

## APPLICATION EXPRESS

## ORACLE APPLICATION EXPRESS 5.0

5.1

## Key Features

- Fully supported no-cost feature of the Oracle Database
- Simple 2-Tier Architecture
- Develop desktop and mobile applications
- 100% Browser-based Development and Runtime
- Very advanced Browser-based IDE
- Packaged Applications
- Runs everywhere Oracle Database runs

5.1

*Oracle Application Express is a declarative, rapid web application development tool for the Oracle Database. It is a fully supported, no cost option available with all editions of the Oracle Database. Oracle Application Express enables you to create database-centric Web applications that are reliable, scalable, and secure. Using only a web browser, and with limited programming experience, you can develop and deploy professional Web-based applications for desktops and mobile devices.*

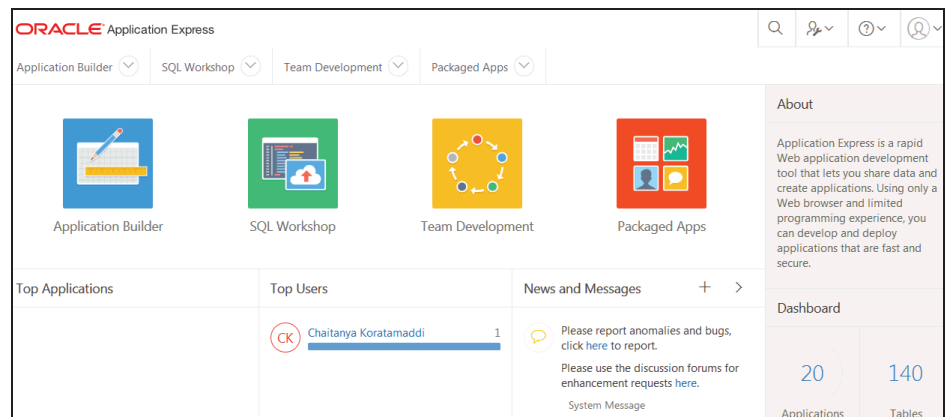


Figure 1: Application Express Development Home Page

## Fully Supported no-cost Feature of the Oracle Database

Oracle Application Express is a fully supported, no cost option of the Oracle Database, and is installed by default in all editions of the Oracle Database. Oracle Application Express runs everywhere and anywhere Oracle Database runs. It is distributed with all editions of Oracle Database 11g and Oracle Database 12c. You can also download the latest Oracle Application Express software from the Oracle Technology Network at <http://otn.oracle.com/apex>.

**Application Development - It's all in the Browser**

To build, deploy, and run applications using Oracle Application Express, you do not need to install any client software. By simply using a browser, you log in to your Workspace, navigate to Application Builder and start building beautiful applications for desktops and mobile devices. End users need just a browser, application URL and their user credentials.

**Utilizing the Declarative Framework**

Oracle Application Express uses a declarative framework for web application development. That is, you specify what to do rather than how to do it. No code is generated or compiled. You interact with a wide variety of built-in wizards and property sheets to define your application.

**Short Learning Curve**

Oracle Application Express enables organizations to capitalize on their existing investment in SQL and PL/SQL skills. Few programming skills are required, and anyone can quickly learn to develop applications. With Oracle Application Express, applications are built faster, with fewer developers.

**Using the Advanced Browser-based IDE**

Oracle Application Express includes, Page Designer, a revolutionary browser-based IDE designed to greatly improve developer's productivity, allowing you to very quickly enhance and maintain pages within Application Express. The key features of Page Designer include:

- Cohesive user experience
- Better visual representation
- Intuitive drag and drop
- Enhanced code editor

5.1

- a. On the Workspace home page, click **App Builder**.
  - b. Select an application.
  - c. On the Application home page, click **Shared Components** in the center of the page.
  - d. Under Security, select **Web Credentials**.
2. On the Web Credentials page, click **History**.

The History page displays recent modifications made to Credentials in the current workspace.

## 18.7 Managing Legacy Web Services

Legacy Web services enable applications to interact with one another over the web in a platform-neutral, language independent environment.

### Note:

The SOAP 1.1 specification is a W3C note. SOAP Version 1.2 specification is a W3C recommendation.

For information about Simple Object Access Protocol (SOAP) see:

<http://www.w3.org/TR/SOAP/>

- [About Web Services](#) (page 18-39)
- [Creating Web Service References](#) (page 18-40)
- [About Working with SSL Enabled Web Services](#) (page 18-41)
- [Creating Web Service References Based on a WSDL](#) (page 18-41)
- [Creating Web Service References Manually](#) (page 18-43)
- [Creating RESTful Web Service References](#) (page 18-45)
- [Using the Web Service Reference Repository](#) (page 18-48)
- [Creating an Input Form and Report on a Web Service](#) (page 18-48)
- [Creating a Form on a Web Service](#) (page 18-50)
- [Creating a Report on a Web Service](#) (page 18-51)
- [How to Invoke a Web Service as a Process](#) (page 18-53)
- [Viewing a Web Service Reference History](#) (page 18-56)

### 18.7.1 About Web Services

5.1

In a typical Web services scenario, a business application sends a request to a service at a given URL by using the protocol over HTTP. The service receives the request, processes it, and returns a response. You can incorporate calls with external Web services in applications developed in App Builder.

