

Oracle® APEX

Accessibility Guide



Release 22.1

F51977-03

June 2022

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Oracle APEX Accessibility Guide, Release 22.1

F51977-03

Copyright © 2017, 2022, Oracle and/or its affiliates.

Primary Author: Caryn Dietrich

Contributing Authors: Terri Jennings, Anthony Rayner, Chaitanya Koratamaddi

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 embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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, Java, and MySQL are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside 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, Epyc, and the AMD 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	vi
Documentation Accessibility	vi
Diversity and Inclusion	vi
Related Documents	vii
Conventions	vii

1 Changes in Release 22.1 for Oracle APEX Accessibility Guide

2 About This Accessibility Guide

2.1 What is Web Accessibility?	2-1
2.2 Why Accessibility is Important	2-2
2.3 About Building for Accessibility	2-2

3 Using Oracle APEX with Assistive Technologies and Keyboard

3.1 Recommended Configuration	3-1
3.2 Accessing Form Pages Using JAWS	3-1
3.2.1 About the Structure of Pages	3-2
3.2.2 Accessing Help Text	3-2
3.2.3 Dealing with Validation Errors	3-2
3.2.3.1 Reviewing the Error	3-2
3.2.3.2 Navigating to the Relevant Fields to Fix the Errors	3-3
3.2.3.3 About WAI-ARIA Support	3-3
3.3 Accessing Oracle APEX Using a Keyboard	3-3
3.3.1 About General Keyboard Shortcuts	3-4
3.3.1.1 General	3-4
3.3.2 About Development Environment Keyboard Shortcuts	3-4
3.3.2.1 General Development Environment	3-4
3.3.2.2 Page Designer	3-5
3.3.2.3 SQL Commands	3-6

3.3.2.4	Code Editor	3-6
3.3.3	About Component-Specific Keyboard Shortcuts	3-6
3.3.3.1	Date Picker (jQuery)	3-7
3.3.3.2	CSS Calendar	3-7
3.3.3.3	Map Region	3-9
3.3.3.4	Interactive Grid Region	3-9
3.3.3.5	About Configuring Interactive Grids to Support Multi Select	3-11
3.3.3.6	Markdown Editor	3-12

4 Developing Accessible Apps

4.1	Developing Accessible Themes	4-1
4.1.1	About Accessibility Features in Universal Theme	4-1
4.1.2	About Accessible Theme Styles	4-2
4.1.3	About Theme Roller	4-2
4.2	Developing Accessible Pages	4-3
4.2.1	Accessible Page Titles	4-4
4.2.2	Accessible Page Modes	4-4
4.2.3	Accessible Cursor Focus	4-5
4.2.4	Accessible Reload on Submit	4-5
4.3	Developing Accessible Regions	4-5
4.3.1	Accessible Page Structure	4-6
4.3.2	Accessible Report Regions	4-9
4.3.3	Accessible Interactive Grid Regions	4-10
4.3.4	Accessible Interactive Report Regions	4-10
4.3.5	Accessible Classic Report Regions	4-11
4.3.6	Accessible JET Chart Regions	4-11
4.4	Developing Accessible Form Pages and Items	4-12
4.4.1	Accessible Form Page Attributes	4-12
4.4.2	Accessible Select Lists and Radio Groups	4-13
4.4.3	Accessible Display Image Alternative Text	4-14
4.5	Developing Accessible Buttons	4-14
4.6	Testing Apps for Accessibility	4-16
4.6.1	About Accessibility Checks in Advisor	4-16
4.6.2	Running Advisor on a Single Page	4-17
4.6.3	Running Advisor on an Entire Application	4-17
4.6.4	Enabling Accessibility Tags in PDF Documents	4-18

5 Open Bugs and Known Issues

6 Further Reading

Index

Preface

Oracle APEX Accessibility Guide offers an introduction to accessibility in Oracle APEX. The guide contains information both for developers who are building applications and also end-users using applications built with APEX.

- [Audience](#)
- [Documentation Accessibility](#)
- [Diversity and Inclusion](#)
- [Related Documents](#)
- [Conventions](#)

Audience

Oracle APEX Accessibility Guide is intended for developers building Oracle APEX applications. [Using Oracle APEX with Assistive Technologies and Keyboard](#) is intended for both developers and end users of applications built with APEX.

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.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Documents

For more information, see these Oracle resources:

- *Oracle APEX Release Notes*
- *Oracle APEX Installation Guide*
- *Oracle APEX App Builder User's Guide*
- *Oracle APEX End User's Guide*
- *Oracle APEX Administration Guide*
- *Oracle APEX SQL Workshop Guide*
- *Oracle APEX API 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.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Changes in Release 22.1 for *Oracle APEX Accessibility Guide*

All content in *Oracle APEX Accessibility Guide* has been updated to reflect release 22.1 functionality and user interface changes.

Deprecated and Desupported Features

See [Deprecated Features](#) and [Desupported Features](#) in *Oracle APEX Release Notes*.

2

About This Accessibility Guide

This guide consists of two distinct sections: general instructions for accessing Oracle APEX with assistive technologies and a keyboard, and guidelines for developing APEX applications that are accessible and highly usable to all users.

"Oracle is committed to creating accessible technologies and products that enhance the overall workplace environment and contribute to the productivity of our employees, our customers, and our customers' customers." —Safra Catz, Chief Executive Officer, Oracle

Accessibility Overview

Web accessibility means people with disabilities can effectively use and contribute to the web. To ensure people with disabilities can use APEX, we need to design and develop more accessible web applications and software.

Accessibility is not just a checklist, but an ongoing and planned effort. The aim of this guide is to help you start the planning process, and understand how to build accessible applications with APEX.

- [What is Web Accessibility?](#)
- [Why Accessibility is Important](#)
- [About Building for Accessibility](#)

2.1 What is Web Accessibility?

This guide refers to the [Web Content Accessibility Guidelines 2.0 \(WCAG 2.0\)](#) to back up some of the guidelines with the relevant standard and provide further reading.

According to W3C, web accessibility means that people with disabilities can perceive, understand, navigate, and interact with the web, and that they can contribute to the web. The disabilities that affect access to the web can be categorized as: visual, auditory, speech, cognitive, physical, and neurological. The goal of web accessibility is to provide equal access to users with disabilities. That is, developers need to focus on building more accessible web applications and products. Apart from people with disabilities, the elderly with increasing impairments, and people in a limiting situation (for example, slow internet connection, no audio etc.) can also benefit with accessible applications and products.

Principles of Accessibility

WCAG 2.0, which is created by W3C, defines the four principles of accessibility as:

- **Perceivable:** Information and user interface components must be presentable to users in ways they can perceive. That is, users must be able to perceive the information presented.
- **Operable:** User interface components and navigation must be operable. That is, users must be able to operate the interface.
- **Understandable:** Users must be able to understand the information as well as the operation of the user interface.

- **Robust:** Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies.

2.2 Why Accessibility is Important

Up to 16% of the world population is disabled, either through birth, aging, illness, or the result of an accident. Accessible products remove obstacles between a company and its existing or potential customers. Accessibility enables employers to recruit from a broader pool of talent. Overall, accessibility generally improves products for everyone (Universal Design).

2.3 About Building for Accessibility

Building for accessibility means coding to standards, and not for a specific technology. Building to standards often leads to better, standardized HTML and cleaner code. The process of building accessible applications at Oracle involves considerable amount of time and resources at all stages of the development cycle. Accessibility guidelines and standards are considered, implemented, and reviewed through the Voluntary Product Accessibility Template (VPAT), a product's statement of conformance to Section 508 of the U.S. Federal Rehabilitation Act.

The Oracle Accessibility Program Office, reporting to the office of the Chief Corporate Architect, is responsible for defining the corporate standards for accessibility, and developing materials to train all employees so that they can successfully create products that meet those standards. To learn more about Oracle's Accessibility Program, see [Oracle Accessibility Information and Resources](#).

Oracle uses the VPAT to represent the degree of conformance to various accessibility standards and guidelines, including Section 508 and WCAG. Depending on when a product was developed and released, different standards may be listed.

3

Using Oracle APEX with Assistive Technologies and Keyboard

Users can access Oracle APEX using only a keyboard or a screen reader such as Freedom Scientific's screen reader JAWS.



Note:

JAWS with all the default settings was used in writing this section. All instructions in this section are based on using a full size keyboard. Note that instructions may vary if using a laptop keyboard.

