilege if APEX_190200 does not have the privilege
  -- yet.

  SELECT ACL INTO ACL_PATH FROM DBA_NETWORK_ACLS
    WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

  IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE(ACL_PATH, 'APEX_190200',
    'connect') IS NULL THEN
    DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE(ACL_PATH,
    'APEX_190200', TRUE, 'connect');
  END IF;

EXCEPTION
  -- When no ACL has been assigned to '*'.
  WHEN NO_DATA_FOUND THEN
    DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('power_users.xml',
    'ACL that lets power users to connect to everywhere',
    'APEX_190200', TRUE, 'connect');
    DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('power_users.xml', '*');
END;
/
COMMIT;
```

The following example demonstrates how to provide less privileged access to local network resources. This example enables access to servers on the local host only, such as email and report servers.

```

DECLARE
  ACL_PATH VARCHAR2(4000);
BEGIN
  -- Look for the ACL currently assigned to 'localhost' and give
  APEX_190200
  -- the "connect" privilege if APEX_190200 does not have the privilege
  yet.
  SELECT ACL INTO ACL_PATH FROM DBA_NETWORK_ACLS
    WHERE HOST = 'localhost' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

  IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE(ACL_PATH, 'APEX_190200',
    'connect') IS NULL THEN
    DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE(ACL_PATH,
    'APEX_190200', TRUE, 'connect');
  END IF;

EXCEPTION
  -- When no ACL has been assigned to 'localhost'.
  WHEN NO_DATA_FOUND THEN
    DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('local-access-users.xml',
    'ACL that lets users to connect to localhost',
    'APEX_190200', TRUE, 'connect');
    DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('local-access-users.xml','localhost');
END;
/
COMMIT;

```

4.7.3 Granting Connect Privileges in Oracle Database 12c or Later

Procedures `CREATE_ACL`, `ASSIGN_ACL`, `ADD_PRIVILEGE` and `CHECK_PRIVILEGE` in `DBMS_NETWORK_ACL_ADMIN` are deprecated in Oracle Database 12c. Oracle recommends to use `APPEND_HOST_ACE`.

The following example demonstrates how to grant connect privileges to any host for the `APEX_190200` database user. This example assumes you connected to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role.

```

BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host => '*',
    ace => xs$ace_type(privilege_list => xs$name_list('connect'),
    principal_name => 'APEX_190200',
    principal_type => xs_acl.ptype_db));
END;
/

```

The following example demonstrates how to provide less privileged access to local network resources. This example enables access to servers on the local host only, such as email and report servers.

```
BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host => 'localhost',
    ace => xs$ace_type(privilege_list => xs$name_list('connect'),
                      principal_name => 'APEX_190200',
                      principal_type => xs_acl.ptype_db));
END;
/
```

4.7.4 Troubleshooting an Invalid ACL Error

Learn how to identify any invalid ACL error by running the query.

If you receive an ORA-44416: Invalid ACL error after running the previous script, use the following query to identify the invalid ACL:

```
REM Show the dangling references to dropped users in the ACL that is
assigned
REM to '*'.

SELECT ACL, PRINCIPAL
  FROM DBA_NETWORK_ACLS NACL, XDS_ACE ACE
 WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL AND
        NACL.ACLID = ACE.ACLID AND
        NOT EXISTS (SELECT NULL FROM ALL_USERS WHERE USERNAME = PRINCIPAL);
```

Next, run the following code to fix the ACL:

```
DECLARE
  ACL_ID  RAW(16);
  CNT     NUMBER;
BEGIN
  -- Look for the object ID of the ACL currently assigned to '*'
  SELECT ACLID INTO ACL_ID FROM DBA_NETWORK_ACLS
     WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

  -- If just some users referenced in the ACL are invalid, remove just
those
  -- users in the ACL. Otherwise, drop the ACL completely.
  SELECT COUNT(PRINCIPAL) INTO CNT FROM XDS_ACE
     WHERE ACLID = ACL_ID AND
           EXISTS (SELECT NULL FROM ALL_USERS WHERE USERNAME = PRINCIPAL);

  IF (CNT > 0) THEN

    FOR R IN (SELECT PRINCIPAL FROM XDS_ACE
              WHERE ACLID = ACL_ID AND
                    NOT EXISTS (SELECT NULL FROM ALL_USERS
                               WHERE USERNAME = PRINCIPAL)) LOOP
```

```
UPDATE XDB.XDB$ACL
  SET OBJECT_VALUE =
    DELETEXML(OBJECT_VALUE,
      '/ACL/ACE[PRINCIPAL="' || R.PRINCIPAL || '"]')
  WHERE OBJECT_ID = ACL_ID;
END LOOP;

ELSE
  DELETE FROM XDB.XDB$ACL WHERE OBJECT_ID = ACL_ID;
END IF;

END;
/

REM commit the changes.

COMMIT;
```

Once the ACL has been fixed, you must run the first script in this section to apply the ACL to the APEX_190200 user.

4.8 Performing Security Tasks

Oracle recommends configuring and using Secure Sockets Layer (SSL) to ensure that passwords and other sensitive data are not transmitted in clear text in HTTP requests.

Without the use of SSL, passwords could potentially be exposed, compromising security.

SSL is an industry standard protocol that uses RSA public key cryptography in conjunction with symmetric key cryptography to provide authentication, encryption, and data integrity.

See Also:

"Configuring HTTP Protocol Attributes" in *Oracle Application Express Administration Guide*

4.9 Controlling the Number of Concurrent Jobs

Learn about specifying the number of concurrently running jobs.

- [About Managing the Number of Concurrent Jobs](#)
Learn about managing maximum number of concurrently running jobs.
- [Viewing the Number of JOB_QUEUE_PROCESSES](#)
You can view number of JOB_QUEUE_PROCESSES in three ways.
- [Changing the Number of JOB_QUEUE_PROCESSES](#)
You can change the number of JOB_QUEUE_PROCESSES by running a SQL statement in SQL*Plus.

4.9.1 About Managing the Number of Concurrent Jobs

Learn about managing maximum number of concurrently running jobs.

`JOB_QUEUE_PROCESSES` determine the maximum number of concurrently running jobs. In Oracle Application Express transactional support and SQL scripts require jobs. If `JOB_QUEUE_PROCESSES` is not enabled and working properly, you cannot successfully execute a script.

4.9.2 Viewing the Number of `JOB_QUEUE_PROCESSES`

You can view number of `JOB_QUEUE_PROCESSES` in three ways.

- [Viewing `JOB_QUEUE_PROCESSES` in the Installation Log File](#)
View `JOB_QUEUE_PROCESSES` in the installation log files.
- [Viewing `JOB_QUEUE_PROCESSES` in Oracle Application Express](#)
View the number of `JOB_QUEUE_PROCESSES` on the About Application Express page.
- [Viewing `JOB_QUEUE_PROCESSES` from SQL*Plus](#)
View the number of `JOB_QUEUE_PROCESSES` from SQL*Plus.

4.9.2.1 Viewing `JOB_QUEUE_PROCESSES` in the Installation Log File

View `JOB_QUEUE_PROCESSES` in the installation log files.



See Also:

["Reviewing a Log of an Installation Session"](#)

4.9.2.2 Viewing `JOB_QUEUE_PROCESSES` in Oracle Application Express

View the number of `JOB_QUEUE_PROCESSES` on the About Application Express page.

To view the About Application Express page:

1. Sign in to Oracle Application Express.
2. Locate the Help menu at the top of the page.
3. From the Help menu, select **About**.
The About Application Express page appears.
4. Scroll down and find `JOB_QUEUE_PROCESSES` at the bottom of the page.



See Also:

["Signing into a Workspace Using Oracle REST Data Services"](#)

4.9.2.3 Viewing JOB_QUEUE_PROCESSES from SQL*Plus

View the number of `JOB_QUEUE_PROCESSES` from SQL*Plus.

Run the following SQL statement:

```
SELECT VALUE FROM v$parameter WHERE NAME = 'job_queue_processes'
```

4.9.3 Changing the Number of JOB_QUEUE_PROCESSES

You can change the number of `JOB_QUEUE_PROCESSES` by running a SQL statement in SQL*Plus.

To update the number of `JOB_QUEUE_PROCESSES`:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. In SQL*Plus run the following SQL statement:

```
ALTER SYSTEM SET JOB_QUEUE_PROCESSES = <number>
```

For example, running the statement `ALTER SYSTEM SET JOB_QUEUE_PROCESSES = 20` sets `JOB_QUEUE_PROCESSES` to 20.

4.10 About Running Oracle Application Express in Other Languages

You can install a single instance of Oracle Application Express with one or more of translated versions.

The Oracle Application Express interface is translated into German, Spanish, French, Italian, Japanese, Korean, Brazilian Portuguese, Simplified Chinese, and Traditional Chinese. A single instance of Oracle Application Express can be installed with one or more of these translated versions. Developers can choose to run the Application Express development environment in any of the installed languages by simply selecting the language from the App Builder log in screen or home page.

In order to install other languages you must use the `apex_19.2.zip` file which contains the extra files referenced below. If you previously downloaded `apex_19.2_en.zip`, then

you do not need to re-install Oracle Application Express. Simply download `apex_19.2.zip` and unzip the file into the same directory where you unzipped `apex_19.2_en.zip`.

The translated version of Oracle Application Express should be loaded into a database that has a character set that supports the specific language. If you attempt to install a translated version of Oracle Application Express into a database that does not support the character encoding of the language, the installation may fail or the translated Oracle Application Express instance may appear corrupt when run. The database character set `AL32UTF8` supports all the translated versions of Oracle Application Express.

You can manually install translated versions of Oracle Application Express using SQL*Plus. The installation files are encoded in `AL32UTF8`.



Note:

Regardless of the target database character set, to install a translated version of Oracle Application Express, you must set the character set value of the `NLS_LANG` environment variable to `AL32UTF8` before starting SQL*Plus.

The following examples illustrate valid `NLS_LANG` settings for loading Oracle Application Express translations:

```
American_America.AL32UTF8  
Japanese_Japan.AL32UTF8
```

4.11 Installing Translated Versions of Oracle Application Express

Learn about installing translated versions of Oracle Application Express.

- [About Installing Translated Versions of Oracle Application Express](#)
You must run `load_lang.sql` script to run a translated version of Oracle Application Express.
- [Installing a Translated Version of Oracle Application Express](#)
Learn how to install a translated version of Oracle Application Express.

4.11.1 About Installing Translated Versions of Oracle Application Express

You must run `load_lang.sql` script to run a translated version of Oracle Application Express.

Whether you are installing for the first time or upgrading from a previous release, you must run the `load_lang.sql` script to run a translated version of Oracle Application Express.

The installation scripts are located in subdirectories identified by a language code in the unzipped distribution `apex/builder`. For example, the German version is located in `apex/builder/de` and the Japanese version is located in `apex/builder/ja`. Within each directory, there is a language loading script identified by the language code (for example, `load_de.sql` or `load_ja.sql`).

4.11.2 Installing a Translated Version of Oracle Application Express

Learn how to install a translated version of Oracle Application Express.

To install a translated version of Oracle Application Express:

1. Set the `NLS_LANG` environment variable, making sure that the character set is `AL32UTF8`. For example:

- Bourne or Korn shell:

```
NLS_LANG=American_America.AL32UTF8
export NLS_LANG
```

- C shell:

```
setenv NLS_LANG American_America.AL32UTF8
```

- For Windows based systems:

```
set NLS_LANG=American_America.AL32UTF8
```

2. Navigate to the directory under `apex/builder` based on the language you need to install. For example for German, navigate to `apex/builder/de`. Start `SQL*Plus` and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following statement:

```
ALTER SESSION SET CURRENT_SCHEMA = APEX_190200;
```

4. Execute the appropriate language specific script. For example:

```
@load_lang.sql
```

Where `lang` is the specific language (for example, `load_de.sql` for German or `load_ja.sql` for Japanese).

4.12 Creating a Workspace and Adding Oracle Application Express Users

Before you can develop or install applications, you must create a workspace, add Oracle Application Express users, and sign in to your workspace.

- [About Workspaces and Users](#)
A workspace enables multiple users to work within the same Oracle Application Express installation while keeping their objects, data, and applications private.
- [Accessing Administration Services Using Oracle REST Data Services](#)
Sign in to Oracle Application Express Administration Services using the Instance administrator account and password created or reset during the installation process.
- [Creating a Workspace Manually](#)
Sign in to Oracle Application Express Administration Services to create workspace manually.
- [Creating Oracle Application Express Users](#)
Create new users in Oracle Application Express Administration Services..
- [Signing into a Workspace Using Oracle REST Data Services](#)
Learn how to sign in to a workspace using the Oracle REST Data Services.

4.12.1 About Workspaces and Users

A workspace enables multiple users to work within the same Oracle Application Express installation while keeping their objects, data, and applications private.

You access the Oracle Application Express home page by logging in to a workspace using a JavaScript enabled Web browser.

Each workspace has a unique ID and name. An instance administrator can create a workspace manually within Oracle Application Express Administration Services or have users submit requests. Oracle Application Express Administration Services is a separate application for managing an entire Oracle Application Express instance.



See Also:

- ["Browser Requirements"](#)
- ["Creating Workspaces" in *Oracle Application Express Administration Guide*](#)
- ["Managing Workspace and Change Requests" in *Oracle Application Express Administration Guide*](#)

4.12.2 Accessing Administration Services Using Oracle REST Data Services

Sign in to Oracle Application Express Administration Services using the Instance administrator account and password created or reset during the installation process.

To manually create a workspace and user accounts, you sign in to a separate application for managing an entire Oracle Application Express instance called Oracle Application Express Administration Services.

To access Oracle Application Express Administration Services:

1. In a Web browser, navigate to the Oracle Application Express Administration Services application.

 **Tip:**

By default, the context root for accessing Oracle Application Express through Oracle REST Data Services is `/ords`. If you wish to have a context root of `/apex` for accessing Oracle Application Express, rename the `ords.war` file to `apex.war` before installing Oracle REST Data Services. See "Installing Oracle REST Data Services" in *Oracle REST Data Services Installation, Configuration, and Development Guide*.

Because your setup uses Oracle REST Data Services go to:

```
http://hostname:port/apex/apex_admin
```

Where:

`hostname` is the name of the system where Oracle REST Data Services is installed.

`port` is the port number assigned when configuring Oracle REST Data Services. In a default installation, this number is 8080. To learn more, see *Oracle REST Data Services Installation, Configuration, and Development Guide*.

`apex` is the service name defined when configuring Oracle REST Data Services.

2. On the Sign In page:
 - Username - Enter the Oracle Application Express administrator account username you specified in Creating or Updating Your Instance Administration Account.
 - Password - Enter the Oracle Application Express administrator account password.
 - Click **Sign In to Administration**.

Note that, depending on your setup, you might be required to change your password when you log in for the first time.

 **See Also:**

- ["Creating or Updating Your Instance Administration Account"](#)
- "Oracle Application Express Administration Services" in *Oracle Application Express Administration Guide*

4.12.3 Creating a Workspace Manually

Sign in to Oracle Application Express Administration Services to create workspace manually.

To manually create a workspace you sign in to a separate application called Oracle Application Express Administration Services. You sign in to Oracle Application Express Administration Services using the `ADMIN` account and password created or reset during the installation process.

To create an Oracle Application Express workspace manually:

1. Access Oracle Application Express Administration Services.
Oracle Application Express Administration Services appears. Next, create a workspace.
2. Click **Manage Workspaces**.
3. Under Workspace Actions, click **Create Workspace**.
The Create Workspace Wizard appears.
4. For Identify Workspace, enter the following:
 - a. Workspace Name - Enter a unique workspace name.
 - b. Workspace ID - Leave Workspace ID blank to have the new Workspace ID automatically generated. A Workspace ID must be a positive integer greater than 100000.
 - c. Workspace Description - Enter a workspace description.
 - d. Click **Next**.
5. For Identify Schema, specify whether you are re-using an existing schema or creating a new one.
If you are using an existing schema:
 - a. For Re-use existing schema, select **Yes**.
 - b. Select a schema from the list.
 - c. Click **Next**.
If you are creating a new schema:
 - a. For Re-use existing schema, select **No**.
 - b. Enter a schema name and password.
 - c. Specify a space quota.
 - d. Click **Next**.

6. For Identify Administrator, enter the Workspace administrator information and click **Next**.
7. Confirm your selections and click **Create Workspace**.

 **See Also:**

- ["Accessing Administration Services Using Oracle REST Data Services"](#)
- "Oracle Application Express Administration Services" in *Oracle Application Express Administration Guide*

4.12.4 Creating Oracle Application Express Users

Create new users in Oracle Application Express Administration Services..

Create new users by signing into Oracle Application Express Administration Services using your administrator password.

To create an Oracle Application Express user account:

1. Sign into Oracle Application Express Administration Services.
Oracle Application Express Administration Services appears.
2. Click **Manage Workspaces**.
3. Under Workspace Actions, click **Manage Developers and Users**.
4. Click **Create User**.
5. Under User Attributes, enter the appropriate information. Fields marked with an asterisk are required.

 **Tip:**

Most attributes in Oracle Application Express include field-level Help. Attributes with field-level Help, have light gray icon that resembles a question mark (?). To view field-level Help, click the Help icon.

6. Under Account Privileges:
 - a. **Workspace** - Select a workspace from the list.
 - b. **Default Schemas** - Specify the default schema used for data browsing, application creation, and SQL script execution.

When using workspaces that have more than one schema available, this schema is the default. This setting does not control security, only the user's preference.
 - c. **User is an administrator** - Specify if this user should have workspace administrator privileges.

Administrators are given access to all components. Additionally, they can manage user accounts, groups, and development services. Components may not be available if they are switched off by Instance Administrators.

- d. **User is a developer** - Specify if this user should have developer privileges.
Developers must have access to either App Builder, SQL Workshop, or both. These components may not be available if they are switched off by the Instance Administrator.
 - e. **App Builder Access** - Determines whether a developer has access to the App Builder.
 - f. **SQL Workshop Access** - Determines whether a developer has access to the SQL Workshop.
 - g. **Team Development Access** - Determines whether a developer has access to the Team Development.
 - h. **Set Account Availability** - Select **Locked** to prevent the account from being used. Select **Unlocked** to allow the account to be used.

If the user has exceeded the maximum log in failures allowed, specified in Workspace Preferences, then their account will be locked automatically.
7. Under Password:
 - **Password** - Enter a case sensitive password.
 - **Confirm Password** - Enter the password again.
 - **Require Change of Password On First Use** - Select **No** to allow the user to use the same password until it expires. Select **Yes** to require the user to change the password immediately when logging in the first time.
 8. Click **Create User** or **Create and Create Another**.



See Also:

["Accessing Administration Services Using Oracle REST Data Services"](#)

4.12.5 Signing into a Workspace Using Oracle REST Data Services

Learn how to sign in to a workspace using the Oracle REST Data Services.

Use your credentials (that is, your workspace name, user name and password) to sign in to the workspace.

To sign in to your workspace:

1. In a Web browser, navigate to the Oracle Application Express Sign In page.

Because your setup uses Oracle REST Data Services, go to:

```
http://hostname:port/apex/
```

Where:

- *hostname* is the name of the system where Oracle REST Data Services is installed.
- *port* is the port number assigned when configuring Oracle REST Data Services. In a default installation, this number is 8080. To learn more, see

Oracle REST Data Services Installation, Configuration, and Development Guide.

- *apex* is the service name defined when configuring Oracle REST Data Services.

The Sign In page appears.

2. On the Sign In page, enter:
 - Workspace - Enter the name of your workspace.
 - Username - Enter your user name.
 - Password - Enter your case-sensitive password.
3. Click **Sign In**.

Note that, depending on your setup, you might be required to change your password when you log in for the first time.

See Also:

- "Creating Workspaces" in *Oracle Application Express Administration Guide*
- "Managing Workspace and Change Requests" in *Oracle Application Express Administration Guide*

4.13 Performing Post Installation Tasks for Upgrade Installations

Once you have verified that your upgrade installation was successful and all upgraded applications function properly, you should remove schemas from prior Oracle Application Express installations.

- [About Removing Prior Oracle Application Express Installations](#)
Learn about removing schemas from a prior installation by verifying if a prior installation exists.
- [Verifying if a Prior Installation Exists](#)
Run the SQL query to verify if a prior Oracle Application Express installation exists.
- [Removing Schemas and SYS Objects from Prior Installations](#)
Start SQL*Plus and connect to database and execute a statement to remove schemas and SYS objects.
- [Removing Schemas from Prior Installations in a CDB](#)
Use `catcon.pl` to remove schemas of prior installations in a CDB.
- [Fixing Invalid ACL in Oracle Database 11g or Later](#)
Learn how to fix an invalid ACL if you are running Oracle Database 11g or later.

**See Also:**

["Upgrading from a Previous Oracle Application Express Release"](#)

4.13.1 About Removing Prior Oracle Application Express Installations

Learn about removing schemas from a prior installation by verifying if a prior installation exists.

The database users associated with schemas from prior installations are privileged users and should be removed when they are no longer necessary. Removing schemas from a prior installation is a two step process. First you verify if a prior installation exists and then you remove the schemas.

4.13.2 Verifying if a Prior Installation Exists

Run the SQL query to verify if a prior Oracle Application Express installation exists.

To verify if a prior installation exists:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following query:

```
SELECT username
FROM dba_users
WHERE ( username LIKE 'FLOWS\_____' ESCAPE '\ '
OR username LIKE 'APEX\_____' ESCAPE '\ ' )
AND username NOT IN ( SELECT schema
FROM dba_registry
WHERE comp_id = 'APEX' );
```

If the results contain entries in the form FLOWS_XXXXXX or APEX_XXXXXX where XXXXXX represents six numbers, those entries are candidates for removal.

4.13.3 Removing Schemas and SYS Objects from Prior Installations

Start SQL*Plus and connect to database and execute a statement to remove schemas and SYS objects.

To remove schemas and SYS objects from prior installations:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Execute statements similar to the following example:

```
DROP USER APEX_190100 CASCADE;
DROP PACKAGE SYS.WWV_DBMS_SQL_APEX_190100;
```

4.13.4 Removing Schemas from Prior Installations in a CDB

Use `catcon.pl` to remove schemas of prior installations in a CDB.

To remove schemas and SYS objects from prior installations, run commands using the following example:

```
$ORACLE_HOME/perl/bin/perl -I $ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/
admin/catcon.pl -b drop_apex190100 -- --x'drop user APEX_190100 cascade'
```

```
$ORACLE_HOME/perl/bin/perl -I $ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/
admin/catcon.pl -b drop_wwv_dbms_sql -- --x'drop package
SYS.WWV_DBMS_SQL_APEX_190100 cascade'
```

4.13.5 Fixing Invalid ACL in Oracle Database 11g or Later

Learn how to fix an invalid ACL if you are running Oracle Database 11g or later.

After following the instructions in "[About Removing Prior Oracle Application Express Installations](#)," you may need to fix an invalid ACL if you are running Oracle Database 11g or later and you enabled network services for the prior Oracle Application Express schema.

To fix an invalid ACL:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute a statement similar to the following:

```
EXEC DBMS_NETWORK_ACL_ADMIN.DELETE_PRIVILEGE('power_users.xml',
'APEX_040200');
```

4.14 About Performance Optimization Tasks

Learn about performance optimization.

Performance of web applications heavily depends on their size and how often a browser has to request static content like images, CSS and JavaScript files. To improve performance, most web servers support on-the-fly HTTP response compression and provide settings that enable you to configure on how long browsers can cache a file before requesting it again. The HTTP response compression is usually implemented using gzip encoding, while browser file caching is enabled by issuing Cache-Control HTTP response header.

Please see your web server documentation to learn how to enable response compression and browser file caching. For optimal performance of the Oracle Application Express development environment and Oracle Application Express applications, Oracle recommends enabling gzip compression of files in the virtual images directory (for example, `/i/`) and responses from the database access descriptor as well as allowing browsers to cache files from the virtual images directory for at least 12 hours.

4.15 Converting Between Runtime and Full Development Environments

Learn about converting between runtime and full development environments.

This section describes how to convert between runtime and full development environments.

- [About Runtime and Full Development Environments](#)
An Oracle Application Express runtime environment enables users to run a production application without supporting the ability to change or edit the application.
- [Converting a Runtime Environment to a Full Development Environment](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdvins.sql`.
- [Converting a Full Development Environment to a Runtime Environment](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrm.sql`.

4.15.1 About Runtime and Full Development Environments

An Oracle Application Express runtime environment enables users to run a production application without supporting the ability to change or edit the application.

A runtime environment includes only the packages necessary to run your applications, making it a more hardened environment. It does not provide a web interface for administration.

You administer an Oracle Application Express runtime environment using SQL*Plus or SQL Developer and the `APEX_INSTANCE_ADMIN` API.

See Also:

- ["About the Oracle Application Express Runtime Environment"](#) in *Oracle Application Express Administration Guide*
- ["Installing Exported Applications into a Runtime Environment"](#) in *Oracle Application Express Administration Guide*

4.15.2 Converting a Runtime Environment to a Full Development Environment

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdvins.sql`.

To convert an Oracle Application Express runtime environment to a full development environment:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxdvins.sql`. For example:

```
@apxdvins.sql
```

4. Follow the instructions in [Creating or Updating Your Instance Administration Account](#).



See Also:

- ["Creating or Updating Your Instance Administration Account"](#)
- *SQL*Plus User's Guide and Reference* for more information about SQL*Plus

4.15.3 Converting a Full Development Environment to a Runtime Environment

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrm.sql`.



Tip:

Before converting a full development environment to a runtime environment, export all Websheet applications. When you convert a development environment to runtime, existing Websheet data grids and reports will be deleted from all workspaces. Converting the environment back to a full development environment will not restore these Websheet data grids or reports.

To convert an Oracle Application Express full development environment to a runtime environment:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

3. Run `apxdevrm.sql`. For example:

