

# Introduction to Oracle APEX

Version 19.2



## Welcome

2

## Agenda

- Introduction
- Data Model
- SQL Workshop
- Application Navigation
- Forms, Reports & Lists
- Advanced Forms & Reports
- Calendars & Charts
- Security
- Themes
- Deployment



3

## Course Objectives

- To build a system from the ground up in the order the authors would do it in the real world
- To build a functional application using as many APEX components as possible
  - Keep in mind that the application will not be 100% perfect



4

## Onsite Logistics

- Morning & Afternoon Breaks
  - 2 at 10-15 minutes each
- Lunch Break at Noon
  - 1 hour
- Conclude around 4:00 PM each day



5

## 1.1: Download & Unzip Course Files

- Download course files
  - Location provided by instructor
- Unzip to your local PC
  - You will be referring to this directory throughout the class



## 1.2: Login to Your Workspace

- Navigate to the APEX login page
  - URL to be provided by instructor
- Login with the credentials provided by the instructor



## Introduction

8

## Question

- Where is your company's mission critical data?



- Answer:

- Probably all or most of the above!



9

## Question

- And why is that necessarily a bad thing?
  - Data may not be secured
  - Data is likely not centrally managed
    - Hard to backup
    - Hard to know which version(s) are the most recent
  - Not Scalable
  - Not Highly Available or Redundant
  - Probably costing more than it needs to
    - Direct & Indirect costs



10

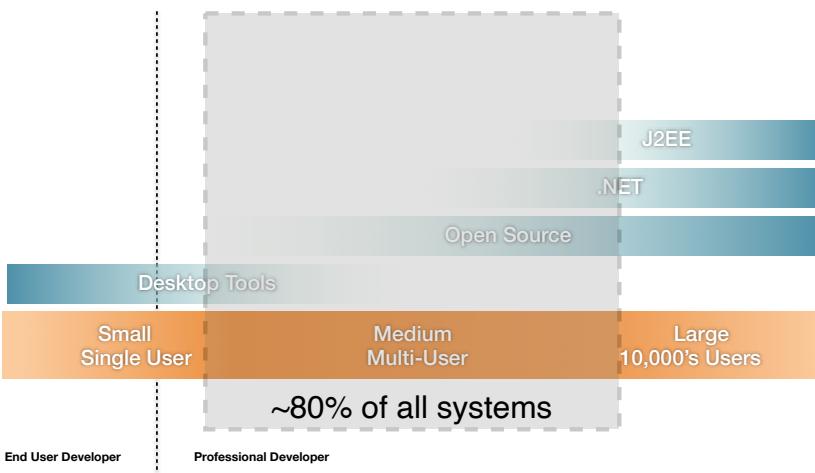
## Reality is...

- Your data is probably
  - In Several Heterogeneous Databases
    - SQL Server, Oracle, Mainframes house dispersed and non-integrated data
  - Not all Managed
    - Desktop Databases, Spreadsheets & eMail messages allow end users to transport critical data with no little to no supervision or security
  - Not Accounted For
    - User created "systems" house data used to run the business, yet IT doesn't know it exists
- The Challenge:
  - Allows access to centrally managed data to authorized users efficiently and without relinquishing control or compromising standards



11

## Traditional IT Landscape



12

## Is IT Partly to Blame?

- Even though most applications are mid-sized, IT often fails to deliver what the business users feel they need. Why?
  - IT has traditionally been a reactive organization
    - Rarely has time to plan proactively
    - Projects not necessarily chosen on merit or need.
  - Technology is more than a full-time job
    - Just when you learn one thing, something newer, cooler, and completely different comes along to completely derail any momentum.
  - Very little in technology “just works” as it should
  - Resources are scarce - Every department is being asked to do more with less
    - People, Skills, Equipment, Capital



13

## Not to mention the fact that...

- High speed internet has become ubiquitous
  - What we used to only be able to access from 9 to 5 is now available 24 x 7 x 365
  - Our employers expect us to react no matter where we are



15

## But IT has had some distractions too...



The Pirate Bay



Windows XP Service Pack 3



14

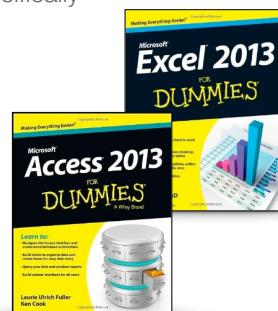
## Even on Vacation



16

## And Since IT “Couldn’t Help”

- Users took things into their own hands.
  - Took a trip to the local bookstore
  - Ordered a **grande-no-foam-mocha-latte-chino-with-whip**
  - Spent a few hours with a book named specifically with them in mind
  - Left the bookstore with just enough knowledge to create a poorly designed, slow, insecure application that solved their problem
- This is where the IT chain of command was broken and things went wrong...



17

## It's a Mess...

- We now live in a world where there are seas of desktop databases and spreadsheets.
  - Sensitive data is floating everywhere waiting to be discovered by malicious forces
  - Users have little to no accountability for the security of the data they access
    - Yet they have superuser-like powers to copy it all to a USB drive and take it anywhere they wish
  - Littering the IT landscape are scores of incomplete or unattempted projects because IT is perpetually too busy responding to day to day emergencies to take on new work
- Every day the sea becomes more and more polluted



