Oracle® Database Free

Installation Guide

23c for Linux x86-64

F77392-07

September 2023

Oracle Database Free Installation Guide, 23c for Linux x86-64

F77392-07

Primary Author: Prakash Jashnani

Contributing Authors: Subhash Chandra, Rhonda Day

Contributors: Santanu Datta, Malai Stalin, Gordon Larimer, Sudeepthi Siddavatam, Gerald Venzl, Christopher Jones, Vasundhara Venkatasubramanian, Alex Keh, Subrahmanyam Kodavaluru

Contents

Preface

This guide explains how to install and configure Oracle Database Free on Linux x86–64.

This guide also provides information about resources available to develop applications and how to remove the database software.

Audience

This guide is intended primarily for application developers who are either developing applications or converting applications to run in the Oracle Database environment.

Oracle Database Free is a free version of the world's most advanced database. Oracle Database Free is easy to install, easy to manage, and easy to develop with. With Oracle Database Free, you use an intuitive, browser-based interface to administer the database, create tables, views, and other database objects, import, export, and view table data, run queries and SQL scripts, and generate reports.

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.

Command Syntax

Refer to these command syntax conventions to understand command examples in this guide.

Convention	Description
$	Bourne or BASH shell prompt in a command example. Do not enter the prompt as part of the command.

Convention	Description
%	C Shell prompt in a command example. Do not enter the prompt as part of the command.
#	Superuser (root) prompt in a command example. Do not enter the prompt as part of the command.
monospace	UNIX command syntax
backslash \	A backslash is the UNIX and Linux command continuation character. It is used in command examples that are too long to fit on a single line. Enter the command as displayed (with a backslash) or enter it on a single line without a backslash: `dd if=/dev/rdsk/c0t1d0s6 of=/dev/rst0 bs=10b \  count=10000`
braces { }	Braces indicate required items: `.DEFINE {macro1}`
brackets [ ]	Brackets indicate optional items: `cvtcrt termname [outfile]`
ellipses ...	Ellipses indicate an arbitrary number of similar items: `CHKVAL fieldname value1 value2 ... valueN`
italic	Italic type indicates a variable. Substitute a value for the variable: *`library_name`*
vertical line |	A vertical line indicates a choice within braces or brackets: `FILE filesize [K|M]`

Related Documents

To help you with your development efforts, consult the books in the development category of the *Oracle Database documentation set.*

For more information, see these documents in the Oracle Database documentation set:

- *Oracle Database SQL Language Reference*
- *Oracle Database PL/SQL Language Reference*
- *Oracle Database PL/SQL Packages and Types Reference*
- *Oracle Database JSON Developer's Guide*
- *Oracle Database Development Guide*

- *Oracle Database Administrator's Guide*
- *Oracle Database SecureFiles and Large Objects Developer's Guide*
- *Oracle Database Object-Relational Developer's Guide*
- *Oracle Database Concepts*
- *Oracle Database Sample Schemas*
- *Oracle APEX Documentation*
- *Application Development with Oracle Database*

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.
italic	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.

1

Installation Guide

Welcome to *Oracle Database Free Installation Guide for Linux x86-64*.

This guide covers the following topics:

2
Introduction

Oracle Database Free is a fully free edition of the Oracle Database.

Development Environments

Oracle Database Free supports the following development environments:

- **Java**: Develop and deploy modern database-bound Java Web applications (Servlets), modules (Microservices) or standalone Java frameworks using the Oracle JDBC Driver, the Universal Connection Pool (UCP), and the Database-embedded JVM (for in-place, server-side processing).

  Visit https://www.oracle.com/database/technologies/appdev/jdbc.html for more information.

- **C and C++** : Developers can use Oracle Call Interface (OCI) and Oracle C++ Call Interface (OCCI) to create high performance programs accessing Oracle Database Free. ODBC and the ODPI-C wrapper over OCI are also usable.

  Visit https://www.oracle.com/database/technologies/appdev/oci.html for more information.

- **.NET, Visual Studio, and Visual Studio Code:**: Developers can use Oracle Data Provider for .NET (ODP.NET) for C# and VB.NET data access to any Oracle Database. At design-time, they can use Oracle Developer Tools for Visual Studio or Oracle Developer Tools for VS Code for full development life cycle support.

  Visit https://www.oracle.com/database/technologies/appdev/dotnet.html for more information

  For walk-throughs on creating a .NET application with Oracle Database Free:

  Visit https://www.oracle.com/tools/technologies/quickstart-dotnet-for-oracle-database.html

  The walk-throughs cover several development scenarios, including with VS Code, Visual Studio, and command line.

- **Oracle SQL Developer**: Oracle SQL Developer is a graphical version of SQL*Plus that gives database developers a convenient way to perform basic tasks. You can connect to any target Oracle Database Free schema using standard Oracle database authentication. Once connected, you can perform operations on objects in the database.

  Download and install Oracle SQL Developer from:

  https://www.oracle.com/database/sqldeveloper/

- **Oracle SQL Developer Web**: Included with Oracle REST Data Services, Oracle SQL Developer Web is the web-based version of Oracle SQL Developer that enables you to run queries and scripts, create database objects, build data models, and monitor database activity.

- **Oracle Developer Tools for VS Code**: This free Visual Studio Code extension enables you to edit and run SQL and PL/SQL for Oracle Database and Oracle Autonomous Database.