```
@apxdevrm.sql
```

 **See Also:**

*SQL*Plus User's Guide and Reference* for more information about SQL*Plus

A

Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway

Install Oracle Application Express by downloading a ZIP file the Oracle Application Express download page and then configuring Embedded PL/SQL Gateway.

These instructions apply to both new and upgrade installations.

- [Oracle Database Requirements](#)
Oracle Application Express release 19.2 requires an Oracle Database release 11.2.0.4 or later, including Enterprise Edition and Express Edition (Oracle Database 18c XE).
- [Performing Pre-installation Tasks for Oracle Application Express](#)
Review and perform pre-installation tasks before installing Oracle Application Express.
- [Downloading and Installing Oracle Application Express](#)
Learn how to download and install Oracle Application Express.
- [Configuring the Embedded PL/SQL Gateway](#)
To configure the embedded PL/SQL gateway, you run a configuration script and unlock the `ANONYMOUS` account.
- [Enabling Network Services in Oracle Database 11g or Later](#)
You must enable network services in Oracle Database 11g or later versions to send outbound mail, use Web services, or use PDF report printing in Oracle Application Express.
- [Performing Security Tasks](#)
Oracle recommends configuring and using Secure Sockets Layer (SSL) to ensure that passwords and other sensitive data are not transmitted in clear text in HTTP requests.
- [Controlling the Number of Concurrent Jobs](#)
Learn about specifying the number of concurrently running jobs.
- [About Running Oracle Application Express in Other Languages](#)
You can install a single instance of Oracle Application Express with one or more of translated versions.
- [Installing Translated Versions of Oracle Application Express](#)
Learn about installing translated versions of Oracle Application Express.
- [About Configuring the SHARED_SERVERS Parameter](#)
Achieve acceptable performance when using the embedded PL/SQL gateway by setting the `SHARED_SERVERS` database initialization parameter to a reasonable value.

- [Creating a Workspace and Adding Oracle Application Express Users](#)
Before you can develop or install applications, you must create a workspace, add Oracle Application Express users and sign in to your workspace.
- [Performing Post Installation Tasks for Upgrade Installations](#)
Remove schemas from prior Oracle Application Express installations.
- [About Performance Optimization Tasks](#)
Learn about performance optimization.
- [Converting Between Runtime and Full Development Environments](#)
Convert between runtime and full development environments.

 **See Also:**

- ["Web Listener Requirements"](#)
- ["Security Considerations When Using the Embedded PL/SQL Gateway"](#)

A.1 Oracle Database Requirements

Oracle Application Express release 19.2 requires an Oracle Database release 11.2.0.4 or later, including Enterprise Edition and Express Edition (Oracle Database 18c XE).

If you are upgrading an Oracle Database version 12.1 CDB, you must download from My Oracle Support the one off patch for bug 20618595. Search for **20618595** on the Patches tab.

A.2 Performing Pre-installation Tasks for Oracle Application Express

Review and perform pre-installation tasks before installing Oracle Application Express.

Before installing Oracle Application Express, Oracle recommends that you complete the following steps:

1. Review and satisfy all Oracle Application Express installation requirements.
2. If you are actively using Oracle Application Express and upgrading the current installation then shut down with normal or immediate priority the Oracle Database instances where you plan to install Oracle Application Express. On Oracle Real Application Clusters (Oracle RAC) systems, shut down all instances on each node.

An alternative to shutting down the database is to prevent all users from accessing Oracle Application Express when upgrading your installation from a previous release of Oracle Application Express. Oracle only recommends this option in high availability production environments where planned outages are not available. For all other scenarios, the database should be shut down.

To disable access to Oracle Application Express when an existing installation is using the Embedded PL/SQL Gateway, disable the Oracle XML DB Protocol

Server by setting the HTTP port to 0. Start SQL*Plus, connect as SYS to the database where Oracle Application Express is installed, and run:

```
EXEC DBMS_XDB.SETHTTPPORT(0);
```

Once you have prevented access from Oracle Application Express users, you should log in to SQL*Plus as SYS, connecting to the database where Oracle Application Express is installed, and query V\$SESSION to ensure there are no long running sessions which would interfere with the upgrade process.

3. Back up the Oracle Database installation.

Oracle recommends that you create a backup of the current Oracle Database installation before you install Oracle Application Express. You can use Oracle Database Recovery Manager, which is included in the Oracle Database installation, to perform the backup.

4. Start the Oracle Database instance that contains the target database.

After backing up the system, you must start the Oracle instance that contains the target Oracle Database. Do not start other processes such as a Web listener. However, if you are performing a remote installation, make sure the Web listener for the remote database has started.

 **Note:**

If you are connecting to a remote database, then start the Web listener.

 **See Also:**

- "[Oracle Application Express Installation Requirements](#)"
- "[Oracle Database Backup and Recovery User's Guide](#)"

A.3 Downloading and Installing Oracle Application Express

Learn how to download and install Oracle Application Express.

This section describes how to download and install Oracle Application Express. How you install Oracle Application Express depends upon by the type of database you into which you are installing.

- [Installing Application Express](#)
Install Oracle Application Express by downloading a ZIP file from the Oracle Application Express download page.
- [Creating or Updating Your Instance Administration Account](#)
Learn how to create or update your Instance Administrator account.
- [Restart Processes](#)
Restart the processes that you stopped before you began the installation.

**See Also:**

"Utilizing the Multitenant Architecture in Oracle Database 12c or Later."

A.3.1 Installing Application Express

Install Oracle Application Express by downloading a ZIP file from the Oracle Application Express download page.

To install Oracle Application Express:

1. For installations where the development will be in English only, download the file `apex_19.2_en.zip` from the Oracle Application Express download page. If the development will include languages other than English, download `apex_19.2.zip` from the Oracle Application Express download page. See:

<http://www.oracle.com/technetwork/developer-tools/apex/downloads/index.html>

Note that the actual file name may differ if a more recent release has shipped since this document was published.

2. Unzip downloaded zip file:
 - If English only, unzip `apex_19.2_en.zip` as follows, preserving directory names:
 - UNIX and Linux: `$ unzip apex_19.2_en.zip`
 - Windows: Double click the file `apex_19.2_en.zip` in Windows Explorer
 - If multiple languages, unzip `apex_19.2.zip` as follows, preserving directory names:
 - UNIX and Linux: `$ unzip apex_19.2.zip`
 - Windows: Double click the file `apex_19.2.zip` in Windows Explorer

**Note:**

You should keep the directory tree where you unzip the files short and not under directories that contain spaces. For example, within Windows unzip to `C:\TEMP`.

3. Change your working directory to `apex`.
4. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:
 - On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

5. Disable any existing password complexity rules for the default profile. See “Configuring Password Protection” in *Oracle Database Security Guide*.
6. Select the appropriate installation option.

Full development environment provides complete access to the App Builder environment to develop applications. A **Runtime environment** enables users to run applications that cannot be modified.

Available installation options include:

- **Full development environment.** Run `apexins.sql` passing the following four arguments in the order shown:

```
@apexins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- `tablespace_apex` is the name of the tablespace for the Oracle Application Express application user.
- `tablespace_files` is the name of the tablespace for the Oracle Application Express files user.
- `tablespace_temp` is the name of the temporary tablespace or tablespace group.
- `images` is the virtual directory for Oracle Application Express images. For installations using EPG, `/i/` is the required value for the images argument. To support future Oracle Application Express upgrades, define the virtual image directory as `/i/`.

Example:

```
@apexins.sql SYSAUX SYSAUX TEMP /i/
```

 **Note:**

If you receive the following error, exit SQL*Plus and change your working directory to where you unzipped the installation file, for example `C:\TEMP` in Windows, before starting SQL*Plus:

```
SP2-0310: unable to open file "apexins.sql"
```

- **Runtime environment.** Run `apxrtins.sql` passing the following arguments in the order shown:

```
@apxrtins.sql tablespace_apex tablespace_files tablespace_temp
images
```

Where:

- *tablespace_apex* is the name of the tablespace for the Oracle Application Express application user.
- *tablespace_files* is the name of the tablespace for the Oracle Application Express files user.
- *tablespace_temp* is the name of the temporary tablespace or tablespace group.
- *images* is the virtual directory for Oracle Application Express images. To support future Oracle Application Express upgrades, define the virtual image directory as */i/*.

Example:

```
@apxrtins.sql SYSAUX SYSAUX TEMP /i/
```

When Application Express installs, it creates the following database accounts:

- APEX_190200 - This account owns the Application Express schema and metadata.
- FLOWS_FILES - This account owns the Application Express uploaded files.
- APEX_PUBLIC_USER - This minimally privileged account is used for Application Express configuration with Oracle REST Data Services or Oracle HTTP Server and `mod_plsql`.
- APEX_INSTANCE_ADMIN_USER - A minimally privileged account used for the REST Administration Interface.

 **Tip:**

To learn more about the REST Administration Interface, see "About the REST Administration Interface" in *Oracle Application Express Administration Guide*

 **Note:**

In an environment where EPG is used to access Oracle Application Express, the existing minimally privileged account `ANONYMOUS` is used.

If you are upgrading from a previous release, then `FLOWS_FILES` already exists and `APEX_PUBLIC_USER` is created if it does not already exist.

 **Tip:**

Oracle Application Express must be installed from a writable directory on the file system. See "[Reviewing a Log of an Installation Session](#)."

 **See Also:**

- "Configuring Password Protection" in *Oracle Database Security Guide*
- ["About the Oracle Application Express Runtime Environment"](#)
- "SQL*Plus User's Guide and Reference for more information about SQL*Plus"
- ["Reviewing a Log of an Installation Session"](#)

A.3.2 Creating or Updating Your Instance Administration Account

Learn how to create or update your Instance Administrator account.

This section describes how to create or update your Instance Administrator account.

 **Tip:**

Skip this section if you are upgrading from a previous release of Oracle Application Express. In an upgrade scenario, the Instance Administrator account and password is preserved and carried over from the prior release.

- [What Is an Instance Administrator?](#)
Instance administrators are superusers that are responsible for managing an entire Oracle Application Express instance, including managing workspace provisioning, configuring features and instance settings, and managing security.
- [About apxchpwd.sql](#)
Run the `apxchpwd.sql` script to create or update your Instance Administrator account.
- [Running apxchpwd.sql](#)
Run the `apxchpwd.sql` script to create and update your Instance Administrator account.

A.3.2.1 What Is an Instance Administrator?

Instance administrators are superusers that are responsible for managing an entire Oracle Application Express instance, including managing workspace provisioning, configuring features and instance settings, and managing security.

To perform these tasks, an Instance administrator signs in to the Oracle Application Express Administration Services application.

 **See Also:**

"Oracle Application Express Administration Services" in *Oracle Application Express Administration Guide*

A.3.2.2 About apxchpwd.sql

Run the `apxchpwd.sql` script to create or update your Instance Administrator account.

Running the `apxchpwd.sql` script enables you to create or update your Instance Administrator account. You must run the `apxchpwd.sql` script in the following scenarios:

- **New Oracle Application Express installations** - Run `apxchpwd.sql` to create an Instance Administrator account and password.
- **Converting of a runtime environment to a development environment** - Run `apxchpwd.sql` to change the Instance Administrator account password.
- **Changing Your Instance Administrator Password** - Run `apxchpwd.sql` to change the password for an existing Instance Administrator account.
- **Unlocking Your Instance Administrator Account** - Run `apxchpwd.sql` to unlock an existing Instance Administrator account.

 **Tip:**

You do not need to run `apxchpwd.sql` when upgrading from a previous release of Oracle Application Express. In an upgrade scenario, the Instance Administrator account password is preserved and carried over from the prior release.

A.3.2.3 Running apxchpwd.sql

Run the `apxchpwd.sql` script to create and update your Instance Administrator account.

To create or update your Instance Administrator account:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

If Application Express was installed in a CDB, change your container to the PDB where you want to change the Administrator account password. For example: `alter session set container = 'PDB1'`

3. If Application Express was installed in a CDB, change your container to the PDB where you want to change the Administrator account password. For example:

```
alter session set container = 'PDB1';
```

4. Run `apxchpwd.sql`. For example:

```
@apxchpwd.sql
```

Follow the on-screen instructions. You will be prompted provide a username, password, and email address. If the account username does not exist, it will be created for you.

See Also:

*SQL*Plus User's Guide and Reference* for more information about SQL*Plus

A.3.3 Restart Processes

Restart the processes that you stopped before you began the installation.

After you install Oracle Application Express, you must restart the processes that you stopped before you began the installation.

A.4 Configuring the Embedded PL/SQL Gateway

To configure the embedded PL/SQL gateway, you run a configuration script and unlock the `ANONYMOUS` account.

The embedded PL/SQL gateway installs with the Oracle Database 11g and later. However, you must configure it before you can use it with Oracle Application Express. To configure the embedded PL/SQL gateway, you run a configuration script and unlock the `ANONYMOUS` account.

Note:

The Oracle XML DB Protocol Server with the Embedded PL/SQL Gateway is not supported before Oracle Database 11g.

- [Running the `apex_epg_config.sql` Configuration Script](#)
Run the `apex_epg_config.sql` script. Then, unlock the `ANONYMOUS` account.
- [Updating the Images Directory When Upgrading from a Previous Release](#)
Run the `apex_epg_config.sql` script to update the `images` directory.

- [Verifying the Port Assigned to HTTP on the Oracle XML DB HTTP Server](#)
Start SQL* Plus and connect to the database and enter the statement to verify the port number assigned to HTTP on the Oracle XML DB HTTP Server.
- [Verifying the Oracle XML DB Protocol Server Port](#)
Start SQL* Plus and connect to the database and enter the statement to verify the port number where the Oracle XML DB Protocol Server is running.
- [Enabling Oracle XML DB Protocol Server](#)
Start SQL* Plus and connect to the database and run a SQL statement to enable Oracle XML DB Protocol Server.
- [Configuring Static File Support](#)
For configuring static files, you must run `apex_rest_config.sql` after a new installation or upgrade of Oracle Application Express.
- [About Configuring Embedded PL/SQL Gateway Behind a Reverse Proxy or Load Balancer](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. Execute the PL/SQL block.



See Also:

- ["Choosing a Web Listener"](#)
- ["About the Embedded PL/SQL Gateway"](#)

A.4.1 Running the `apex_epg_config.sql` Configuration Script

Run the `apex_epg_config.sql` script. Then, unlock the `ANONYMOUS` account.

You configure the embedded PL/SQL gateway by running the configuration script `apex_epg_config.sql`. Then, you unlock the `ANONYMOUS` account.

To run the `apex_epg_config.sql` configuration script:



WARNING:

This script has to be executed only on the database server . It's not possible to perform a remote connect to the database server. The database server should have access to the specified directory. It's also not possible to map a network directory.

1. Change your working directory to the `apex` directory where you unzipped the Oracle Application Express software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. If you are configuring a PDB, enter the following statement:

```
ALTER SESSION SET CONTAINER = <PDB_NAME>;
```

4. Run `apex_epg_config.sql` passing the file system path to the base directory where the Oracle Application Express software was unzipped as shown in the following example:

- On Windows:

```
@apex_epg_config.sql SYSTEM_DRIVE:\TEMP
```

- On UNIX and Linux:

```
@apex_epg_config.sql /tmp
```

5. Unlock the `ANONYMOUS` account. If configuring a PDB, run this statement in `CDB$ROOT`. Enter the following statement:

```
ALTER USER ANONYMOUS ACCOUNT UNLOCK;
```

A.4.2 Updating the Images Directory When Upgrading from a Previous Release

Run the `apex_epg_config.sql` script to update the `images` directory.

If you are upgrading Oracle Application Express from a previous release, you must run the `apex_epg_config.sql` script to update the `images` directory.

Tip:

If this is a new installation, this step is unnecessary. The images will be loaded by running `apex_epg_config.sql` as described in the prior section.

To run the `apex_epg_config.sql` script:

1. Change your working directory to the `apex` directory where you unzipped the Oracle Application Express software.

2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apex_epg_config.sql` passing the file system path to the base directory where the Oracle Application Express software was unzipped as shown in the following example:

- On Windows:

```
@apex_epg_config.sql SYSTEM_DRIVE:\TEMP
```

- On UNIX and Linux:

```
@apex_epg_config.sql /tmp
```



Tip:

The above examples assume that you unzipped Oracle Application Express in a directory called `TEMP` on Windows and `tmp` on UNIX or Linux.

A.4.3 Verifying the Port Assigned to HTTP on the Oracle XML DB HTTP Server

Start SQL* Plus and connect to the database and enter the statement to verify the port number assigned to HTTP on the Oracle XML DB HTTP Server.

The embedded PL/SQL gateway runs in the Oracle XML DB Protocol Server in the Oracle database.

To verify the port number assigned to HTTP on the Oracle XML DB HTTP Server:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Enter the following statement to verify the port number:

```
SELECT DBMS_XDB.GETHTTPPORT FROM DUAL;
```

If the port number returns 0, the Oracle XML DB HTTP Server is disabled.

A.4.4 Verifying the Oracle XML DB Protocol Server Port

Start SQL* Plus and connect to the database and enter the statement to verify the port number where the Oracle XML DB Protocol Server is running.

The Embedded PL/SQL Gateway runs in the Oracle XML DB Protocol Server in the Oracle database. You can determine if the Oracle XML DB Protocol Server is enabled by verifying the associated port number.

To verify the port number where the Oracle XML DB Protocol Server is running:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Enter the following statement to verify the port number:

```
SELECT DBMS_XDB.GETHTTPPORT FROM DUAL;
```

If the port number returns 0, the Oracle XML DB Protocol Server is disabled.

3. To enable the Oracle XML DB Protocol Server.

See Also:

["Enabling Oracle XML DB Protocol Server"](#)

A.4.5 Enabling Oracle XML DB Protocol Server

Start SQL* Plus and connect to the database and run a SQL statement to enable Oracle XML DB Protocol Server.

The embedded PL/SQL gateway runs in the Oracle XML DB Protocol Server in the Oracle database.

To enable Oracle XML DB Protocol Server:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Enter a statement similar to the following:

```
EXEC DBMS_XDB.SETHTTPPORT(port);
```

For example:

```
EXEC DBMS_XDB.SETHTTPPORT(8080);
```

Note:

Port numbers less than 1024 are reserved for use by privileged processes on many operating systems. To enable the XML DB HTTP listener on a port less than 1024, such as 80, review the documentation.

See Also:

- "Repository Access Using Protocols" in *Oracle XML DB Developer's Guide*.
- "Protocol Address Configuration and Port Number Limitations " in *Oracle Database Net Services Reference* .

A.4.6 Configuring Static File Support

For configuring static files, you must run `apex_rest_config.sql` after a new installation or upgrade of Oracle Application Express.

Oracle Application Express enables application developers to include static files with their applications. Static files can be associated with a workspace, an application, a plug-in, or an application theme. When using the embedded PL/SQL gateway as your web listener, the necessary configuration to support static files is performed as part of the embedded PL/SQL gateway configuration.

See Also:

- For new installations, see "[Running the apex_epg_config.sql Configuration Script](#)"
- For upgrade installations, see "[Updating the Images Directory When Upgrading from a Previous Release](#)"

A.4.7 About Configuring Embedded PL/SQL Gateway Behind a Reverse Proxy or Load Balancer

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. Execute the PL/SQL block.

If Oracle Application Express is running behind a reverse proxy or load balancer, you set the CGI environment variables `PROTOCOL` and `HTTP_HOST`. For embedded PL/SQL gateway, you must call the following PL/SQL block:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following PL/SQL block:

```
BEGIN
  DBMS_EPG.set_dad_attribute (
    dad_name => 'APEX',
    attr_name => 'cgi-environment-list',
    attr_value => 'REQUEST_PROTOCOL=https');
  DBMS_EPG.set_dad_attribute (
    dad_name => 'APEX',
    attr_name => 'cgi-environment-list',
    attr_value => 'HTTP_HOST=external-name:external-port');
  commit;
END;
/
```

To learn more, see your Java EE application servers documentation.

A.5 Enabling Network Services in Oracle Database 11g or Later

You must enable network services in Oracle Database 11g or later versions to send outbound mail, use Web services, or use PDF report printing in Oracle Application Express.

- [When and Why Network Services Must be Enabled](#)
Enabling network services enables support for sending outbound mail in Oracle Application Express, use of Web services in Oracle Application Express, and PDF report printing.
- [Granting Connect Privileges Prior to Oracle Database 12c](#)
Demonstrates how to grant connect privileges to any host for the APEX_190200 database user.
- [Granting Connect Privileges in Oracle Database 12c or Later](#)
Procedures CREATE_ACL, ASSIGN_ACL, ADD_PRIVILEGE and CHECK_PRIVILEGE in DBMS_NETWORK_ACL_ADMIN are deprecated in Oracle Database 12c. Oracle recommends to use APPEND_HOST_ACE.
- [Troubleshooting an Invalid ACL Error](#)
Learn how to identify any invalid ACL error by running the query.

A.5.1 When and Why Network Services Must be Enabled

Enabling network services enables support for sending outbound mail in Oracle Application Express, use of Web services in Oracle Application Express, and PDF report printing.

By default, the ability to interact with network services is disabled in Oracle Database 11g Release 2 or later. Therefore, if you are running Oracle Application Express with Oracle Database 11g Release 2 or later, you must use the new DBMS_NETWORK_ACL_ADMIN package to grant connect privileges to any host for the APEX_190200 database user. Failing to grant these privileges results in issues with:

- Sending outbound mail in Oracle Application Express.

Users can call methods from the `APEX_MAIL` package, but issues arise when sending outbound email.

- Using Web services in Oracle Application Express.
- PDF report printing.

 **Note:**

When upgrading Application Express on a database 12c or newer, based on the configuration of the old Application Express version the upgrade automatically configures Network Services.

 **Tip:**

To run the examples described in this section, the compatible initialization parameter of the database must be set to at least 11.1.0.0.0. By default an 11g or 12c database will already have the parameter set properly, but a database upgraded to 11g or 12c from a prior version may not. For information about changing database initialization parameters, see "Creating and Configuring an Oracle Database" in *Oracle Database Administrator's Guide*.

A.5.2 Granting Connect Privileges Prior to Oracle Database 12c

Demonstrates how to grant connect privileges to any host for the `APEX_190200` database user.

The following example demonstrates how to grant connect privileges to any host for the `APEX_190200` database user. This example assumes you connected to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role.

```
DECLARE
    ACL_PATH VARCHAR2(4000);
BEGIN
    -- Look for the ACL currently assigned to '*' and give APEX_190200
    -- the "connect" privilege if APEX_190200 does not have the privilege
    yet.

    SELECT ACL INTO ACL_PATH FROM DBA_NETWORK_ACLS
        WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

    IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE(ACL_PATH, 'APEX_190200',
        'connect') IS NULL THEN
        DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE(ACL_PATH,
        'APEX_190200', TRUE, 'connect');
    END IF;

EXCEPTION
    -- When no ACL has been assigned to '*'.
```

```

WHEN NO_DATA_FOUND THEN
  DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('power_users.xml',
    'ACL that lets power users to connect to everywhere',
    'APEX_190200', TRUE, 'connect');
  DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('power_users.xml','*');
END;
/
COMMIT;

```

The following example demonstrates how to provide less privileged access to local network resources. This example enables access to servers on the local host only, such as email and report servers.

```

DECLARE
  ACL_PATH VARCHAR2(4000);
BEGIN
  -- Look for the ACL currently assigned to 'localhost' and give
  APEX_190200
  -- the "connect" privilege if APEX_190200 does not have the privilege
  yet.
  SELECT ACL INTO ACL_PATH FROM DBA_NETWORK_ACLS
    WHERE HOST = 'localhost' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

  IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE(ACL_PATH, 'APEX_190200',
    'connect') IS NULL THEN
    DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE(ACL_PATH,
    'APEX_190200', TRUE, 'connect');
  END IF;

EXCEPTION
  -- When no ACL has been assigned to 'localhost'.
  WHEN NO_DATA_FOUND THEN
    DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('local-access-users.xml',
    'ACL that lets users to connect to localhost',
    'APEX_190200', TRUE, 'connect');
    DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('local-access-users.xml','localhost');
END;
/
COMMIT;

```

A.5.3 Granting Connect Privileges in Oracle Database 12c or Later

Procedures `CREATE_ACL`, `ASSIGN_ACL`, `ADD_PRIVILEGE` and `CHECK_PRIVILEGE` in `DBMS_NETWORK_ACL_ADMIN` are deprecated in Oracle Database 12c. Oracle recommends to use `APPEND_HOST_ACE`.

The following example demonstrates how to grant connect privileges to any host for the `APEX_190200` database user. This example assumes you connected to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role.

```

BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(

```

```

host => '*',
ace => xs$ace_type(privilege_list => xs$name_list('connect'),
                  principal_name => 'APEX_190200',
                  principal_type => xs_acl.ptype_db));
END;
/

```

The following example demonstrates how to provide less privileged access to local network resources. This example enables access to servers on the local host only, such as email and report servers.

```

BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host => 'localhost',
    ace => xs$ace_type(privilege_list => xs$name_list('connect'),
                      principal_name => 'APEX_190200',
                      principal_type => xs_acl.ptype_db));
END;
/

```

A.5.4 Troubleshooting an Invalid ACL Error

Learn how to identify any invalid ACL error by running the query.