- [Recommended Configuration](#)
- [Accessing Form Pages Using JAWS](#)
- [Accessing Oracle APEX Using a Keyboard](#)
This section describes how to access components in Oracle APEX using a keyboard.



See Also:

[Oracle's Accessibility Program](#)

3.1 Recommended Configuration

Oracle recommends the screen reader JAWS with the most recent version of Google Chrome for the best user experience.

When reading PDF files exported from APEX (for example, data exports from our report-based region types), Oracle recommends using Adobe Acrobat Reader DC for the best user experience with a screen reader.

3.2 Accessing Form Pages Using JAWS

Many of the pages in Oracle APEX are form pages designed for maintaining application metadata. This section describes how you can gain an understanding of the structure of these pages and interact with them using a screen reader such as JAWS.

- [About the Structure of Pages](#)
- [Accessing Help Text](#)
- [Dealing with Validation Errors](#)

3.2.1 About the Structure of Pages

Many form pages are logically broken up into different regions. For example, the Edit Application Definition page has the following regions containing form fields for maintaining information about the current application:

- Application (app ID)
- Name
- Properties
- Availability
- Error Handling
- Global Notification
- Substitutions
- Build Options
- Report Printing
- Copyright Banner
- About
- Tasks

Oracle APEX uses Heading (Hn) tags throughout its development interface to define page structure. Please use standard screen reader heading navigation to navigate such pages.

3.2.2 Accessing Help Text

Most items within the Oracle APEX development environment have field-level help defined. To activate field-level help, focus on the item for which you need help and press the keyboard shortcut **ALT+F1**. Pressing this keyboard shortcut displays a popup window containing the help text.

3.2.3 Dealing with Validation Errors

Oracle APEX provides many built-in features that help a JAWS user deal with validation errors on form pages.

- [Reviewing the Error](#)
- [Navigating to the Relevant Fields to Fix the Errors](#)
- [About WAI-ARIA Support](#)

3.2.3.1 Reviewing the Error

To navigate to the section of the page where the error messages displays, locate the H2 (Heading Level 2) tag with the text "n error has occurred" (where 'n' is the number of errors). Immediately after this heading there will be an unordered list containing all the errors that have occurred on the page.

3.2.3.2 Navigating to the Relevant Fields to Fix the Errors

For validations that are associated with a particular page item, Oracle APEX renders the actual error message as a link in the notification section. When a user activates the link, focus shifts to the associated item. This enables all users to easily navigate straight to the page item related to the error.

3.2.3.3 About WAI-ARIA Support

Oracle APEX includes some Web Accessibility Initiative - Accessible Rich Internet Applications (WAI-ARIA) markup to help screen reader usability in dealing effectively with item errors. This enhancement improves the screen reader user experience for screen reader and browser combinations that support WAI-ARIA.

When an error occurs relating to a specific page item, that page item is given the following additional attributes:

- `aria-invalid="true"`
This means a screen reader announces "Invalid Entry" when the screen reader user is focused on the page item.
- `aria-describedby="[page_item_name]_error"`
This attribute value matches up with the item's associated error messages so when the screen reader user focuses on the page item, the screen reader announces the actual error.

Before this enhancement, if a screen reader user navigated to a page item that had an error, the screen reader might announce "Job, Edit, Type of Text". Now, the screen reader announces "Job Edit, Invalid Entry, Job must have some value, Type of Text".

3.3 Accessing Oracle APEX Using a Keyboard

This section describes how to access components in Oracle APEX using a keyboard.

Note:

The Oracle APEX interface is intended for use with the US keyboard layout. Users of other keyboard layouts should refer to the US keyboard layout in conjunction with this documentation as there may be incompatibilities regarding number and special character keys, such as the ' (quote) key.

- [About General Keyboard Shortcuts](#)
Learn about keyboard shortcuts for Oracle APEX.
- [About Development Environment Keyboard Shortcuts](#)
Learn about keyboard shortcuts for the development environment, including shortcuts for Page Designer, SQL Commands, and Code Editor.
- [About Component-Specific Keyboard Shortcuts](#)
Learn about keyboard shortcuts for specific components, including Date Picker, CSS Calendar, Map Region, Interactive Grid, Interactive Report, and Markdown Editor.

3.3.1 About General Keyboard Shortcuts

Learn about keyboard shortcuts for Oracle APEX.

General keyboard shortcuts are used throughout Oracle APEX.

If a page incorporates specialized keyboard shortcuts, click the **Help** icon located in the top-right of the page and select **Shortcuts**, or press **Alt+Shift+F1** to display a list of valid keyboard shortcuts. Not all pages have specialized shortcuts.

- [General](#)

3.3.1.1 General

Table 3-1 General Keyboard Shortcuts

Action	Where	Keys
Display the list of valid Keyboard Shortcuts for the page	Any page with special keyboard shortcuts	Alt+Shift+F1
Open field help dialog	Focus on any field with a (?) help icon	Alt+F1
Move focus back to field without closing dialog	Focus on a field help dialog	Alt+F6
Close dialog	Focus in any dialog	Escape
Select the previous/next tab	Any page with focus in region display selector tabs	Arrow keys

3.3.2 About Development Environment Keyboard Shortcuts

Learn about keyboard shortcuts for the development environment, including shortcuts for Page Designer, SQL Commands, and Code Editor.

Development Environment Keyboard shortcuts are specific to the development environment.

If a page incorporates specialized keyboard shortcuts, click the **Help** icon located in the top-right of the page and select **Shortcuts**, or press **Alt+Shift+F1** to display a list of valid keyboard shortcuts. Not all pages have specialized shortcuts.

- [General Development Environment](#)
- [Page Designer](#)
- [SQL Commands](#)
- [Code Editor](#)

3.3.2.1 General Development Environment

Table 3-2 General Development Environment Keyboard Shortcuts

Action	Where	Keys
Display Global Search	Any page	Ctrl+Quote

Table 3-2 (Cont.) General Development Environment Keyboard Shortcuts

Action	Where	Keys
Move splitter	Focus on a splitter handle	Arrow keys
Expand or collapse splitter, if supported by splitter	Focus on a splitter handle	Enter
Move to next splitter	On pages that have a splitter	Ctrl+F6
Move to previous splitter	On pages that have a splitter	Ctrl+Shift+F6

3.3.2.2 Page Designer

Table 3-3 Page Designer Keyboard Shortcuts

Action	Keys
Display from here, in Layout	Ctrl+/, D, H
Display from page, in Layout	Ctrl+/, D, T
Go to Dynamic Actions	Ctrl+/, A
Go to Gallery Buttons	Ctrl+/, G, B
Go to Gallery Items	Ctrl+/, G, I
Go to Gallery Regions	Ctrl+/, G, R
Go to Help	Alt+F1
Go to Layout	Ctrl+/, L
Go to Messages	Ctrl+/, M
Go to Page Shared Components	Ctrl+/, S
Go to Processing	Ctrl+/, P
Go to Property Editor Attributes	Ctrl+/, E, A
Go to Property Editor Main	Ctrl+/, E, M
Go to Property Editor Printing	Ctrl+/, E, P
Go to Rendering	Ctrl+/, R
List Keyboard Shortcuts	Alt+Shift+F1
Navigate to Next Page	Alt+Shift+Page Down
Navigate to Previous Page	Alt+Shift+Page Up
Page Search	Ctrl+/, F
Redo	Ctrl+Y
Restore/Expand	Alt+F11
Save	Alt+F7
Save and Run Page	Alt+F8
Toggle Hide Empty Positions	Ctrl+/, D, E
Undo	Ctrl+Z
Utilities: Advisor	Ctrl+/, U, A
Open context menu, with focus anywhere that has a context menu	Shift+F10 or Context menu key
Move selection, in Layout Gallery and Icon List wizards	Arrow keys
Move to previous group, in Property Editor with focus on a group heading	Up Arrow key

Table 3-3 (Cont.) Page Designer Keyboard Shortcuts

Action	Keys
Move to next group, in Property Editor with focus on a group heading	Down Arrow key
Move to first group, in Property Editor with focus on a group heading	Home key
Move to last group, in Property Editor with focus on a group heading	End key
Expand or collapse the group, in Property Editor with focus on a group heading	Left or Right Arrow keys or Enter or Space
Select previous/next tab, with focus on any tab	Arrow keys

3.3.2.3 SQL Commands

Table 3-4 SQL Commands Keyboard Shortcuts

Action	Keys
Run the current command	Ctrl+Enter

3.3.2.4 Code Editor

Table 3-5 Code Editor Keyboard Shortcuts

Action	Keys
Undo	Ctrl+Z
Redo	Ctrl+Shift+Z
Find	Ctrl+F
Replace	Ctrl+Shift+F
Find previous instance, when focus is in Find field	Up Arrow
Find next instance, when focus is in Find field	Down Arrow
Auto Complete, when available	Ctrl+Space
Leave code editor and go to next tab stop	Alt+F6
Leave code editor and go to previous tab stop	Alt+Shift+F6

3.3.3 About Component-Specific Keyboard Shortcuts

Learn about keyboard shortcuts for specific components, including Date Picker, CSS Calendar, Map Region, Interactive Grid, Interactive Report, and Markdown Editor.