  Download and install Oracle Developer Tools for VS Code from the Visual Studio Code Marketplace:

https://marketplace.visualstudio.com/items?itemName=Oracle.oracledevtools

- **Oracle Application Express**: Oracle Application Express (APEX) is a rapid web application development tool for the Oracle database.

  Download and install Application Express from:

  https://www.oracle.com/database/technologies/appdev/rest.html

- **Oracle REST Data Services (ORDS)**: ORDS makes it easy to develop modern REST interfaces for relational data in the Oracle Database and the Oracle Database JSON Document Store.

  Download and install ORDS from:

  https://www.oracle.com/database/technologies/appdev/rest.html

- **SODA** (Simple Oracle Document Access) APIs that let you develop NoSQL-style applications against collections of JSON documents. Native language SODA drivers are available for common languages.

  Visit https://docs.oracle.com/en/database/oracle/simple-oracle-document-access/ for more information.

Scripting Languages

You can use **scripting languages** such as:

- **Python**

  The `python-oracledb` driver is a Python programming language extension module allowing Python programs to connect to Oracle Database.

  Visit https://oracle.github.io/python-oracledb/ for more information.

- **Node.js**

  The node-oracledb driver allows Node.js applications to access Oracle Database.

  Visit https://oracle.github.io/node-oracledb/ for more information.

- **PHP**

  Access Oracle Database with the PHP OCI8 extension or the PDO_OCI Driver. PHP OCI8 and PDO_OCI are part of the PHP open source project.

  Visit https://www.php.net/oci8 for more information on PHP OCI8 and https://www.php.net/pdo_oci for more information on PDO_OCI.

- **Go**

  Access Oracle Database using the open source `godror` driver.

  Visit https://pkg.go.dev/github.com/godror/godror for more information.

- **ROracle**

  ROracle is an open source R package supporting a DBI-compliant Oracle driver based on the high performance OCI library.

  Visit https://www.oracle.com/database/technologies/roracle-downloads.html for more information about ROracle.

- **Ruby**

  Build Ruby and Ruby on Rails applications using the `ruby-oci8` driver or `JRuby` with the Oracle Enhanced Adapter for ActiveRecord.

Visit:

- https://www.rubydoc.info/gems/ruby-oci8/ for information about `ruby-oci8`.
- https://www.jruby.org/ for information about JRuby.
- https://github.com/rsim/oracle-enhanced for information about Oracle Enhanced Adapter.

- **Rust**

  The open source `rust-oracle` driver lets Rust programs access Oracle Database.

  Visit https://crates.io/crates/oracle for more information.

- **Others**

  Accessing Oracle Database Free from other languages is possible using community drivers.

Oracle Call Interface (OCI) Demonstration Programs

A set of OCI demonstration programs and their corresponding project files are available in the *$ORACLE_HOME*`/demo` subdirectory after an Oracle Database Free installation.

You can run these OCI demonstration programs to familiarize yourself with the steps involved in developing OCI applications. Oracle Database Free does not support generating the client shared library. The build option in `demo_rdbms.mk` is not valid for Oracle Database Free. You can compile and link application and demo programs with the provided header files. Because the object (`.o's`) and archive (`.a's`) libraries are not available in the installed location, you cannot use `genclntsh` and `genclntst`.

Examples

You can download and install Oracle Database Examples in an existing Oracle home to view the product demonstrations.

See Examples Installation Guide for more information about products available on the Oracle examples media

Examples are also available from https://github.com/oracle/oracle-db-examples

Learn More About Oracle Database Free

- Oracle Database Free home page:

  https://www.oracle.com/database/free/

- Oracle Database Free Discussion Forum:

  You can search the Oracle Database Free Discussion Forum for answers to problems already discussed and post new questions to the community for answers.

  https://forums.oracle.com/ords/apexds/domain/dev-community/category/oracle-database-free

3
Requirements

You must have `root` user credentials to install Oracle Database Free.

The system must meet the following software requirements:

- System Requirements
- Swap Space Requirements
- Server Component Kernel Parameter Requirements

System Requirements

This table lists the system requirements for Oracle Database Free.

Table 3-1 Oracle Database Free System Requirements

Requirement	Value
Operating system	See Oracle Database Installation Guide for Linux for the list of supported Linux distributions and the minimum operating system requirements for each x86-64 Linux platform.
Network protocol	The following protocols are supported: • IPC • UDP • TCP/IP • TCP/IP with SSL
RAM	1 GB RAM minimum. 2 GB RAM recommended.
Disk space	10 GB minimum.

Swap Space Requirements

For Oracle Database Free, Oracle recommends a minimum swap space of 2 GB or twice the size of RAM, whichever is lesser.

Server Component Kernel Parameter Requirements

The Oracle Database Preinstallation RPM checks your system for kernel parameter settings. If the kernel parameter values of your system are less than the values listed in this table, then the Oracle Database Preinstallation RPM sets the recommended minimum kernel parameter values for you.

The values set in the `/etc/sysctl.d/97-oracle-database-sysctl.conf` file persist on system restarts.