If you receive an `ORA-44416: Invalid ACL error` after running the previous script, use the following query to identify the invalid ACL:

```

REM Show the dangling references to dropped users in the ACL that is
assigned
REM to '*'.

SELECT ACL, PRINCIPAL
  FROM DBA_NETWORK_ACLS NACL, XDS_ACE ACE
 WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL AND
        NACL.ACLID = ACE.ACLID AND
        NOT EXISTS (SELECT NULL FROM ALL_USERS WHERE USERNAME = PRINCIPAL);

```

Next, run the following code to fix the ACL:

```

DECLARE
  ACL_ID  RAW(16);
  CNT     NUMBER;
BEGIN
  -- Look for the object ID of the ACL currently assigned to '*'
  SELECT ACLID INTO ACL_ID FROM DBA_NETWORK_ACLS
     WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

  -- If just some users referenced in the ACL are invalid, remove just
those
  -- users in the ACL. Otherwise, drop the ACL completely.
  SELECT COUNT(PRINCIPAL) INTO CNT FROM XDS_ACE
     WHERE ACLID = ACL_ID AND
           EXISTS (SELECT NULL FROM ALL_USERS WHERE USERNAME = PRINCIPAL);

```

```
IF (CNT > 0) THEN

    FOR R IN (SELECT PRINCIPAL FROM XDS_ACE
              WHERE ACLID = ACL_ID AND
                    NOT EXISTS (SELECT NULL FROM ALL_USERS
                                WHERE USERNAME = PRINCIPAL)) LOOP

        UPDATE XDB.XDB$ACL
           SET OBJECT_VALUE =
               DELETEXML(OBJECT_VALUE,
                          '/ACL/ACE[PRINCIPAL="' || R.PRINCIPAL || '"]')
           WHERE OBJECT_ID = ACL_ID;
    END LOOP;

ELSE
    DELETE FROM XDB.XDB$ACL WHERE OBJECT_ID = ACL_ID;
END IF;

END;
/

REM commit the changes.

COMMIT;
```

Once the ACL has been fixed, you must run the first script in this section to apply the ACL to the APEX_190200 user.

A.6 Performing Security Tasks

Oracle recommends configuring and using Secure Sockets Layer (SSL) to ensure that passwords and other sensitive data are not transmitted in clear text in HTTP requests.

Without the use of SSL, passwords could potentially be exposed, compromising security.

SSL is an industry standard protocol that uses RSA public key cryptography in conjunction with symmetric key cryptography to provide authentication, encryption, and data integrity.



See Also:

"Configuring HTTP Protocol Attributes" in *Oracle Application Express Administration Guide*

A.7 Controlling the Number of Concurrent Jobs

Learn about specifying the number of concurrently running jobs.

- [About Managing the Number of Concurrent Jobs](#)
Learn about managing maximum number of concurrently running jobs.

- [Viewing the Number of JOB_QUEUE_PROCESSES](#)
You can view number of JOB_QUEUE_PROCESSES in three ways.
- [Changing the Number of JOB_QUEUE_PROCESSES](#)
You can change the number of JOB_QUEUE_PROCESSES by running a SQL statement in SQL*Plus.

A.7.1 About Managing the Number of Concurrent Jobs

Learn about managing maximum number of concurrently running jobs.

JOB_QUEUE_PROCESSES determine the maximum number of concurrently running jobs. In Oracle Application Express transactional support and SQL scripts require jobs. If JOB_QUEUE_PROCESSES is not enabled and working properly, you cannot successfully execute a script.

A.7.2 Viewing the Number of JOB_QUEUE_PROCESSES

You can view number of JOB_QUEUE_PROCESSES in three ways.

- [Viewing JOB_QUEUE_PROCESSES in the Installation Log File](#)
View JOB_QUEUE_PROCESSES in the installation log files.
- [Viewing JOB_QUEUE_PROCESSES in Oracle Application Express](#)
View the number of JOB_QUEUE_PROCESSES on the About Application Express page.
- [Viewing JOB_QUEUE_PROCESSES from SQL*Plus](#)
View the number of JOB_QUEUE_PROCESSES from SQL*Plus.

A.7.2.1 Viewing JOB_QUEUE_PROCESSES in the Installation Log File

View JOB_QUEUE_PROCESSES in the installation log files.



See Also:

["Reviewing a Log of an Installation Session"](#)

A.7.2.2 Viewing JOB_QUEUE_PROCESSES in Oracle Application Express

View the number of JOB_QUEUE_PROCESSES on the About Application Express page.

To view the About Application Express page:

1. Sign in to Oracle Application Express.
2. Locate the Help menu at the top of the page.
3. From the Help menu, select **About**.
The About Application Express page appears.
4. Scroll down and find JOB_QUEUE_PROCESSES at the bottom of the page.



See Also:

["Signing into a Workspace Using Oracle REST Data Services"](#)

A.7.2.3 Viewing JOB_QUEUE_PROCESSES from SQL*Plus

View the number of JOB_QUEUE_PROCESSES from SQL*Plus.

Run the following SQL statement:

```
SELECT VALUE FROM v$parameter WHERE NAME = 'job_queue_processes'
```

A.7.3 Changing the Number of JOB_QUEUE_PROCESSES

You can change the number of JOB_QUEUE_PROCESSES by running a SQL statement in SQL*Plus.

To update the number of JOB_QUEUE_PROCESSES:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

2. In SQL*Plus run the following SQL statement:

```
ALTER SYSTEM SET JOB_QUEUE_PROCESSES = <number>
```

For example, running the statement ALTER SYSTEM SET JOB_QUEUE_PROCESSES = 20 sets JOB_QUEUE_PROCESSES to 20.

A.8 About Running Oracle Application Express in Other Languages

You can install a single instance of Oracle Application Express with one or more of translated versions.

The Oracle Application Express interface is translated into German, Spanish, French, Italian, Japanese, Korean, Brazilian Portuguese, Simplified Chinese, and Traditional Chinese. A single instance of Oracle Application Express can be installed with one or

more of these translated versions. Developers can choose to run the Application Express development environment in any of the installed languages by simply selecting the language from the App Builder log in screen or home page.

In order to install other languages you must use the `apex_19.2.zip` file which contains the extra files referenced below. If you previously downloaded `apex_19.2_en.zip`, then you do not need to re-install Oracle Application Express. Simply download `apex_19.2.zip` and unzip the file into the same directory where you unzipped `apex_19.2_en.zip`.

The translated version of Oracle Application Express should be loaded into a database that has a character set that supports the specific language. If you attempt to install a translated version of Oracle Application Express into a database that does not support the character encoding of the language, the installation may fail or the translated Oracle Application Express instance may appear corrupt when run. The database character set `AL32UTF8` supports all the translated versions of Oracle Application Express.

You can manually install translated versions of Oracle Application Express using SQL*Plus. The installation files are encoded in `AL32UTF8`.

 **Note:**

Regardless of the target database character set, to install a translated version of Oracle Application Express, you must set the character set value of the `NLS_LANG` environment variable to `AL32UTF8` before starting SQL*Plus.

The following examples illustrate valid `NLS_LANG` settings for loading Oracle Application Express translations:

```
American_America.AL32UTF8  
Japanese_Japan.AL32UTF8
```

A.9 Installing Translated Versions of Oracle Application Express

Learn about installing translated versions of Oracle Application Express.

- [About Installing Translated Versions of Oracle Application Express](#)
You must run `load_lang.sql` script to run a translated version of Oracle Application Express.
- [Installing a Translated Version of Oracle Application Express](#)
Learn how to install a translated version of Oracle Application Express.

A.9.1 About Installing Translated Versions of Oracle Application Express

You must run `load_lang.sql` script to run a translated version of Oracle Application Express.

Whether you are installing for the first time or upgrading from a previous release, you must run the `load_lang.sql` script to run a translated version of Oracle Application Express.

The installation scripts are located in subdirectories identified by a language code in the unzipped distribution `apex/builder`. For example, the German version is located in `apex/builder/de` and the Japanese version is located in `apex/builder/ja`. Within each directory, there is a language loading script identified by the language code (for example, `load_de.sql` or `load_ja.sql`).

A.9.2 Installing a Translated Version of Oracle Application Express

Learn how to install a translated version of Oracle Application Express.

To install a translated version of Oracle Application Express:

1. Set the `NLS_LANG` environment variable, making sure that the character set is `AL32UTF8`. For example:

- Bourne or Korn shell:

```
NLS_LANG=American_America.AL32UTF8
export NLS_LANG
```

- C shell:

```
setenv NLS_LANG American_America.AL32UTF8
```

- For Windows based systems:

```
set NLS_LANG=American_America.AL32UTF8
```

2. Navigate to the directory under `apex/builder` based on the language you need to install. For example for German, navigate to `apex/builder/de`. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following statement:

```
ALTER SESSION SET CURRENT_SCHEMA = APEX_190200;
```

4. Execute the appropriate language specific script. For example:

```
@load_lang.sql
```

Where `lang` is the specific language (for example, `load_de.sql` for German or `load_ja.sql` for Japanese).

A.10 About Configuring the SHARED_SERVERS Parameter

Achieve acceptable performance when using the embedded PL/SQL gateway by setting the `SHARED_SERVERS` database initialization parameter to a reasonable value.

The embedded PL/SQL gateway uses the shared server architecture of the Oracle Database. To achieve acceptable performance when using the embedded PL/SQL gateway, ensure the `SHARED_SERVERS` database initialization parameter is set to a reasonable value (that is, not 0 or 1). For a small group of concurrent users, Oracle recommends a value of 5 for `SHARED_SERVERS`.

Consider the following example:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following statement:

```
ALTER SYSTEM SET SHARED_SERVERS = 5 SCOPE=BOTH;
```

A.11 Creating a Workspace and Adding Oracle Application Express Users

Before you can develop or install applications, you must create a workspace, add Oracle Application Express users and sign in to your workspace.

- [About Workspaces and Users](#)
A workspace enables multiple users to work within the same Oracle Application Express installation while keeping their objects, data, and applications private.
- [Accessing Administration Services Using Embedded PL/SQL Gateway](#)
You can sign in to Oracle Application Express Administration Services using the `ADMIN` account and password created or reset during the installation process.

- [Creating a Workspace Manually](#)
Sign in to Oracle Application Express Administration Services to create workspace manually.
- [Creating Oracle Application Express Users](#)
Create new users in Oracle Application Express Administration Services..
- [Signing into a Workspace Using the Embedded PL/SQL Gateway](#)
Learn how to sign in to a workspace using the embedded PL/SQL gateway.

A.11.1 About Workspaces and Users

A workspace enables multiple users to work within the same Oracle Application Express installation while keeping their objects, data, and applications private.

You access the Oracle Application Express home page by logging in to a workspace using a JavaScript enabled Web browser.

Each workspace has a unique ID and name. An instance administrator can create a workspace manually within Oracle Application Express Administration Services or have users submit requests. Oracle Application Express Administration Services is a separate application for managing an entire Oracle Application Express instance.

See Also:

- ["Browser Requirements"](#)
- ["Creating Workspaces" in *Oracle Application Express Administration Guide*](#)
- ["Managing Workspace and Change Requests" in *Oracle Application Express Administration Guide*](#)

A.11.2 Accessing Administration Services Using Embedded PL/SQL Gateway

You can sign in to Oracle Application Express Administration Services using the `ADMIN` account and password created or reset during the installation process.

To manually create a workspace and user accounts, you sign in to a separate application for managing an entire Oracle Application Express instance called Oracle Application Express Administration Services. You sign in to Oracle Application Express Administration Services using the `ADMIN` account and password created or reset during the installation process.

To access Oracle Application Express Administration Services:

1. In a Web browser, navigate to the Oracle Application Express Administration Services application.

Because your setup uses embedded PL/SQL gateway, go to:

`http://hostname:port/apex/apex_admin`

Where:

hostname is the name of the system where Oracle XML DB Protocol server is installed.

port is the port number assigned to HTTP on the Oracle XML DB Protocol server. In a default installation, this number is 8080. If you are using the Oracle Database 12c or later multitenant architecture, then each pluggable database (PDB) will have a distinct port number.

apexis the database access descriptor (DAD) defined in the configuration file.

2. On the Sign In page:

- Username - Enter `admin`.
- Password - is the database access descriptor (DAD) defined in the configuration file.
- Click **Sign In to Administration**.

Note that, depending on your setup, you might be required to change your password when you log in for the first time.

 **See Also:**

- ["Creating or Updating Your Instance Administration Account"](#)
- "Oracle Application Express Administration Services" in *Oracle Application Express Administration Guide*.

A.11.3 Creating a Workspace Manually

Sign in to Oracle Application Express Administration Services to create workspace manually.

To manually create a workspace you sign in to a separate application called Oracle Application Express Administration Services. You sign in to Oracle Application Express Administration Services using the `ADMIN` account and password created or reset during the installation process.

To create an Oracle Application Express workspace manually:

1. Access Oracle Application Express Administration Services.
Oracle Application Express Administration Services appears. Next, create a workspace.
2. Click **Manage Workspaces**.
3. Under Workspace Actions, click **Create Workspace**.
The Create Workspace Wizard appears.
4. For Identify Workspace, enter the following:
 - a. Workspace Name - Enter a unique workspace name.

 **Tip:**

Most attributes in Oracle Application Express include field-level Help. Attributes with field-level Help, have light gray icon that resembles a question mark (?). To view field-level Help, click the Help icon.

6. Under Account Privileges:
 - a. **Workspace** - Select a workspace from the list.
 - b. **Default Schemas** - Specify the default schema used for data browsing, application creation, and SQL script execution.

When using workspaces that have more than one schema available, this schema is the default. This setting does not control security, only the user's preference.
 - c. **User is an administrator** - Specify if this user should have workspace administrator privileges.

Administrators are given access to all components. Additionally, they can manage user accounts, groups, and development services. Components may not be available if they are switched off by Instance Administrators.
 - d. **User is a developer** - Specify if this user should have developer privileges.

Developers must have access to either App Builder, SQL Workshop, or both. These components may not be available if they are switched off by the Instance Administrator.
 - e. **App Builder Access** - Determines whether a developer has access to the App Builder.
 - f. **SQL Workshop Access** - Determines whether a developer has access to the SQL Workshop.
 - g. **Team Development Access** - Determines whether a developer has access to the Team Development.
 - h. **Set Account Availability** - Select **Locked** to prevent the account from being used. Select **Unlocked** to allow the account to be used.

If the user has exceeded the maximum log in failures allowed, specified in Workspace Preferences, then their account will be locked automatically.
7. Under Password:
 - **Password** - Enter a case sensitive password.
 - **Confirm Password** - Enter the password again.
 - **Require Change of Password On First Use** - Select **No** to allow the user to use the same password until it expires. Select **Yes** to require the user to change the password immediately when logging in the first time.
8. Click **Create User** or **Create and Create Another**.



See Also:

["Accessing Administration Services Using Oracle REST Data Services"](#)

A.11.5 Signing into a Workspace Using the Embedded PL/SQL Gateway

Learn how to sign in to a workspace using the embedded PL/SQL gateway.

Use your credentials (that is, your workspace name, user name and password) to sign in to the workspace.

To sign in to your workspace:

1. In a Web browser, navigate to the Oracle Application Express Sign In page.

Because your setup uses Embedded PL/SQL Gateway, go to:

```
http://hostname:port/apex/
```

Where:

- *hostname* is the name of the system where Oracle XML DB Protocol server is installed.
- *port* is the port number assigned to HTTP on the Oracle XML DB Protocol server. In a default installation, this number is 8080. If you are using the Oracle Database 12c or later multitenant architecture, then each pluggable database (PDB) will have a distinct port number.
- *apex* is the database access descriptor (DAD) defined in the configuration file.

The Sign In page appears.

2. On the Sign In page, enter:
 - Workspace - Enter the name of your workspace.
 - Username - Enter your user name.
 - Password - Enter your case-sensitive password.
3. Click **Sign In**.

Note that, depending on your setup, you might be required to change your password when you log in for the first time.



See Also:

- "Creating Workspaces" in *Oracle Application Express Administration Guide*
- "Managing Workspace and Change Requests in Oracle Application Express Administration Guide"

A.12 Performing Post Installation Tasks for Upgrade Installations

Remove schemas from prior Oracle Application Express installations.

Once you have verified that your upgrade installation was successful and all upgraded applications function properly, you should remove schemas from prior Oracle Application Express installations.

- [About Removing Prior Oracle Application Express Installations](#)
Verifying if a prior installation exists and removing the schema.
- [Verifying if a Prior Installation Exists](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` and run the query.
- [Removing Schemas and SYS Objects from Prior Installations](#)
Start SQL*Plus and connect to database and execute a statement to remove schemas and `SYS` objects.
- [Fixing Invalid ACL in Oracle Database 11g or Later](#)
Learn how to fix an invalid ACL if you are running Oracle Database 11g or later.



See Also:

["Upgrading from a Previous Oracle Application Express Release"](#)

A.12.1 About Removing Prior Oracle Application Express Installations

Verifying if a prior installation exists and removing the schema.

The database users associated with schemas from prior installations are privileged users and should be removed when they are no longer necessary. Removing schemas from a prior installation is a two step process. First you verify if a prior installation exists and then you remove the schemas.

A.12.2 Verifying if a Prior Installation Exists

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` and run the query.

To verify if a prior installation exists:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS`. For example:
 - On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following query:

```
SELECT username FROM dba_users WHERE ( username LIKE 'FLOWS\_____'
ESCAPE '\' OR username LIKE 'APEX\_____' ESCAPE '\' ) AND
username NOT IN ( SELECT schema FROM
dba_registry WHERE comp_id = 'APEX' );
```

If the results contain entries in the form FLOWS_XXXXXX or APEX_XXXXXX where XXXXXX represents six numbers, those entries are candidates for removal.

A.12.3 Removing Schemas and SYS Objects from Prior Installations

Start SQL*Plus and connect to database and execute a statement to remove schemas and SYS objects.

To remove schemas and SYS objects from prior installations:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Execute statements similar to the following example:

```
DROP USER APEX_190100 CASCADE;
DROP PACKAGE SYS.WWV_DBMS_SQL_APEX_190100;
```

A.12.4 Fixing Invalid ACL in Oracle Database 11g or Later

Learn how to fix an invalid ACL if you are running Oracle Database 11g or later.

After following the instructions in "[About Removing Prior Oracle Application Express Installations](#)," you may need to fix an invalid ACL if you are running Oracle Database 11g or later and you enabled network services for the prior Oracle Application Express schema.

To fix an invalid ACL:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute a statement similar to the following:

```
EXEC DBMS_NETWORK_ACL_ADMIN.DELETE_PRIVILEGE('power_users.xml',
'APEX_040200');
```

A.13 About Performance Optimization Tasks

Learn about performance optimization.

Performance of web applications heavily depends on their size and how often a browser has to request static content like images, CSS and JavaScript files. To improve performance, most web servers support on-the-fly HTTP response compression and provide settings that enable you to configure on how long browsers can cache a file before requesting it again. The HTTP response compression is usually implemented using gzip encoding, while browser file caching is enabled by issuing Cache-Control HTTP response header.

Please see your web server documentation to learn how to enable response compression and browser file caching. For optimal performance of the Oracle Application Express development environment and Oracle Application Express applications, Oracle recommends enabling gzip compression of files in the virtual images directory (for example, `/i/`) and responses from the database access descriptor as well as allowing browsers to cache files from the virtual images directory for at least 12 hours.

A.14 Converting Between Runtime and Full Development Environments

Convert between runtime and full development environments.

- [About Runtime and Full Development Environments](#)
An Oracle Application Express runtime environment enables users to run a production application without supporting the ability to change or edit the application.

- [Converting a Runtime Environment to a Full Development Environment in a non-CDB or locally in a PDB](#)
 Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run `apxdvins.sql`.
- [Converting a Full Development Environment to a Runtime Environment in a non-CDB or locally in a PDB](#)
 Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrn.sql`.
- [Converting a Runtime Environment to a Full Development Environment in a CDB](#)
 Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdvins_con.sql`.
- [Converting a Full Development Environment to a Runtime Environment in a CDB](#)
 Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrn_con.sql`.

A.14.1 About Runtime and Full Development Environments

An Oracle Application Express runtime environment enables users to run a production application without supporting the ability to change or edit the application.

A runtime environment includes only the packages necessary to run your applications, making it a more hardened environment. It does not provide a web interface for administration.

You administer an Oracle Application Express runtime environment using SQL*Plus or SQL Developer and the `APEX_INSTANCE_ADMIN` API.



See Also:

- ["About the Oracle Application Express Runtime Environment"](#) in *Oracle Application Express Administration Guide*
- ["Installing Exported Applications into a Runtime Environment"](#) in *Oracle Application Express Administration Guide*

A.14.2 Converting a Runtime Environment to a Full Development Environment in a non-CDB or locally in a PDB

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run `apxdvins.sql`.

To convert an Oracle Application Express runtime environment to a full development SQL*Plus or SQL Developer and the `APEX_INSTANCE_ADMIN` API:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxdvins.sql`. For example:

```
@apxdvins.sql
```

4. Follow the instructions in [Creating or Updating Your Instance Administration Account](#).

See Also:

- ["Creating or Updating Your Instance Administration Account."](#)
- *SQL*Plus User's Guide and Reference* for more information about SQL*Plus

A.14.3 Converting a Full Development Environment to a Runtime Environment in a non-CDB or locally in a PDB

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrm.sql`.

Tip:

Before converting a full development environment to a runtime environment, export all Websheet applications. When you convert a development environment to runtime, existing Websheet data grids and reports will be deleted from all workspaces. Converting the environment back to a full development environment will not restore these Websheet data grids or reports.

To convert an Oracle Application Express full development environment to a runtime environment in a non-CDB or locally in a PDB:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxdevm.sql`. For example:

```
@apxdevm.sql
```



See Also:

*SQL*Plus User's Guide and Reference* for more information about SQL*Plus

A.14.4 Converting a Runtime Environment to a Full Development Environment in a CDB

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdvins_con.sql`.

To convert an Oracle Application Express runtime environment to a full development environment in a CDB:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxdvins_con.sql`. For example:

```
@apxdvins_con.sql
```

4. Follow the instructions in [Creating or Updating Your Instance Administration Account](#).

See Also:

- ["Creating or Updating Your Instance Administration Account."](#)
- *SQL*Plus User's Guide and Reference* for more information about SQL*Plus

A.14.5 Converting a Full Development Environment to a Runtime Environment in a CDB

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrm_con.sql`.

Tip:

Before converting a full development environment to a runtime environment, export all Websheet applications. When you convert a development environment to runtime, existing Websheet data grids and reports will be deleted from all workspaces. Converting the environment back to a full development environment will not restore these Websheet data grids or reports.

To convert an Oracle Application Express full development environment to a runtime environment in a CDB:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxdevrm_con.sql`. For example:

```
@apxdevrm_con.sql
```

 **See Also:**

*SQL*Plus User's Guide and Reference* for more information about SQL*Plus

B

Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server

Install Oracle Application Express by downloading a ZIP file from the Oracle Application Express download page and then configuring Oracle HTTP Server.

Note:

`mod_plsql` is deprecated as of Oracle HTTP Server 12c (12.1.3). For more information, see My Oracle Support Note 1576588.1. Oracle recommends using Oracle REST Data Services. See "[Installing Application Express and Configuring Oracle REST Data Services](#)."

This section describes how to install Oracle Application Express by downloading a ZIP file from Oracle Technology Network (OTN) and then configuring legacy Web listener Oracle HTTP Server with `mod_plsql` distributed with Oracle Database 11g or Oracle Application Server 10g and Oracle Database 12c or later. These instructions apply to both new and upgrade installations.

- [Oracle Database Requirements](#)
Oracle Application Express release 19.2 requires an Oracle Database release 11.2.0.4 or later, including Enterprise Edition and Express Edition (Oracle Database 18c XE).
- [Performing Pre-installation Tasks for Oracle Application Express](#)
Review and perform pre-installation tasks before installing Oracle Application Express.
- [Downloading and Installing Oracle Application Express](#)
Learn about downloading and installing Oracle Application Express.
- [Configuring the Oracle HTTP Server](#)
Configuring Oracle HTTP Server requires that you copy the images directory, edit the `dads.conf` file, configure static files support, stop and restart Oracle HTTP Server, disable Oracle XML DB Protocol Server, and configuring Oracle HTTP server behind a reverse proxy or load balancer.
- [Enabling Network Services in Oracle Database 11g or Later](#)
You must enable network services in Oracle Database 11g or later versions to send outbound mail, use Web services, or use PDF report printing in Oracle Application Express.
- [Performing Security Tasks](#)
Oracle recommends configuring and using Secure Sockets Layer (SSL) to ensure that passwords and other sensitive data are not transmitted in clear text in HTTP requests.

- [Controlling the Number of Concurrent Jobs](#)
Learn about specifying the number of concurrently running jobs.
- [About Running Oracle Application Express in Other Languages](#)
You can install a single instance of Oracle Application Express with one or more of translated versions.
- [Installing Translated Versions of Oracle Application Express](#)
Learn about installing translated versions of Oracle Application Express.
- [Creating a Workspace and Adding Oracle Application Express Users](#)
Learn about creating workspace and adding Oracle Application Express users.
- [Performing Post Installation Tasks for Upgrade Installations](#)
Remove schemas from prior Oracle Application Express installations.
- [About Performance Optimization Tasks](#)
Learn about performance optimization.
- [Converting Between Runtime and Full Development Environments](#)
Learn how to convert between runtime and full development environments.



See Also:

["Web Listener Requirements"](#)

B.1 Oracle Database Requirements

Oracle Application Express release 19.2 requires an Oracle Database release 11.2.0.4 or later, including Enterprise Edition and Express Edition (Oracle Database 18c XE).

If you are upgrading an Oracle Database version 12.1 CDB, you must download from My Oracle Support the one off patch for bug 20618595. Search for **20618595** on the Patches tab.