Component-specific keyboard shortcuts only work for specific components.

If a page incorporates specialized keyboard shortcuts, click the **Help** icon located in the top-right of the page and select **Shortcuts**, or press **Alt+Shift+F1** to display a list of valid keyboard shortcuts. Not all pages have specialized shortcuts.

- [Date Picker \(jQuery\)](#)
- [CSS Calendar](#)

- [Map Region](#)
- [Interactive Grid Region](#)
- [About Configuring Interactive Grids to Support Multi Select](#)
- [Markdown Editor](#)

3.3.3.1 Date Picker (jQuery)

Table 3-6 Date Picker (jQuery) Keyboard Shortcuts

Action	Keys
Open the Date Picker, when focus is on the relevant input field	Ctrl+Home
Go to the previous month	Page Up
Go to the next month	Page Down
Go to the current month	Ctrl+Home
Go to the previous week	Ctrl+Up Arrow
Go to the next week	Ctrl+Down Arrow
Go to the previous day	Ctrl+Left Arrow
Go to the next day	Ctrl+Right Arrow
Accept the selected date	Enter
Close the date picker without selecting a date	Escape
Close the date picker and erase the date	Ctrl+End

3.3.3.2 CSS Calendar

When you focus on the calendar grid (month, week or day view), you can use the arrow keys to navigate within the calendar. There are two calendar browsing modes:

1. Event Browsing - Focuses on existing calendar events
2. Calendar Browsing - Allows you to freely select dates or date ranges

The keyboard shortcut behavior depends on the current view.

Table 3-7 Event Browsing CSS Calendar Keyboard Shortcuts

Action	Keys
Go to the previous event. If on the first event in the given view, move to the last event.	Up Arrow
Go to the next event. If on the last event in the given view, move to the first one.	Down Arrow
Switch to Calendar Browsing mode	N
Leave browsing mode.	Escape

Table 3-8 Month View CSS Calendar Keyboard Shortcuts

Action	Keys
Go to the previous day. If on the first day of a month, switch to the previous month.	Left Arrow

Table 3-8 (Cont.) Month View CSS Calendar Keyboard Shortcuts

Action	Keys
Go to the next day. If on the last day of a month, switch to the next month.	Right Arrow
Go to the next week; same day.	Down Arrow
Go to the previous week; same day.	Up Arrow
Extend the selection by one day, up to the end of the current view.	Shift+Right Arrow
Decrease the selection by one day, only until the selection spans one day.	Shift+Left Arrow
Move to the next/previous month.	Page Up / Page Down
Execute the Create Link given as Plug-in attribute by the developer or fire Calendar Date Select event	Enter / Space
Leave Calendar Browsing and go back to Event Browsing	Escape

Table 3-9 Week View CSS Calendar Keyboard Shortcuts

Action	Keys
Go to the previous day. If on the first day of a week, switch to the previous week.	Left Arrow
Go to the next day. If on the last day of a month, switch to the next month.	Right Arrow
Extend the selection by one day, up to the end of the current view	Shift+Right Arrow
Decrease the selection by one day until the selection spans one day.	Shift+Left Arrow
Move selection 30 minutes back in time	Up Arrow
Move selection 30 minutes forward	Down Arrow
Extend selection range by 30 minutes	Shift+ Down Arrow
Decrease selection by 30 minutes until the selection is 30 minutes long	Shift+Up Arrow
Move to the next month	Page Up
Move to the previous month	Page Down
Execute the Create Link given as plug-in attribute by the developer or fire Calendar Date Select event	Enter / Space
Leave Calendar Browsing and go back to Event Browsing	Escape

Table 3-10 Day View CSS Calendar Keyboard Shortcuts

Action	Keys
Go to the previous day	Left Arrow
Go to the next day	Right Arrow
Move selection 30 minutes back in time	Up Arrow
Move selection 30 minutes forward	Down
Extend selection range by 30 minutes	Shift+Down Arrow

Table 3-10 (Cont.) Day View CSS Calendar Keyboard Shortcuts

Action	Keys
Decrease selection by 30 minutes until the selection is 30 minutes long	Shift+Up Arrow
Move to the next day	Page Up
Move to the previous day	Page Down
Execute the Create Link given as plug-in attribute by the developer or fire Calendar Date Select event	Enter / Space
Leave Calendar Browsing and go back to Event Browsing	Escape

3.3.3.3 Map Region

Table 3-11 Map Region Keyboard Shortcuts

Action	Keys
Increase the zoom level by 1.	= / +
Increase the zoom level by 2.	Shift+= / Shift++
Decrease the zoom level by 1.	-
Pan by 100 pixels.	Arrow keys
Increase the rotation by 15 degrees.	Shift+Right Arrow
Decrease the rotation by 15 degrees.	Shift+Left Arrow
Increase the pitch by 10 degrees.	Shift+Up Arrow
Decrease the pitch by 10 degrees.	Shift+Down Arrow

3.3.3.4 Interactive Grid Region

Interactive grid regions come with many keyboard shortcuts and keyboard-specific features to help the keyboard-only and power user. Specifically, they are designed to provide all functionality that is available with the mouse to keyboard users.

In addition, the grid containing the data provides two distinct modes of interaction:

Navigation and **Edit** mode. **Navigation** mode is designed for use when the grid is not currently editable and allows fast and easy navigation around the grid using keyboard.

Navigation mode is the default and the only mode if the interactive grid is not editable. **Edit** mode is designed for use when the grid is currently editable, and keyboard behavior differs slightly.

The interactive grid region consists of multiple separate user interface elements, some of which handle tab stops differently depending on the element type and the expected keyboard semantics. For example, the toolbar and pagination controls are implemented as separate tab stops, while the grid view and icon view are implemented as a single tab stop where cursor key navigation is used to move around the view elements when the view has focus.

The following tables list available keyboard shortcuts in interactive grid regions. General shortcuts work in both navigation mode and in edit mode.

Table 3-12 General Interactive Grid Keyboard Shortcuts

Action	Keys
Move to the next tab stop	Tab
Move to the previous tab stop	Shift+Tab
Toggle between Row Selection and Cell Selection	F8
Open the column header menu, when focus is on the column header	Enter or Space
Increase column width, when focus is on the column header	Windows: Ctrl+Left Arrow Mac: Ctrl+Cmd+Left Arrow
Decrease column width, when focus is on the column header	Windows: Ctrl+Right Arrow Mac: Ctrl+Cmd+Right Arrow
Move column to next column position, when focus is on the column header	Shift+Left Arrow
Move column to the previous column position, when focus is on the column header	Shift+Right Arrow
Sort ascending on the current column, when focus is on the column header	Alt+Up Arrow
Sort ascending on the current column in addition to the existing search columns, when focus is on the column header	Shift+Alt+Up Arrow
Sort descending on the current column, when focus is on the column header	Alt+Down Arrow
Sort descending on the current column in addition to the existing search columns, when focus is on the column header	Shift+Alt+Down Arrow
Move to next tab stop in column header menu, when column header menu is open	Tab
Move to previous tab stop in the column header menu, when column header menu is open	Shift+Tab
Close the column header menu, when the column header menu is open	Escape
Show help (if defined) for an item, when a single row is viewed and focus is on the column	Alt+F1

Table 3-13 Navigation Mode Interactive Grid Keyboard Shortcuts

Action	Keys
Move around the grid, including column headers	Arrow keys
Move without changing the selection	Windows: Ctrl+Up Arrow or Ctrl+Down Arrow Mac: Ctrl+Alt+Up Arrow or Ctrl+Alt+Down Arrow
Select	Space
Toggle Selection	Windows: Ctrl+Space Mac: Ctrl+Alt+Space
Range select and de-select multiple rows	Shift+Up Arrow or Shift+Down Arrow