Table 3-2 Kernel Parameter Settings Required for Oracle Database Free

Kernel Parameter	Setting
semmsl	250
semmns	32000
semopm	100
semmni	128
shmmax	4398046511104
shmmni	4096
shmall	1073741824
file-max	6815744
aio-max-nr	1048576
ip_local_port_range	9000–65500
panic_on_oops	1
rmem_default	262144
rmem_max	4194304
wmem_default	262144
wmem_max	1048576

4

Licensing Restrictions

For more information on licensing details, see *Oracle Database Database Licensing Information User Manual*

This section covers the following topics:

Oracle Database Free CPU Limitations

Oracle Database Free limits itself automatically to two cores for processing. For example, on a computer with 2 dual-core CPUs (four cores), if a large number of database clients try to simultaneously run CPU-intensive queries, then Oracle Database Free will process the queries at the rate of just two cores even if more CPU capacity is available.

Oracle Database Free Installation and Runtime Restrictions

Oracle Database Free restricts itself to only one installation per logical environment. The logical environment can either be a virtual host such as a VM or container, or a physical host. If you attempt to start more than one Oracle Database Free installation in such a logical environment, then an `ORA-00442: Oracle Database Free single instance violation error` is displayed and your database will not start.

This does not affect any existing installation or new installations of Oracle Database Standard Edition 2 or Oracle Database Enterprise Edition.

Oracle Database Free User Data Limitations

The maximum amount of user data in Oracle Database Free cannot exceed 12 GB. If the user data grows beyond this limit, then the system displays an `ORA-12954: The request exceeds the maximum allowed database size of 12 GB` error.

Oracle Database Free RAM Limitation

The maximum amount of RAM for Oracle Database Free cannot exceed 2 GB, even if more is available.

5

Installing Oracle Database Free

You can install Oracle Database Free using RPM packages.

An RPM-based installation performs preinstallation checks, extracts the database software, reassigns ownership of the extracted software to the preconfigured user and groups, maintains the Oracle inventory, and runs all root operations required to configure the Oracle Database software for a single-instance Oracle Database creation and configuration.

The RPM–based installation process detects when the minimum requirements for an installation are not met and prompts you to finish these minimum preinstallation requirements.

This section covers the following topics:

- Installing Oracle Database Free Using RPM Packages
- Performing a Silent Installation
- Setting Oracle Database Free Environment Variables

Installing Oracle Database Free Using RPM Packages

Perform the following steps to install and configure Oracle Database Free using RPM packages.

Before you install Oracle Database 23c Free, uninstall any existing Oracle Database XE or Oracle Database Free or database with the SID `XE` or `FREE` from the target system. See, Deinstalling Oracle Database Free.

Oracle Database Free installation uses approximately 9 GB disk space under `/opt`. If this disk partition does not have the required disk space available, you must add space or mount an alternative partition as `/opt/oracle`. This disk partition is the defined as Oracle base where the software and database will reside.

> **Note:**
>
> The Oracle Database Free installation does not support symbolic links (symlinks) for that disk.

Installing Oracle Database Free RPM

1. Use `sudo` to log in as `root`.

   ```
   sudo -s
   ```

2. Enable the Oracle Linux 8 Developer Channel.

For Oracle Linux 8 Minimal or Server Install:

```
dnf install -y oraclelinux-developer-release-el8
```

For Oracle Linux 8 Oracle Cloud-Based Instances:

```
dnf config-manager --set-enabled ol8_developer
```

For Oracle Linux 8 Cloud Developer Image:

No additional steps required.

For Oracle Linux 8 Container Image:

Depending on your host, enter the appropriate command from the preceding scenarios.

3. Install the Oracle Database Preinstallation RPM.

   Oracle Linux 8

   ```
   dnf -y install oracle-database-preinstall-23c
   ```

   Red Hat Enterprise Linux 8

   a. Go to the Oracle `yum` site:

      https://yum.oracle.com/repo/OracleLinux/OL8/developer/x86_64/

   b. Download the latest 23c Oracle Database Preinstallation RPM. For example, `oracle-database-preinstall-23c-1.0-1.el8.x86_64.rpm`

   c. Install the latest Preinstallation RPM. For example:

      ```
      dnf -y localinstall  oracle-database-
      preinstall-23c-1.0-1.el8.x86_64.rpm
      ```

   > **Note:**
   >
   > - The Oracle Database Preinstallation RPM automatically creates the Oracle installation owner and groups. It also sets up other kernel configuration settings as required for Oracle Database installations. If you plan to use job-role separation, then create the extended set of database users and groups depending on your requirements.
   > - Use the `-y` option if you want `dnf` to skip the package confirmation prompt.
   > - See, About DNF for more information about the `dnf` command.

4. Access the Oracle Database Free software download page:

   https://www.oracle.com/database/technologies/free-downloads.html

5. Download the `oracle-database-free-23c-1.0-1.el8.x86_64.rpm` RPM file required for performing an RPM-based installation to a directory of your choice.

6. Install the database software.

```
dnf -y localinstall oracle-database-free-23c-1.0-1.el8.x86_64.rpm
```

> **Note:**
>
> Review the RPM log files to determine the system configuration changes. For example, review `/var/log/oracle-database-preinstall-23c/results/orakernel.log`.

The installation of the Oracle Database software is now complete.

Creating and Configuring an Oracle Database

The configuration script creates a container database (`FREE`) with one pluggable database (`FREEPDB1`) and configures the listener at the default port (1521).

You can modify the configuration parameters by editing the `/etc/sysconfig/oracle-free-23c.conf` file.