B.2 Performing Pre-installation Tasks for Oracle Application Express

Review and perform pre-installation tasks before installing Oracle Application Express.

Before installing Oracle Application Express, Oracle recommends that you complete the following steps:

1. Review and satisfy all Oracle Application Express installation requirements.
2. If you are actively using Oracle Application Express and upgrading the current installation then shut down with normal or immediate priority the Oracle Database instances where you plan to install Oracle Application Express. On Oracle Real Application Clusters (Oracle RAC) systems, shut down all instances on each node.

An alternative to shutting down the database is to prevent all users from accessing Oracle Application Express when upgrading your installation from a previous release of Oracle Application Express. Oracle only recommends this option in high availability production environments where planned outages are not available. For all other scenarios, the database should be shut down.

To disable user access to Oracle Application Express when the existing installation uses the Oracle HTTP Server with `mod_plsql`, you should either shut down the Web server or disable the Application Express Database Access Descriptor of the Web server.

Once you have prevented access from Oracle Application Express users, you should log in to SQL*Plus as `SYS`, connecting to the database where Oracle Application Express is installed, and query `V$SESSION` to ensure there are no long running sessions which would interfere with the upgrade process.

3. Back up the Oracle Database installation.

Oracle recommends that you create a backup of the current Oracle Database installation before you install Oracle Application Express. You can use Oracle Database Recovery Manager, which is included in the Oracle Database installation, to perform the backup.

4. Start the Oracle Database instance that contains the target database.

After backing up the system, you must start the Oracle instance that contains the target Oracle Database. Do not start other processes such as a Web listener. However, if you are performing a remote installation, make sure the Web listener for the remote database has started.

 **Note:**

If you are connecting to a remote database, then start the Web listener.

 **See Also:**

- "[Oracle Application Express Installation Requirements](#)"
- "[Oracle Database Backup and Recovery User's Guide](#)"

B.3 Downloading and Installing Oracle Application Express

Learn about downloading and installing Oracle Application Express.

This section describes how to download and install Oracle Application Express. How you install Oracle Application Express depends upon by the type of database you into which you are installing.

- [Installing Application Express](#)
Install Oracle Application Express by downloading a ZIP file from the Oracle Application Express download page.
- [Creating or Updating Your Instance Administration Account](#)
Learn how to create or update your instance administrator account.
- [Restarting Processes](#)
Restart the processes that you stopped before you began the installation.

- [Configuring the APEX_PUBLIC_USER Account](#)
It is important to correctly configure the `APEX_PUBLIC_USER` account to enable proper operation of Oracle Application Express.



See Also:

["Utilizing the Multitenant Architecture in Oracle Database 12c or Later."](#)

B.3.1 Installing Application Express

Install Oracle Application Express by downloading a ZIP file from the Oracle Application Express download page.

To install Oracle Application Express:

1. For installations where the development will be in English only, download the file `apex_19.2_en.zip` from the Oracle Application Express download page. If the development will include languages other than English, download `apex_19.2.zip` from the Oracle Application Express download page. See:

<http://www.oracle.com/technetwork/developer-tools/apex/downloads/index.html>

Note that the actual file name may differ if a more recent release has shipped since this document was published.

2. Unzip downloaded zip file:
 - If English only, unzip `apex_19.2_en.zip` as follows, preserving directory names:
 - UNIX and Linux: `$ unzip apex_19.2_en.zip`
 - Windows: Double click the file `apex_19.2_en.zip` in Windows Explorer
 - If multiple languages, unzip `apex_19.2.zip` as follows, preserving directory names:
 - UNIX and Linux: `$ unzip apex_19.2.zip`
 - Windows: Double click the file `apex_19.2.zip` in Windows Explorer



Note:

You should keep the directory tree where you unzip the files short and not under directories that contain spaces. For example, within Windows unzip to `C:\TEMP`.

3. Change your working directory to `apex`.
4. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:
 - On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

5. Disable any existing password complexity rules for the default profile.
6. Select the appropriate installation option.

Full development environment provides complete access to the App Builder environment to develop applications. A **Runtime environment** enables users to run applications that cannot be modified.

Available installation options include:

- **Full development environment.** Run `apexins.sql` passing the following four arguments in the order shown:

```
@apexins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- `tablespace_apex` is the name of the tablespace for the Oracle Application Express application user.
- `tablespace_files` is the name of the tablespace for the Oracle Application Express files user.
- `tablespace_temp` is the name of the temporary tablespace or tablespace group.
- `images` is the virtual directory for Oracle Application Express images. For installations using EPG, `/i/` is the required value for the images argument. To support future Oracle Application Express upgrades, define the virtual image directory as `/i/`.

Example:

```
@apexins.sql SYSAUX SYSAUX TEMP /i/
```

 **Note:**

If you receive the following error, exit SQL*Plus and change your working directory to where you unzipped the installation file, for example `C:\TEMP` in Windows, before starting SQL*Plus:

```
SP2-0310: unable to open file "apexins.sql"
```

- **Runtime environment.** Run `apxrtins.sql` passing the following arguments in the order shown:

```
@apxrtins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- `tablespace_apex` is the name of the tablespace for the Oracle Application Express application user.

- `tablespace_files` is the name of the tablespace for the Oracle Application Express files user.
- `tablespace_temp` is the name of the temporary tablespace or tablespace group.
- `images` is the virtual directory for Oracle Application Express images. To support future Oracle Application Express upgrades, define the virtual image directory as `/i/`.

Example:

```
@apxrtins.sql SYSAUX SYSAUX TEMP /i/
```

When Application Express installs, it creates the following database accounts:

- `APEX_190200` - This account owns the Application Express schema and metadata.
- `FLows_FILES` - This account owns the Application Express uploaded files.
- `APEX_PUBLIC_USER` - This minimally privileged account is used for Application Express configuration with Oracle REST Data Services or Oracle HTTP Server and `mod_plsql`.
- `APEX_INSTANCE_ADMIN_USER` - A minimally privileged account used for the REST Administration Interface.

 **Tip:**

To learn more about the REST Administration Interface, see "About the REST Administration Interface" in *Oracle Application Express Administration Guide*

If you configured RESTful Web services, then these additional accounts will be created:

- `APEX_REST_PUBLIC_USER` - The account used when invoking RESTful Services definitions stored in Oracle Application Express.
- `APEX_LISTENER` - The account used to query RESTful Services definitions stored in Oracle Application Express.

If you are upgrading from a previous release, then `FLows_FILES` already exists and `APEX_PUBLIC_USER` is created if it does not already exist.

 **See Also:**

- "Configuring Password Protection" in *Oracle Database Security Guide*
- "[About the Oracle Application Express Runtime Environment](#)"
- "*SQL*Plus User's Guide and Reference*" for more information about SQL*Plus
- Oracle Application Express must be installed from a writable directory on the file system. See "[Reviewing a Log of an Installation Session](#)"

B.3.2 Creating or Updating Your Instance Administration Account

Learn how to create or update your instance administrator account.

This section describes how to create or update your Instance Administrator account.

Tip:

Skip this section if you are upgrading from a previous release of Oracle Application Express. In an upgrade scenario, the Instance Administrator account and password is preserved and carried over from the prior release.

- [What Is an Instance Administrator?](#)
Instance administrators are superusers that are responsible for managing an entire Oracle Application Express instance, including managing workspace provisioning, configuring features and instance settings, and managing security.
- [About apxchpwd.sql](#)
Run the `apxchpwd.sql` script to create or update your Instance Administrator account.
- [Running apxchpwd.sql](#)
Run the `apxchpwd.sql` script to create and update your Instance Administrator account.

B.3.2.1 What Is an Instance Administrator?

Instance administrators are superusers that are responsible for managing an entire Oracle Application Express instance, including managing workspace provisioning, configuring features and instance settings, and managing security.

To perform these tasks, an Instance administrator signs in to the Oracle Application Express Administration Services application.

See Also:

"Oracle Application Express Administration Services" in *Oracle Application Express Administration Guide*

B.3.2.2 About apxchpwd.sql

Run the `apxchpwd.sql` script to create or update your Instance Administrator account.

Running the `apxchpwd.sql` script enables you to create or update your Instance Administrator account. You must run the `apxchpwd.sql` script in the following scenarios:

- **New Oracle Application Express installations** - Run `apxchpwd.sql` to create an Instance Administrator account and password.

- **Converting of a runtime environment to a development environment** - Run `apxchpwd.sql` to change the Instance Administrator account password.
- **Changing Your Instance Administrator Password** -Run `apxchpwd.sql` to change the password for an existing Instance Administrator account.
- **Unlocking Your Instance Administrator Account** - Run `apxchpwd.sql` to unlock an existing Instance Administrator account.

 **Tip:**

You do not need to run `apxchpwd.sql` when upgrading from a previous release of Oracle Application Express. In an upgrade scenario, the Instance Administrator account password is preserved and carried over from the prior release.

B.3.2.3 Running `apxchpwd.sql`

Run the `apxchpwd.sql` script to create and update your Instance Administrator account.

To create or update your Instance Administrator account:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxchpwd.sql`. For example:

```
@apxchpwd.sql
```

Follow the on-screen instructions. You will be prompted provide a username, password, and email address. If the account username does not exist, it will be created for you.

 **See Also:**

*SQL*Plus User's Guide and Reference* for more information about SQL*Plus

B.3.3 Restarting Processes

Restart the processes that you stopped before you began the installation.

After you install Oracle Application Express, you must restart the processes that you stopped before you began the installation. In addition, restart Oracle HTTP Server.

B.3.4 Configuring the APEX_PUBLIC_USER Account

It is important to correctly configure the APEX_PUBLIC_USER account to enable proper operation of Oracle Application Express.

- [About the APEX_PUBLIC_USER Account](#)
Learn about the random password that is created upon installation in APEX_PUBLIC_USER account.
- [Unlocking the APEX_PUBLIC_USER Account](#)
Unlock the APEX_PUBLIC_USER account by running a SQL statement.
- [Changing the Password for the APEX_PUBLIC_USER Account](#)
Change the password for the APEX_PUBLIC_USER account by running a SQL statement.
- [About Password Expiration in Oracle Database 11g](#)
Set PASSWORD_LIFE_TIME parameter to unlimited by altering APEX_PUBLIC_USER to prevent password expiration.

B.3.4.1 About the APEX_PUBLIC_USER Account

Learn about the random password that is created upon installation in APEX_PUBLIC_USER account.

The APEX_PUBLIC_USER account is created with a random password in a new installation of Oracle Application Express. You must change the password for this account before configuring the database access descriptor (DAD) in a new installation.

B.3.4.2 Unlocking the APEX_PUBLIC_USER Account

Unlock the APEX_PUBLIC_USER account by running a SQL statement.

 **Tip:**

If you are upgrading from a prior release of Oracle Application Express, this step is unnecessary.

To unlock the APEX_PUBLIC_USER account:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. If you are configuring an Oracle Database 12c or later CDB, make sure you are connected to the root container, `CDB$ROOT` when completing this step.

For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following statement:

```
ALTER USER APEX_PUBLIC_USER ACCOUNT UNLOCK
```

B.3.4.3 Changing the Password for the APEX_PUBLIC_USER Account

Change the password for the `APEX_PUBLIC_USER` account by running a SQL statement.



Tip:

If you are upgrading from a prior release of Oracle Application Express, this step is unnecessary.

To change the password for the `APEX_PUBLIC_USER` account:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. If you are configuring an Oracle Database 12c or later CDB, make sure you are connected to the root container, `CDB$ROOT` when completing this step.

For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following statement:

```
ALTER USER APEX_PUBLIC_USER IDENTIFIED BY new_password
```

Where *new_password* is the new password you are setting for `APEX_PUBLIC_USER`. You will use this password when creating the DAD in the sections that follow.

B.3.4.4 About Password Expiration in Oracle Database 11g

Set `PASSWORD_LIFE_TIME` parameter to unlimited by altering `APEX_PUBLIC_USER` to prevent password expiration.

In the default profile in Oracle Database 11g, the parameter `PASSWORD_LIFE_TIME` is set to 180. If you are using Oracle Database 11g with Oracle Application Express, this causes the password for `APEX_PUBLIC_USER` to expire in 180 days. As a result, your Oracle Application Express instance will become unusable until you change the password.

To prevent this behavior, create another profile in which the `PASSWORD_LIFE_TIME` parameter is set to unlimited and alter the `APEX_PUBLIC_USER` account and assign it to the new profile.



See Also:

Oracle Database Security Guide for information on creating profiles and assigning them to database users.

B.4 Configuring the Oracle HTTP Server

Configuring Oracle HTTP Server requires that you copy the images directory, edit the `dads.conf` file, configure static files support, stop and restart Oracle HTTP Server, disable Oracle XML DB Protocol Server, and configuring Oracle HTTP server behind a reverse proxy or load balancer.

This section describes how to configure Oracle HTTP Server with `mod_plsql`.

- [About Copying the Images Directory](#)
Learn about the importance of copying the images directory.
- [Copying the Images Directory](#)
Learn how to copy images directory.
- [Editing the dads.conf File](#)
Learn how to edit `dads.conf` file.
- [Configuring Static File Support](#)
Learn how to configure static file support in Oracle Application Express.
- [Stopping and Restarting Oracle HTTP Server](#)
Learn how to stop and restart Oracle HTTP Server.

- [Disabling Oracle XML DB Protocol Server](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role and run the EXEC DBMS_XDB.SETHTTPPORT(0); command.
- [About Configuring Oracle HTTP Server Behind a Reverse Proxy or Load Balancer](#)
Learn about configuring Oracle HTTP server behind a reverse proxy or load balancer.

B.4.1 About Copying the Images Directory

Learn about the importance of copying the images directory.

Whether you are loading a new installation or upgrading from a previous release, you must copy the `images` directory from the top level of the `apex\images` directory, for example `C:\TEMP`, to the location on the file system containing the Oracle home for Oracle HTTP Server.

During an upgrade, you overwrite your existing images directory. Before you begin the upgrade, to ensure that you can revert to the previous version, Oracle recommends that you create a copy of your existing `images` directory for Oracle Application Express, indicating the release number of the images (for example, `images_3_1`).

B.4.2 Copying the Images Directory

Learn how to copy images directory.

To locate the `images` directory on the file system, review the following files for the text alias `/i/`:

- Oracle HTTP Server distributed Oracle9i Release 2—see the `httpd.conf` file.
- Oracle Application Server 10g—see the `marvel.conf` or `dads.conf` files.
- Oracle HTTP Server distributed with Oracle Database 11g or 12c —see the `marvel.conf` or `dads.conf` files.

When you locate the `images` directory path, rename the existing `images` directory to a backup location. Doing so enables you to revert to the previous release, if that becomes necessary.

Use the following command syntax to copy the `apex\images` directory from the Oracle Database home to a path in the Oracle HTTP Server home:

- Oracle HTTP Server distributed with Oracle Database 12c or later:
 - Windows:

```
xcopy /E /I APEX_HOME\apex\images ORACLE_HTTPSERVER_HOME
\user_projects\domains\base_domain\images
```

- UNIX and Linux:

```
cp -rf $APEX_HOME/apex/images ORACLE_HTTPSERVER_HOME/user_projects/
domains/base_domain
```

In the preceding syntax examples:

- `APEX_HOME` is the location of where the installation Zip file was unzipped
- `ORACLE_HTTPSERVER_HOME` is the existing Oracle Application Server or Oracle HTTP Server Oracle home

B.4.3 Editing the `dads.conf` File

Learn how to edit `dads.conf` file.

The `dads.conf` file contains the information about the Database Access Descriptor (DAD) to access Oracle Application Express.

Note:

Within the context of this document, `ORACLE_HTTPSERVER_HOME` is the location where Oracle HTTP Server is installed.

To edit the `dads.conf` file:

1. Use a text editor and open the `dads.conf`.

- Oracle Application Server 10g:

- Windows:

```
ORACLE_HTTPSERVER_HOME\Apache\modplsql\conf\dads.conf
```

- UNIX and Linux:

```
ORACLE_HTTPSERVER_HOME/Apache/modplsql/conf/dads.conf
```

- Oracle HTTP Server distributed with Oracle Database 11g:

- Windows:

```
ORACLE_INSTANCE\config\OHS/ohs<1>\mod_plsql
```

- UNIX and Linux:

```
ORACLE_INSTANCE/config/OHS/ohs<1>/mod_plsql
```

- Oracle HTTP Server 12c:

- Windows:

```
ORACLE_HTTPSERVER_HOME\user_projects\domains\base_domain\config\fmwconfig\components\OHS\instances\ohs1\mod_plsql\dads.conf
```

- UNIX and Linux:

```
ORACLE_HTTPSERVER_HOME/user_projects/domains/base_domain/config/fmwconfig/components/OHS/instances/ohs1/mod_plsql/dads.conf
```

 **Note:**

In the examples above, ohs1 refers to the name of the Oracle HTTP Server instance. When running the Oracle HTTP Server Configuration Wizard, the default name is ohs1. It may have been changed to a name other than ohs1.

2. In the `dads.conf` file, replace `ORACLE_HTTPSERVER_HOME`, `host`, `port`, `service_name`, and `apex_public_user_password` with values appropriate for your environment. Note that the `apex_public_user_password` is the password you changed in `APEX_PUBLIC_USER` Account.

Note that the path listed is only an example. The path in the `dads.conf` file should reference the file system path described in About Copying the Images Directory.

```
Alias /i/ "ORACLE_HTTPSERVER_HOME/user_projects/domains/base_domain/
images/"
```

```
<Location /pls/apex>
  Order deny,allow
  AllowOverride None
  PlsqlDatabaseConnectString      host:port:service_name ServiceNameFormat
  PlsqlNLSLanguage                AMERICAN_AMERICA.AL32UTF8
  PlsqlAuthenticationMode        Basic
  SetHandler                      pls_handler
  PlsqlDocumentTablename          wwv_flow_file_objects$
  PlsqlDatabaseUsername           APEX_PUBLIC_USER
  PlsqlDefaultPage                apex
  PlsqlDatabasePassword           apex_public_user_password
  PlsqlRequestValidationFunction  wwv_flow_epg_include_modules.authorize
  PlsqlPathAlias                  r
  PlsqlPathAliasProcedure         wwv_flow.resolve_friendly_url
  Allow from all
</Location>
```

 **Note:**

`PlsqlPathAlias` and `PlsqlPathAliasProcedure` are required new parameters that must be added to an existing DAD. This is important when upgrading to Application Express release 5.0.

3. Locate the line containing `PlsqlNLSLanguage`.

The `PlsqlNLSLanguage` setting determines the language setting of the DAD. The character set portion of the `PlsqlNLSLanguage` value must be set to `AL32UTF8`, regardless of whether the database character set is `AL32UTF8`. For example:

```
...
PlsqlNLSLanguage                AMERICAN_AMERICA.AL32UTF8
...
```

4. Save and exit the `dads.conf` file.

 **See Also:**

- ["Changing the Password for the APEX_PUBLIC_USER Account"](#)
- ["About Copying the Images Directory"](#)

B.4.4 Configuring Static File Support

Learn how to configure static file support in Oracle Application Express.

Oracle Application Express enables application developers to include static files with their applications. Static files can be associated with a workspace, an application, a plug-in, or an application theme. When using the Oracle HTTP Server as your web listener, static files are served using Path Aliasing functionality of the `mod_plsql` plug-in.

 **See Also:**

["Editing the dads.conf File"](#) for the required configuration parameters

B.4.5 Stopping and Restarting Oracle HTTP Server

Learn how to stop and restart Oracle HTTP Server.

This section describes how to stop and restart Oracle HTTP Server.

- [Stopping and Restarting Oracle HTTP Server with Oracle Database 10g and Lower](#)
Learn how to stop and restart Oracle HTTP Server.
- [Stopping and Restarting Oracle HTTP Server 12c](#)
Learn how to stop and restart the Oracle HTTP Server distributed with Oracle Database 12c or later.

B.4.5.1 Stopping and Restarting Oracle HTTP Server with Oracle Database 10g and Lower

Learn how to stop and restart Oracle HTTP Server.

 **Note:**

Within the context of this document, `ORACLE_HTTPSERVER_HOME` is the location where Oracle HTTP Server is installed.

To stop and restart Oracle HTTP Server:

- Windows:

Execute the following:

```
ORACLE_HTTPSERVER_HOME\opmn\bin\opmnctl stopproc ias-
component=HTTP_Server
ORACLE_HTTPSERVER_HOME\opmn\bin\opmnctl startproc ias-
component=HTTP_Server
```

- UNIX and Linux

Execute the following:

```
ORACLE_HTTPSERVER_HOME/opmn/bin/opmnctl stopproc ias-
component=HTTP_Server
ORACLE_HTTPSERVER_HOME/opmn/bin/opmnctl startproc ias-
component=HTTP_Server
```

 **Note:**

If you are using Oracle HTTP Server that is included with Oracle Fusion Middleware 11.1.1, you may need to replace the `ias-component` name above if the name given to the component was something other than `HTTP_Server` when it was installed. The default name is `ohs1`.

B.4.5.2 Stopping and Restarting Oracle HTTP Server 12c

Learn how to stop and restart the Oracle HTTP Server distributed with Oracle Database 12c or later.

To stop and restart the Oracle HTTP Server distributed with Oracle Database 12c or later, enter commands using the following syntax, where `DOMAIN_HOME` is the path to the Oracle HTTP Server domain:

- Windows:

Execute the following:

```
%DOMAIN_HOME%\bin\stopComponent.cmd componentName
%DOMAIN_HOME%\bin\startComponent.cmd componentName
```

- UNIX and Linux

Execute the following:

```
$DOMAIN_HOME/bin/stopComponent.sh componentName
$DOMAIN_HOME/bin/startComponent.sh componentName
```

In a default installation, `componentName` is `ohs1` but may have been changed when running the Oracle HTTP Server Configuration Wizard.

B.4.6 Disabling Oracle XML DB Protocol Server

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `EXEC DBMS_XDB.SETHTTPPORT(0);` command.

If you were previously using the Embedded PL/SQL Gateway and now want to use Oracle HTTP server and `mod_plsql`, you need to disable the Oracle XML DB Protocol Server.

To disable Oracle XML DB Protocol Server:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following command:

```
EXEC DBMS_XDB.SETHTTPPORT(0);
```

B.4.7 About Configuring Oracle HTTP Server Behind a Reverse Proxy or Load Balancer

Learn about configuring Oracle HTTP server behind a reverse proxy or load balancer.

When Oracle Application Express is running behind a reverse proxy or load balancer, it is important to communicate the original HTTP hostname and protocol as seen by user's browser to the Application Express engine. The engine will use this information to generate valid URL in HTML responses and HTTP redirects that user's browser can successfully follow. For Oracle HTTP server and `mod_plsql`, use `PlsqlCGIEnvironmentList` directive in the `dads.conf` configuration file. For example:

```
PlsqlCGIEnvironmentList REQUEST_PROTOCOL=https
PlsqlCGIEnvironmentList HTTP_HOST=www.example.com:443
```



See Also:

["Editing the dads.conf File"](#)

B.5 Enabling Network Services in Oracle Database 11g or Later

You must enable network services in Oracle Database 11g or later versions to send outbound mail, use Web services, or use PDF report printing in Oracle Application Express.

- [When and Why Network Services Must be Enabled](#)
Enabling network services enables support for sending outbound mail in Oracle Application Express, use of Web services in Oracle Application Express, and PDF report printing.
- [Granting Connect Privileges Prior to Oracle Database 12c](#)
Demonstrates how to grant connect privileges to any host for the APEX_190200 database user.
- [Granting Connect Privileges in Oracle Database 12c or Later](#)
Procedures CREATE_ACL, ASSIGN_ACL, ADD_PRIVILEGE and CHECK_PRIVILEGE in DBMS_NETWORK_ACL_ADMIN are deprecated in Oracle Database 12c. Oracle recommends to use APPEND_HOST_ACE.
- [Troubleshooting an Invalid ACL Error](#)
Learn how to identify any invalid ACL error by running the query.

B.5.1 When and Why Network Services Must be Enabled

Enabling network services enables support for sending outbound mail in Oracle Application Express, use of Web services in Oracle Application Express, and PDF report printing.

By default, the ability to interact with network services is disabled in Oracle Database 11g Release 2 or later. Therefore, if you are running Oracle Application Express with Oracle Database 11g Release 2 or later, you must use the new DBMS_NETWORK_ACL_ADMIN package to grant connect privileges to any host for the APEX_190200 database user. Failing to grant these privileges results in issues with:

- Sending outbound mail in Oracle Application Express.
Users can call methods from the APEX_MAIL package, but issues arise when sending outbound email.
- Using Web services in Oracle Application Express.
- PDF report printing.

Note:

When upgrading Application Express on a database 12c or newer, based on the configuration of the old Application Express version the upgrade automatically configures Network Services.

 **Tip:**

To run the examples described in this section, the compatible initialization parameter of the database must be set to at least 11.1.0.0.0. By default an 11g or 12c database will already have the parameter set properly, but a database upgraded to 11g or 12c from a prior version may not. For information about changing database initialization parameters, see "Creating and Configuring an Oracle Database" in *Oracle Database Administrator's Guide*.

B.5.2 Granting Connect Privileges Prior to Oracle Database 12c

Demonstrates how to grant connect privileges to any host for the APEX_190200 database user.