Table 3-13 (Cont.) Navigation Mode Interactive Grid Keyboard Shortcuts

Action	Keys
Move one page up in the current column	Page Up
Move one page down in the current column	Page Down
Move to first cell in current row	Home
Move to last cell in current row	End
Move to the first cell in the grid	Windows: Ctrl+Home Mac: Option+Up Arrow
Move to the last cell in the grid	Windows: Ctrl+End Mac: Option+Down Arrow
Add a row after the last selected row, if grid is editable	Insert
Delete the selected rows, if grid is editable	Delete
Move around the icons, while in icon view	Arrow keys
Move to the first icon, while in icon view	Home
Move to the last icon, while in icon view	End
Switch from Navigation mode to Edit mode, if the interactive grid is editable	Enter or F2

Table 3-14 Edit Mode Interactive Grid Keyboard Shortcuts

Action	Keys
Move to the next cell in the same row in Edit mode	Tab
Move to the previous cell in the same row in Edit mode	Shift+Tab
Move to the next cell in the same column	Enter
Move to the previous cell in the same column	Shift+Enter
Exit Edit mode and return to Navigation mode	Escape

3.3.3.5 About Configuring Interactive Grids to Support Multi Select

Interactive grids must be configured to support multi select. When multi select is enabled, users can use arrow keys to move around an entire grid, use the Space key to select, or use Ctrl+Space to toggle selections.

To configure an interactive grid to support multi-select:

- Editable interactive grid - Configure the `APEX$ROW_SELECTOR` column which is automatically created for you when you make an interactive grid editable. Under **Settings**, set the **Enable Multi Select** attribute to **Yes** (the default).
- Non-editable interactive grid - Use an advanced JavaScript configuration. To see an example, install the *Sample Interactive Grids* sample application, and see the **Multiple Selection** example under **Reporting**.

 **See Also:**

- Making an Existing Interactive Grid Editable in *Oracle APEX App Builder User's Guide*
- Installing Apps from the Gallery in *Oracle APEX App Builder User's Guide*

3.3.3.6 Markdown Editor

Table 3-15 Markdown Editor Keyboard Shortcuts

Action	Keys
Leave markdown editor and go to next tab stop	Ctrl+Shift+.
Leave markdown editor and go to previous tab stop	Ctrl+Shift+,

4

Developing Accessible Apps

This section provides guidelines for creating an accessible app with Oracle APEX.

- [Developing Accessible Themes](#)
Themes are collections of templates that enable you to define the layout and style of an entire application. Oracle APEX apps use Universal Theme by default.
- [Developing Accessible Pages](#)
Enable easy navigation of your app for users of assistive technology by properly configuring some common attributes of all your pages, such as the title, page mode, cursor focus, and reload on submit.
- [Developing Accessible Regions](#)
Ensure the regions on a page provide sufficient navigational information for users of assistive technology.
- [Developing Accessible Form Pages and Items](#)
Ensure that your form pages and items provide sufficient information to users of assistive technology.
- [Developing Accessible Buttons](#)
Buttons in Universal Theme offer great display flexibility while supporting accessibility.
- [Testing Apps for Accessibility](#)
APEX includes a built-in tool called the Advisor, which developers can run as part of quality assurance of an application.

4.1 Developing Accessible Themes

Themes are collections of templates that enable you to define the layout and style of an entire application. Oracle APEX apps use Universal Theme by default.

Important:

Oracle recommends using Universal Theme for creating accessible apps.

This guide is written for use with the Universal Theme. This guide does not support customized themes directly. While some discussion applies generally to all apps, many discussions are specific to Universal Theme.

- [About Accessibility Features in Universal Theme](#)
- [About Accessible Theme Styles](#)
- [About Theme Roller](#)

4.1.1 About Accessibility Features in Universal Theme

Universal Theme (also called Theme 42, or abbreviated as UT) was first introduced in Oracle Application Express 5.0 to improve out of the box usability and accessibility for Oracle APEX apps.

While you can create and use your own fully accessible custom themes in APEX, this guide only provides instructions for apps using Universal Theme.

Oracle strongly recommends migrating from older themes to Universal Theme. See the [Universal Theme migration guide](#) for further assistance.

4.1.2 About Accessible Theme Styles

Theme styles enable you to change the look and feel of an app without switching to an entirely different theme.

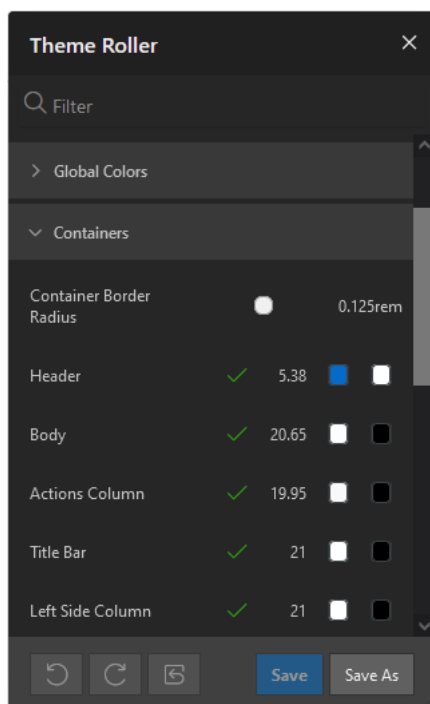
Oracle recommends using the default Oracle APEX theme style Vita for best accessibility.

You can mark theme styles in APEX as **accessibility tested**. Because the underlying CSS of a theme style affects the overall accessibility of an app, you can mark a theme style as accessibility tested if you are confident that it meets the appropriate accessibility standards that you or your organization must adhere to. The Vita theme style is accessibility tested.

4.1.3 About Theme Roller

Theme Roller is a developer tool in the application runtime for analyzing and declaratively adjusting the style settings of Universal Theme without needing to know the underlying CSS.

Some color combinations cause accessibility issues due to a lack of contrast. To help prevent this, Theme Roller has automatic color contrast checking that analyzes your app's compliance with WCAG 2.0 contrast standards. Theme Roller notifies you if a setting passes or fails this contrast check.



A check mark appears next to color combinations for settings to indicate that the color contrast defined meets the minimum required ratio. The ratio is also included next to the check mark.

To learn more about minimum color contrast, see [WCAG 2.0 Understanding 1.4.3 Color Contrast](#).

4.2 Developing Accessible Pages

Enable easy navigation of your app for users of assistive technology by properly configuring some common attributes of all your pages, such as the title, page mode, cursor focus, and reload on submit.

- [Accessible Page Titles](#)
- [Accessible Page Modes](#)
- [Accessible Cursor Focus](#)
- [Accessible Reload on Submit](#)

See Also:

- Right Pane of Page Designer (Property Editor) in *Oracle APEX App Builder User's Guide*
- Accessing Property Editor in *Oracle APEX App Builder User's Guide*

4.2.1 Accessible Page Titles

A good page title helps users of assistive technology understand the purpose of the page. Since the page title is usually the first thing a user of a screen reader hears when loading the page, every page title should be meaningful and unique.

Consider the following example. The title of a Form page should include some information about the record the user is currently editing, such as "Create / Edit Customer: *[customer name]*." In Page Designer, select the page in the Rendering tree and do the following in the Property Editor:

- To replace P1_CUSTOMER_NAME with the page item containing the item that best identifies the current record, in the **Identification – Title** field enter `Create / Edit Customer: &P1_CUSTOMER_NAME.`, with the period included.
- Form pages have a Fetch Row process that fetches the row in the Execution Options. By default, this is set to fire after the page header (**After Header**). Change this to fire **Before Header**.

Tip:

You can review and update all page titles in an app with the Grid Edit of all Pages tool. You should review all titles for accuracy and meaning, especially different pages with the same title and null page titles.

See Also:

- Controlling When a Page Process Executes in *Oracle APEX App Builder User's Guide*
- Accessing Cross Page Utilities in *Oracle APEX App Builder User's Guide*

4.2.2 Accessible Page Modes

Oracle APEX supports multiple page mode types:

- **Normal**
- **Modal Dialog**
- **Non-Modal Dialog**

All of these page types can be used and are coded to the accessibility standards.

For example, modal dialog pages follow guidelines for pages including constraining keyboard focus inside of the dialog and ensuring screen reader users only hear information about the current modal dialog page (and not the base page where the modal dialog page was launched).