19

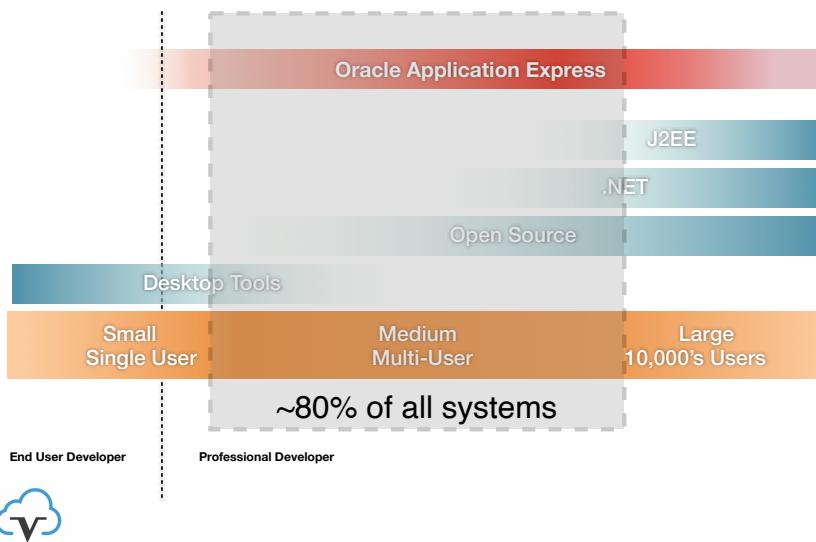
## They Became Instant Business Heroes

- And cast IT as the Villains
- They did what IT “couldn’t”
  - Solve their immediate problem with a tactical solution tailored to their specific business requirements
- What they didn’t realize was that their **short term gain** would become a **long term negative impact** to both IT and the business



18

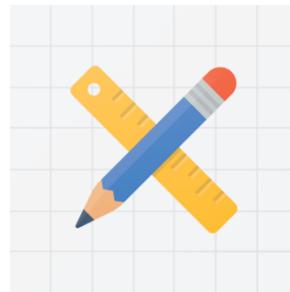
## Traditional IT Landscape



20

## What is APEX

- Application Express enables you to design, develop and deploy **secure, beautiful & responsive, database-driven** applications using **only your web browser**.
  - Quickly & easily create full featured web based applications
  - Target applications at Desktop users, Mobile users or both
  - Seamlessly integrate with all database options
  - Includes jQuery
  - Leverages prior knowledge of SQL & PL/SQL



21

## Database-Centric Development Framework



Develop desktop and mobile web apps



Visualize and maintain database data



Leverage SQL Skills and database capabilities



22

## Use Cases



Developing opportunistic & self service web apps



Extending enterprise application solutions



Migrating file based and client server apps to the web



23

## 3-Tier Architecture



Browser



Mid Tier



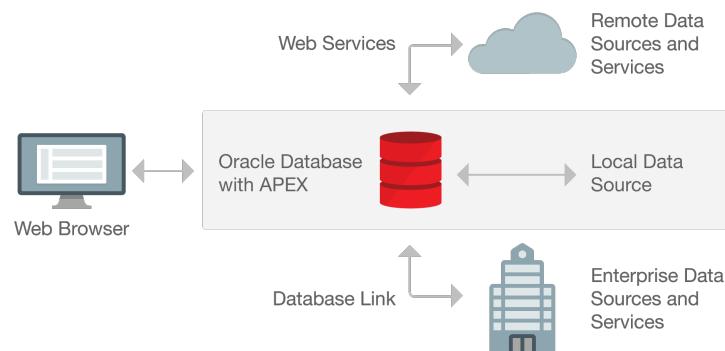
Database Tier

Data Schemas  
Oracle APEX  
Oracle Database



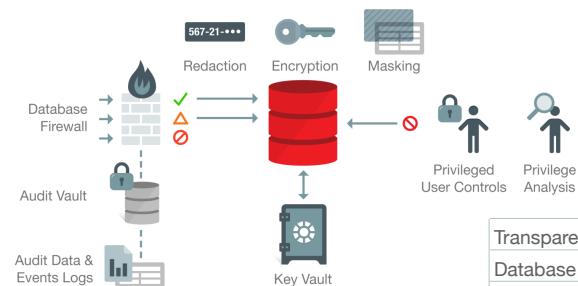
24

## Data Sources



25

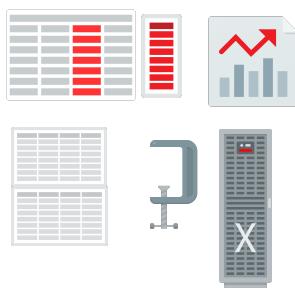
## Security Features



Transparent Data Encryption	✓
Database Vault	✓
Audit Vault	✓
Database Firewall	✓
Key Vault	✓
Data Redaction	✓
Data Masking	✓
Auditing	✓

26

## High Performance Features

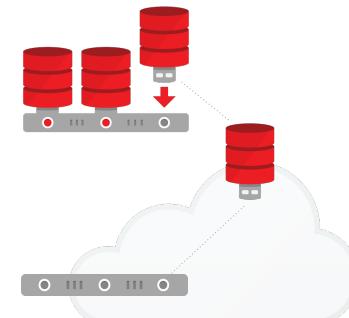


Multi version read consistency	✓
Row level locking	✓
In-Memory	✓
Compression	✓
Partitioning	✓
Hadoop, Big Data SQL, R	✓
Cost based query optimizer	✓
Scale-out with Oracle RAC	✓
Exadata Smartscan & Infiniband	✓

54

27

## Multitenant Features



PDB Deployment	✓
PDB Cloning	✓
PDB Self Service App	✓
PDB Migration	✓
High Density Consolidation	✓
Automated Provisioning and Configuration	✓
PDB Resource Management	✓

28

## Database Development Features



SQL and PL/SQL	✓
Java in Database	✓
Analytics	✓
Text	✓
Regular Expressions	✓
Spatial	✓
Pattern Matching	✓
Database XML	✓
Database JSON	✓
AQ	✓