The parameters set in this file are explained in detail in the silent mode installation procedure: Performing a Silent Installation.
To create the database with the default settings:

1. Log in as `root` using `sudo`.

```
sudo -s
```

2. Run the service configuration script:

```
/etc/init.d/oracle-free-23c configure
```

At the command prompt, specify a password for the `SYS`, `SYSTEM`, and `PDBADMIN` administrative user accounts. Oracle recommends that your password should be at least 8 characters in length, contain at least 1 upper case character, 1 lower case character and, 1 digit [0-9].

> **See Also:**
>
> The same password will be used for these accounts. The password should conform to the Oracle recommended standards. See Oracle Database Security Guide for more information about guidelines for securing passwords

After the configuration completes, the database and listener are started.

Table 5-1 Configuration, Database Files and Logs Location

File Name and Location	Purpose
`/opt/oracle`	Oracle base. This is the root of the Oracle Database Free directory tree.

Table 5-1 (Cont.) Configuration, Database Files and Logs Location

File Name and Location	Purpose
`/opt/oracle/product/23c/dbhomeFree`	Oracle home. This home is where the Oracle Database Free is installed. It contains the directories of the Oracle Database Free executables and network files.
`/opt/oracle/oradata/FREE`	Database files.
`/opt/oracle/diag` subdirectories	Diagnostic logs. The database alert log is `/opt/oracle/diag/rdbms/free/FREE/trace/alert_FREE.log`
`/opt/oracle/cfgtoollogs/dbca/FREE`	Database creation logs. The `FREE.log` file contains the results of the database creation script execution.
`/etc/sysconfig/oracle-free-23c.conf`	Configuration default parameters.
`/etc/init.d/oracle-free-23c`	Configuration and services script.

> **Note:**
>
> If a host does not have any IP address other than loop back address assigned (typically in a docker or in an another container environment), Oracle Net Configuration Assistant (Oracle NETCA) may fail during the installation with the error `No valid IP Address returned for the host` *`hostname`* `in netca trace log`. Please assign an IP address and retry the installation.

Performing a Silent Installation

You can install Oracle Database Free using silent mode. You can use this mode to perform an embedded install of Oracle Database Free with your application, or unattended operation.

To perform a silent installation, you must enter a password for the administrative accounts as a parameter to the script, or specify it in the configuration file.

1. Create a wrapper shell script to perform the silent installation. The script should contain commands similar to the following:

   For Oracle Linux 8

   ```
   #!/bin/bash

   yum -y localinstall /downloads/oracle-database-
   free-23c-1.0-1.el8.x86_64.rpm > /free_logs/FREEsilentinstall.log
   2>&1

   /etc/init.d/oracle-free-23c configure >> /free_logs/
   FREEsilentinstall.log 2>&1
   ```

Alternatively, you can enter the password in the script, such as:

```
(echo "password"; echo "password";) | /etc/init.d/oracle-free-23c
configure >> /free_logs/FREEsilentinstall.log 2>&1
```

Replace *password* with a password that is secure. The password entered should be at least 8 characters in length, contain at least 1 uppercase character, 1 lower case character, and 1 digit [0-9].

2. Make the wrapper script executable.

```
chmod +x myscript.sh
```

3. Run the script as `root` using `sudo`.

```
sudo ./myscript.sh
```

The Oracle Database free Oracle home is `/opt/oracle/product/23c/dbhomeFREE`

For details of the installation, review the `/free_logs/FREEsilentinstall.log` file.

The database is configured with the default settings. It is not necessary to modify these parameters unless you have specific requirements. Make a copy of the configuration file `/etc/sysconfig/oracle-free-23c.conf` before modifying it. Make your modifications after the RPM install and before configuring the database.

The `/etc/sysconfig/oracle-free-23c.conf` configuration file sets the following:

- `LISTENER_PORT`: A valid listener numeric port value for the database listener. Do not specify any value for automatic port assignment.
- `CHARSET`: Character set of the database. This is set to `AL32UTF8`.
- `DBFILE_DEST` Database file directory. By default, the database files are stored in the Oracle base `/opt/oracle/oradata` subdirectory. You can also create your own database file directory. However, the permissions for this file path should be owned by the `oracle` user.
- `SKIP_VALIDATIONS`: Skip validation for memory and disk space. Default is `false`.

Caution:

When you modify and save a file containing the plain text password, provide the ownership of the file only to the Oracle software installation owner (`oracle`) user. Change the permissions on the file to `600`. Oracle recommends that database administrators or other administrators delete or secure such files containing plain text passwords when they are not in use.

Note:

The password should conform to the Oracle recommended standards. See Oracle Database Security Guide for more information about guidelines for securing passwords

The database creation logs are located under Oracle base in the `/opt/oracle/cfgtoollogs/dbca/` subdirectory.

Setting Oracle Database Free Environment Variables

After you install and configure Oracle Database Free, set the environment before you use Oracle Database Free.

Use the `oraenv` and `coraenv` scripts to set your environment variables.

For example, to set your environment variables in Bourne, Bash, or Korn shell without being prompted by the script:

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
```

```
ORACLE_HOME = [] ? /opt/oracle/product/23c/dbhomeFree
The Oracle base has been set to /opt/oracle
```

For C shell:

```
setenv ORACLE_SID FREE
setenv ORACLEENV_ASK NO
source /opt/oracle/product/23c/dbhomeFree/bin/coraenv
```

6
Connecting to Oracle Database Free

Connecting Locally using OS Authentication

When you install Oracle Database Free, the oracle user is granted SYSDBA privileges. You can use the following commands to connect to the database.