The following example demonstrates how to grant connect privileges to any host for the APEX_190200 database user. This example assumes you connected to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role.

```

DECLARE
    ACL_PATH VARCHAR2(4000);
BEGIN
    -- Look for the ACL currently assigned to '*' and give APEX_190200
    -- the "connect" privilege if APEX_190200 does not have the privilege
    yet.

    SELECT ACL INTO ACL_PATH FROM DBA_NETWORK_ACLS
        WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

    IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE(ACL_PATH, 'APEX_190200',
        'connect') IS NULL THEN
        DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE(ACL_PATH,
        'APEX_190200', TRUE, 'connect');
    END IF;

EXCEPTION
    -- When no ACL has been assigned to '*'.
    WHEN NO_DATA_FOUND THEN
        DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('power_users.xml',
        'ACL that lets power users to connect to everywhere',
        'APEX_190200', TRUE, 'connect');
        DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('power_users.xml', '*');
END;
/
COMMIT;

```

The following example demonstrates how to provide less privileged access to local network resources. This example enables access to servers on the local host only, such as email and report servers.

```

DECLARE
  ACL_PATH VARCHAR2(4000);
BEGIN
  -- Look for the ACL currently assigned to 'localhost' and give
  APEX_190200
  -- the "connect" privilege if APEX_190200 does not have the privilege
  yet.
  SELECT ACL INTO ACL_PATH FROM DBA_NETWORK_ACLS
  WHERE HOST = 'localhost' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

  IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE(ACL_PATH, 'APEX_190200',
  'connect') IS NULL THEN
    DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE(ACL_PATH,
    'APEX_190200', TRUE, 'connect');
  END IF;

EXCEPTION
  -- When no ACL has been assigned to 'localhost'.
  WHEN NO_DATA_FOUND THEN
    DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('local-access-users.xml',
    'ACL that lets users to connect to localhost',
    'APEX_190200', TRUE, 'connect');
    DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('local-access-users.xml','localhost');
END;
/
COMMIT;

```

B.5.3 Granting Connect Privileges in Oracle Database 12c or Later

Procedures `CREATE_ACL`, `ASSIGN_ACL`, `ADD_PRIVILEGE` and `CHECK_PRIVILEGE` in `DBMS_NETWORK_ACL_ADMIN` are deprecated in Oracle Database 12c. Oracle recommends to use `APPEND_HOST_ACE`.

The following example demonstrates how to grant connect privileges to any host for the `APEX_190200` database user. This example assumes you connected to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role.

```

BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host => '*',
    ace => xs$ace_type(privilege_list => xs$name_list('connect'),
    principal_name => 'APEX_190200',
    principal_type => xs_acl.ptype_db));
END;
/

```

The following example demonstrates how to provide less privileged access to local network resources. This example enables access to servers on the local host only, such as email and report servers.

```
BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host => 'localhost',
    ace => xs$ace_type(privilege_list => xs$name_list('connect'),
                      principal_name => 'APEX_190200',
                      principal_type => xs_acl.ptype_db));
END;
/
```

B.5.4 Troubleshooting an Invalid ACL Error

Learn how to identify any invalid ACL error by running the query.

If you receive an ORA-44416: Invalid ACL error after running the previous script, use the following query to identify the invalid ACL:

```
REM Show the dangling references to dropped users in the ACL that is
assigned
REM to '*'.

SELECT ACL, PRINCIPAL
  FROM DBA_NETWORK_ACLS NACL, XDS_ACE ACE
 WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL AND
        NACL.ACLID = ACE.ACLID AND
        NOT EXISTS (SELECT NULL FROM ALL_USERS WHERE USERNAME = PRINCIPAL);
```

Next, run the following code to fix the ACL:

```
DECLARE
  ACL_ID  RAW(16);
  CNT     NUMBER;
BEGIN
  -- Look for the object ID of the ACL currently assigned to '*'
  SELECT ACLID INTO ACL_ID FROM DBA_NETWORK_ACLS
     WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

  -- If just some users referenced in the ACL are invalid, remove just
those
  -- users in the ACL. Otherwise, drop the ACL completely.
  SELECT COUNT(PRINCIPAL) INTO CNT FROM XDS_ACE
     WHERE ACLID = ACL_ID AND
           EXISTS (SELECT NULL FROM ALL_USERS WHERE USERNAME = PRINCIPAL);

  IF (CNT > 0) THEN

    FOR R IN (SELECT PRINCIPAL FROM XDS_ACE
              WHERE ACLID = ACL_ID AND
                    NOT EXISTS (SELECT NULL FROM ALL_USERS
                                WHERE USERNAME = PRINCIPAL)) LOOP
```

```
UPDATE XDB.XDB$ACL
  SET OBJECT_VALUE =
        DELETEXML(OBJECT_VALUE,
            '/ACL/ACE[PRINCIPAL="' || R.PRINCIPAL || '"]')
  WHERE OBJECT_ID = ACL_ID;
END LOOP;

ELSE
  DELETE FROM XDB.XDB$ACL WHERE OBJECT_ID = ACL_ID;
END IF;

END;
/

REM commit the changes.

COMMIT;
```

Once the ACL has been fixed, you must run the first script in this section to apply the ACL to the APEX_190200 user.

B.6 Performing Security Tasks

Oracle recommends configuring and using Secure Sockets Layer (SSL) to ensure that passwords and other sensitive data are not transmitted in clear text in HTTP requests.

Without the use of SSL, passwords could potentially be exposed, compromising security.

SSL is an industry standard protocol that uses RSA public key cryptography in conjunction with symmetric key cryptography to provide authentication, encryption, and data integrity.



See Also:

"Configuring HTTP Protocol Attributes" in *Oracle Application Express Administration Guide*

B.7 Controlling the Number of Concurrent Jobs

Learn about specifying the number of concurrently running jobs.

- [About Managing the Number of Concurrent Jobs](#)
Learn about managing maximum number of concurrently running jobs.
- [Viewing the Number of JOB_QUEUE_PROCESSES](#)
You can view number of JOB_QUEUE_PROCESSES in three ways.
- [Changing the Number of JOB_QUEUE_PROCESSES](#)
You can change the number of JOB_QUEUE_PROCESSES by running a SQL statement in SQL*Plus.

B.7.1 About Managing the Number of Concurrent Jobs

Learn about managing maximum number of concurrently running jobs.

`JOB_QUEUE_PROCESSES` determine the maximum number of concurrently running jobs. In Oracle Application Express transactional support and SQL scripts require jobs. If `JOB_QUEUE_PROCESSES` is not enabled and working properly, you cannot successfully execute a script.

B.7.2 Viewing the Number of `JOB_QUEUE_PROCESSES`

You can view number of `JOB_QUEUE_PROCESSES` in three ways.

- [Viewing `JOB_QUEUE_PROCESSES` in the Installation Log File](#)
View `JOB_QUEUE_PROCESSES` in the installation log files.
- [Viewing `JOB_QUEUE_PROCESSES` in Oracle Application Express](#)
View the number of `JOB_QUEUE_PROCESSES` on the About Application Express page.
- [Viewing `JOB_QUEUE_PROCESSES` from SQL*Plus](#)
View the number of `JOB_QUEUE_PROCESSES` from SQL*Plus.

B.7.2.1 Viewing `JOB_QUEUE_PROCESSES` in the Installation Log File

View `JOB_QUEUE_PROCESSES` in the installation log files.



See Also:

["Reviewing a Log of an Installation Session"](#)

B.7.2.2 Viewing `JOB_QUEUE_PROCESSES` in Oracle Application Express

View the number of `JOB_QUEUE_PROCESSES` on the About Application Express page.

To view the About Application Express page:

1. Sign in to Oracle Application Express.
2. Locate the Help menu at the top of the page.
3. From the Help menu, select **About**.
The About Application Express page appears.
4. Scroll down and find `JOB_QUEUE_PROCESSES` at the bottom of the page.



See Also:

["Signing into a Workspace Using Oracle REST Data Services"](#)

B.7.2.3 Viewing JOB_QUEUE_PROCESSES from SQL*Plus

View the number of JOB_QUEUE_PROCESSES from SQL*Plus.

Run the following SQL statement:

```
SELECT VALUE FROM v$parameter WHERE NAME = 'job_queue_processes'
```

B.7.3 Changing the Number of JOB_QUEUE_PROCESSES

You can change the number of JOB_QUEUE_PROCESSES by running a SQL statement in SQL*Plus.

To update the number of JOB_QUEUE_PROCESSES:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

2. In SQL*Plus run the following SQL statement:

```
ALTER SYSTEM SET JOB_QUEUE_PROCESSES = <number>
```

For example, running the statement ALTER SYSTEM SET JOB_QUEUE_PROCESSES = 20 sets JOB_QUEUE_PROCESSES to 20.

B.8 About Running Oracle Application Express in Other Languages

You can install a single instance of Oracle Application Express with one or more of translated versions.

The Oracle Application Express interface is translated into German, Spanish, French, Italian, Japanese, Korean, Brazilian Portuguese, Simplified Chinese, and Traditional Chinese. A single instance of Oracle Application Express can be installed with one or more of these translated versions. Developers can choose to run the Application Express development environment in any of the installed languages by simply selecting the language from the App Builder log in screen or home page.

In order to install other languages you must use the apex_19.2.zip file which contains the extra files referenced below. If you previously downloaded apex_19.2_en.zip, then

you do not need to re-install Oracle Application Express. Simply download `apex_19.2.zip` and unzip the file into the same directory where you unzipped `apex_19.2_en.zip`.

The translated version of Oracle Application Express should be loaded into a database that has a character set that supports the specific language. If you attempt to install a translated version of Oracle Application Express into a database that does not support the character encoding of the language, the installation may fail or the translated Oracle Application Express instance may appear corrupt when run. The database character set `AL32UTF8` supports all the translated versions of Oracle Application Express.

You can manually install translated versions of Oracle Application Express using `SQL*Plus`. The installation files are encoded in `AL32UTF8`.

 **Note:**

Regardless of the target database character set, to install a translated version of Oracle Application Express, you must set the character set value of the `NLS_LANG` environment variable to `AL32UTF8` before starting `SQL*Plus`.

The following examples illustrate valid `NLS_LANG` settings for loading Oracle Application Express translations:

```
American_America.AL32UTF8  
Japanese_Japan.AL32UTF8
```

B.9 Installing Translated Versions of Oracle Application Express

Learn about installing translated versions of Oracle Application Express.

- [About Installing Translated Versions of Oracle Application Express](#)
You must run `load_lang.sql` script to run a translated version of Oracle Application Express.
- [Installing a Translated Version of Oracle Application Express](#)
Learn how to install a translated version of Oracle Application Express.

B.9.1 About Installing Translated Versions of Oracle Application Express

You must run `load_lang.sql` script to run a translated version of Oracle Application Express.

Whether you are installing for the first time or upgrading from a previous release, you must run the `load_lang.sql` script to run a translated version of Oracle Application Express.

The installation scripts are located in subdirectories identified by a language code in the unzipped distribution `apex/builder`. For example, the German version is located in `apex/builder/de` and the Japanese version is located in `apex/builder/ja`. Within each directory, there is a language loading script identified by the language code (for example, `load_de.sql` or `load_ja.sql`).

B.9.2 Installing a Translated Version of Oracle Application Express

Learn how to install a translated version of Oracle Application Express.

To install a translated version of Oracle Application Express:

1. Set the `NLS_LANG` environment variable, making sure that the character set is `AL32UTF8`. For example:

- Bourne or Korn shell:

```
NLS_LANG=American_America.AL32UTF8
export NLS_LANG
```

- C shell:

```
setenv NLS_LANG American_America.AL32UTF8
```

- For Windows based systems:

```
set NLS_LANG=American_America.AL32UTF8
```

2. Navigate to the directory under `apex/builder` based on the language you need to install. For example for German, navigate to `apex/builder/de`. Start `SQL*Plus` and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following statement:

```
ALTER SESSION SET CURRENT_SCHEMA = APEX_190200;
```

4. Execute the appropriate language specific script. For example:

```
@load_lang.sql
```

Where `lang` is the specific language (for example, `load_de.sql` for German or `load_ja.sql` for Japanese).

B.10 Creating a Workspace and Adding Oracle Application Express Users

Learn about creating workspace and adding Oracle Application Express users.

Before you can develop or install applications, you must create a workspace, add Oracle Application Express users and sign in to your workspace.

- [About Workspaces and Users](#)
A workspace enables multiple users to work within the same Oracle Application Express installation while keeping their objects, data, and applications private.
- [Accessing Administration Services Using Oracle HTTP Server](#)
Learn how to access Oracle Application Express administration services.
- [Creating Oracle Application Express Users](#)
Create new users in Oracle Application Express Administration Services..
- [Signing into a Workspace Using the Oracle HTTP Server](#)
Learn how to sign in to a workspace using the Oracle HTTP Server.

B.10.1 About Workspaces and Users

A workspace enables multiple users to work within the same Oracle Application Express installation while keeping their objects, data, and applications private.

You access the Oracle Application Express home page by logging in to a workspace using a JavaScript enabled Web browser.

Each workspace has a unique ID and name. An instance administrator can create a workspace manually within Oracle Application Express Administration Services or have users submit requests. Oracle Application Express Administration Services is a separate application for managing an entire Oracle Application Express instance.

See Also:

- ["Browser Requirements"](#)
- ["Creating Workspaces" in *Oracle Application Express Administration Guide*](#)
- ["Managing Workspace and Change Requests" in *Oracle Application Express Administration Guide*](#)

B.10.2 Accessing Administration Services Using Oracle HTTP Server

Learn how to access Oracle Application Express administration services.

To manually create a workspace and user accounts, you sign in to a separate application for managing an entire Oracle Application Express instance called Oracle

Application Express Administration Services. You sign in to Oracle Application Express Administration Services using the `ADMIN` account and password created or reset during the installation process.

To access Oracle Application Express Administration Services:

1. In a Web browser, navigate to the Oracle Application Express Administration Services application.

Because your setup uses Oracle HTTP Server with `mod_plsql` go to:

```
http://hostname:port/pls/apex/apex_admin
```

Where:

`hostname` is the name of the system where Oracle HTTP Server is installed.

`port` is the port number assigned to HTTP on the Oracle HTTP Server. In a default installation, this number is 7777.

`pls` is the indicator to use the `mod_plsql` cartridge.

`apex` is the database access descriptor (DAD) defined in the `mod_plsql` configuration file.

2. On the Sign In page:
 - Username - Enter `admin`.
 - Password - Enter the Oracle Application Express administrator account password you specified in your instance administration account.
 - Click **Sign In to Administration**.

Note that, depending on your setup, you might be required to change your password when you log in for the first time.

- [Creating a Workspace Manually](#)
Sign in to Oracle Application Express Administration Services to create workspace manually.



See Also:

- ["Creating or Updating Your Instance Administration Account"](#)
- "Oracle Application Express Administration Services" in *Oracle Application Express Administration Guide*

B.10.2.1 Creating a Workspace Manually

Sign in to Oracle Application Express Administration Services to create workspace manually.

To manually create a workspace you sign in to a separate application called Oracle Application Express Administration Services. You sign in to Oracle Application Express Administration Services using the `ADMIN` account and password created or reset during the installation process.

To create an Oracle Application Express workspace manually:

1. Access Oracle Application Express Administration Services.
Oracle Application Express Administration Services appears. Next, create a workspace.
2. Click **Manage Workspaces**.
3. Under Workspace Actions, click **Create Workspace**.
The Create Workspace Wizard appears.
4. For Identify Workspace, enter the following:
 - a. Workspace Name - Enter a unique workspace name.
 - b. Workspace ID - Leave Workspace ID blank to have the new Workspace ID automatically generated. A Workspace ID must be a positive integer greater than 100000.
 - c. Workspace Description - Enter a workspace description.
 - d. Click **Next**.
5. For Identify Schema, specify whether you are re-using an existing schema or creating a new one.
If you are using an existing schema:
 - a. For Re-use existing schema, select **Yes**.
 - b. Select a schema from the list.
 - c. Click **Next**.
If you are creating a new schema:
 - a. For Re-use existing schema, select **No**.
 - b. Enter a schema name and password.
 - c. Specify a space quota.
 - d. Click **Next**.
6. For Identify Administrator, enter the Workspace administrator information and click **Next**.
7. Confirm your selections and click **Create Workspace**.

 **See Also:**

- ["Accessing Administration Services Using Oracle REST Data Services"](#)
- "Oracle Application Express Administration Services" in *Oracle Application Express Administration Guide*

B.10.3 Creating Oracle Application Express Users

Create new users in Oracle Application Express Administration Services..

Create new users by signing into Oracle Application Express Administration Services using your administrator password.

To create an Oracle Application Express user account:

1. Sign into Oracle Application Express Administration Services.
Oracle Application Express Administration Services appears.
2. Click **Manage Workspaces**.
3. Under Workspace Actions, click **Manage Developers and Users**.
4. Click **Create User**.
5. Under User Attributes, enter the appropriate information. Fields marked with an asterisk are required.

 **Tip:**

Most attributes in Oracle Application Express include field-level Help. Attributes with field-level Help, have light gray icon that resembles a question mark (?). To view field-level Help, click the Help icon.

6. Under Account Privileges:
 - a. **Workspace** - Select a workspace from the list.
 - b. **Default Schemas** - Specify the default schema used for data browsing, application creation, and SQL script execution.

When using workspaces that have more than one schema available, this schema is the default. This setting does not control security, only the user's preference.
 - c. **User is an administrator** - Specify if this user should have workspace administrator privileges.

Administrators are given access to all components. Additionally, they can manage user accounts, groups, and development services. Components may not be available if they are switched off by Instance Administrators.
 - d. **User is a developer** - Specify if this user should have developer privileges.

Developers must have access to either App Builder, SQL Workshop, or both. These components may not be available if they are switched off by the Instance Administrator.
 - e. **App Builder Access** - Determines whether a developer has access to the App Builder.
 - f. **SQL Workshop Access** - Determines whether a developer has access to the SQL Workshop.
 - g. **Team Development Access** - Determines whether a developer has access to the Team Development.
 - h. **Set Account Availability** - Select **Locked** to prevent the account from being used. Select **Unlocked** to allow the account to be used.

If the user has exceeded the maximum log in failures allowed, specified in Workspace Preferences, then their account will be locked automatically.
7. Under Password:
 - **Password** - Enter a case sensitive password.

- **Confirm Password** - Enter the password again.
 - **Require Change of Password On First Use** - Select **No** to allow the user to use the same password until it expires. Select **Yes** to require the user to change the password immediately when logging in the first time.
8. Click **Create User** or **Create and Create Another**.



See Also:

["Accessing Administration Services Using Oracle REST Data Services"](#)

B.10.4 Signing into a Workspace Using the Oracle HTTP Server

Learn how to sign in to a workspace using the Oracle HTTP Server.

Use your credentials (that is, your workspace name, user name and password) to sign in to the workspace.

To sign in to your workspace:

1. In a Web browser, navigate to the Oracle Application Express Sign In page.

Because your setup uses Oracle HTTP Server (Apache) and `mod_plsqli`, go to:

```
http://hostname:port/pls/apex
```

Where:

- `hostname` is the name of the system where Oracle HTTP Server is installed.
- `port` is the port number assigned to HTTP on the Oracle HTTP Server. In a default installation, this number is 7777. You can find information about your Oracle HTTP Server installation's port number from either of the following files:

```
ORACLE_BASE\ORACLE_HOME\install\portlist.ini
ORACLE_BASE\ORACLE_HTTPSERVER_HOME\Apache\Apache\conf\httpd.conf
```

Be aware that if you change a port number, it is not updated in the `portlist.ini` file. You can only rely on this file immediately after installation.

- `pls` is the indicator to use the `mod_plsqli` cartridge.
- `apex` is the database access descriptor (DAD) defined in the `mod_plsqli` configuration file.

The Sign In page appears.

2. On the Sign In page, enter:
 - Workspace - Enter the name of your workspace.
 - Username - Enter your user name.
 - Password - Enter your case-sensitive password.
3. Click **Sign In**.

Note that, depending on your setup, you might be required to change your password when you log in for the first time.

 **See Also:**

- "Creating Workspaces" in *Oracle Application Express Administration Guide*
- "Managing Workspace and Change Requests" in *Oracle Application Express Administration Guide*

B.11 Performing Post Installation Tasks for Upgrade Installations

Remove schemas from prior Oracle Application Express installations.

Once you have verified that your upgrade installation was successful and all upgraded applications function properly, you should remove schemas from prior Oracle Application Express installations.

- [About Removing Prior Oracle Application Express Installations](#)
Learn about removing schemas from a prior installation by verifying if a prior installation exists.
- [Verifying if a Prior Installation Exists](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` and run the query.
- [Removing Schemas and SYS Objects from Prior Installations](#)
Start SQL*Plus and connect to database and execute a statement to remove schemas and `SYS` objects.
- [Fixing Invalid ACL in Oracle Database 11g or Later](#)
Learn how to fix an invalid ACL if you are running Oracle Database 11g or later.

 **See Also:**

["Upgrading from a Previous Oracle Application Express Release"](#)

B.11.1 About Removing Prior Oracle Application Express Installations

Learn about removing schemas from a prior installation by verifying if a prior installation exists.

The database users associated with schemas from prior installations are privileged users and should be removed when they are no longer necessary. Removing schemas from a prior installation is a two step process. First you verify if a prior installation exists and then you remove the schemas.

B.11.2 Verifying if a Prior Installation Exists

Start SQL*Plus and connect to the database where Oracle Application Express is installed as *SYS* and run the query.

To verify if a prior installation exists:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as *SYS*. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following query:

```
SELECT username FROM dba_users WHERE ( username LIKE 'FLOWS\_____' ESCAPE
'\ ' OR username LIKE 'APEX\_____' ESCAPE '\ ' ) AND USERNAME NOT IN
( SELECT SCHEMA FROM
dba_registry WHERE comp_id = 'APEX' );
```

If the results contain entries in the form *FLOWS_XXXXXX* or *APEX_XXXXXX* where *XXXXXX* represents six numbers, those entries are candidates for removal.

B.11.3 Removing Schemas and SYS Objects from Prior Installations

Start SQL*Plus and connect to database and execute a statement to remove schemas and *SYS* objects.

To remove schemas and *SYS* objects from prior installations:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as *SYS* specifying the *SYSDBA* role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Execute statements similar to the following example:

```
DROP USER APEX_190100 CASCADE;  
DROP PACKAGE SYS.WWV_DBMS_SQL_APEX_190100;
```

B.11.4 Fixing Invalid ACL in Oracle Database 11g or Later

Learn how to fix an invalid ACL if you are running Oracle Database 11g or later.

After following the instructions in "[About Removing Prior Oracle Application Express Installations](#)," you may need to fix an invalid ACL if you are running Oracle Database 11g or later and you enabled network services for the prior Oracle Application Express schema.