29

## Data Model



30

## Identifying Requirements

- Our example application will be a **Project Tracking** system
  - Almost every company has one & almost everyone has used one
  - Allows us to **focus on the design and implementation** as opposed to system concepts
- Requirements are fairly standard
  - Projects** have **Milestones**, **Resources** and **Tasks**
  - Tasks
    - Are assigned to specific Milestones
    - Worked on by a specific Resource
    - May have **Task Details**



31

## Identifying System Requirements

- Now that we've identified the objects, we can go a little further and decide what attributes we want to track

PROJECTS	MILESTONES	
PROJECT_NAME	TEXT	
PROJECT_DESCRIPTIONS	TEXT	
RESOURCES	TASKS	
FIRST_NAME	TEXT	
LAST_NAME	TEXT	
USERNAME	TEXT	
PASSWORD	TEXT	
IS_ADMIN	TEXT	
IS_DEVELOPER	TEXT	
REQUIREMENTS	TASK_CATEGORIES	
REQUIREMENT_NAME	TEXT	
REQUIREMENT_DESCRIPTION	TEXT	
	CATEGORY_NAME	
	TEXT	
	CATEGORY_DESCRIPTION	DATE



32

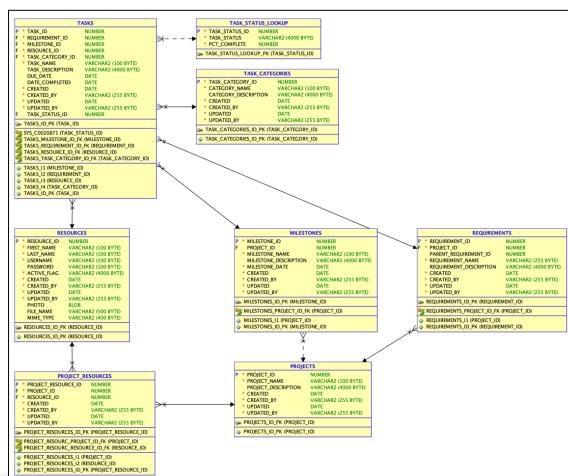
## APEX and Primary Keys

- Things to keep in mind when designing for APEX
  - APEX is set up to make use of **Primary Keys** that are
    - Surrogate Primary Keys
    - Sequence-Based
    - Made up of no more than 2 columns
    - Populated by a Before Insert trigger
  - Not that you can't use other methods
    - ROWIDs
    - Natural Primary Keys
  - APEX just does more for you if you work with it



33

## Relational Model



35

## Table Definitions and UI Defaults

- User Interface (UI) Defaults
  - provide a way to define **standard ways** for APEX to display each attribute
  - Set them **before you start coding** your APEX app
  - Will help with **consistency across the app** and across developers
- Design your tables with this in mind and APEX can extrapolate UI Defaults
  - Column comments on a table can become field level help
  - Marking a column as NOT NULL makes the field required
  - DATE or TIMESTAMP data types are set up as Date Pickers
  - Size of a VARCHAR column will dictate Text Field or Text Area
  - Column order dictates the order things appear within a form or report
  - Defining a column as a BLOB turns on Declarative Blob functionality



34

## Engineer & Refine the Relational Model

Table Properties - PROJECTS

General

Primary Key

Unique Constraints

Indexes

Table Level Constraints

Existing Dependencies

Foreign Keys

Nested Columns

Valid Time Dimensions

Volume Properties

Spatial Properties

Column Groups

Comments

Comments in RDBMS

Notes

Impact Analysis

Measurements

Change Requests

Responsible Parties

Documents

Scripts

Dynamic Properties

Redaction Policy

Classification Types

Summary

Columns

Name: PROJECT\_ID

Data type: NUMBER

Datatype: NUMERIC

Type: NUMERIC

Precision:

Scale:

PK

FK

Mandatory

Deprecated

Comments

Comments in RDBMS

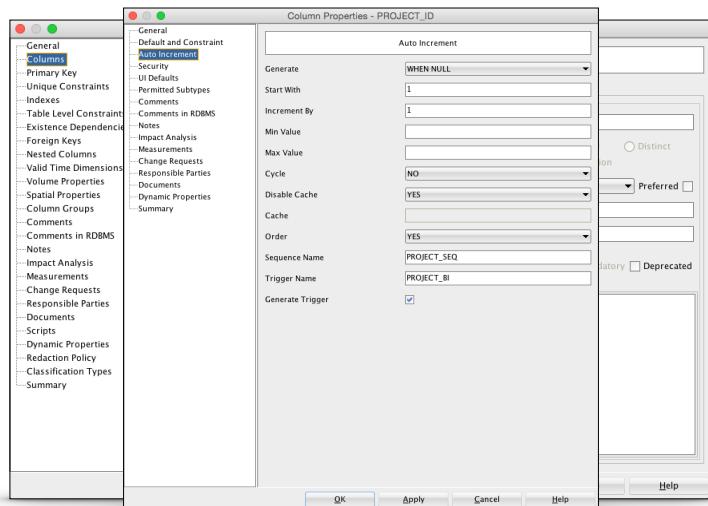
Notes

OK Apply Naming Rules Cancel Help



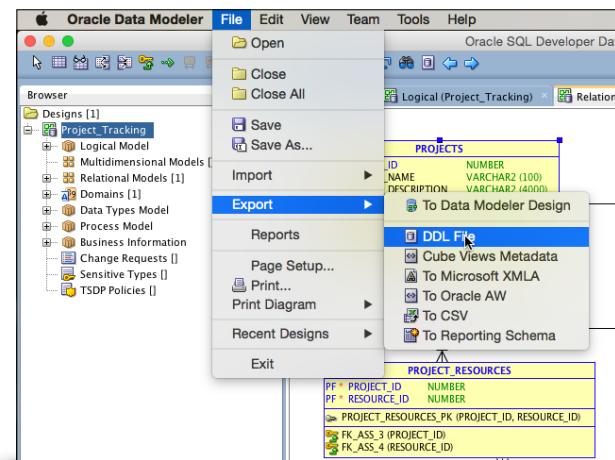
36

## Engineer & Refine the Relational Model



37

## Generate DDL



38

## Generate DDL

```
1 -- Generated by Oracle SQL Developer Data Modeler 4.0.3.853
2 -- at: 2015-09-22 12:05:52 CDT
3 -- site: Oracle Database 11g
4 -- type: Oracle Database 11g
5
6
7
8
9--CREATE
10 TABLE MILESTONES
11 (
12   MILESTONE_ID NUMBER NOT NULL ,
13   PROJECT_ID NUMBER NOT NULL ,
14   MILESTONE_NAME VARCHAR2 (100) NOT NULL ,
15   MILESTONE_DATE DATE NOT NULL ,
16   CREATED_ON DATE ,
17   CREATED_BY VARCHAR2 (50) ,
18   UPDATED_ON DATE ,
19   UPDATED_BY VARCHAR2 (50)
20 );
21 ALTER TABLE MILESTONES ADD CONSTRAINT MILESTONE_PK PRIMARY KEY ( MILESTONE_ID )
22 ;
23
24--CREATE
25 TABLE PROJECTS
26 (
27   PROJECT_ID NUMBER NOT NULL ,
28   PROJECT_NAME VARCHAR2 (100) NOT NULL ,
29   PROJECT_DESCRIPTION VARCHAR2 (4000) ,
30   CREATED_ON DATE ,
31   CREATED_BY VARCHAR2 (50) ,
32   UPDATED_ON DATE ,
33   UPDATED_BY VARCHAR2 (50)
34 );
```

39

## SQL Workshop

40

## SQL Workshop

- One of the main components of APEX, the **SQL Workshop** is a tool that interacts with database objects
- Provides most **commonly needed functionality** when building APEX applications
  - Create and manage commonly used objects
  - Execution of ad-hoc and scripted SQL commands
  - Visually assemble and execute SQL statements
  - Load data from spreadsheets and text files



SQL Workshop

41

## SQL Workshop

- SQL Workshop is an **alternative** - not a replacement - for desktop GUI development tools such as TOAD or SQL Developer

### – Advantages:

- No client configuration or schema password needed
- Can be run from on any device - phone, tablet, browser
- Easier to get access to

### – Drawbacks

- Not all objects & object types supported
- Less robust UI
- Single window



42

## SQL Developer

- Oracle SQL Developer is a free & supported IDE
  - Runs on Windows, Mac & Linux
- Very feature-competitive with TOAD
- Only IDE that has built-in APEX integration
  - Manage Applications
  - Export Pages
  - Remote Debugging
  - Application Reports



43

## SQL Workshop Components

- SQL Workshop consists of five major components
  - Object Browser
  - SQL Commands
  - SQL Scripts
  - Utilities
  - RESTful Services



Object Browser



SQL Commands



SQL Scripts



Utilities



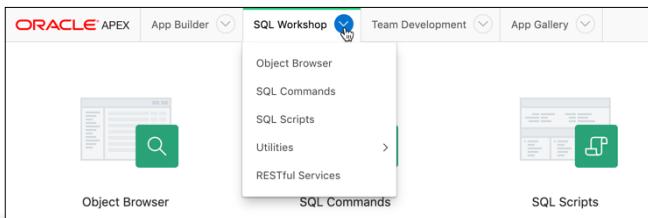
RESTful Services



44

## SQL Workshop Navigation

- Any of the modules of SQL Workshop can be accessed via the SQL Workshop tab
  - Click on the caret to expand the menu



45

## SQL Scripts

- Allows the developer to **Create, Upload, Manage, Edit** and **Execute** SQL Scripts
- Scripts run in the **background via DBMS\_JOB** so you can continue to work while they process
- Results stored for later retrieval
- Some limitations:
  - SQL\*Plus commands are ignored
  - No support for Bind Variables



SQL Scripts



46

## SQL Scripts

- Initial data model was created with **SQL Data Modeler**
- The object creation script was then generated
- Script can be uploaded and run using APEX's SQL Scripts interface



47

## QuickSQL

- Originally only found at <https://quicksql.oracle.com>
  - Now incorporated as a feature in APEX 18
- QuickSQL is a **SQL markup tool** that allows developers to **quickly create tables & views** with brief syntax
  - Definitely **not a substitute** for data modeling
  - Good for **quick prototyping**
  - Results can be **pulled back** into Oracle Data Modeler or any other tool



48

## QuickSQL

- Simply enter shorthand about your tables, columns and relationships
- For example:

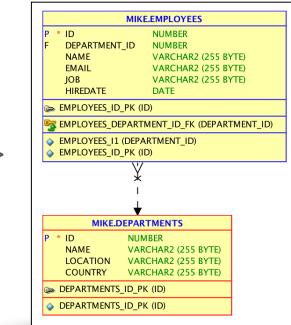
```
emp /insert 10
ename vc(100) /nn
deptno number /nn
sal
mgr number
bonus vc(100)
```

```
/insert 10 = Insert 10 rows
vc(100) = VARCHAR2(100)
/nn = NOT NULL
number = NUMBER
```

## QuickSQL Example

```
departments /insert 4
name /nn
location
country
employees /insert 14
name /nn vc50
email /lower
cost_center num
date_hired
job

view emp_v departments employees
```