```
$ cd $ORACLE_HOME/bin
$ ./sqlplus / as sysdba
```

These commands connect you to the root container `CDB$ROOT` of the multitenant database (CDB) as database user `SYS`. This method of connecting to the database works even if the Net Services listener is not running.

An output similar to the following confirms that you are now connected to the database.

```
SQL*Plus: Release 23.0.0.0.0 - Production on Tue Aug 15 03:27:05 2023
Version 23.3.0.23.09

Copyright (c) 1982, 2023, Oracle.  All rights reserved.

Connected to:
Oracle Database 23c Free Release 23.0.0.0.0 - Develop, Learn, and Run for
Free
Version 23.3.0.23.09
```

Net Services Listener and Default Services

The Net Services database listener for Oracle Database Free allows you to connect to the database over TCP/IP from the same machine or other machines on the network. The configuration of the Listener can be viewed using the following commands run from the command prompt:

```
$ cd $ORACLE_HOME/bin
$ lsnrctl status


LSNRCTL for Linux: Version 23.0.0.0.0 - Production on 15-AUG-2023 03:30:11

Copyright (c) 1991, 2023, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=dbhost.example.com)
(PORT=1521)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 23.0.0.0.0 - Production
Start Date                15-AUG-2023 03:19:38
Uptime                    0 days 0 hr. 8 min. 3 sec
```

```
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Default Service           FREE
Listener Parameter File /opt/oracle/product/23c/dbhomeFree/network/
admin/listener.ora
Listener Log File /opt/oracle/diag/tnslsnr/dbhost/listener/alert/
log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=dbhost.example.com)
(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
Services Summary...
Service "FREE.dbhost.example.com" has 1 instance(s).
  Instance "FREE", status READY, has 1 handler(s) for this service...
Service "FREEXDB.dbhost.example.com" has 1 instance(s).
  Instance "FREE", status READY, has 1 handler(s) for this service...
Service "dbhost.example.com" has 1 instance(s).
  Instance "FREE", status READY, has 1 handler(s) for this service...
Service "freepdb1.dbhost.example.com" has 1 instance(s).
  Instance "FREE", status READY, has 1 handler(s) for this service...
The command completed successfully
```

The output from the `lsnrctl` command shows values of a number of important parameters:

- the port the listener listens on
- the list of services registered with the listener
- the name of the configuration file used by the listener
- the name of the log file

You must specify a service when connecting to the database through the listener. The default services created by Oracle Database Free are FREE and `FREEPDB1`. The Oracle Database Free service connects you to the `root` container of the database (`CDB$ROOT`) and the `FREEPDB1` service connects you to the default pluggable database `FREEPDB1`, created during installation. For each new pluggable database (PDB), there is a new default service created with the same name as the PDB.

> **Note:**
>
> If you shut down the Oracle Database Free instance, then the `lsnrctl` status command does not show any services that you can connect to.

Connecting to Oracle Database Using Easy Connect Naming Method

You can connect to the database using the following Easy Connect strings:

- Multitenant container database: *host*[:*port*]
- Pluggable database: *host*[:*port*]/*service_name*

`FREEPDB1` is the service name defined for the first PDB created by default. You can replace `FREEPDB1` with the name of another PDB you want to connect to.

Specifying the port is optional when the listener is setup with the default port `1521`. You must specify the port number if you use another port.

Connection strings for local connections were provided on the final screen of the install. If you are connecting from a remote computer, you must enter the `hostname` (where Oracle Database Free is installed) instead of `localhost`.

The Net Services database listener must be running on the database host of the specified port for the connections to succeed.

For example, you can connect to the `root` container of the database from a client computer with SQL*Plus using the following commands:

```
$ cd $ORACLE_HOME/bin
$ ./sqlplus system@dbhost.example.com:1521
```

You can connect to the default PDB `FREEPDB1` using the following commands:

```
$ cd $ORACLE_HOME/bin
$ ./sqlplus system@dbhost.example.com:1521/FREEPDB1
```

Replace `dbhost.example.com` with your database host name. If required, replace `1521` with the port number the listener listens on. You can replace `FREEPDB1` with the name of another PDB that you want to connect to.

To shorten connect strings or to avoid hard coding of the host name and the port in the application code and DBA scripts, you can define an alias for the connect string in the configuration file `ORACLE_HOME/network/admin/tnsnames.ora` on the database clients. See Configuring the Local Naming Method for more details.

7

Starting and Stopping Oracle Database Free

You can start and stop the database manually or set it to automatically start when the system shuts down or starts.

Shut Down and Start-Up Using SQL*Plus

You can shut down and start the database using SQL*Plus.

To shut down the database, log in as the `oracle` user with its environment variables set for access to Oracle Database Free, and run the following SQL*Plus command:

```
$ sqlplus / as sysdba
SQL> SHUTDOWN IMMEDIATE
```

To start the database:

```
SQL> STARTUP
SQL> ALTER PLUGGABLE DATABASE ALL OPEN;
```

Automating Shutdown and Start-Up

Oracle recommends that you configure the system to automatically start Oracle Database Free when the system starts, and to automatically shut it down when the system shuts down. Automating database shutdown guards against incorrect database shutdowns.

To automate the start up and shutdown of the listener and database, run the following commands as `root`:

```
$ sudo -s
```

For Oracle Linux 8:

```
# systemctl daemon-reload
# systemctl enable oracle-free-23c
```

Shutting Down and Starting Up Using the Configuration Services Script

After you configure the listener, you can run the Configuration Services Script to check the status of the database and listener.

Log in as `root` using `sudo`.