To fix an invalid ACL:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog  
SQL> CONNECT SYS as SYSDBA  
Enter password: SYS_password
```

3. Execute a statement similar to the following:

```
EXEC DBMS_NETWORK_ACL_ADMIN.DELETE_PRIVILEGE('power_users.xml',  
'APEX_040200');
```

B.12 About Performance Optimization Tasks

Learn about performance optimization.

Performance of web applications heavily depends on their size and how often a browser has to request static content like images, CSS and JavaScript files. To improve performance, most web servers support on-the-fly HTTP response compression and provide settings that enable you to configure how long browsers may cache a file before requesting it again. The HTTP response compression is usually implemented using gzip encoding (for example, `mod_deflate` in Oracle HTTP Server), while browser file caching is enabled by issuing Cache-Control HTTP response header (for example, using `mod_headers` in Oracle HTTP Server).

See your web server documentation to learn how to enable response compression and browser file caching. For optimal performance of the Oracle Application Express development environment and Oracle Application Express applications, Oracle

recommends enabling gzip compression of files in the virtual images directory (for example, `/i/`) and responses from the database access descriptor (for example, `/pls/apex`), as well as allowing browsers to cache files from the virtual images directory for at least 12 hours.

B.13 Converting Between Runtime and Full Development Environments

Learn how to convert between runtime and full development environments.

This section describes how to convert between runtime and full development environments.

- [About Runtime and Full Development Environments](#)
An Oracle Application Express runtime environment enables users to run a production application without supporting the ability to change or edit the application.
- [Converting a Runtime Environment to a Full Development Environment in a non-CDB or locally in a PDB](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdvins.sql`.
- [Converting a Full Development Environment to a Runtime Environment in a non-CDB or locally in a PDB](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrm.sql`.
- [Converting a Runtime Environment to a Full Development Environment in a CDB](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdvins_con.sql`.
- [Converting a Full Development Environment to a Runtime Environment in a CDB](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrm_con.sql`.

B.13.1 About Runtime and Full Development Environments

An Oracle Application Express runtime environment enables users to run a production application without supporting the ability to change or edit the application.

A runtime environment includes only the packages necessary to run your applications, making it a more hardened environment. It does not provide a web interface for administration.

You administer an Oracle Application Express runtime environment using SQL*Plus or SQL Developer and the `APEX_INSTANCE_ADMIN` API.

 **See Also:**

- ["About the Oracle Application Express Runtime Environment"](#) in *Oracle Application Express Administration Guide*
- ["Installing Exported Applications into a Runtime Environment"](#) in *Oracle Application Express Administration Guide*

B.13.2 Converting a Runtime Environment to a Full Development Environment in a non-CDB or locally in a PDB

Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role and run the `apxdvins.sql`.

To convert an Oracle Application Express runtime environment to a full development environment in a non-CDB or locally in a PDB:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxdvins.sql`. For example:

```
@apxdvins.sql
```

4. Follow the instructions in [Creating or Updating Your Instance Administration Account](#).

 **See Also:**

- ["Creating or Updating Your Instance Administration Account."](#)
- ["SQL*Plus User's Guide and Reference"](#) for more information about SQL*Plus

B.13.3 Converting a Full Development Environment to a Runtime Environment in a non-CDB or locally in a PDB

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrm.sql`.

Tip:

Before converting a full development environment to a runtime environment, export all Websheet applications. When you convert a development environment to runtime, existing Websheet data grids and reports will be deleted from all workspaces. Converting the environment back to a full development environment will not restore these Websheet data grids or reports.

To convert an Oracle Application Express full development environment to a runtime environment in a non-CDB or locally in a PDB:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxdevrm.sql`. For example:

```
@apxdevrm.sql
```

See Also:

*SQL*Plus User's Guide and Reference* for more information about SQL*Plus

B.13.4 Converting a Runtime Environment to a Full Development Environment in a CDB

Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role and run the `apxdvins_con.sql`.

To convert an Oracle Application Express runtime environment to a full development environment in a CDB:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxdvins_con.sql`. For example:

```
@apxdvins_con.sql
```

4. Follow the instructions in [Creating or Updating Your Instance Administration Account](#).

See Also:

- ["Creating or Updating Your Instance Administration Account."](#)
- ["SQL*Plus User's Guide and Reference"](#) for more information about SQL*Plus

B.13.5 Converting a Full Development Environment to a Runtime Environment in a CDB

Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and run the `apxdevrm_con.sql`.

Tip:

Before converting a full development environment to a runtime environment, export all Websheet applications. When you convert a development environment to runtime, existing Websheet data grids and reports will be deleted from all workspaces. Converting the environment back to a full development environment will not restore these Websheet data grids or reports.

To convert an Oracle Application Express full development environment to a runtime environment in a CDB:

1. Change your working directory to the `apex` directory where you unzipped the installation software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run `apxdevrm_con.sql`. For example:

```
@apxdevrm_con.sql
```

See Also:

*SQL*Plus User's Guide and Reference* for more information about SQL*Plus

C

Automating the Oracle Application Express Installation Process

Automate the process of installing and configuring an Oracle Application Express instance.

- [About apxsilentins.sql](#)
Run the `apxsilentins.sql` script to automate the installation and configuration of an Oracle Application Express instance.
- [Running apxsilentins.sql](#)
Run the `apxsilentins.sql` script.

C.1 About `apxsilentins.sql`

Run the `apxsilentins.sql` script to automate the installation and configuration of an Oracle Application Express instance.

Traditionally you run the `apexins.sql` script to install Oracle Application Express and then perform a multiple other steps to configure the `APEX_PUBLIC_USER` account. The `apxsilentins.sql` script simplifies the installation and configuration process. `apxsilentins.sql` accepts additional parameters so that passwords can be passed for following database users associated with the Oracle Application Express schema: `APEX_PUBLIC_USER`, `APEX_LISTENER`, `APEX_REST_PUBLIC_USER` and the Oracle Application Express Instance Administration user, `ADMIN`. You can also use these passwords for the configuration of middle tiers and other processes. `apxsilentins.sql` also completes other installation steps such as creating and setting the password for the Instance Administration user, `ADMIN`, configuring a network ACL, and configuring Oracle REST Data Services.

By running the `apxsilentins.sql` script, removes the need for completing the following topics:

- "Installing Application Express"
- "Creating or Updating Your Instance Administration Account"
- "Configuring the `APEX_PUBLIC_USER` Account"
- "Enabling Network Services in Oracle Database 11g or Later"
- "Configuring Static File Support" (`apex_rest_config.sql`)

C.2 Running `apxsilentins.sql`

Run the `apxsilentins.sql` script.

1. To run `apxsilentins.sql`:
1. Change your working directory to `apex`.

2. Start SQL*Plus and connect as user SYS to the database where Oracle Application Express is installed. You will need to specify the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Disable any existing password complexity rules for the default profile.
4. Run apxsilentins.sql passing the following eight arguments in the order shown:

```
@apxsilentins.sql tablespace_apex tablespace_files tablespace_temp images
password_apex_pub_user password_apex_listener password_apex_rest_pub_user
password_internal_admin
```

Where:

- *tablespace_apex* is the name of the tablespace for the Oracle Application Express application user.
- *tablespace_files* is the name of the tablespace for the Oracle Application Express files user.
- *tablespace_temp* is the name of the temporary tablespace or tablespace group.
- *images* is the virtual directory for Oracle Application Express images. For installations using EPG, */i/* is the required value for the images argument. To support future Oracle Application Express upgrades, define the virtual image directory as */i/*.
- *password_apex_pub_user* is the password for the APEX_PUBLIC_USER database account.
- *password_apex_listener* is the password for the APEX_LISTENER database account.
- *password_apex_rest_pub_user* is the password for the APEX_REST_PUBLIC_USER database account.
- *password_internal_admin* is the password for the Instance Administration ADMIN Application Express account. This password must meet the following requirements:
 - Contain at least 6 characters.
 - Contain at least one numeric character (0123456789).
 - Contain at least one punctuation character (!"#\$%&()``*+,-/;:~?_).
 - Contain at least one uppercase alphabetic character.

For example:

```
@apxsilentins.sql SYSAUX SYSAUX TEMP /i/ Passw0rd!1 Passw0rd!2 Passw0rd!3 Passw0rd!4
```

Once `apxsilentins.sql` completes, you complete the steps in "Downloading and Installing Oracle REST Services" and "Configuring Oracle REST Data Services" (except for "Configuring Static File Support.")

Use the passwords you supplied to `apxsilentins.sql` when completing these steps. Then, move on to "Creating a Workspace and Adding Oracle Application Express Users."

 **See Also:**

- ["Downloading and Installing Oracle REST Data Services"](#)
- ["Configuring Oracle REST Data Services"](#)
- [Creating a Workspace and Adding Oracle Application Express Users](#)

D

Maximizing Uptime During an Application Express Upgrade

Learn how to maximize uptime during an Application Express upgrade.

Previously, Oracle Application Express could only be upgraded by completely disabling application usage for an extended length of time. The following is an overview of the additional steps you can take to keep your applications usable for end users during most portions of an Oracle Application Express upgrade.

This advanced procedure is an alternative to the following “Installing Application Express” subsections of the *Oracle Application Express Installation Guide*:

- [Installing Application Express and Configuring Oracle REST Data Services](#)
- [Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway](#)
- [Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server](#)

To upgrade the instance, administrators typically run these phases in one step by executing one of the following:

- For full development environment:

```
@apexins.sql tablespace_apex tablespace_files tablespace_temp images
```

- For runtime-only environment:

```
@apxrtins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- *tablespace_apex* is the name of the tablespace for the Oracle Application Express application user.
- *tablespace_files* is the name of the tablespace for the Oracle Application Express files user.
- *tablespace_temp* is the name of the temporary tablespace or tablespace group.
- *images* is the virtual directory for Oracle Application Express images.

The upgrade of an Application Express instance runs in four phases:

1. Create database schemas and database objects (tables, packages).
2. Migrate application metadata.
3. Migrate data that runtime applications modify and switch to the new version.
4. Migrate additional log and summary data.

Phases 1 and 4 do not disable end users using the instance. Phase 2 only affects developers who modify applications. Phase 3 affects all access to Application Express.

Oracle now also provides alternative upgrade scripts to run the phases independently. Administrators can use these scripts instead of `apexins.sql` and `apxrtins.sql`, to reduce the effective downtime of an Application Express instance from potentially hours to just a few minutes (depending on hardware performance).



Note:

This feature is not supported when Application Express is installed in CDB\$ROOT.

Administrators must sequentially execute the following scripts to start phases 1, 2 and 3, respectively. At the end of phase 3, a scheduler job automatically starts to execute phase 4.

To reduce downtime during an Oracle Application Express upgrade:

1. Execute phase 1 script: Development and runtime usage is not affected.

- For full development environment:

```
@apexins1.sql tablespace_apex tablespace_files tablespace_temp
images
```

- For runtime-only environment:

```
@apxrtins1.sql tablespace_apex tablespace_files tablespace_temp
images
```

Example: `@apexins1.sql sysaux sysaux temp /i/`

2. Execute phase 2 script: Development is disabled, but runtime usage is not affected.

- For full development environment:

```
@apexins2.sql tablespace_apex tablespace_files tablespace_temp
images
```

- For runtime-only environment:

```
@apxrtins2.sql tablespace_apex tablespace_files tablespace_temp
images
```

Example: `@apexins2.sql sysaux sysaux temp /i/`

3. Disable web access for the web listener (Oracle REST Data Services, Embedded PL/SQL Gateway, or Oracle HTTP Server).

4. Execute phase 3 script: Application Express can not be used.

- For full development environment:

```
@apexins3.sql tablespace_apex tablespace_files tablespace_temp
images
```

- For runtime-only environment:

```
@apxrtins3.sql tablespace_apex tablespace_files tablespace_temp
images
```

Example: `@apexins3.sql sysaux sysaux temp /i/`

5. Install images of the new Application Express version in your web listener. Administrators can do this while phase 3 is running or even earlier, if the new version's images directory is different to the previous Application Express version's (for example: `/i51/` for the new version vs. `/i50/` for the old version).

For details refer to the installation instructions for Oracle REST Data Services, Embedded PL/SQL Gateway, or Oracle HTTP Server.

6. Re-enable web access for the web listener (restart Oracle REST Data Services, Embedded PL/SQL Gateway, or Oracle HTTP Server).

After web access is restarted, developers and users can access the instance again, while phase 4 finishes in the background.

See Also:

- [" Installing Application Express and Configuring Oracle REST Data Services"](#)
- ["Installing Application Express and Configuring Web Listener - Embedded PL/SQL Gateway"](#)
- [" Installing Application Express and Configuring Legacy Web Listener – Oracle HTTP Server"](#)

E

Oracle Application Express Installation Troubleshooting

Learn about troubleshooting Oracle Application Express Installation.

This section contains information on troubleshooting.

- [Reviewing a Log of an Installation Session](#)
The `apexins.sql` script creates a log file in the `apex` directory using the naming convention `installYYYY-MM-DD_HH24-MI-SS.log`.
- [Verifying the Validity of an Oracle Application Express Installation](#)
Verify the validity of an Oracle Application Express installation by running a query.
- [Cleaning Up After a Failed Installation](#)
Learn about best practices for troubleshooting and cleaning up after a failed installation.
- [About Images Displaying Incorrectly in Oracle Application Express](#)
Learn about troubleshooting if images in Oracle Application Express do not display correctly.
- [About Page Protection Violation](#)
A page protection violation may be caused by manual alteration of protected page items.



See Also:

["Upgrading from a Previous Oracle Application Express Release"](#)

E.1 Reviewing a Log of an Installation Session

The `apexins.sql` script creates a log file in the `apex` directory using the naming convention `installYYYY-MM-DD_HH24-MI-SS.log`.

In a successful installation, the log file contains the following text:

```
Thank you for installing Oracle Application Express.  
Oracle Application Express is installed in the APEX_190200 schema.
```

If the log file contains a few errors, it does not mean that your installation failed. Note that acceptable errors are noted as such in the log file.

E.2 Verifying the Validity of an Oracle Application Express Installation

Verify the validity of an Oracle Application Express installation by running a query.

You can verify the validity of an Oracle Application Express installation by running the following query:

```
SELECT STATUS FROM DBA_REGISTRY  
WHERE COMP_ID = 'APEX';
```

If the result is `VALID`, you can assume the installation was successful.

E.3 Cleaning Up After a Failed Installation

Learn about best practices for troubleshooting and cleaning up after a failed installation.

In a successful installation the following banner displays near the end of the installation:

```
Thank you for installing Oracle Application Express.  
Oracle Application Express is installed in the APEX_190200 schema.
```

To reinstall, you must either drop the Oracle Application Express database schemas, or run a script to completely remove Application Express from the database, depending upon the installation type.

- [Reverting to a Previous Release After a Failed Upgrade Installation](#)
Learn about reverting to Oracle Application Express to a previous release in the case of a failed upgrade installation.
- [Removing Oracle Application Express from the Database](#)
Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role and execute the `SQL> @apxremov.sql` command.

E.3.1 Reverting to a Previous Release After a Failed Upgrade Installation

Learn about reverting to Oracle Application Express to a previous release in the case of a failed upgrade installation.

In the case of a failed upgrade installation, you may want to revert Oracle Application Express to a previous release and then remove the schemas associated with the current release.

- [Verifying If You Have a Previous Release of Oracle Application Express](#)
- [Reverting the Images Directory](#)
- [Reverting to a Previous Release](#)
- [Removing the Oracle Application Express Release Schema](#)

E.3.1.1 Verifying If You Have a Previous Release of Oracle Application Express

To verify whether you have a previous release of Oracle Application Express:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Execute the following command in SQL*Plus:

```
select username from dba_users
where regexp_like(username, '(FLOWS|APEX)_\d{6}')
and username <> (select table_owner from all_synonyms
where synonym_name = 'WWV_FLOW'
and owner = 'PUBLIC')
```

If the query above returns any rows, the database contains a previous release of Oracle Application Express.

E.3.1.2 Reverting the Images Directory

If you altered your images directory, revert it back to the release you want to revert to using the following methods:

- **Oracle REST Data Services or Oracle HTTP Server with mod_plsql:**

If you altered your images directory, you must point the text alias `/i/` back to images directory for the release you want to revert to.

- **Embedded PL/SQL Gateway:**

Run `apex_epg_config.sql` from the release directory you want to revert to, to restore the images of that release. See the *Oracle Application Express Installation Guide* for the release you want to revert to and find the section "Updating the Images Directory When Upgrading from a Previous Release."



See Also:

"Copying the Images Directory "

E.3.1.3 Reverting to a Previous Release

This section describes how to revert to a previous release Oracle Application Express.

- [Reverting to Release 1.5](#)

- [Reverting to Release 1.6](#)
- [Reverting to Release 2.0](#)
- [Reverting to Release 2.2](#)
- [Reverting to Release 3.0](#)
- [Reverting to Release 3.1](#)
- [Reverting to Release 3.2](#)
- [Reverting to Release 4.0](#)
- [Reverting to Release 4.1](#)
- [Reverting to Release 4.2 in a non-CDB](#)
- [Reverting to Release 4.2 in a CDB](#)
- [Reverting to Release 5.0 in a non-CDB](#)
- [Reverting to Release 5.0 in a CDB](#)
- [Reverting to Release 5.1 in a non-CDB](#)
- [Reverting to Release 5.1 in a CDB](#)
- [Reverting to Release 18.1 in a non-CDB](#)
- [Reverting to Release 18.1 in a CDB](#)
- [Reverting to Release 18.2 in a non-CDB](#)
- [Reverting to Release 18.2 in a CDB](#)
- [Reverting to Release 19.1 in a non-CDB](#)
- [Reverting to Release 19.1 in a CDB](#)

E.3.1.3.1 Reverting to Release 1.5

To revert to a previous Oracle Application Express release 1.5:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as *SYS* specifying the *SYSDBA* role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following:

```
ALTER SESSION SET CURRENT_SCHEMA = FLOWS_010500;
exec flows_010500.www_flow_upgrade.switch_schemas('APEX_190200','FLOWS_010500');
```

4. See "Removing the Oracle Application Express Release Schema."

 **See Also:**

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.2 Reverting to Release 1.6

To revert to a Oracle Application Express release 1.6:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as *SYS* specifying the *SYSDBA* role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following:

```
ALTER SESSION SET CURRENT_SCHEMA = FLOWS_010600;
exec flows_010600.wvw_flow_upgrade.switch_schemas('APEX_190200','FLOWS_010600');
```

4. Depending upon the release you are reverting to, execute the appropriate command in SQL*Plus.
5. See "Removing the Oracle Application Express Release Schema."

 **See Also:**

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.3 Reverting to Release 2.0

To revert to a previous Oracle Application Express release 2.0:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as *SYS* specifying the *SYSDBA* role:
 - On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following:

```
ALTER SESSION SET CURRENT_SCHEMA = FLOWS_020000;
exec flows_020000.www_flow_upgrade.switch_schemas('APEX_190200','FLOWS_020000');
```

4. See "Removing the Oracle Application Express Release Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.4 Reverting to Release 2.2

To revert to a previous Oracle Application Express release 2.2:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following:

```
ALTER SESSION SET CURRENT_SCHEMA = FLOWS_020200;
exec flows_020200.www_flow_upgrade.switch_schemas('APEX_190200','FLOWS_020200');
```

4. See "Removing the Oracle Application Express Release Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.5 Reverting to Release 3.0

To revert to a previous Oracle Application Express release 3.0:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex` in the 3.0 source.
3. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

4. Execute the following:

```
set define '^'

@apexvalidate x x FLOWS_030000
ALTER SESSION SET CURRENT_SCHEMA = FLOWS_030000;
exec flows_030000.wv_flow_upgrade.switch_schemas('APEX_190200','FLOWS_030000');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
declare
    l_apex_version varchar2(30);
begin
    l_apex_version := flows_030000.wv_flows_release;
    dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','FLOWS_030000');
    dbms_registry.downgraded('APEX',l_apex_version);
    validate_apex;
end;
/
```

5. See "Removing the Oracle Application Express Release 19.1 Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.6 Reverting to Release 3.1

To revert to a previous Oracle Application Express release 3.1:

1. If you altered your images directory, revert it back to the release you want to revert to.

2. Change your working directory to `apex/core` in the 3.1 source.
3. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

4. Execute the following commands:

```
@wwv_flow_val.plb
@wwv_dbms_sql.sql
@wwv_dbms_sql.plb
```

5. Change your working directory to `apex` in the 3.1 source.
6. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

7. Execute the following:

```
set define '^'

@apexvalidate x x FLOWS_030100
ALTER SESSION SET CURRENT_SCHEMA = FLOWS_030100;
exec flows_030100.wwv_flow_upgrade.switch_schemas('APEX_190200','FLOWS_030100');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
declare
    l_apex_version varchar2(30);
begin
    l_apex_version := flows_030100.wwv_flows_release;
    dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','FLOWS_030100');
    dbms_registry.downgraded('APEX',l_apex_version);
    validate_apex;
end;
/
```

8. See "Removing the Oracle Application Express Release Schema."

 **See Also:**

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.7 Reverting to Release 3.2

To revert to a previous Oracle Application Express release 3.2:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex/core` in the 3.2 source.
3. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

4. Execute the following commands:

```
@wv_flow_val.plb
@wv_dbms_sql.sql
@wv_dbms_sql.plb
```

5. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

6. Execute the following:

```
set define '^'

@apexvalidate x x APEX_030200
ALTER SESSION SET CURRENT_SCHEMA = APEX_030200;
exec apex_030200.wv_flow_upgrade.switch_schemas('APEX_190200','APEX_030200');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
declare
  l_apex_version varchar2(30);
```

```
begin
  l_apex_version := apex_030200.wv_flows_release;
  dbms_registry.downgrading('APEX', 'Oracle Application
Express', 'validate_apex', 'APEX_030200')
  dbms_registry.downgraded('APEX', l_apex_version);
  validate_apex;
end;
/
```

7. See "Removing the Oracle Application Express Release Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.8 Reverting to Release 4.0

To revert to a previous Oracle Application Express release 4.0:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following commands:

```
@wv_flow_val.sql
@wv_flow_val.plb
@wv_dbms_sql.sql
@wv_dbms_sql.plb
```

4. Change your working directory to apex in the 4.0 source.
5. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

6. Execute the following:

```
set define '^'

@apexvalidate x x APEX_040000
ALTER SESSION SET CURRENT_SCHEMA = APEX_040000;
exec apex_040000.wv_flow_upgrade.switch_schemas('APEX_190200','APEX_040000');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
declare
  l_apex_version varchar2(30);
begin
  l_apex_version := apex_040000.wv_flows_release;
  dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','APEX_040000');
  dbms_registry.downgraded('APEX',l_apex_version);
  validate_apex;
end;
/
```

7. See "Removing the Oracle Application Express Release Schema."

 **See Also:**

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.9 Reverting to Release 4.1

To revert to a previous Oracle Application Express release 4.1:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex/core` in the 4.1 source.
3. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

4. Execute the following commands:

```
@wv_flow_val.sql
@wv_flow_val.plb
```

```
@wwv_dbms_sql.sql
@wwv_dbms_sql.plb
```

5. Change your working directory to `apex` in the 4.1 source.
6. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

7. Execute the following:

```
set define '^'

@apexvalidate x x APEX_040100
ALTER SESSION SET CURRENT_SCHEMA = APEX_040100;
exec apex_040100.wwv_flow_upgrade.switch_schemas('APEX_190200','APEX_040100');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
declare
    l_apex_version varchar2(30);
begin
    l_apex_version := apex_040100.wwv_flows_release;
    dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','APEX_040100');
    dbms_registry.downgraded('APEX',l_apex_version);
    validate_apex;
end;
/
```

8. See "Removing the Oracle Application Express Release Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.10 Reverting to Release 4.2 in a non-CDB

To revert to a previous Oracle Application Express release 4.2 in a non-CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex/core` in the 4.2 source.
3. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

4. Execute the following commands:

```
alter session set current_schema = SYS;

@core_sys_views.sql

grant select on sys.wv_flow_gv$session to APEX_040200;

@wv_flow_val.sql
@wv_flow_val.plb
@wv_dbms_sql.sql
grant execute on wv_dbms_sql to APEX_040200;
@wv_dbms_sql.plb

begin
    dbms_utility.compile_schema('APEX_040200');
end;
/
```

5. Change your working directory to apex in the 4.2 source.

6. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

7. Execute the following:

```
set define '^'

@apexvalidate x x APEX_040200
ALTER SESSION SET CURRENT_SCHEMA = APEX_040200;
exec apex_040200.wv_flow_upgrade.switch_schemas('APEX_190200','APEX_040200');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
declare
    l_apex_version varchar2(30);
begin
    l_apex_version := apex_040200.wv_flows_release;
    dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','APEX_040200');
    dbms_registry.downgraded('APEX',l_apex_version);
    validate_apex;
end;
/
```

8. See "Removing the Oracle Application Express Release Schema."

 **See Also:**

- "Reverting the Images Directory"
- "Removing the Oracle Application Express Release Schema"

E.3.1.3.11 Reverting to Release 4.2 in a CDB

To revert to a previous Oracle Application Express release 4.2 in a CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex/core` in the 4.2 source.
3. Create a new text file in that directory named `apx42dgrd1.sql` consisting of the following:

```
alter session set current_schema = SYS;

@core_sys_views.sql

grant select on sys.wv_flow_gv$session to APEX_040200;

@wv_flow_val.sql
@wv_flow_val.plb
@wv_dbms_sql.sql
grant execute on wv_dbms_sql to APEX_040200;
@wv_dbms_sql.plb

begin
  dbms_utility.compile_schema('APEX_040200');
end;
/
```

4. Create a second new text file in that directory named `apx42dgrd.sql` consisting of the following:

```
set define '^'

whenever sqlerror exit

column :xe_home new_value OH_HOME NOPRINT
variable xe_home varchar2(255)

set serverout on
begin
  -- get oracle_home
  sys.dbms_system.get_env('ORACLE_HOME', :xe_home);
  if length(:xe_home) = 0 then
    sys.dbms_output.put_line(lpad('-', 80, '-'));
    raise_application_error (
      -20001,
      'Oracle Home environment variable not set' );
  end if;
end;
```

```

/
whenever sqlerror continue

set termout off
select :xe_home from sys.dual;
set termout on

host ^OH_HOME/perl/bin/perl -I ^OH_HOME/rdbms/admin ^OH_HOME/rdbms/admin/
catcon.pl -b apx42dgrd apx42dgrd1.sql

```

5. Start SQL*Plus and connect to CDB\$ROOT of the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```

SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

On UNIX and Linux:

```

$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

6. Execute the following commands:

```
@apx42dgrd.sql
```

7. Change your working directory to apex in the 4.2 source.

8. Create a new text file in that directory name apx42dgrd1.sql with the following contents:

```

set define '^'

ALTER SESSION SET CURRENT_SCHEMA = SYS;

@apexvalidate x x APEX_040200

ALTER SESSION SET CURRENT_SCHEMA = APEX_040200;
exec apex_040200.wv_flow_upgrade.switch_schemas('APEX_190200','APEX_040200');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
declare
    l_apex_version varchar2(30);
begin
    l_apex_version := apex_040200.wv_flows_release;
    dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','APEX_040200');
    dbms_registry.downgraded('APEX',l_apex_version);
    validate_apex;
end;
/

```

9. Create a second new text file in that directory named apx42dgrd.sql consisting of the following:

```

set define '^'

whenever sqlerror exit

column :xe_home new_value OH_HOME NOPRINT
variable xe_home varchar2(255)

```

```

set serverout on
begin
-- get oracle_home
  sys.dbms_system.get_env('ORACLE_HOME',:xe_home);
  if length(:xe_home) = 0 then
    sys.dbms_output.put_line(lpad('-',80,'-'));
    raise_application_error (
      -20001,
      'Oracle Home environment variable not set' );
  end if;
end;
/
whenever sqlerror continue

set termout off
select :xe_home from sys.dual;
set termout on

host ^OH_HOME/perl/bin/perl -I ^OH_HOME/rdbms/admin ^OH_HOME/rdbms/admin/
catcon.pl -b apx42dgrd apx42dgrd1.sql

```

10. Start SQL*Plus and connect to CDB\$ROOT of the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```

SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

On UNIX and Linux:

```

$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

11. Execute the following:

```
@apx42dgrd.sql
```

12. See "Removing the Oracle Application Express Release Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.12 Reverting to Release 5.0 in a non-CDB

To revert to a previous Oracle Application Express release 5.0 in a non-CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex/core` in the 5.0 source.
3. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