4.2.3 Accessible Cursor Focus

The **Navigation – Cursor Focus** attribute specifies where the cursor appears for the user on page load. You can select from the following options:

- Do not focus cursor
- First item on page

Select **Do not focus cursor** to focus on nothing on page load.

Select **First item on page** to focus the cursor in the first field on the page on page load.

Oracle recommends setting your cursor focus to **First item on page** wherever your users want it most of the time, such as in a wizard that has multiple pages with multiple options on each page. Ensure that your **First item on page** design does not misplace important information that users could miss.

4.2.4 Accessible Reload on Submit

This attribute controls how errors display on the page. You can select from the following options:

- Always
- Only for Success

Oracle recommends using **Only for Success**. When **Only for Success** is selected, Oracle APEX renders errors on the client-side without reloading the entire page. This option is faster for all users, and if there are errors, screen reader users are immediately notified and read the error information because the error notification is using the ARIA role of *alert*.

Note:

Tabular form regions can only be set to Always. You cannot use this client-side error display on tabular form pages.

See Also:

[Dealing with Validation Errors](#)

4.3 Developing Accessible Regions

Ensure the regions on a page provide sufficient navigational information for users of assistive technology.

- [Accessible Page Structure](#)
- [Accessible Report Regions](#)
- [Accessible Interactive Grid Regions](#)

- [Accessible Interactive Report Regions](#)
- [Accessible Classic Report Regions](#)
- [Accessible JET Chart Regions](#)

4.3.1 Accessible Page Structure

Users of assistive technology require a clearly defined structure on every page. After a page loads, screen reader users review all of the heading elements on the page. Correctly defined headings orientate your users by providing a structural overview of the page. You should nest page headings so that there is one H1 on a page to uniquely identify the page contents, then nest H2s, H3s, and so forth accordingly.

You achieve this structure with regions in Oracle APEX. Follow one or more of the following guidances as best practices for structuring pages in your app.

H1 Headings

Include a breadcrumb region on your pages to create a unique H1. You can use the Create Application Wizard to generate breadcrumbs automatically based on the hierarchy of pages defined in the wizard, or you can configure the breadcrumbs in your app manually. By default, the last breadcrumb entry on a page is the H1.

Alternatively, use the Hero region template in Universal Theme. This region template renders the region title as the H1 text on the page by default, which is useful if your current app layout and design does not warrant breadcrumbs.

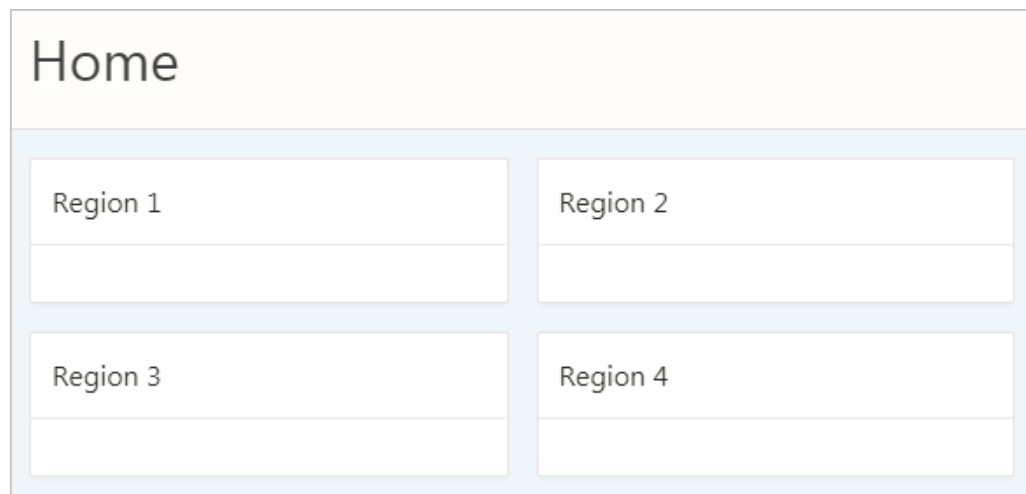


Tip:

Only use one Hero region per page to avoid creating more than one H1 element.

H2 Headings

After the H1, most region templates in Universal Theme render other content on the page with an H2 around the region title



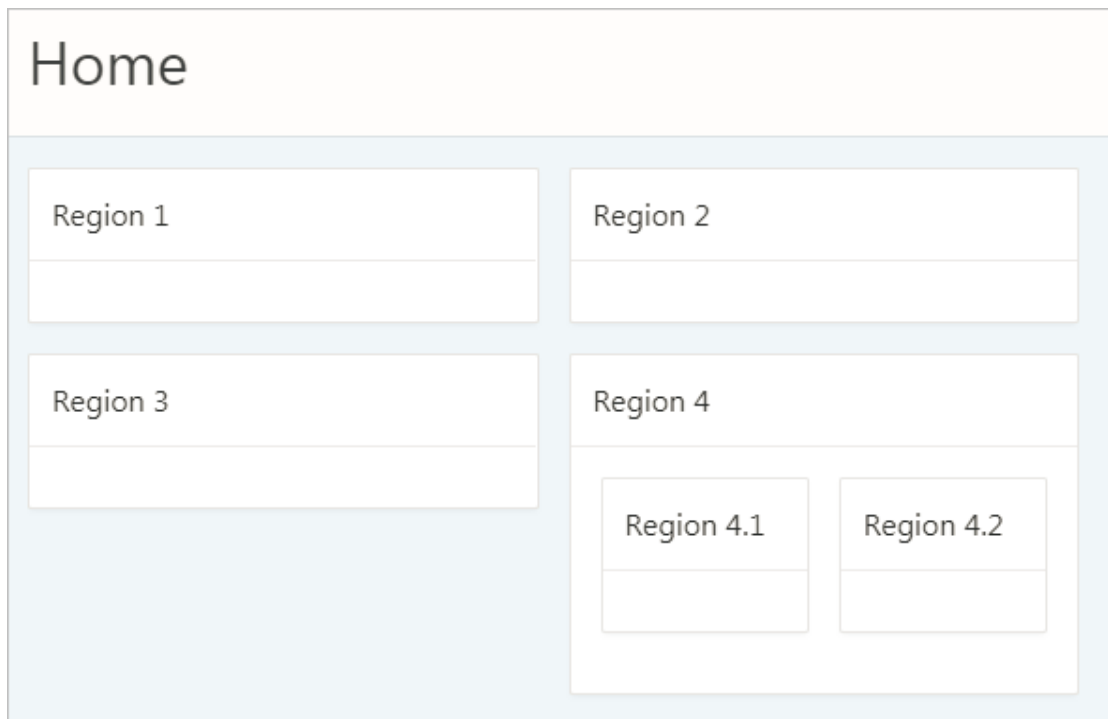
H2s are appropriate for non-nested regions on a page, such as the following example heading structure:

```
H1: Charts
  H2: region 1
  H2: region 2
  H2: region 3
  H2: region 4
```

This semantic structure helps all users scan the current page, both visually using the eyes and audibly using a screen reader.

H3 and Other Sub-Region Headings

Universal Theme does not automatically support headings beyond H2, such as when you have sub-regions nested within an H2 region.



By default, APEX applies H2s to nested sub-regions (regions 4.1 and 4.2 in the above example) in the Standard region template. This creates an accessibility problem because screen reader users may incorrectly believe that these sub-regions are of similar importance to the parent region or that the regions are not ancestral to the parent region.