```
# /etc/init.d/oracle-free-23c status
```

The output of this command is similar to the following:

```
Status of the Oracle FREE 23c service:

LISTENER status: RUNNING
FREE Database status: RUNNING
```

Run the following commands as `root` using `sudo`.

```
$ sudo -s
```

Oracle Linux 8:

To start the listener and the database:

```
# systemctl start oracle-free-23c
```

To stop the database and the listener:

```
# systemctl stop oracle-free-23c
```

To stop and start the listener and the database:

```
# systemctl restart oracle-free-23c
```

8

Moving from Previous Versions of Oracle Database XE to Oracle Database Free

This topic explains how to export and import data between Oracle Database 21c Express Edition (XE) and Oracle Database 23c Free.

- Exporting and Importing Data between Oracle Database 21c XE and Oracle Database 23c Free
- Exporting and Importing Data between Oracle Database 23c (23.2) Free and Oracle Database 23c (23.3) Free

Note these points before you start the export and import process:

- You cannot use Oracle Database Upgrade Assistant (Oracle DBUA) to perform an upgrade.
- You cannot use Oracle Database Configuration Assistant (Oracle DBCA) to plug PDBs of previous versions to Oracle Database Free.
- Oracle Database 18c XE users must first move to 21c XE (See, Moving from Previous Versions of Oracle Database XE to XE 21c), and then export data from 21c XE to import them into Oracle Database 23c Free.

Exporting and Importing Data between Oracle Database 21c XE and Oracle Database 23c Free

Learn how to export and import data between Oracle Database 21c Express Edition (XE) and Oracle Database 23c Free.

Exporting Data

To export data from your 21c XE database:

1. As the `root` user, create a `/opt/dump` directory on the local file system for the `DUMP_DIR` directory object.

   ```
   mkdir /opt/dump
   chown -R oracle:oinstall /opt/dump
   chmod -R 760 /opt/dump
   ```

2. Perform the following steps for each pluggable database (PDB). The steps in this section are for the PDB `xepdb1`.

   a. Set the ORACLE_HOME and ORACLE_SID environment variables.

      ```
      export ORACLE_SID=XE
      export ORACLE_HOME=/opt/oracle/product/21c/dbhomeXE
      ```

b. Connect to the 21c XE database as user `SYS` using the `SYSDBA` privilege as the `oracle` user, and switch the container to `xepdb1`.

c. Create the directory object `DUMP_DIR` and grant `READ` and `WRITE` privileges on the `DUMP_DIR` directory to the `SYSTEM` user.

```
/opt/oracle/product/21c/dbhomeXE/bin/sqlplus / AS SYSDBA
SQL> ALTER SESSION SET CONTAINER=xepdb1;
SQL> CREATE DIRECTORY DUMP_DIR AS '/opt/dump';
SQL> GRANT READ, WRITE ON DIRECTORY DUMP_DIR TO SYSTEM;
```

d. Export data from your 21c XE PDB `xepdb1` to the dump folder.

```
/opt/oracle/product/21c/dbhomeXE/bin/expdp system/
system_password@dbhost.example.com:1521/xepdb1 full=Y
directory=DUMP_DIR dumpfile=expdb21c_xepdb1.dmp
logfile=expdb21c_xepdb1.log
```

> **Note:**
>
> - Replace `dbhost.example.com` with your database host name. If required, replace `1521` with the port number the listener listens on.
> - The default PDB name created is `xepdb1` in 21c and `freepdb1` in 23c.

3. Deinstall Oracle Database 21c XE if you plan to install 23c Free on the same system. See Deinstalling Oracle Database XE for more information
4. Install Oracle Database 23c Free.

Importing Data

To import data to your Oracle Database 23c Free, perform the following steps for each PDB. The steps in this section are for importing data from PDB `xepdb1` to `freepdb1`.

1. Set the ORACLE_HOME and ORACLE_SID environment variables.

```
export ORACLE_SID=FREE
export ORACLE_HOME=/opt/oracle/product/23c/dbhomeFree
```

2. Connect to the 23c Free database as user `SYS` using the `SYSDBA` privilege as the `oracle` user and switch the container to `freepdb1`.
3. Create the directory object `DUMP_DIR` and grant `READ` and `WRITE` privileges on the `DUMP_DIR` directory to the `SYSTEM` user.

```
/opt/oracle/product/23c/dbhomeFree/bin/sqlplus / AS SYSDBA
SQL> ALTER SESSION SET CONTAINER=freepdb1;
SQL> CREATE DIRECTORY DUMP_DIR AS '/opt/dump';
SQL> GRANT READ, WRITE ON DIRECTORY DUMP_DIR TO SYSTEM;
```

4. Import data to the 23c Free PDB `freepdb1` from the dump folder created during the export operation.