4. Execute the following commands:

```
alter session set current_schema = SYS;

@wwv_flow_val.sql
@wwv_flow_val.plb

begin
  dbms_utility.compile_schema('APEX_050000');
end;
/

set define '^'
@validate_apex x x APEX_050000

begin
  for i in ( select owner, trigger_name
            from sys.dba_triggers
            where owner      = 'APEX_050000'
              and trigger_name like 'WWV_FLOW_UPGRADE_%'
            order by 1 )
  loop
    sys.dbms_output.put_line('Dropping trigger '||i.owner||'.'||i.trigger_name);
    execute immediate 'drop trigger '||i.owner||'.'||i.trigger_name;
  end loop;
end;
/

ALTER SESSION SET CURRENT_SCHEMA = APEX_050000;
exec apex_050000.wwv_flow_upgrade.switch_schemas('APEX_190200','APEX_050000');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
drop context APEX$SESSION;
create context APEX$SESSION using APEX_050000.WWV_FLOW_SESSION_CONTEXT;
declare
  l_apex_version varchar2(30);
  l_schemas sys.dbms_registry.schema_list_t;
begin
  l_apex_version := apex_050000.wwv_flows_release;
  dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','APEX_050000');
  dbms_registry.downgraded('APEX',l_apex_version);
  select username
  bulk collect into l_schemas
  from all_users
  where username in
('FLOWS_FILES','APEX_PUBLIC_USER','APEX_LISTENER','APEX_REST_PUBLIC_USER','APEX_I
NSTANCE_ADMIN_USER')
  order by 1;
  sys.dbms_registry.update_schema_list('APEX', l_schemas);
  validate_apex;
```

```
end;
/
```

5. See "Removing the Oracle Application Express Release Schema."

See Also:

- "Reverting the Images Directory"
- "Removing the Oracle Application Express Release Schema"

E.3.1.3.13 Reverting to Release 5.0 in a CDB

To revert to a previous Oracle Application Express release 5.0 in a CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex/core` in the 5.0 source.
3. Create a new text file in that directory named `apx50dgrd1.sql` consisting of the following:

```
alter session set current_schema = SYS;

@wwv_flow_val.sql
@wwv_flow_val.plb

begin
  dbms_utility.compile_schema('APEX_050000');
end;
/

set define '^'
@validate_apex x x APEX_050000

begin
  for i in ( select owner, trigger_name
            from sys.dba_triggers
            where owner      = 'APEX_050000'
              and trigger_name like 'WWV_FLOW_UPGRADE_%'
            order by 1 )
  loop
    sys.dbms_output.put_line('Dropping trigger '||i.owner||'.'||i.trigger_name);
    execute immediate 'drop trigger '||i.owner||'.'||i.trigger_name;
  end loop;
end;
/

ALTER SESSION SET CURRENT_SCHEMA = APEX_050000;
exec apex_050000.wwv_flow_upgrade.switch_schemas('APEX_190200','APEX_050000');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
drop context APEX$SESSION;
create context APEX$SESSION using APEX_050000.WWV_FLOW_SESSION_CONTEXT;
declare
  l_apex_version varchar2(30);
  l_schemas sys.dbms_registry.schema_list_t;
begin
```

```

l_apex_version := apex_050000.wv_flows_release;
dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','APEX_050000');
dbms_registry.downgraded('APEX',l_apex_version);
select username
  bulk collect into l_schemas
  from all_users
  where username in
('FLOWS_FILES','APEX_PUBLIC_USER','APEX_LISTENER','APEX_REST_PUBLIC_USER','APEX_I
NSTANCE_ADMIN_USER')
  order by 1;
sys.dbms_registry.update_schema_list('APEX', l_schemas);
validate_apex;
end;
/

```

4. Create a second new text file in that directory named `apx50dgrd.sql` consisting of the following:

```

set define '^'
whenever sqlerror exit
column :xe_home new_value OH_HOME NOPRINT
variable xe_home varchar2(255)
set serverout on
begin
-- get oracle_home
sys.dbms_system.get_env('ORACLE_HOME',:xe_home);
if length(:xe_home) = 0 then
  sys.dbms_output.put_line(lpad('-',80,'-'));
  raise_application_error (-20001,'Oracle Home environment variable not
set' );
end if;
end;
/
whenever sqlerror continue
set termout off
select :xe_home from sys.dual;
set termout on
host ^OH_HOME/perl/bin/perl -I ^OH_HOME/rdbms/admin ^OH_HOME/rdbms/admin/
catcon.pl -b apx50dgrd apx50dgrd1.sql

```

5. Start SQL*Plus and connect to CDB\$ROOT of the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```

SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

On UNIX and Linux:

```

$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

6. Execute the following commands:

```
@apx50dgrd.sql
```

7. See "Removing the Oracle Application Express Release Schema."

 **See Also:**

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.14 Reverting to Release 5.1 in a non-CDB

To revert to a previous Oracle Application Express release 5.1 in a non-CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex/core` in the 5.1 source.
3. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

4. Execute the following commands:

```
alter session set current_schema = SYS;

@wwv_flow_val.sql
@wwv_flow_val.plb

begin
    dbms_utility.compile_schema('APEX_050100');
end;
/

set define '^'
@validate_apex x x APEX_050100

begin
    for i in ( select owner, trigger_name
              from sys.dba_triggers
              where owner          = 'APEX_050100'
                and trigger_name like 'WWV_FLOW_UPGRADE_%'
              order by 1 )
    loop
        sys.dbms_output.put_line('Dropping trigger '||i.owner||'.'||i.trigger_name);
        execute immediate 'drop trigger '||i.owner||'.'||i.trigger_name;
    end loop;
end;
/

ALTER SESSION SET CURRENT_SCHEMA = APEX_050100;
exec apex_050100.wwv_flow_upgrade.switch_schemas('APEX_190200','APEX_050100');
```

```

ALTER SESSION SET CURRENT_SCHEMA = SYS;
drop context APEX$SESSION;
create context APEX$SESSION using APEX_050100.WWV_FLOW_SESSION_CONTEXT;
declare
    l_apex_version varchar2(30);
    l_schemas sys.dbms_registry.schema_list_t;
begin
    l_apex_version := apex_050100.wwv_flows_release;
    dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','APEX_050100');
    dbms_registry.downgraded('APEX',l_apex_version);
    select username
        bulk collect into l_schemas
        from all_users
        where username in
        ('FLOWS_FILES','APEX_PUBLIC_USER','APEX_LISTENER','APEX_REST_PUBLIC_USER','APEX_I
NSTANCE_ADMIN_USER')
        order by 1;
    sys.dbms_registry.update_schema_list('APEX', l_schemas);
    validate_apex;
end;
/

```

5. See the next section, "Removing the Oracle Application Express Release 19.1 Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.15 Reverting to Release 5.1 in a CDB

To revert to a previous Oracle Application Express release 5.1 in a CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex/core` in the 5.1 source.
3. Create a new text file in that directory named `apx51dgrd1.sql` consisting of the following:

```

alter session set current_schema = SYS;

@wwv_flow_val.sql
@wwv_flow_val.plb

begin
    dbms_utility.compile_schema('APEX_050100');
end;
/

set define '^'
@validate_apex x x APEX_050100

begin

```

```

for i in ( select owner, trigger_name
          from sys.dba_triggers
          where owner      = 'APEX_050100'
            and trigger_name like 'WWV_FLOW_UPGRADE_%'
          order by 1 )
loop
  sys.dbms_output.put_line('Dropping trigger '||i.owner||'.'||i.trigger_name);
  execute immediate 'drop trigger '||i.owner||'.'||i.trigger_name;
end loop;
end;
/

ALTER SESSION SET CURRENT_SCHEMA = APEX_050100;
exec apex_050100.wv_flow_upgrade.switch_schemas('APEX_190200','APEX_050100');
ALTER SESSION SET CURRENT_SCHEMA = SYS;
drop context APEX$SESSION;
create context APEX$SESSION using APEX_050100.WWV_FLOW_SESSION_CONTEXT;
declare
  l_apex_version varchar2(30);
  l_schemas sys.dbms_registry.schema_list_t;
begin
  l_apex_version := apex_050100.wv_flows_release;
  dbms_registry.downgrading('APEX','Oracle Application
Express','validate_apex','APEX_050100');
  dbms_registry.downgraded('APEX',l_apex_version);
  select username
         bulk collect into l_schemas
         from all_users
         where username in
('FLOWS_FILES','APEX_PUBLIC_USER','APEX_LISTENER','APEX_REST_PUBLIC_USER','APEX_I
NSTANCE_ADMIN_USER')
         order by 1;
  sys.dbms_registry.update_schema_list('APEX', l_schemas);
  validate_apex;
end;
/

```

4. Create a second new text file in that directory named `apx51dgrd.sql` consisting of the following:

```

set define '^'
whenever sqlerror exit
column :xe_home new_value OH_HOME NOPRINT
variable xe_home varchar2(255)
set serverout on
begin
  -- get oracle_home
  sys.dbms_system.get_env('ORACLE_HOME',:xe_home);
  if length(:xe_home) = 0 then
    sys.dbms_output.put_line(lpad('-',80,'-'));
    raise_application_error (-20001,'Oracle Home environment variable not
set' );
  end if;
end;
/
whenever sqlerror continue
set termout off
select :xe_home from sys.dual;
set termout on
host ^OH_HOME/perl/bin/perl -I ^OH_HOME/rdbms/admin ^OH_HOME/rdbms/admin/
catcon.pl -b apx51dgrd apx51dgrd1.sql

```

5. Start SQL*Plus and connect to CDB\$ROOT of the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

6. Execute the following commands:

```
@apx51dgrd.sql
```

7. See "Removing the Oracle Application Express Release Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.16 Reverting to Release 18.1 in a non-CDB

To revert to a previous Oracle Application Express release 18.1 in a non-CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex` in the 18.1 source.
3. Create a new text file in that directory named `apxdwngrd.sql` consisting of the following:

```
set define '^'
set concat on
set concat .
set verify off

set termout off
column foo new_val LOG
select 'apxdwngrd_' ||to_char(sysdate, 'YYYY-MM-DD_HH24-MI-SS') ||
'.log' as foo
  from sys.dual;
set termout on
spool ^LOG

@@core/scripts/set_appun.sql

whenever sqlerror exit
set serveroutput on size unlimited
```

```
declare
    l_cnt    number := 0;
begin
    select count(*) into l_cnt from sys.dba_users where username =
'^APPUN';
    if l_cnt = 0 then
        dbms_output.put_line('^APPUN not found in this database.');
```

raise program_error;

```
    end if;
end;
/
whenever sqlerror continue

prompt ...Create validate procedure in SYS schema and start registration
@@core/validate_apex.sql x x ^APPUN

grant inherit any privileges to ^APPUN;

prompt Installing SYS views

@@core/sys_core_views.sql

@@core/wwv_flow_val.sql
grant execute on sys.wwv_flow_val to ^APPUN.;

@@core/wwv_flow_val.plb

ALTER SESSION SET CURRENT_SCHEMA = ^APPUN;

exec
sys.dbms_session.modify_package_state(sys.dbms_session.reinitialize);

begin
    ^APPUN..wwv_flow_upgrade.remove_jobs();
    ^APPUN..wwv_flow_upgrade.create_jobs('^APPUN');
    ^APPUN..wwv_flow_upgrade.create_public_synonyms('^APPUN');
    ^APPUN..wwv_flow_upgrade.grant_public_synonyms('^APPUN');
    ^APPUN..wwv_flow_upgrade.flows_files_objects_remove('^APPUN');
    ^APPUN..wwv_flow_upgrade.flows_files_objects_create('^APPUN');
end;
/

ALTER SESSION SET CURRENT_SCHEMA = SYS;

drop context APEX$SESSION;
create context APEX$SESSION using ^APPUN..WWV_FLOW_SESSION_CONTEXT;

alter package sys.wwv_dbms_sql_^APPUN. compile;
alter package sys.wwv_dbms_sql_^APPUN. compile body;

exec
sys.dbms_session.modify_package_state(sys.dbms_session.reinitialize);
```

```

set serveroutput on size unlimited

declare
    l_apex_version varchar2(30);
    l_schemas sys.dbms_registry.schema_list_t;
begin
    execute immediate 'drop package ^APPUN..WWV_FLOW_DB_VERSION';
    l_apex_version := ^APPUN..wwv_flows_release;
    sys.dbms_registry.loading('APEX','Oracle Application
Express','validate_apex', '^APPUN');
    select username
        bulk collect into l_schemas
        from sys.all_users
        where username in
('FLOWS_FILES','APEX_PUBLIC_USER','APEX_LISTENER','APEX_REST_PUBLIC_USER
','APEX_INSTANCE_ADMIN_USER')
        order by 1;
    sys.dbms_registry.update_schema_list('APEX', l_schemas);
    sys.dbms_registry.loaded('APEX',l_apex_version);
    commit;
    sys.validate_apex;
end;
/

```

4. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```

SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

On UNIX and Linux:

```

$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

5. Run the apxdwngrd.sql script:

```

SQL> @apxdwngrd.sql

```

6. See "Removing the Oracle Application Express Release Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.17 Reverting to Release 18.1 in a CDB

To revert to a previous Oracle Application Express release 18.1 in a CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to apex in the 18.1 source.
3. Create a new text file in that directory named apxdwngrd.sql consisting of the following:

```

set define '^'
set concat on
set concat .
set verify off

set termout off
column foo new_val LOG
select 'apxdwngrd_' ||to_char(sysdate, 'YYYY-MM-DD_HH24-MI-SS') || '.log' as foo
  from sys.dual;
set termout on
spool ^LOG

@@core/scripts/set_appun.sql

whenever sqlerror exit
set serveroutput on size unlimited

declare
  l_cnt    number := 0;
begin
  select count(*) into l_cnt from sys.dba_users where username = '^APPUN';
  if l_cnt = 0 then
    dbms_output.put_line('^APPUN not found in this database. ');
    raise program_error;
  end if;
end;
/
whenever sqlerror continue

prompt ...Create validate procedure in SYS schema and start registration
@@core/validate_apex.sql x x ^APPUN

grant inherit any privileges to ^APPUN;

prompt Installing SYS views

@@core/sys_core_views.sql

@@core/wwv_flow_val.sql
grant execute on sys.wwv_flow_val to ^APPUN.;

@@core/wwv_flow_val.plb

ALTER SESSION SET CURRENT_SCHEMA = ^APPUN;

exec sys.dbms_session.modify_package_state(sys.dbms_session.reinitialize);

begin
  ^APPUN..wwv_flow_upgrade.remove_jobs();
  ^APPUN..wwv_flow_upgrade.create_jobs('^APPUN');
  ^APPUN..wwv_flow_upgrade.create_public_synonyms('^APPUN');
  ^APPUN..wwv_flow_upgrade.grant_public_synonyms('^APPUN');
  ^APPUN..wwv_flow_upgrade.flows_files_objects_remove('^APPUN');
```

```

        ^APPUN..wwv_flow_upgrade.flows_files_objects_create('^APPUN');
end;
/

ALTER SESSION SET CURRENT_SCHEMA = SYS;

drop context APEX$SESSION;
create context APEX$SESSION using ^APPUN..WWV_FLOW_SESSION_CONTEXT;

alter package sys.wwv_dbms_sql_ ^APPUN. compile;
alter package sys.wwv_dbms_sql_ ^APPUN. compile body;

exec sys.dbms_session.modify_package_state(sys.dbms_session.reinitialize);

set serveroutput on size unlimited

declare
    l_apex_version varchar2(30);
    l_schemas sys.dbms_registry.schema_list_t;
begin
    execute immediate 'drop package ^APPUN..WWV_FLOW_DB_VERSION';
    l_apex_version := ^APPUN..wwv_flows_release;
    sys.dbms_registry.loading('APEX','Oracle Application
Express','validate_apex', '^APPUN');
    select username
        bulk collect into l_schemas
        from sys.all_users
        where username in
('FLOWS_FILES','APEX_PUBLIC_USER','APEX_LISTENER','APEX_REST_PUBLIC_USER','APEX_I
NSTANCE_ADMIN_USER')
        order by 1;
    sys.dbms_registry.update_schema_list('APEX', l_schemas);
    sys.dbms_registry.loaded('APEX',l_apex_version);
    commit;
    sys.validate_apex;
end;
/

```

4. Create a second new text file in that directory named `apxdwngrd_cdb.sql` consisting of the following:

```

set define '^'
whenever sqlerror exit
column :xe_home new_value OH_HOME NOPRINT
variable xe_home varchar2(255)
set serverout on
begin
-- get oracle_home
sys.dbms_system.get_env('ORACLE_HOME',:xe_home);
if length(:xe_home) = 0 then
    sys.dbms_output.put_line(lpad('-',80,'-'));
    raise_application_error (-20001,'Oracle Home environment variable not
set' );
end if;
end;
/
whenever sqlerror continue
set termout off
select :xe_home from sys.dual;
set termout on
host ^OH_HOME/perl/bin/perl -I ^OH_HOME/rdbms/admin ^OH_HOME/rdbms/admin/
catcon.pl -b apx18ldgrd apxdwngrd.sql

```

5. Start SQL*Plus and connect to CDB\$ROOT of the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

6. Run the `apxdwngrd_cdb.sql` script:

```
SQL> @apxdwngrd_cdb.sql
```

7. See "Removing the Oracle Application Express Release Schema."



See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.18 Reverting to Release 18.2 in a non-CDB

To revert to a previous Oracle Application Express release 18.2 in a non-CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex` in the 18.2 source.
3. Create a new text file in that directory named `apxdwngrd.sql` consisting of the following:

```
set define '^'
set concat on
set concat .
set verify off

set termout off
column foo new_val LOG
select 'apxdwngrd_' ||to_char(sysdate, 'YYYY-MM-DD_HH24-MI-SS') ||
'.log' as foo
  from sys.dual;
set termout on
spool ^LOG

@@core/scripts/set_appun.sql

whenever sqlerror exit
set serveroutput on size unlimited
```

```

declare
    l_cnt    number := 0;
begin
    select count(*) into l_cnt from sys.dba_users where username =
'^APPUN';
    if l_cnt = 0 then
        dbms_output.put_line('^APPUN not found in this database.');
```

raise program_error;

```

    end if;
end;
/
whenever sqlerror continue

prompt ...Create validate procedure in SYS schema and start registration
@@core/validate_apex.sql x x ^APPUN

grant inherit any privileges to ^APPUN;

prompt Installing SYS views

@@core/sys_core_views.sql

@@core/wwv_flow_val.sql
grant execute on sys.wwv_flow_val to ^APPUN.;

@@core/wwv_flow_val.plb

ALTER SESSION SET CURRENT_SCHEMA = ^APPUN;

exec
sys.dbms_session.modify_package_state(sys.dbms_session.reinitialize);

begin
    ^APPUN..wwv_flow_upgrade.remove_jobs();
    ^APPUN..wwv_flow_upgrade.create_jobs('^APPUN');
    ^APPUN..wwv_flow_upgrade.create_public_synonyms('^APPUN');
    ^APPUN..wwv_flow_upgrade.grant_public_synonyms('^APPUN');
    ^APPUN..wwv_flow_upgrade.flows_files_objects_remove('^APPUN');
    ^APPUN..wwv_flow_upgrade.flows_files_objects_create('^APPUN');
end;
/

ALTER SESSION SET CURRENT_SCHEMA = SYS;

drop context APEX$SESSION;
create context APEX$SESSION using ^APPUN..WWV_FLOW_SESSION_CONTEXT;

alter package sys.wwv_dbms_sql_^APPUN. compile;
alter package sys.wwv_dbms_sql_^APPUN. compile body;

exec
sys.dbms_session.modify_package_state(sys.dbms_session.reinitialize);
```

```
set serveroutput on size unlimited

declare
    l_apex_version varchar2(30);
    l_schemas sys.dbms_registry.schema_list_t;
begin
    execute immediate 'drop package ^APPUN..WWV_FLOW_DB_VERSION';
    l_apex_version := ^APPUN..wwv_flows_release;
    sys.dbms_registry.loading('APEX','Oracle Application
Express','validate_apex', '^APPUN');
    select username
        bulk collect into l_schemas
        from sys.all_users
        where username in
('FLOWS_FILES','APEX_PUBLIC_USER','APEX_LISTENER','APEX_REST_PUBLIC_USER
','APEX_INSTANCE_ADMIN_USER')
        order by 1;
    sys.dbms_registry.update_schema_list('APEX', l_schemas);
    sys.dbms_registry.loaded('APEX',l_apex_version);
    commit;
    sys.validate_apex;
end;
/
```

4. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

5. Run the `apxdwngrd.sql` script:

```
SQL> @apxdwngrd.sql
```

6. See "Removing the Oracle Application Express Release Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.19 Reverting to Release 18.2 in a CDB

To revert to a previous Oracle Application Express release 18.2 in a CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex` in the 18.2 source.
3. Create a new text file in that directory named `apxdwngrd.sql` consisting of the following:

```
set define '^'
set concat on
set concat .
set verify off

set termout off
column foo new_val LOG
select 'apxdwngrd_' ||to_char(sysdate, 'YYYY-MM-DD_HH24-MI-SS') || '.log' as foo
  from sys.dual;
set termout on
spool ^LOG

@@core/scripts/set_appun.sql

whenever sqlerror exit
set serveroutput on size unlimited

declare
  l_cnt    number := 0;
begin
  select count(*) into l_cnt from sys.dba_users where username = '^APPUN';
  if l_cnt = 0 then
    dbms_output.put_line('^APPUN not found in this database. ');
    raise program_error;
  end if;
end;
/
whenever sqlerror continue

prompt ...Create validate procedure in SYS schema and start registration
@@core/validate_apex.sql x x ^APPUN

grant inherit any privileges to ^APPUN;

prompt Installing SYS views

@@core/sys_core_views.sql

@@core/wwv_flow_val.sql
grant execute on sys.wwv_flow_val to ^APPUN.;

@@core/wwv_flow_val.plb

ALTER SESSION SET CURRENT_SCHEMA = ^APPUN;

exec sys.dbms_session.modify_package_state(sys.dbms_session.reinitialize);

begin
  ^APPUN..wwv_flow_upgrade.remove_jobs();
  ^APPUN..wwv_flow_upgrade.create_jobs('^APPUN');
  ^APPUN..wwv_flow_upgrade.create_public_synonyms('^APPUN');
  ^APPUN..wwv_flow_upgrade.grant_public_synonyms('^APPUN');
  ^APPUN..wwv_flow_upgrade.flows_files_objects_remove('^APPUN');
```

```

        ^APPUN..wv_flow_upgrade.flows_files_objects_create('^APPUN');
end;
/

ALTER SESSION SET CURRENT_SCHEMA = SYS;

drop context APEX$SESSION;
create context APEX$SESSION using ^APPUN..WV_FLOW_SESSION_CONTEXT;

alter package sys.wv_dbms_sql_^APPUN. compile;
alter package sys.wv_dbms_sql_^APPUN. compile body;

exec sys.dbms_session.modify_package_state(sys.dbms_session.reinitialize);

set serveroutput on size unlimited

declare
    l_apex_version varchar2(30);
    l_schemas sys.dbms_registry.schema_list_t;
begin
    execute immediate 'drop package ^APPUN..WV_FLOW_DB_VERSION';
    l_apex_version := ^APPUN..wv_flows_release;
    sys.dbms_registry.loading('APEX','Oracle Application
Express','validate_apex', '^APPUN');
    select username
        bulk collect into l_schemas
        from sys.all_users
        where username in
('FLOWS_FILES','APEX_PUBLIC_USER','APEX_LISTENER','APEX_REST_PUBLIC_USER','APEX_I
NSTANCE_ADMIN_USER')
        order by 1;
    sys.dbms_registry.update_schema_list('APEX', l_schemas);
    sys.dbms_registry.loaded('APEX',l_apex_version);
    commit;
    sys.validate_apex;
end;
/

```

4. Create a second new text file in that directory named `apxdwngrd_cdb.sql` consisting of the following:

```

set define '^'
whenever sqlerror exit
column :xe_home new_value OH_HOME NOPRINT
variable xe_home varchar2(255)
set serverout on
begin
    -- get oracle_home
    sys.dbms_system.get_env('ORACLE_HOME',:xe_home);
    if length(:xe_home) = 0 then
        sys.dbms_output.put_line(lpad('-',80,'-'));
        raise_application_error (-20001,'Oracle Home environment variable not
set' );
    end if;
end;
/
whenever sqlerror continue
set termout off
select :xe_home from sys.dual;
set termout on
host ^OH_HOME/perl/bin/perl -I ^OH_HOME/rdbms/admin ^OH_HOME/rdbms/admin/
catcon.pl -b apx182dgrd apxdwngrd.sql

```

5. Start SQL*Plus and connect to CDB\$ROOT of the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

6. Run the `apxdwngrd_cdb.sql` script:

```
SQL> @apxdwngrd_cdb.sql
```

7. See "Removing the Oracle Application Express Release Schema."

See Also:

- ["Reverting the Images Directory"](#)
- ["Removing the Oracle Application Express Release Schema"](#)

E.3.1.3.20 Reverting to Release 19.1 in a non-CDB

To revert to Oracle Application Express release 19.1 in a non-CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex` in the 19.1 source.
3. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS specifying the SYSDBA role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

4. Run the `apxdwngrd.sql` script:

```
SQL> @apxdwngrd.sql
```

(Optional) Enter the result of the procedure here.