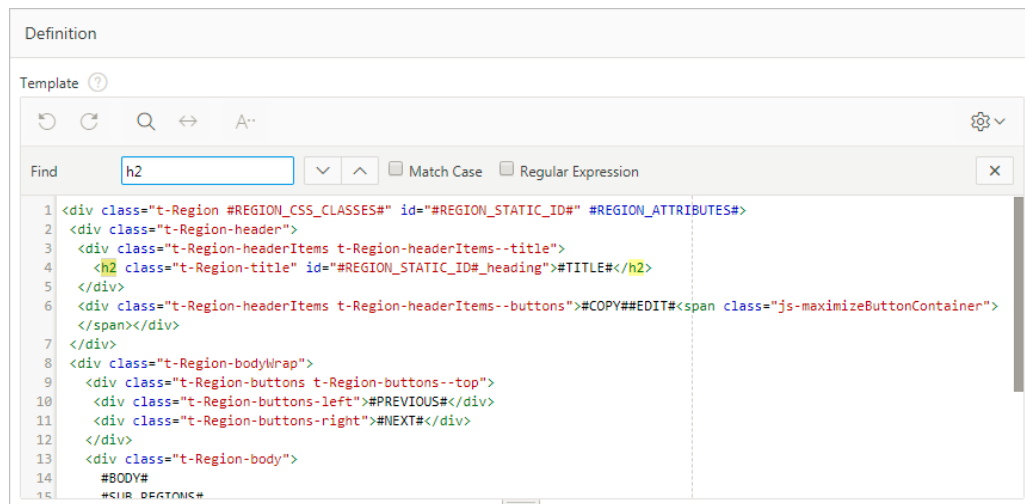
```
H1: Charts
  H2: region 1
  H2: region 2
  H2: region 3
  H2: region 4
  H2: region 4.1
  H2: region 4.2
```

To properly structure these sub-regions, you must duplicate the Standard region template, then use the Code Editor to manually replace the H2 tags with H3s in the template's definition.

 **Tip:**

To copy a region, see *Creating a New Template in Oracle APEX App Builder User's Guide* and select the **As a Copy of an Existing Template** option.

Changing the sub-regions to H3s only affects the structure of the element (not the style), which is what we want for accessibility.



```
Definition
Template ?
Find h2
1 <div class="t-Region #REGION_CSS_CLASSES#" id="#REGION_STATIC_ID#" #REGION_ATTRIBUTES#>
2 <div class="t-Region-header">
3 <div class="t-Region-headerItems t-Region-headerItems--title">
4 <h2 class="t-Region-title" id="#REGION_STATIC_ID#_heading">#TITLE#</h2>
5 </div>
6 <div class="t-Region-headerItems t-Region-headerItems--buttons">#COPY##EDIT#<span class="js-maximizeButtonContainer">
7 </span></div>
8 </div>
9 <div class="t-Region-bodyWrap">
10 <div class="t-Region-buttons t-Region-buttons--top">
11 <div class="t-Region-buttons-left">#PREVIOUS#</div>
12 <div class="t-Region-buttons-right">#NEXT#</div>
13 </div>
14 <div class="t-Region-body">
15 #BODY#
16 #CUR_REGION#
```

This results in the following heading structure:

```
H1: Charts
  H2: region 1
  H2: region 2
  H2: region 3
  H2: region 4
    H3: region 4.1
    H3: region 4.2
```

 **Important:**

Be aware of the following issues when copying the Standard region template:

- While a copied template keeps the theme subscription (which enables you to refresh the theme when APEX is upgraded), the copied template does not upgrade automatically on refresh. You may need to manually change or re-copy the template post-refresh if there are any markup changes in that template.
- You must manually add an additional template for every deeper heading level in your app (H3, H4, H5...). We plan to make it easier to control the heading level without requiring an additional template in a future APEX release.

Region Titles

You should define meaningful titles for most region types because Universal Theme uses the region title as the heading text for the region.

Descriptive titles are essential for accessibility. By default, titles are visible on screen. However, sometimes you may not want a title for a region, such as when sighted users can quickly discern the region's content from surrounding page content at a glance. In such cases, you must still provide heading text for screen reader users. To make such a page accessible, select the region, in Property Editor open the Template Options, and set the Header attribute to **Hidden but accessible**.

 **Tip:**

You can review and update all region titles in an app with the Grid Edit of all Region Titles page. You should review all titles for accuracy and meaning, especially regions with the same title on the same page which may be an issue depending on the context.

 **See Also:**

Accessing Page Specific Utilities in *Oracle APEX App Builder User's Guide*

4.3.2 Accessible Report Regions

Classic reports, Interactive reports, and Interactive Grids all provide a way of defining the report row header column(s), which greatly improves screen reader experience. Row header columns are defined using the **Value Identifies Row** column attribute.

Define the column value, or combination of column values, that help identify the row of data. For person data, this could be the first and last name; for customer data, this could be the customer name, and so on.

When a screen reader user navigates to different rows of the report, the column, or columns, defined as identifying the row are announced as the row header as soon as the users moves

to the new row. If **Value Identifies Row** is not defined, it is much harder for these users to orient themselves when moving around the report.

4.3.3 Accessible Interactive Grid Regions

Introduced in Oracle Application Express 5.1, interactive grids combine powerful reporting with grid-based editing in a single component. Users can access all interactive grid functionality with just a keyboard, enabling all users to navigate around the grid and edit their data using the keyboard. For a complete list of keyboard support in interactive grids, see [Interactive Grid Region](#).

4.3.4 Accessible Interactive Report Regions

Data Table Accessible Label

Oracle APEX generates the following SUMMARY text for the rendered data table:

```
Region = [region title], Report = [current saved report name], View =  
[current View], Displayed Rows Start = [current start of rows displayed],  
Displayed Rows End = [current end of rows displayed], Total Rows = [total  
number of rows]
```

This text identifies the data table, and therefore provides a great deal of contextual information to users of assistive technology.

Edit Icon Link Text

When creating a Report and Form, APEX generates the links between the two pages to allow you to edit specific records. The link text generated for all rows is `Edit` by default. You should make this text a meaningful label so that screen reader users (who listen to all the links on the current page) can understand the purpose of the link and where it takes them. The link text should be specific to the current row in the report. For example, for a report of customers, the link text could be `Edit Customer: SMITH`.

You can define the link text of an interactive report region by selecting the **Attributes** node, then in the Property Editor expand the **Link** column, set **Link Column** to **Link to Single Row View** or **Link to Custom Target**, and update the alt text in the **Link Icon** field.

Because there is no visible link text used, the link uses the inline image's alt text as the text for the link. Note that you can use a substitution string such as `#CUSTOMER_NAME#` to target the column that best identifies the current row.



See Also:

Editing Interactive Report Attributes in *Oracle APEX App Builder User's Guide*

4.3.5 Accessible Classic Report Regions

Data Table

By default, Oracle APEX generates SUMMARY text equal to the region title. This identifies the data table to users of assistive technology.

Edit Link Text

When creating a Report and Form, APEX generates the links between the two pages to allow you to edit specific records. The link text generated for all rows is `Edit` by default. You should make this text a meaningful label so that screen reader users (who listen to all the links on the current page) can understand the purpose of the link and where it takes them. The link text should be specific to the current row in the report. For example, for a report of customers, the link text could be `Edit Customer: SMITH`.

You can define the link text of an interactive report region by selecting the Attributes node, then in the Property Editor expand the Link column, set Link Column to **Link to Single Row View** or **Link to Custom Target**, and update the alt text in the Link Icon field.

Because there is no visible link text used, the link uses the inline image's alt text as the text for the link. Note that you can use a substitution string such as `#ENAME#` to target the column that best identifies the current row.

4.3.6 Accessible JET Chart Regions

Oracle Application Express 5.1 introduced charts based on the Oracle JET Data Visualization components (JET charts). Releases prior to 5.1 used charts based on AnyChart, where there were many accessibility issues.

For more information about accessibility of Oracle JET Data Visualization components including support for keyboard and touch, see the [Oracle JET documentation](#).

The following sections detail some chart settings that are particularly useful for ensuring chart accessibility.

Chart Title

The chart title displays in the chart region next to the chart. It is also the accessible label for the chart page section. In Page Designer, the chart title is defined in the chart Attributes node.

If there is only one chart displayed within a region, consider hiding the region title to avoid visual duplication. You can hide the region title by changing the Header region appearance setting in the Template Options dialog to **Hidden but accessible**. This removes the region title from the page but preserves the heading for assistive technologies (if users scan the current headings on the page to orientate themselves, they can perceive this title).

Tooltip

The tooltip displays when the mouse hovers over a specific chart section or the keyboard has focus on this section. In addition, the tooltip is the accessible label for that section of the chart. For example, when screen reader users navigate around the chart sections, this tooltip is announced.

To display the tooltip, select the chart Attributes node in the Rendering tree; in the Property Editor, set the Show field to **Yes**; and finally, select what information to include in the tooltip. You can define this in two ways:

- Adjust the other attributes displayed in the Tooltip group in Page Designer. This allows you to control display of Series Name, Group Name, Value, and Label information where applicable for your chart. For example, if the chart only has one series, then you may want to exclude the series name in the tooltip, so you would set Show Series Name to **No**.
- For a highly customized tooltip, set a column in your database table to be used as the source for the tooltip. You can configure this on the individual Series node nested under the chart in the Rendering tree.