49

## 2.1 Create Base Objects Using QuickSQL

- Use QuickSQL to generate a SQL Script
- Inspect and run the SQL Script

## Augmenting what Data Modeler Created

- Not everything can be generated exactly how you want it using SQL Data Modeler
  - Can't model Stored PL/SQL (Packages, Procedures and Functions)
  - Limited Control over Triggers
    - Before Insert Triggers ONLY (No Before Update)
    - No way to inject code of your own.
- Often you'll need to augment what is generated using
  - Scripting
  - SQL Commands Interface
  - Object Browser



50

## Augmenting what Data Modeler Created

- **2.2\_ddl.sql**
  - Creates
    - `hash_password` function
    - `authenticate_user` function
    - `project_resource_pkg` package
  - Updates
    - `RESOURCE_BIU` trigger
    - `MILESTONES_PROJECT_ID_FK` constraint



53

## SQL Commands

- An interactive tool that allows users to enter and execute **ad-hoc SQL statements**
  - Both DDL and DML are supported
  - Transactional and anonymous PL/SQL blocks
- Commands can be saved for later use
- Results can be easily exported to CSV
- Users can view Explain Plans
- A historical log of the last 100 commands is automatically kept



SQL Commands



55

## 2.2 Altering Objects & Creating Functions

- Use the provided script to alter the generated tables and add PL/SQL objects you'll need later



## Object Browser

- Interactive, web-based tool used to create and manage database objects
  - Helpful wizards assist with the creation of objects
  - Most database object types supported
- No need for SQL\*Net or database schema passwords
  - SQL Workshop allows APEX users access to any associated schema
  - But schema roles and privileges will be respected



Object Browser



56

## Object Browser

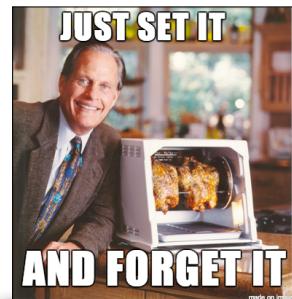
- Key functionality includes:
  - Create, manage and drop database objects
  - View and edit table data
  - Manage Indexes & Constraints
  - Manage UI Defaults
  - Generate DDL
  - Create and Edit PL/SQL program units
    - Triggers, Functions, Procedures & Packages
  - Gather & Analyze Statistics
  - Display Object Dependencies



57

## User Interface Defaults

- Define **default attribute values** for APEX components
  - When components are generated via wizards, the defaults will be applied automatically
  - Result is a more consistent looking application
- Accessed from either Object Browser or Utilities



59



## 2.3 Altering Objects manually

- **2.3.1- Adding Columns Using the SQL Commands Interface**
  - Add 3 columns which allow for the attachment of files to a **RESOURCES** record
    - **PHOTO**
    - **FILE\_NAME**
    - **MIME\_TYPE**
- **2.3.2 - Altering Object Structure Using the Object Browser**
  - Extract **TASK\_STATUS** from **TASKS** and create a new lookup table, making for a more consistent and more easily extensible application



## User Interface Defaults

- Allows you to **define the default display attributes for tables & views** when they are used by an APEX Wizard
  - APEX's wizards will always look to see if UI Defaults are defined, and use them if so
  - Helps provide consistency across one or many applications
- Managed via SQL Workshop
  - the **Object Browser**
  - **Utilities > User Interface Defaults**



60

## User Interface Defaults

- Specify attributes of how a column will render in three different places in APEX:
  - Reports
  - Tabular Forms
  - Forms
- Specify a friendly **Label** for a column
  - Give the column a “Friendly” name that will display on APEX reports & forms
  - Example: **Description** vs. **DESCR**



61

## User Interface Defaults

- Additional attributes include:
  - Alignment (Left/Center/Right)
  - Searchable
  - Display (Y/N)
  - Display Sequence
  - Display As (Text Box, Select List, etc.)
  - Format Mask
  - Default Value
  - Height, Width & Maximum Width
  - Required (Y/N)
  - Help Text
  - And many others...



62

## User Interface Defaults Gotcha

- When an item is created using UI defaults, there is **no relationship established between that item and the UI default**
  - If you change a UI defaults, the changes **will not automatically propagate** to previously created items
- Items created before UI Defaults have been established **will NOT inherit properties** of the UI Defaults
- **Developers can override** any UI default at or after an APEX component is created



63

## User Interface Defaults

- Divided into two categories
  - **Table Dictionary**
    - Create defaults based on specific tables and columns
    - **Overrides any Attribute level definition**
  - **Attribute Dictionary**
    - Create defaults based on generic attribute or column names
    - Will be applied to any column matching the Attribute Name
    - Can create “aliases” for the attribute name



64

## User Interface Defaults

### • Table Dictionary

- Initially created by synchronizing with the database definition
  - Can be synchronized table by table
  - Can use **SYNCHRONIZE** button to synch all tables in a schema
- When synchronizing APEX looks at the table definition and
  - Uses the data type to decide what type of APEX Item should be used
  - Uses the column comments to fill in the Item Level Help
  - Makes NOT NULL columns required in APEX
  - Makes the Label a Title case version of the column name
    - The label for **START\_DATE** would be **Start Date**
- A great way to seed your UI Defaults is to synchronize with the database



65

## User Interface Defaults

### • Attribute Dictionary

- Because they are more generic, they are not created from the underlying database
  - Often entered by hand for columns that are used frequently
  - Can have several “aliases” for the same definition
  - Definitions can be migrated from a table’s UI Defaults
  - Definitions can be migrated from the definitions of objects on a page
- Using the Attribute Dictionary you can
  - Synchronize a Page’s Display attributes to the Attribute Dictionary
  - Push Attribute Dictionary definitions to a specific Page