```
/opt/oracle/product/23c/dbhomeFree/bin/impdp system/
system_password@dbhost.example.com:1521/freepdb1 full=Y
directory=DUMP_DIR dumpfile=expdb21c_xepdb1.dmp
logfile=impdb23c_freepdb1.log
```

> **Note:**
>
> - Replace `dbhost.example.com` with your database host name. If required, replace `1521` with the port number the listener listens on.
> - The default PDB name created is `xepdb1` in 21c and `freepdb1` in 23c.

Ignore the following errors:

- ORA-31684: Object type TABLESPACE:"UNDOTBS1" already exists
- ORA-31684: Object type TABLESPACE:"TEMP" already exists
- ORA-31684: Object type TABLESPACE:"USERS" already exists
- ORA-31684: Object type USER:"PDBADMIN" already exists
- ORA-39083: Object type PROC_SYSTEM_GRANT failed to create with error:
- ORA-29393: user EM_EXPRESS_ALL does not exist or is not logged on
- ORA-39083: Object type NETWORK_ACL:TABLE:NACL$_ACE_EXP failed to create with error:
- ORA-01007: Reference to a variable not in SELECT clause.
- ORA-39342: Data Pump did not import dependent objects for NETWORK_ACL due to the previous error
- ORA-01007: Reference to a variable not in SELECT clause.

Exporting and Importing Data between Oracle Database 23c (23.2) Free and Oracle Database 23c (23.3) Free

Learn how to export and import data between Oracle Database 23c (23.2) Free and Oracle Database 23c (23.3) Free.

Exporting Data

To export data from Oracle Database 23.2 Free:

1. As the `root` user, create a `/opt/dump` directory on the local file system for the `DUMP_DIR` directory object.

```
mkdir /opt/dump
chown -R oracle:oinstall /opt/dump
chmod -R 760 /opt/dump
```

2. Perform the following steps for each pluggable database (PDB). The steps in this section are for the PDB `freepdb1`.

   a. Set the ORACLE_HOME and ORACLE_SID environment variables.

   ```
   export ORACLE_SID=FREE
   export ORACLE_HOME=/opt/oracle/product/23c/dbhomeFree
   ```

   b. Connect to the 23.2 Free database as user `SYS` using the `SYSDBA` privilege as the `oracle` user, and switch the container to `freepdb1`.

   c. Create the directory object `DUMP_DIR` and grant `READ` and `WRITE` privileges on the `DUMP_DIR` directory to the `SYSTEM` user.

   ```
   /opt/oracle/product/23c/dbhomeFree/bin/sqlplus / AS SYSDBA
   SQL> ALTER SESSION SET CONTAINER=freepdb1;
   SQL> CREATE DIRECTORY DUMP_DIR AS '/opt/dump';
   SQL> GRANT READ, WRITE ON DIRECTORY DUMP_DIR TO SYSTEM;
   ```

   d. Export data from your 23.2 Free PDB `freepdb1` to the dump folder.

   ```
   /opt/oracle/product/23c/dbhomeFree/bin/expdp system/
   system_password@dbhost.example.com:1521/freepdb1 full=Y
   directory=DUMP_DIR dumpfile=expdb23c_freepdb1.dmp
   logfile=expdb23c_freepdb1.log
   ```

   > **Note:**
   >
   > - Replace `dbhost.example.com` with your database host name. If required, replace `1521` with the port number the listener listens on.
   > - The default PDB name created is `freepdb1` in 23.2 Free and 23.3 Free.

3. Deinstall Oracle Database 23.2 Free if you plan to install 23.3 Free on the same system. See Deinstalling Oracle Database Free for more information

4. Install Oracle Database 23c (23.3) Free.

Importing Data

To import data to your Oracle Database 23c (23.3) Free, perform the following steps for each PDB. The steps in this section are for importing data from 23.2 Free PDB `freepdb1` and overwriting to the same PDB `freepdb1` in 23.3 Free.

1. Set the ORACLE_HOME and ORACLE_SID environment variables.

   ```
   export ORACLE_SID=FREE
   export ORACLE_HOME=/opt/oracle/product/23c/dbhomeFree
   ```

2. Connect to the 23.3 Free database as user `SYS` using the `SYSDBA` privilege as the `oracle` user and switch the container to `freepdb1`.

3. Create the directory object `DUMP_DIR` and grant `READ` and `WRITE` privileges on the `DUMP_DIR` directory to the `SYSTEM` user.

```
/opt/oracle/product/23c/dbhomeFree/bin/sqlplus / AS SYSDBA
SQL> ALTER SESSION SET CONTAINER=freepdb1;
SQL> CREATE DIRECTORY DUMP_DIR AS '/opt/dump';
SQL> GRANT READ, WRITE ON DIRECTORY DUMP_DIR TO SYSTEM;
```

4. Import data to the 23.3 Free PDB `freepdb1` from the dump folder created during the export operation.