E.3.1.3.21 Reverting to Release 19.1 in a CDB

To revert to Oracle Application Express release 19.1 in a CDB:

1. If you altered your images directory, revert it back to the release you want to revert to.
2. Change your working directory to `apex` in the 19.1 source.
3. Create a script in the apex directory called `apxdwngrd_cdb.sql` with the following contents:

```
set define '^'
whenever sqlerror exit
column :xe_home new_value OH_HOME NOPRINT
variable xe_home varchar2(255)
set serverout on
begin
-- get oracle_home
sys.dbms_system.get_env('ORACLE_HOME',:xe_home);
if length(:xe_home) = 0 then
    sys.dbms_output.put_line(lpad('-',80,'-'));
    raise_application_error (-20001,'Oracle Home environment variable
not set' );
end if;
end;
/
whenever sqlerror continue
set termout off
select :xe_home from sys.dual;
set termout on
host ^OH_HOME/perl/bin/perl -I ^OH_HOME/rdbms/admin ^OH_HOME/rdbms/
admin/catcon.pl -b apx191dgrd apxdwngrd.sql
```

4. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

5. Run the `apxdwngrd_cdb.sql` script:

```
SQL> @apxdwngrd_cdb.sql
```

E.3.1.4 Removing the Oracle Application Express Release Schema

After you have reverted to the prior release, you can remove the Oracle Application Express schema.

- [Removing the Oracle Application Express Release 19.2 Schema from a Non-CDB](#)
Start SQL*Plus and connect to the database and execute `DROP USER APEX_190200 CASCADE;` .
- [Removing the Oracle Application Express Release 19.2 Schema from a CDB](#)
Create text files, start SQL*Plus and connect to the database execute `@remove_apx192_usr.sql`.

E.3.1.4.1 Removing the Oracle Application Express Release 19.2 Schema from a Non-CDB

Start SQL*Plus and connect to the database and execute `DROP USER APEX_190200 CASCADE;` .

To remove the release 19.2 schema from a non-CDB:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Execute the following command:

```
DROP USER APEX_190200 CASCADE;
```

Once you have removed the Oracle Application Express 19.2 schema, you can now attempt the upgrade again.

E.3.1.4.2 Removing the Oracle Application Express Release 19.2 Schema from a CDB

Create text files, start SQL*Plus and connect to the database execute `@remove_apx192_usr.sql`.

To remove the release 19.2 schema from a CDB:

1. Create a new text file named `remove_apx192_usr1.sql` with the following contents:

```
alter session set current_schema = SYS;
drop user APEX_190200 cascade;
```

2. Create a second new text file named `remove_apx192_usr.sql` with the following contents:

```
set define '^'
whenever sqlerror exit
```

```

column :xe_home new_value OH_HOME NOPRINT
variable xe_home varchar2(255)

set serverout on
begin
-- get oracle_home
  sys.dbms_system.get_env('ORACLE_HOME',:xe_home);
  if length(:xe_home) = 0 then
    sys.dbms_output.put_line(lpad('-',80,'-'));
    raise_application_error (
      -20001,
      'Oracle Home environment variable not set' );
  end if;
end;
/
whenever sqlerror continue

set termout off
select :xe_home from sys.dual;
set termout on

host ^OH_HOME/perl/bin/perl -I ^OH_HOME/rdbms/admin ^OH_HOME/rdbms/admin/
catcon.pl -b remove_apx192_usr remove_apx192_usr1.sql

```

3. Start SQL*Plus and connect to the database where Oracle Application Express is installed as *SYS* specifying the *SYSDBA* role:

- On Windows:

```

SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

- On UNIX and Linux:

```

$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password

```

4. Execute the following command:

```
@remove_apx192_usr.sql
```

Once you have removed the Oracle Application Express 19.2 schema, you can now attempt the upgrade again.

E.3.2 Removing Oracle Application Express from the Database

Start SQL*Plus and connect to the database where Oracle Application Express is installed as *SYS* specifying the *SYSDBA* role and execute the `SQL> @apxremov.sql` command.

This section describes how to remove the Oracle Application Express schema, synonyms, and users from the database without deleting the database.

 **Note:**

Do NOT follow these steps if you have upgraded your database from a prior release, and still want to use the prior release of Oracle Application Express. For information about reverting to a prior release, see [Reverting to a Previous Release](#). If you are not sure whether you have completed a new installation or an upgrade installation, review [Cleaning Up After a Failed Installation](#) to verify if a previous release of Oracle Application Express exists in the database.

To remove Oracle Application Express from the database:

1. Change your working directory to the `apex` directory where you unzipped the Oracle Application Express software.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Execute the following command:

```
SQL> @apxremov.sql
```

4. After successfully removing Application Express using `apxremov.sql`, you must exit your current SQL*Plus session and reconnect before attempting another install using `apexins.sql`.

E.4 About Images Displaying Incorrectly in Oracle Application Express

Learn about troubleshooting if images in Oracle Application Express do not display correctly.

If images in Oracle Application Express do not display correctly, you may have more than one definition of the `/i/` alias. To address this issue:

- If possible, rename the first instance of `/i/` to a different alias name.
- Alternatively, copy the images from the directory where Oracle Application Express was downloaded or the images copied for Oracle REST Data Services (ORDS) to the directory defined by the first `/i/` alias.

E.5 About Page Protection Violation

A page protection violation may be caused by manual alteration of protected page items.

If this error occurs after installation when trying to log into Oracle Application Express, then stop and start Oracle REST Data Services. If you are unsure of what caused this error, contact the application administrator for assistance.

F

Upgrading Oracle Application Express within Oracle Database Express Edition

Learn about upgrading Oracle Application Express included with Oracle Database XE.

Oracle Database 11g Express Edition (Oracle Database XE) includes Oracle Application Express release 4.0. Oracle strongly recommends that you upgrade to the latest Oracle Application Express release to take advantage of the latest features. This section describes how to upgrade Oracle Application Express included with Oracle Database XE.

Tip:

Upgrading Oracle Application Express does not change the Oracle Support policy for Oracle Database XE. Oracle Database XE is only supported on the Oracle OTN forums. Oracle Support will not answer questions about Oracle Application Express on Oracle Database XE.

- [Upgrading to the Latest Oracle Application Express Release](#)
Learn how to upgrade to the latest Oracle Application Express release.
- [Understanding the Differences Between Later Oracle Application Express Releases](#)
Learn about Oracle Database 11g Express Edition that includes Oracle Application Express 4.0 and the difference between 4.0 and later releases.

See Also:

Oracle Database 11g Express Edition OTN site at:

<http://www.oracle.com/technetwork/database/database-technologies/express-edition/overview/index.html>

F.1 Upgrading to the Latest Oracle Application Express Release

Learn how to upgrade to the latest Oracle Application Express release.

To upgrade to the latest Oracle Application Express release:

1. Download the latest version of Oracle Application Express from the download page. See:

<http://www.oracle.com/technetwork/developer-tools/apex/downloads/index.html>

2. Unzip downloaded zip file:

- UNIX and Linux: `$ unzip filename.zip`
- Windows: Double click `filename.zip` in Windows Explorer

 **Tip:**

Keep the directory tree where you unzip the files short and not under directories that contain spaces. For example, within Windows unzip to `C:\`.

3. Change your working directory to `apex`.

4. Start SQL*Plus and connect to the Oracle Database XE where Oracle Application Express is installed as `SYS` specifying the `SYSDBA` role. For example:

- On Windows:

```
SYSTEM_DRIVE:\ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

- On UNIX and Linux:

```
$ sqlplus /nolog
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

 **Tip:**

Keep the directory tree where you unzip the files short and not under directories that contain spaces. For example, within Windows unzip to `C:\`.

5. Install Oracle Application Express:

```
@apexins.sql tablespace_apex tablespace_files tablespace_temp images
```

Where:

- `tablespace_apex` is the name of the tablespace for the Oracle Application Express application user.
- `tablespace_files` is the name of the tablespace for the Oracle Application Express files user.
- `tablespace_temp` is the name of the temporary tablespace or tablespace group.
- `images` is the virtual directory for Oracle Application Express images. To support future Oracle Application Express upgrades, define the virtual image directory as `/i/`.

Example

```
@apexins.sql SYSAUX SYSAUX TEMP /i/
```

6. Log back into SQL*Plus (as described in step 4) and load images:

```
@apex_epg_config.sql APEX_HOME
```

 **Tip:**

APEX_HOME is the directory you specified when unzipping the file. For example, on Windows C:\.

7. Upgrade the Oracle Application Express password by running `apxchpwd.sql`:

```
@apxchpwd.sql
```

When prompted, enter a password for the `ADMIN` account.

8. Navigate to the Oracle Application Express Administration Services application:

- a. In a Web browser, navigate to:

```
http://hostname:port/apex/apex_admin
```

Where:

hostname is the name of the system where Oracle XML DB Protocol server is installed.

port is the port number assigned to HTTP on the Oracle XML DB Protocol server. In a default installation, this number is 8080. If you are using the Oracle Database 12c or later multitenant architecture, then each pluggable database (PDB) will have a distinct port number.

apex is the database access descriptor (DAD) defined in the configuration file.

- b. On the Sign In page:

- Username - Enter `admin`.
- Password - Enter the Oracle Application Express administrator account password you specified in step 4.
- Click **Sign In to Administration**.

Note that, depending on your setup, you might be required to change your password when you log in for the first time.

F.2 Understanding the Differences Between Later Oracle Application Express Releases

Learn about Oracle Database 11g Express Edition that includes Oracle Application Express 4.0 and the difference between 4.0 and later releases.

Oracle Database 11g Express Edition includes Oracle Application Express 4.0. This section describes the differences between Oracle Application Express release 4.0 and later releases.

- [User Interface Differences](#)
Learn about the differences between Oracle Application Express release 4.0 and later releases.

F.2.1 User Interface Differences

Learn about the differences between Oracle Application Express release 4.0 and later releases.

This section describes the differences between Oracle Application Express release 4.0 and later releases:

- **Workspace Home Page** - The Workspace home page displays differently in newer versions of Oracle Application Express. In the newer versions of Application Express, the components displayed on the Workspace home page are **App Builder**, **SQL Workshop**, **Team Development**, and **App Gallery**.
- **Administration** - In the newer versions of Oracle Application Express, **Administration** appears as a menu on the Workspace home page.
- **Account** - In the newer versions of Oracle Application Express, the **Account Menu** appears at the top of the Application Express home page and allows you to set the following preferences:
 - Default Schema
 - Default Date Format
 - Enable Legacy Component View
 - Run application in new window - Specify how Application Express manages windows or tabs when running applications from Application Builder
 - Use single window to run all applications - Specify whether different applications reuse the same application runtime window or tab
- **Accessibility Mode** - In the newer versions of Oracle Application Express, the Workspace home page includes Accessibility Mode list that allows you to select either **Standard** or **Screen Reader** mode.

Index

A

Accessing Oracle Application Express
Oracle Database Cloud, [2-1](#)

ACL
fixing invalid, [4-33](#), [A-32](#), [B-34](#)

Administration Services
accessing with Embedded PL/SQL Gateway, [A-26](#)
accessing with Oracle HTTP Server, [B-27](#)
accessing with Oracle REST Data Services, [4-27](#)

apex_epg_config.sql, [A-10](#), [A-11](#)

apex_epg_config.sql script, [A-11](#)

APEX_PUBLIC_USER account
about, [4-9](#), [B-9](#)
changing password, [4-10](#), [B-10](#)
configuring, [4-9](#), [B-9](#)
expiration in Oracle Database 11g, [4-11](#), [B-11](#)
unlocking, [4-10](#), [B-9](#)

apexins.sql, [4-5](#), [A-5](#), [B-5](#)

application container
about, [3-3](#)

Application Express
about users, [4-26](#), [A-26](#), [B-27](#)
about workspaces, [4-26](#), [A-26](#), [B-27](#)
Administration Services, [4-27](#), [A-26](#), [B-27](#)
browser requirement, [1-3](#)
creating workspace manually, [4-28](#), [A-27](#), [B-28](#)
database requirement, [1-1](#), [4-2](#), [A-2](#), [B-2](#)
disk space requirements, [1-4](#)
download and install in non-CDB, [4-4](#), [A-4](#), [B-3](#), [B-4](#)
download and install locally in PDB, [4-4](#), [A-4](#), [B-3](#), [B-4](#)
getting started, [4-26](#), [A-25](#), [B-27](#)
incompatible versions, [3-16](#)
installing translated versions, [4-24](#), [A-23](#), [B-25](#)
Oracle REST Data Services requirement, [4-2](#)
patching in CDBs, [3-16](#)
patching in PDBs, [3-17](#), [3-18](#)

Application Express (*continued*)
pre-installation tasks, [4-2](#), [4-3](#), [A-2](#), [A-3](#), [B-2](#), [B-3](#)
reinstalling in CDBs, [3-10](#)
release number, [2-9](#)
signing in to workspace, [4-30](#), [A-30](#), [B-31](#)
uninstalling in CDBs, [3-7](#)
XML DB requirement, [1-4](#)

Application Express Administration Services, [4-28](#), [A-27](#), [B-29](#)

applications
specifying compatibility mode, [2-11](#)
upgrading, [2-11](#)

apxchpwd.sql, [4-9](#), [A-9](#), [B-8](#), [F-3](#)

changing Instance Administrator account
password, [4-8](#), [A-8](#), [B-7](#)

creating Instance Administrator account, [4-8](#), [A-8](#), [B-7](#)

running, [4-8](#), [A-8](#), [B-8](#)

unlockin Instance Administrator account, [4-8](#), [A-8](#), [B-7](#)

updating Instance Administrator account, [4-8](#), [A-8](#), [B-7](#)

apxdevrm_con.sql, [A-37](#), [B-39](#)

apxdevrm.sql, [4-37](#), [A-36](#), [B-37](#)

apxdvins_con.sql, [A-36](#), [B-38](#)

apxdvins.sql, [4-36](#), [A-35](#), [B-36](#)

apxrtins.sql, [4-5](#), [A-5](#), [B-5](#)

B

browser
requirement, [1-3](#)

C

CDB
application container, [3-3](#)

CDBs
patching Application Express, [3-16](#)
uninstalling Application Express, [3-7](#)

CDBs reinstalling Application Express, [3-10](#)

common Application Express from another CDB, [3-20](#)

compatibility mode
 specifying for, [2-11](#)

configure
 HTTP access, [3-6](#)

configuring
 embedded PL/SQL gateway, [A-9](#)
 Oracle HTTP Server, [B-11](#)

creating
 application PDB, application root seed, [3-6](#)
 application seed, [3-5](#)

D

database installation
 SHARED_SERVERS parameter, [A-25](#)

database requirement
 Application Express, [1-1](#), [4-2](#), [A-2](#), [B-2](#)

development environment
 changing to runtime, [4-36](#), [A-35](#), [A-37](#), [B-37](#),
[B-39](#)

disk space
 requirements, [1-4](#)

E

embedded PL/SQL gateway
 about, [2-16](#)
 apex_epg_config.sql, [A-10](#)
 configuring, [A-1](#), [A-9](#)
 configuring behind load balancer, [A-15](#)
 configuring behind reverse proxy, [A-15](#)
 installing, [A-1](#)
 security considerations, [2-15](#)
 SHARED_SERVERS parameter, [A-25](#)
 verifying HTTP server port, [A-13](#)

Embedded PL/SQL Gateway, [A-30](#)

environment, configuring, [2-4](#)

F

full development environment
 about, [4-35](#), [A-34](#), [B-35](#)
 converting to runtime, [4-36](#), [A-35](#), [A-37](#),
[B-37](#), [B-39](#)
 installing, [4-5](#), [A-5](#), [B-5](#)

G

getting started
 Application Express, [4-26](#), [A-25](#), [B-27](#)

I

images
 copying, [B-12](#)
 copying when upgrading, [B-12](#)

installation
 about copying images directory, [B-12](#)
 configuring embedded PL/SQL gateway, [A-9](#)
 configuring Oracle HTTP Server, [B-11](#)
 downloading Oracle REST Data Services,
[4-12](#), [4-13](#)
 enabling network services, [4-17](#), [A-16](#), [B-18](#)
 installing in other languages, [4-23](#), [A-22](#),
[B-24](#)
 managing JOB_QUEUE_PROCESSES,
[4-21](#), [A-20](#), [B-22](#)
 overview, [2-3](#)
 patch sets, [2-3](#)
 performing security tasks, [4-21](#), [A-20](#), [B-22](#)
 planning, [2-2](#)
 point releases, [2-3](#)
 process, [2-3](#)
 requirements, [1-1](#), [2-2](#)
 restart processes, [4-9](#), [A-9](#), [B-9](#)
 signing in, [4-26](#), [A-25](#), [B-27](#)
 understanding, [2-2](#)
 updating images directory, [A-11](#)
 verifying validity, [E-2](#)

installation option
 full development environment, [4-5](#), [A-5](#), [B-5](#)
 runtime environment, [4-5](#), [A-5](#), [B-5](#)

installing
 Application Express in application container,
[3-3](#)
 failures, [E-2](#)
 Oracle REST Data Services, [4-12](#), [4-13](#)
 other languages, [4-23](#), [A-22](#), [B-24](#)

Instance Administrator
 about, [B-7](#)

Instance administrator account
 about, [4-7](#), [A-7](#)
 creating, [4-7](#), [A-7](#), [B-7](#)
 running apxchpwd.sql, [4-8](#), [A-8](#), [B-7](#)
 updating, [4-7](#), [A-7](#), [B-7](#)

Instance Administrator account
 changing password, [4-8](#), [B-7](#)
 creating, [4-8](#), [B-7](#)
 creating password, [4-8](#), [B-7](#)
 unlocking, [4-8](#), [B-7](#)

J

JOB_QUEUE_PROCESSES, [4-21](#), [A-20](#), [B-22](#)
 changing number of, [4-23](#), [A-22](#), [B-24](#)
 viewing from SQL*Plus, [4-23](#), [A-22](#), [B-24](#)

JOB_QUEUE_PROCESSES (*continued*)
 viewing in installation log, [4-22](#), [A-21](#), [B-23](#)
 viewing number of, [4-22](#), [A-21](#), [B-23](#)
 viewing on About page, [4-22](#), [A-21](#), [B-23](#)

L

log file, [E-1](#)
 login credentials
 recovering workspace name, [2-7](#)

M

MEMORY_TARGET
 checking, [1-2](#)
 Multitenant Architecture, [3-1](#)

N

network services
 enabling, [4-17](#), [A-16](#), [B-18](#)
 granting connect privileges, [4-18](#), [A-17](#), [B-19](#)
 granting connect privileges, Oracle Database
 12c, [4-19](#), [A-18](#), [B-20](#)
 invalid ACL error, [4-20](#), [A-19](#), [B-21](#)
 non-CDB
 installing in, [4-4](#), [A-4](#), [B-3](#), [B-4](#)

O

Oracle Application Express
 configuring your environment, [2-4](#)
 creating users, [4-29](#), [A-28](#), [B-29](#)
 installing, [2-10](#)
 Oracle Database, [2-10](#)
 Oracle Application Server 10g
 editing dads.conf, [B-13](#)
 Oracle HTTP Server
 about, [2-17](#)
 configuring, [B-1](#), [B-11](#)
 configuring behind load balancer, [B-17](#)
 configuring behind reverse proxy, [B-17](#)
 copying images directory, [B-12](#)
 editing dads.conf, [B-13](#)
 installing, [B-1](#)
 signing in, [B-31](#)
 stopping and starting, [B-15](#)
 using with Oracle RAC, [2-18](#)
 Oracle HTTP Server 11g
 editing dads.conf, [B-13](#)
 Oracle HTTP Server 12c
 editing dads.conf, [B-13](#)
 Oracle Real Application Clusters (Oracle RAC)
 selecting an HTTP Server, [2-18](#)

Oracle Real Application Clusters (Oracle RAC) (*continued*)
 shutting down instances, [4-2](#), [A-2](#), [B-2](#)
 Oracle REST Data Services
 about, [2-15](#)
 configuring, [4-1](#)
 configuring behind load balancer, [4-13](#)
 configuring behind reverse proxy, [4-13](#)
 configuring RESTful Services, [4-11](#), [4-15](#)
 Copying the Images Directory, [4-15](#)
 determining, [2-10](#)
 downloading, [4-12](#), [4-13](#)
 installing, [4-1](#), [4-12](#), [4-13](#)
 release number, [2-9](#)
 Validating the Oracle REST Data Services
 Installation, [4-15](#)
 Web Server HTTP POST Request Limits,
 [4-14](#)
 Oracle REST Data Services requirement, [4-2](#)
 Oracle XML DB Protocol server
 disabling, [4-14](#), [B-17](#)
 updating images directory, [A-11](#)
 verifying HTTP server port, [A-13](#)
 Oracle XML DB Protocol Server
 enabling, [A-14](#)
 Oracle XML DB Protocol Server, verifying HTTP
 server port, [A-12](#)
 ords.version.number.zip, [4-13](#)
 OTN installation
 copying images, [B-12](#)
 overview, [2-3](#)

P

password
 resetting from Sign In page, [2-6](#)
 patch sets, [2-3](#)
 PDB
 application container, [3-3](#)
 installing in, [4-4](#), [A-4](#), [B-3](#), [B-4](#)
 PDBs
 patching Application Express, [3-17](#), [3-18](#)
 performance
 optimizing, [4-34](#), [A-33](#), [B-34](#)
 performance optimization
 about, [4-34](#), [A-33](#), [B-34](#)
 expired header attribute, [4-34](#), [A-33](#), [B-34](#)
 gzip compression, [4-34](#), [A-33](#), [B-34](#)
 Plug-in non-CDB, [3-16](#)
 Plug-in PDB, [3-16](#)
 plugging in PDBs
 Application Express from another CDB, [3-14](#)
 Application Express in root, [3-13](#)

plugging in PDBs (*continued*)

- Application Express is not contained in the root container
 - local Application Express from another CDB, [3-21](#)
- Application Express is not in the root container of the target CDB
 - non-CDB or PDB with no Application Express, [3-21](#)
 - non-CDB with Application Express, [3-20](#)
- Application Express not contained root, [3-20](#)
- Application Express not in the root of CDB, [3-19](#)
- Application Express root another CDB, [3-15](#)
- non-CDB with Application Express, [3-13](#)

point releases, [2-3](#)

post-installation tasks

- installing other languages, [4-23](#), [A-22](#), [B-24](#)

pre-installation tasks, [4-2](#), [A-2](#), [B-2](#)

prior installations

- removing when upgrading, [4-32](#), [A-31](#), [B-32](#)

R

release number

- Oracle REST Data Services, [2-9](#)
- viewing, [2-9](#)

release numbering

- convention, [2-8](#)

removing schemas from prior installation, [4-33](#)

requirements, [1-1](#)

- browser, [1-3](#)
- database, [1-1](#), [4-2](#), [A-2](#), [B-2](#)
- disk space, [1-4](#)
- MEMORY_TARGET, [1-2](#)
- Oracle XML DB, [1-4](#)

RESTful Services

- configuring, [4-11](#)

reverting to previous release, [2-12](#)

running

- apex_epg_config.sql, [A-11](#)
- apexins.sql, [4-5](#), [A-5](#), [B-5](#)
- apxchpwd.sql, [4-9](#), [A-9](#), [B-8](#), [F-3](#)
- apxdevm_con.sql, [A-37](#), [B-39](#)
- apxdevm.sql, [4-37](#), [A-36](#), [B-37](#)
- apxdvins_con.sql, [A-36](#), [B-38](#)
- apxdvins.sql, [4-36](#), [A-35](#), [B-36](#)
- apxrtins.sql, [4-5](#), [A-5](#), [B-5](#)

runtime environment

- about, [2-18](#), [4-35](#), [A-34](#), [B-35](#)
- converting to development environment, [4-35](#), [A-34](#), [A-36](#), [B-36](#), [B-38](#)
- installing, [4-5](#), [A-5](#), [B-5](#)

S

Secure Sockets Layer (SSL), [4-21](#), [A-20](#), [B-22](#)

security considerations

- embedded PL/SQL gateway, [2-15](#)

Sign In dialog

- requesting a workspace, [2-5](#)

SSL, [4-21](#), [A-20](#), [B-22](#)

static file support, configuring, [B-15](#)

T

translated version

- installing, [4-23](#), [A-22](#), [B-24](#)

translated versions

- about installing, [4-24](#), [A-23](#), [B-25](#)

troubleshooting, [E-1](#)

- cleaning up after failed installation, [E-2](#)
- images, [E-37](#)
- reviewing log file, [E-1](#)

U

upgrade post installation tasks, [4-31](#), [A-31](#), [B-32](#)

- fixing invalid ACL, [4-33](#), [A-32](#), [B-34](#)
- removing prior installation, [4-32](#), [A-31](#), [B-32](#)
- removing schemas from prior installation, [4-33](#), [A-32](#), [B-33](#)
- verifying prior installation, [4-32](#), [A-31](#), [B-33](#)

upgrading

- about, [2-7](#)
- Application Express in application container, [3-3](#)
- compatibility mode, [2-11](#)
- copying images, [B-12](#)
- environment clean-up, [2-12](#)
- existing applications, [2-11](#)
- reverting, [2-12](#)
- sample scenarios, [2-9](#)
- testing, [2-12](#)

user accounts

- creating, [4-29](#), [A-28](#), [B-29](#)

V

verifying

- application container installation, [3-4](#)

version, [2-10](#)

W

Web browser

- requirements, [1-3](#)

web listener
 choosing, [2-13](#)
 comparing each, [2-14](#)
 embedded PL/SQL gateway, [2-16](#)
 Oracle HTTP Server, [2-17](#)
 Oracle REST Data Services, [2-15](#)

Web listener
 embedded PL/SQL gateway, [A-1](#)
 Oracle HTTP Server, [B-1](#)
 Oracle REST Data Services, [4-1](#)

Web Listener
 embedded PL/SQL gateway, [1-3](#)
 HTTP server, [1-3](#)
 Oracle REST Data Services, [1-3](#)

Web Listener (*continued*)

workspace
 creating, [4-28](#), [A-27](#), [B-28](#)
 recovering workspace name, [2-7](#)
 requesting from Sign In dialog, [2-5](#)
 signing in, [4-30](#), [A-30](#)

workspace name
 recovering, [2-7](#)

X

XML DB
 requirement, [1-4](#)