66

## 2.4 UI Defaults

- **2.4.1 - Synchronizing User Interface Defaults**
  - Create “default” UI Defaults
- **2.4.2 - Customizing User Interface Defaults**
  - Add some UI defaults to various tables & columns



## Application Navigation

68

# Creating Applications

69

## Applications

- Applications are the top-level component in an APEX application
  - All other components map back to an application
  - Applications map to Workspaces
- Applications are created and managed from the App Builder
  - You'll spend 90% of your time in APEX here, so get used to it



70

## Applications

- Applications are **collections of pages** which perform some set of functionality
  - Can be very specific and single-focused
  - Can be very sophisticated and robust
  - Best Practice:
    - Don't let any one application get too large
    - Use several smaller applications instead
    - Easier to manage and patch this way



71

## Applications

- Applications can integrate with almost any user repository
  - Some require no code; others will require some code and/or configuration
- Applications contain **Shared Components**
  - Common components that can be referenced from multiple places within an application or in some cases, across applications
- Applications can be designed for a desktop, mobile device or both
  - Based on which User Interface Type is included



72

## Application Hierarchy

- Workspace
  - Application
    - Shared Components
    - Pages
    - Regions
      - Items
      - Buttons
    - Processes
    - Computations
    - Dynamic Actions
    - Branches



73

## Application User Interfaces

- Previous releases of APEX supported multiple User Interfaces
  - Desktop & Mobile
- Now, there is only one
  - Still contains essential options that control an application's UI



74

## Creating Applications

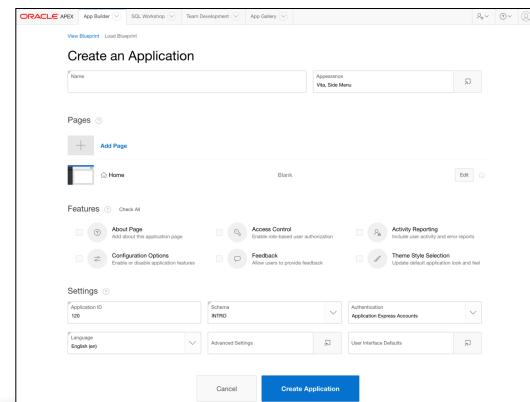
- Several ways to create an applications
  - New Application
  - From a Spreadsheet
  - Productivity App
  - Websheet
  - QuickSQL
  - Copy Application



75

## New Application

- APEX 18 introduces a new **Create Application** “wizard”



76

## New Application

### • Appearance

- Choose menu location, color, theme style, icon

### • Pages

- Select high-level properties for pages

### • Features

- Automatically add pages with embedded functionality

### • Settings

- High level settings of the application



77

## Create From a Spreadsheet

- Quickly create a **CRUD** (CReate Update & Delete) Application based on data in a Spreadsheet
- Application can automatically include
  - **Searchable Report**
  - **Form to Update Data**
  - **Summary Report & Chart**
- Application can be **extended & modified** via Application Builder



79

## New Application

### • Application Blueprint

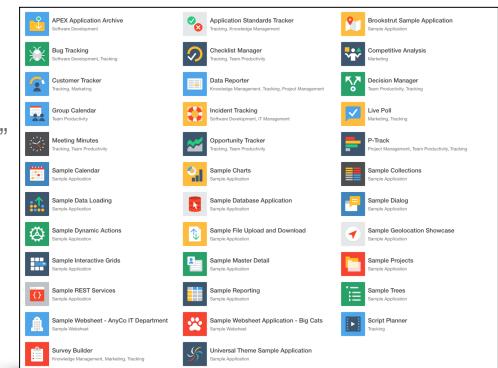
- JSON file that contains all of the metadata specified in the Create Application “wizard”
- Can be modified and used to create a new application

```
1  "application": {
2     "name": "Sample Application",
3     "appIcon": "app-color-1",
4     "schema": "INTRO",
5     "features": [
6         {"activityReporting": true,
7          "configurationOptions": true,
8          "helpLinks": true,
9          "helpPages": true,
10         "themeSelection": true
11     },
12     "appearance": {
13         "themeStyle": "Vista - Red",
14         "themeColor": "#E64A19",
15         "icon": "app-icon-gauge",
16         "iconBackgroundColor": "app-color-4",
17         "iconColor": "#FFFFFF"
18     },
19     "settings": {
20         "baseTablePrefix": "",
21         "primaryLanguage": "en",
22         "authentication": "Application Express"
23     }
24   },
25   "pages": [
26     {"name": "*"}
27 ]}
```

78

## App Gallery

- Previously called **Sample/Packages Applications** and **Productivity Apps** the **App Gallery** contains a menu of **fully-functional APEX applications**
  - Some of them are designed to illustrate a concept
  - These will have “**Sample**” in their name



80

## Websheets

- **Websheets** are simple applications that are designed to simply track data
  - No logic or flows
  - Limited functionality otherwise
- Their **future is uncertain**, and therefore, are not discussed in this class



81

## Copy Application

- Makes a **copy of an existing application**
  - Useful for when you want to try something out but aren't 100% sure that it will work
  - Does not copy associated database objects
  - Also useful to create a “framework” application with all core shared components predefined and start with that vs. starting from scratch each time



83

## QuickSQL

- Creates an application based on the **results of a QuickSQL script**
  - First, create the tables & views with QuickSQL
  - Next, run the script
  - Finally, create the application based on the tables & views created



82

## 3.1 Create an Application

- Create a new application, selecting some of the UI options and add all of the attribute pages