Automatic Refresh Functionality

Developers can define an **Automatic Refresh, Interval** in the chart attributes which causes the chart region to automatically refresh according to the interval defined. Oracle recommends avoiding this functionality for accessibility purposes since this type of automatic refresh is very disruptive for users of assistive technology.

4.4 Developing Accessible Form Pages and Items

Ensure that your form pages and items provide sufficient information to users of assistive technology.



Note:

The following overview is not a comprehensive list of all item types in Oracle APEX . This section only cites items that have specific settings or issues related to accessibility.

- [Accessible Form Page Attributes](#)
- [Accessible Select Lists and Radio Groups](#)
- [Accessible Display Image Alternative Text](#)

4.4.1 Accessible Form Page Attributes

Item Labels

Define meaningful labels for all items. Users of assistive technology require meaningful labels to understand and interact with a field.



Tip:

You can review and update all item in an app with the Grid Edit of all Item Labels page. You should review all labels for accuracy and meaning, especially items with null labels and items with the same label on the same page.

 **See Also:**

Viewing Item Utilities in *Oracle APEX App Builder User's Guide*

Value Placeholder

Oracle recommends against using just a value placeholder as the only label for an item because it is insufficient for accessibility.

To hide a visible label from display for design reasons, Oracle recommends using a Hidden label template in conjunction with a label.

Item Templates

Universal Theme includes the following item templates to control how the label renders for the item:

Hidden

Hides the label from display but preserves it in the markup for use by assistive technologies. Use this template to preserve accessibility rather than not providing a label.

Optional / Optional - Above

Labels the field optional without any further decoration.

Required / Required - Above

Labels the field required and displays an asterisk adjacent to the label.

For accessibility, Oracle APEX ignores the asterisk and appends hidden text "Value Required" to the label. Assistive technology announces this label as "Label (Value Required)" so that the user can immediately tell if the field is required.

Additionally, APEX adds "Value Required" to the end of the label rather than the beginning, so that screen readers users can still search for a specific field by label name.

Value Required

Oracle does not recommend use of the Value Required attribute for accessibility purposes. If the Value Required attribute is set to **Yes** and the item's label template is **Required**, APEX duplicates reporting of the required state to Assistive Technologies (see the VPAT to learn more).

If you have a simple required field, use a **Not Null** style validation on the page item.

 **See Also:**

- Validating User Input in Forms in *Oracle APEX App Builder User's Guide*
- [Dealing with Validation Errors](#)

4.4.2 Accessible Select Lists and Radio Groups

With the Page Action on Selection attribute, you can define select lists and radio groups to redirect or submit the page when the value of the control changes. However, this behavior

could be disruptive to users who may not expect a page reload or change of context. There are a few ways this can be handled in a more accessible way.

Instead of redirecting or submitting the page automatically, you could:

- Use dynamic actions or cascading LOVs to achieve similar functionality
- Add a button next to the page item that handles the redirect/submit explicitly

Consider the following example, where you have a select list on a page to filter a report region, and you submit the page with this attribute in order to update the report and show the filtered results. Instead, use a Dynamic Action on Change for the select list, and then define an action to refresh the report region. For further examples, see the **Sample Dynamic Action** app, available to install from the Gallery.

If redirecting or submitting the page on a change cannot be avoided, you must ensure the label for the item lets the user know this will happen so that the change of context is expected. For example, a label of `Select Employee (Reloads Page)` indicates that making a selection for this item will reload the content.



See Also:

Installing Apps from the Gallery in *Oracle APEX App Builder User's Guide*

4.4.3 Accessible Display Image Alternative Text

The display image item type enables you to display alternative text with an image on a form page, which is essential to users of assistive technology.

Alternative text settings for the display image item type relevant to accessibility include:

Alternative Text Column

Displayed when Based On is set to **BLOB Column Specified in Item Source**

Defines the column that stores the alternative text for the image.

Alternative Text

Displayed when Based On is set to **Image URL stored in Page Item Value**

Defines alternative text to describe the image. This can also be dynamic by using substitution syntax, such as `&P1_ALT_TEXT`. (ensure you include the period)

4.5 Developing Accessible Buttons

Buttons in Universal Theme offer great display flexibility while supporting accessibility.

Label

The label is crucial for button accessibility. Assistive technologies such as screen readers announce the button label, and users with speech recognition software can speak the name of the label to activate the button. Ensure your button labels are clear, unique, and meaningful.

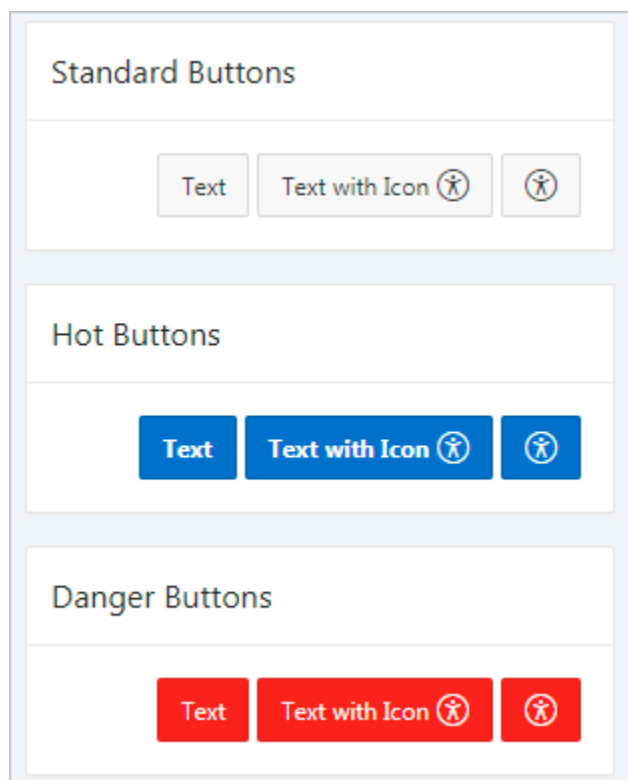
Button Template

Buttons can be based on the following templates:

- Icon
- Text
- Text with Icon

All of these are accessible. Labels for icon-based buttons are hidden from display, but they still exist in the markup for screen readers.

Template Options



Be aware that in the Template Options dialog, the **Type** option changes the button design in ways that can affect accessibility. For example, the Danger type colors the button background red and the foreground text white. Conveying information by color alone is an accessibility issue, so only use the Type template options with this potential problem in mind. For example, the Danger type can still be accessible if the button text clearly indicates the purpose of the button (as opposed to only an exclamation point).

Tip:

You can review and update all items in an app with the **Grid Edit of all Button Labels** page. You should review all labels for accuracy and meaning, especially items with the same label on the same page.

**See Also:**

Accessing Page Specific Utilities in *Oracle APEX App Builder User's Guide*

4.6 Testing Apps for Accessibility

APEX includes a built-in tool called the Advisor, which developers can run as part of quality assurance of an application.

To help with other non-mandatory attributes defined by a developer, the Advisor includes a set of accessibility checks to highlight possible common accessibility problems (such as page items with no labels or page regions with no row headers defined). Developers should review and fix such problems as part of their ongoing development process. We recommend fixing issues identified by the Advisor before performing other accessibility testing (such as testing with Assistive Technologies).

- [About Accessibility Checks in Advisor](#)
The Advisor performs the following accessibility checks when you run it. These checks identify some common configuration errors which have a negative impact on accessibility.
- [Running Advisor on a Single Page](#)
Perform accessibility checks (and others) on a page in an app.
- [Running Advisor on an Entire Application](#)
Perform accessibility checks (and others) on an entire app.
- [Enabling Accessibility Tags in PDF Documents](#)
Manually enable and disable accessibility preferences that persist beyond the user's current session using the `APEX_UTIL.SET_PREFERENCE` procedure.

4.6.1 About Accessibility Checks in Advisor

The Advisor performs the following accessibility checks when you run it. These checks identify some common configuration errors which have a negative impact on accessibility.

Theme Style tested for accessibility

Theme Styles that have not been tested for accessibility may contain more issues, such as insufficient color contrast.

Page has page title

Meaningful page titles help users understand the content and purpose of the current page. (*Note:* Global pages, and pages with no regions are excluded from this check.)

Region has Row Header

Regions that support row headers should have a column with the **Value Identifies Row** attribute set to Yes.