```
$ /opt/oracle/product/23c/dbhomeFree/bin/impdp system/
system_password@dbhost.example.com:1521/freepdb1 full=Y
directory=DUMP_DIR dumpfile=expdb23c_freepdb1.dmp
logfile=impdb23c_freepdb1.log
```

> **Note:**
>
> - Replace `dbhost.example.com` with your database host name. If required, replace `1521` with the port number the listener listens on.
> - The default PDB name created is `freepdb1` in 23.2 Free and 23.3 Free.

Ignore the following errors:

- ORA-39342: Data Pump did not import dependent objects for NETWORK_ACL due to the previous error
- ORA-31684: Object type TABLESPACE:"UNDOTBS1" already exists
- ORA-31684: Object type TABLESPACE:"TEMP" already exists
- ORA-31684: Object type TABLESPACE:"USERS" already exists
- ORA-31684: Object type USER:"PDBADMIN" already exists
- ORA-31684: Object type DIRECTORY:"DUMP_DIR" already exists

9
Deinstalling Oracle Database Free

When you deinstall Oracle Database Free, all components, including data files, the database, and the software, are removed.

If you want to save your data files but remove the Oracle Database Free software and database, then first export the data before you deinstall.

Because the deinstallation process removes all files from the directory in which Oracle Database Free is installed, back up any files from the directory (if needed) before you deinstall. The database will no longer be operational after deinstallation.

Run this procedure as `root` or with `root` privileges.

```
$ sudo -s
```

Run the following commands to deinstall Oracle Database Free:

- Delete all the Oracle Database Free data files, the listener, and configuration files. After this operation, only logs and the Oracle home software remains.

  ```
  # /etc/init.d/oracle-free-23c delete
  ```

- This command removes the software. After this operation, some content under Oracle base `/opt/oracle` will remain and you can manually delete it.

  ```
  # yum remove oracle-database-free-23c
  ```

- (Optional) If you only installed Oracle Database Free on the system and have no further Oracle Database software installed, you can also remove the Oracle Database Preinstallation RPM:

  ```
  # yum remove oracle-database-preinstall-23c
  ```

- To delete the downloaded RPM files, for example:

  ```
  # rm oracle-database-preinstall-23c*
  # rm oracle-database-free-23c*
  ```

10

Reporting Security Vulnerabilities

If you find any security vulnerabilities with Oracle Database Free, then email a description of the issue to Oracle at secalert_us@oracle.com.

Include the following information in your email:

- A complete description of the problem.
- The version of Oracle Database Free you are using.
- The platform on which you are running Oracle Database Free.
- Any scripts or examples that may be helpful in tracking down the security problem.

11

Globalization Support

Oracle Database Free is configured by default to process character data in all supported languages simultaneously:

- The database is created with the Unicode `AL32UTF8` character set. `AL32UTF8` is the recommended database character set suitable for storing data in practically any language. Multiple languages can be mixed even in a single character value. While not a recommended option, you can modify the `CHARSET` parameter in the `/etc/sysconfig/oracle-free-23c.conf` configuration file to any other supported database character set before running `/etc/init.d/oracle-free-23c configure` .

  Supported database character sets are listed in tables A-4 and A-6 in Appendix A of the *Oracle Database Globalization Support Guide* . Character sets from Table A-4 are preferred over character sets from Table A-6 because they contain more comprehensive character repertoires.
- Oracle Database Free supports the same globalization features that Oracle Database Enterprise Edition (EE) provides.

Setting Language and Locale Preferences for Client Connections

Configure client applications connecting to an Oracle Database according to your locale preferences and your I/O device character set.

You must configure client applications connecting to an Oracle Database according to your locale preferences and your I/O device character set. If your applications do not have their own specific methods to configure locale preferences, then the method you use to configure an Oracle database client connection depends on the access API you use to connect to the database. Check your application documentation, before you configure locale preferences for your applications.

For applications that connect to Oracle Databases using Oracle Call Interface (OCI) use NLS_LANG and other client settings with names that start with `NLS_` to set the locale conventions and client character set for Oracle Database sessions. It is important that you set the character set part of the NLS_LANG value properly. The Oracle character set whose name you set in NLS_LANG must correspond to the character encoding used by your I/O devices. For Linux, it is the character encoding of the terminal or terminal emulator, and usually corresponds to what the `locale` command reports. For Microsoft Windows, it is either the ANSI Code Page (for GUI applications), such as WE8MSWIN1252, or the OEM Code Page (for Console mode applications), such as US8PC437. By doing this, the OCI API is notified about the character set of data that it receives from the application. OCI can then convert this data correctly to and from the database character set.

You can specify NLS_LANG and the other NLS settings as environment variables. On Microsoft Windows, you can also specify them as Registry settings. Environment variable values take precedence over Registry values.

Oracle Universal Installer sets a default value for the NLS_LANG setting in Registry when it creates a new Oracle home on Microsoft Windows. The NLS_LANG value is based on the language of the Windows user interface, which is the language of Windows menu items and dialog box labels. The installer does not set NLS_LANG on Linux and other UNIX system-based operating systems.

Caution:

Failure to set the client character set correctly can cause data loss.

Java applications that connect to Oracle Databases by using Oracle JDBC do not use NLS_LANG. Instead, Oracle JDBC maps the default locale of the Java VM in which the application runs to the Oracle Database language and territory settings. Oracle JDBC then configures the connected database session using these settings. Because Java works internally in Unicode, the client character set is always set to Unicode. Unless an application explicitly changes it, the default locale of the Java VM is set based on the locale of the user operating system on which the Java VM runs. Check your Java VM documentation for information about configuring the Java VM default locale.

Note:

In 3-tier architecture deployments, application servers that are database clients can have settings in their configuration files that specify the NLS_LANG value or the Java VM locale. Check the documentation accompanying these servers.

See Also:

Oracle Database Globalization Support Guide for more information about configuring user locale preferences