## Running a New Application

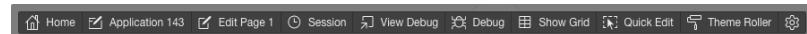
- Every Application has an **Authentication Scheme** Associated with it
- We chose **APEX Developers** as our Authentication Scheme
  - OK for class, not recommended for Real World Applications
- Even though we're logged in to APEX, the Application requires us to authenticate
  - Therefore we must provide our Username and Password



85

## Developers Toolbar

- Allows Developers to interact with the APEX Builder environment
- **Only Available When:**
  - User is Signed on to both Oracle APEX development environment and to another APEX application in the same workspace
- **Not Available to the casual user!**



86

## Our Application Does Nothing!

- That's not entirely true...
- We get all of the following for free:
  - **Session State Management**
  - **Page & Tab Management**
  - **Authentication & Authorization Schemes**
  - **Built-in Look & Feel**
  - **Conditional Processing Engine**
  - **User Management**



87

## Application Features

88

## Application Features

- New in APEX 18, you can now add “Features” to an application when creating it
- Features include:
  - About Page
  - Configuration Page
  - Access Control
  - Feedback
  - Activity Reporting
  - Theme Style Selection



89

## Configuration

- Allows you to **control which Application Features pages are active**
  - Does not remove them; simply enables & disables them
- Pages includes are:
  - **Activity Reporting**
  - **Feedback**
  - **About Page**
  - **Theme Style Selection**



91

## About

- A static page that can be configured with a **simple message about the application**
  - Useful for displaying information about the application, who to contact in the case of an error, planned downtime, etc.



90

## Access Control

- Perhaps the most functional feature, Access Control allows you to **manage user to role mappings**
- By default, **three roles** are created
  - **Admin**
  - **Contributor**
  - **Reader**
- **Roles** and **Role Mappings** are stored two tables
  - **APEX\_APPL\_ACL\_ROLES**
  - **APEX\_APPL\_ACL\_USER\_ROLES**



92

## Feedback

- Enabling **Feedback** creates an additional icon on the Navigation Bar
  - Clicking on this allows the user to create Feedback
- Feedback will be recorded in the table  
**APEX\_TEAM\_FEEDBACK**
  - Reports can be created on top of that if needed
- Feedback is part of the **Team Development** module
  - But can be used independently of that



93

## Activity Reporting

- A variety of reports based on the APEX logs
  - **APEX\_ACTIVITY\_LOG**
- Five different reports
  - **Activity Dashboard**
  - **Top Users**
  - **Application Error Log**
  - **Page Performance**
  - **Page Views**



94

## Theme Style Selection

- Allows the user to **select the default Theme** for the application
  - Can also allow end users to choose their own theme
- **New in APEX 19.2:** end users can select **Vita - Dark Mode** for a dark mode theme



95

## 3.2: Application Features

- Explore the different options of the Application Features that were installed in your application
  - Create and view Feedback
  - Change the default Theme
  - Add a new user and associate the Reader role to that user



# Public Pages

97

## Public Pages

- APEX pages can either require a user to be authenticated or not
- Setting the **Authentication** property to **Page is Public** will allow any user - authenticated or not - to view that page
- Public Pages are useful for:
  - Splash Screens
  - Home Pages
  - Dashboards
  - Login Pages
  - Applications with no need for Authentication for a portion of their pages



98

### 3.3: Public Pages

- Set Page 1 of your Application to a Public Page



## Navigation Bars

100

## Navigation Bar

- Each APEX application has one **Navigation Bar**
  - Set of links that are typically displayed on every page
    - Login
    - Logout
    - Help
    - My Account
- As developers, we can create and modify **Navigation Bar Entries**

## Navigation Bar Entries

- Each Navigation Bar Entry can:
  - Contain an image, text or both
  - Optionally have a parent entry
  - Link to an APEX page or external URL
  - Be associated with:
    - An APEX condition
    - An Authorization Scheme
    - A Build Option



101



102

## 3.4: Navigation Bar

- Create a Login item in the Navigation Bar
- Set the Log Out, Feedback Help & &APP\_USER. Navigation Bar Entry so that it only shows when logged in

## Global Pages



104

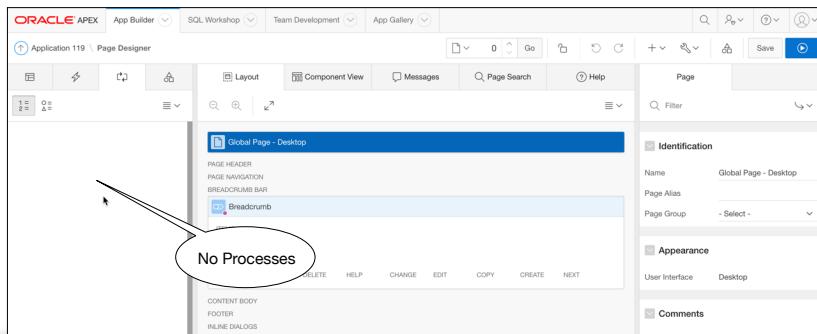
## Global Page

- The **Global Page** is a special page
  - They only contain Page Rendering UI components
    - Do not include page Computations or Processes
  - In the Component View, it will also display Application Items, Processes & Computations
- Prior to APEX 4.2 called Page Zero
  - Can be any Page Number you like, but typically still set to zero
- **Items on a Global Page display on ALL pages of the associated User Interface**
  - Unless conditionally restricted to do otherwise



105

## Global Page - Page Designer



107

## Global Pages

- Common uses for Global Pages:
  - Breadcrumb Regions
  - Lists
  - Common Regions
  - JavaScript
  - CSS