Page item has label

The item should have a defined label. For example, only defining the Value Placeholder text is not sufficient in labelling an item for accessibility.

Page item does not cause an unexpected context change

Some page item settings can cause an unexpected change of context for the user, such as select lists that submit the page after a value is selected.

Consider the following ways to retain the context of the page:

- Remove page submit or redirect behavior from page items, and replace the functionality with Dynamic Actions or Cascading LOVs (if you need to stay on the same page).
- Notify users what happens when they click an item. For example, adding "Launches new page" in the label.

Display Image item has image ALT text defined

Display Image page items must provide text or a column (depending on the Based On setting) to serve as the image's alternative text. This is important for accessibility, as it is the only way for some users to perceive the content of the image.

4.6.2 Running Advisor on a Single Page

Perform accessibility checks (and others) on a page in an app.

To run Advisor on a single page:

1. Navigate to the appropriate page.
2. Click the **Utilities** menu and select **Advisor**.
Advisor appears.
3. Under Checks to Perform, review the selected options. Select and deselect options as appropriate.
4. Click **Perform Check** at the top of the page.
A Results page appears.
5. To alter the existing preferences:
 - a. Under Filter Result, deselect the appropriate options.
 - b. Click **Apply Filter**
6. To re-run the Advisor and see if reported problems are fixed, click **Perform Check**.

4.6.3 Running Advisor on an Entire Application

Perform accessibility checks (and others) on an entire app.

To run Advisor on an entire application:

1. Navigate to the appropriate application:
 - a. On the Workspace home page, click the **App Builder** icon.
The App Builder home page appears.
 - b. Select the application.
The Application home page appears.
2. Click **Utilities**.
3. Click **Advisor**.
The Advisor page appears.

4. Under Checks to Perform:
 - a. Expand Checks to Perform.
 - b. Review the selected options. Select and deselect options as appropriate.
5. In Pages(s), enter a comma separated list of pages in the field provided. To check all pages, leave this option blank.
6. Click **Perform Check** at the top of the page.
A Results page appears.

 **Tip:**

Current Advisor settings are used next time the check is performed.

7. To alter the existing preferences:
 - a. Under Filter Result, deselect the appropriate options.
 - b. Click **Apply Filter**.
8. To run Advisor again, click **Perform Check**.

4.6.4 Enabling Accessibility Tags in PDF Documents

Manually enable and disable accessibility preferences that persist beyond the user's current session using the `APEX_UTIL.SET_PREFERENCE` procedure.

Creating an accessible PDF enables the document to be read aloud by a screen reader and other text-to-speech tools and increases the PDF file size. Setting the `p_preference` parameter `APEX_PDF_ACCESSIBLE` to `Y` adds accessibility tags for native PDF printing. Alternately, setting `APEX_PDF_ACCESSIBLE` to `N` disables accessibility tags. This preference is set automatically when the **Include Accessibility Tags** switch is enabled in the download dialog of an interactive report or interactive grid.

 **See Also:**

`SET_PREFERENCE` Procedure in *Oracle APEX API Reference*

5

Open Bugs and Known Issues

Review known issues in Oracle APEX in the following documents.

VPAT

[VPATs for Oracle Database related products](#) - VPAT for APEX.

Click the link, click **Database Products** or **Open All**, then find the relevant version of APEX.

The VPAT lists exceptions in the product, along with their corresponding bug number. Bugs are categorized as one of the following:

- Affecting the Development Environment
- Affecting Applications
- Affecting Documentation

6

Further Reading

Additional resources for the assessment and enhancement of accessibility in Oracle APEX apps include the following documents.

- [Oracle Accessibility Program](#) - General information about accessibility at Oracle. Click **Database Products** or **Open all** and locate the relevant version of APEX.
- [Plug-ins](#) - Plug-ins enable developers to readily extended with custom item types, region types, processes, and dynamic actions. Once defined, plug-in based components are created and maintained very much like standard APEX components.
- [Universal Theme Sample App](#) - Sample app hosted on apex.oracle.com with information about Universal Theme, including a Migration Guide to move from older themes.

Index

A

- accessibility
 - at Oracle, [2-2](#)
 - definition, [2-1](#)
 - developing apps for, [4-1](#)
 - on the web, [2-1](#)
 - recommended system configuration, [3-1](#)
 - resources, [6-1](#)
 - testing apps for, [4-16](#)
- Accessible Rich Internet Applications
 - See WAI-ARIA support
- Advisor
 - about, [4-16](#)
 - running on entire app, [4-17](#)
 - running on single page, [4-17](#)
- alt text
 - See images
- attributes
 - Link to Custom Target, [4-10](#), [4-11](#)
 - Link to Single Row View, [4-10](#), [4-11](#)
 - Page Action on Selection, [4-13](#)
 - Reload on Submit, [4-5](#)
 - Value Identifies Row, [4-9](#)

B

- browser
 - recommended, [3-1](#)
- bugs
 - See VPAT
- buttons
 - labelling, [4-14](#)
 - templates, [4-14](#)

C

- charts, [4-11](#)
- Code Editor, [3-6](#)
- components
 - accessing with keyboard
 - See keyboard shortcuts component-specific
- CSS Calendar, [3-7](#)
- cursor focus, [4-5](#)

D

- Date Picker, [3-7](#)
- development environment
 - accessing with keyboard, [3-4–3-6](#)

F

- forms, [4-12](#)
 - accessing using JAWS, [3-1](#)
 - attributes in, [4-12](#)
 - dynamic actions in, [4-13](#)
 - regions in, [3-2](#)

G

- general environment
 - accessing with keyboard, [3-4](#)
 - accessing with keyboards, [3-4](#)

H

- headings, [3-2](#), [4-6](#)
- help
 - field-level, [3-2](#)

I

- images, [4-14](#)
 - alternative text for, [4-14](#)
- items
 - labeling, [4-13](#)
 - labelling, [4-12](#)
 - templates for, [4-12](#)
 - value placeholders for, [4-12](#)

J

- JAWS
 - accessing forms, [3-1](#)

K

keyboard shortcuts,
 code editor, [3-6](#)
 component-specific, [3-6](#), [3-7](#), [3-9](#), [3-12](#)
 CSS Calendar, [3-7](#)
 Date Picker, [3-7](#)
 development environment, [3-4–3-6](#)
 field-level help, [3-2](#)
 general, [3-4](#)
 interactive grid, [3-9](#)
 map regions, [3-9](#)
 markdown editor, [3-12](#)
 page designer, [3-5](#)
 SQL Commands, [3-6](#)

known issues
 See VPAT

M

Map Region, [3-9](#)
 markdown editor, [3-12](#)

P

Page Designer, [3-5](#)

pages
 cursor focus, [4-5](#)
 forms
 See forms
 headings, [3-2](#)
 Reload on Submit, [4-5](#)
 running accessibility checks on, [4-17](#)
 structure of, [3-2](#), [4-6](#)
 titles, [4-4](#)

PDFs
 accessibility tags in, [4-18](#)
 recommended software, [3-1](#)

R

radio groups, [4-13](#)

regions
 developing, [4-5](#)
 hero, [4-6](#)

reports
 classic reports, [4-9](#), [4-11](#)
 interactive grids, [3-9](#), [4-9](#), [4-10](#)
 interactive reports, [4-9](#), [4-10](#)

reports (*continued*)

S

screen readers
 edit icon link text for, [4-10](#), [4-11](#)
 fixing validation errors, [3-3](#)
 item errors, [3-3](#)
 recommended, [3-1](#)
 report row headers for, [4-9](#)
 reviewing validation errors, [3-2](#)

select lists, [4-13](#)

SQL Commands, [3-6](#)

T

templates
 buttons, [4-14](#)

Theme 42
 See Universal Theme

themes
 accessibility tested, [4-2](#)
 custom, [4-1](#)
 styles, [4-2](#)
 Theme Roller, [4-2](#)
 Universal Theme, [4-1](#)

U

Universal Theme, [4-1](#), [4-6](#), [4-12](#)

UT
 See Universal Theme

V

validation errors
 fixing, [3-3](#)
 rendering, [4-5](#)
 reviewing, [3-2](#)

Voluntary Product Accessibility Template
 See VPAT

VPAT, [2-2](#), [4-12](#), [5-1](#)

W

WAI-ARIA support, [3-3](#)

Web Content Accessibility Guidelines (WCAG),
[2-1](#)