- [Adding a Legacy Chart in Page Designer](#) (page 25-6)  
Add a legacy chart to an existing page in Page Designer.
- [Editing Legacy Charts](#) (page 25-8)  
Edit legacy charts in Page Designer.
- [About Saving Flash Charts](#) (page 25-11)  
You can add support to save Flash charts locally.
- [Using Custom XML with Flash and HTML5 Charts](#) (page 25-11)  
Further customize the look and feel of a legacy chart by adding custom XML.

## 25.1.1 About Chart Legacy Types

HTML5 charts and Flash charts support in Oracle Application Express is based on the AnyChart HTML5 Chart Component and AnyChart Flash Chart Component. AnyChart is a flexible Flash and JavaScript-based solution that enables developers to create animated, compact, interactive charts.

- [About Legacy HTML5 Charts](#) (page 25-2)
- [About Legacy Flash Charts](#) (page 25-2)

### 25.1.1.1 About Legacy HTML5 Charts

HTML5 chart support in Oracle Application Express is based on the AnyChart HTML5 Chart Component. HTML5 charts use a JavaScript chart engine, rendering the chart in SVG format. Flash cannot be rendered on most of the modern mobile devices, however you can now take advantage of our new HTML5 charting solution to incorporate charts in your mobile applications. HTML5 charts are compatible with popular browsers for the following mobile platforms:

- Android: Versions 3.1, 3.2, 4.0, 4.0.3, and 4.1
- IOS (iPhone, iPad, iPod Touch): Safari 3.2 and higher is required



#### Tip:

To learn more, see <http://6.anychart.com/products/anychart/docs>.

### 25.1.1.2 About Legacy Flash Charts

5.1

Flash chart support in Oracle Application Express is based on the AnyChart Flash Chart Component. Flash charts are rendered by a browser and require Flash player 9 or later. With AnyChart 6.0, Flash charts are rendered using the FLASH\_PREFERRED rendering type. If Flash Player is not available on the device displaying the chart (for example, pages displayed on an iPhone), AnyChart automatically switches to the HTML5 chart engine and displays an SVG-based chart.

## 25.1.2 Creating SQL Queries for Legacy Charts

Use the following syntax when creating legacy charts.

- [Legacy Chart Syntax](#) (page 25-3)

### 25.1.2.4 Legacy Range Chart Syntax (Flash and HTML5)

Range charts require two values for each bar. To create a range chart, create a chart and provide a SQL query using the following syntax:

```
SELECT link, label, low_value, high_value
FROM ...
```

### 25.1.2.5 Legacy Scatter Chart Syntax (Flash and HTML5)

Legacy scatter charts require an x value and y value for each point. To create a range chart, create a chart and provide a SQL query using the following syntax:

```
SELECT link, label, x_value, y_value
FROM ...
```

### 25.1.2.6 Legacy Candlestick Chart Syntax (Flash and HTML5)

Legacy candlestick charts require open, low, high, and close values for each candlestick. To create a candlestick chart, create a chart and provide a SQL query using the following syntax:

```
SELECT link, label, open, low, high, close
FROM ...
```

### 25.1.2.7 Legacy Gantt Chart Syntax (Flash Only)

Project Gantt charts require a task name, task id, parent task id, actual start date, actual\_end\_date, and progress value for each task. Two optional values for planned start date and planned end date can also be used. To create a Project Gantt chart, **create a Flash chart** and provide a SQL query using the following syntax:

5.1

```
SELECT link, task_name, task_id, parent_task_id, actual_start_date, actual_end_date,
progress
FROM ...
```

Resource Gantt charts require a resource id, resource name, parent resource id, actual start date, and actual end date value for each task. To create a Resource Gantt chart, create a Flash chart and provide a SQL query using the following syntax:

```
SELECT link, resource_id, resource_name, resource_parent_id, actual_start_date,
actual_end_date
FROM ...
```

To represent parent-child hierarchical data on a Resource Gantt chart, provide a SQL query using `START WITH...CONNECT BY` syntax:

```
SELECT link,
       resource_id,
       resource_name,
       resource_parent_id,
       actual_start_date,
       actual_end_date,
FROM ...
START WITH resource_parent_id IS NULL
CONNECT BY PRIOR resource_id = resource_parent_id
ORDER SIBLINGS BY resource_name
```