106

## Breadcrumbs

108

## Breadcrumbs

- Hierarchical structures used for navigation
  - For example:  
**Admin > Resources > Manage Resources**
- When clicked, can redirect to another page or external URL



109

## Breadcrumbs Best Practice

- **Place one Breadcrumb Region on a Global Page**
  - When using the Create Application Wizard, a Breadcrumb Region is placed on each page
  - This is not scalable, as each time you add a new page, you will have to add a new Breadcrumb region
  - And if you want to change a region attribute, you will need to change it for each one



111

## Breadcrumb Entries

- Each **Breadcrumb** has one or more **Breadcrumb Entries**
- Each Breadcrumb Entry:
  - Corresponds to a page
    - Only One Entry per Page
  - Links to either a Page or URL
  - May map to a Parent Entry
    - No Mapping indicates that it is a top-level node



110

## Breadcrumbs Best Practice

- Remove the **Breadcrumb** region from all other pages and add it back the **Global Page**
  - When any page is run, the Breadcrumb will behave as if it was placed on that page



112

### 3.5: Breadcrumbs

- Copy the **Breadcrumb Region** from Page 10000 of your application to your Global Page
- Delete the **Breadcrumb Region** from Page 10000



114

## Lists of Values

### List of Values

- Predefined lists of valid **value & attribute pairs** available throughout the application
- Commonly referred to as **LOVs**
- Associated with specific items:
  - Select Lists & Multi-Select Lists
  - Checkboxes
  - Radio Groups
  - Shuttle Widget
  - Popup LOVs



115

### List of Values

- Each LOV must contain a **Display Value & Return Value**
  - We see the Display Value; the database sees the Return Value
- Optionally alias your queries to use d & r:

```
SELECT
    status d,
    status_id r
FROM
    status_lookup
```



116

## Types of Lists of Values

- **Static**

- Manually enter display and return values or use the popup tool to enter them
- **Sort at Runtime** will sort the list; otherwise it will be displayed “as entered”
- **Not scalable**, as if you need to add a new value, you have to re-import the entire APEX application
- Use when you are 100% sure the values will not change



117

## Types of Lists of Values

- **PL/SQL Function Body Returning SQL Query**

- Alternatively, you can provide a **PL/SQL function** that will return a valid SQL statement for the source of an LOV
- SQL statement that is returned must still adhere to the same rules as one that is entered

```
RETURN myFunction(:P1_ITEM);
```

- **Note:** if you want to use bind variables, include them in the string - do NOT evaluate them while concatenating the string in PL/SQL
  - Otherwise, you may be susceptible to a SQL Injection attack



119

## Types of Lists of Values

- **SQL Query (Dynamic)**

- Provide a SQL query with two columns:
  - **Display Value**
  - **Return Value**
- If they are the same, then simply alias the columns differently
- You’ll also have to use aliases if using a concatenated string as a column

```
SELECT
    first_name || ' ' || last_name full_name,
    user_id
FROM
    employees
```



118

### 3.6.1 Static LOVs

- Create a Static List of Values that will be used to indicate Yes/No



## 3.6.2 Dynamic LOVs

- Create a Dynamic lists of values for Project Milestones



## Setup Home Page

123

## Forms, Reports & Lists

122

## Setup Pages

- The Setup section of an application is **often created first**
  - Need to be able to create and maintain lookup tables
  - May need to enter a certain amount of seed data before the application can function correctly
  - Likely need to create and maintain end users
- In your case, you'll start by creating a single blank page as your Setup Home Page and then adding the other forms & reports



124

## 4.1 Admin Home Page

- Create a blank admin home page following the steps in the exercise.

## Setup Data



126

## Setup Data

- We'll **load and run another script** that will populate the setup tables with seed data
  - This way, you don't have to enter it by hand

## 4.2 Setup Data

- Load and run **4.2\_setup\_data.sql** from the SQL Scripts section of the SQL Workshop



127

## Form on a Table with Report

129

### Form on Table with Report

- Single wizard that creates 2 pages
  - Report that shows all records in the base table
  - Form that allows creation and edit of records
- The report will contain links to the edit form
  - One button to create new records
  - One link per record to enable editing
- All created without writing any code
  - Simply fill out the appropriate data in the wizard



130

## APEX Reports

131

### APEX Reports

- Perhaps the most powerful region in APEX
- Supply a SQL statement, and the APEX reporting engine does the rest
  - Can also simply choose columns declaratively
- Built-in support for:
  - Sorting
  - Formatting
  - Pagination
  - Bind Variables
  - Breaking
  - Templates
  - Conditional Columns



132

## Four Types of APEX Reports

- **Classic Report**

- Enter a SQL Query

- **PL/SQL Function Returning SQL Query**

- Specify a function which returns a valid SQL query
  - Useful for dynamic SQL reports

- **Interactive Reports**

- Ad-Hoc Query Tool

- **Interactive Grids**

- Replacement for Tabular Forms
  - Edit multiple rows & columns at once



133

## Reports

- Reports have three main components:

- **Region Definition**

- Title, SQL Query, Region Template & Layout, Conditions, Caching, Header & Footer,

- **Report Attributes**

- Report Template, Pagination, Break Formatting, CSV Export parameters, Printing Enabled

- **Individual Column Attributes**

- Column Formatting, Linking, Sorting, Inclusion in Export or Print, Authorization Schemes



134

## APEX Forms

135

## Forms

- Forms are used to display, edit and collect data, which is then sent back to the database for processing

- Forms can update:

- **Local Database or Remote Database**

- Table / View
    - SQL Query
    - PL/SQL Function Returning SQL Query

- **Web Source Modules**



136

## Forms

- New in APEX 19.1, there is a **new region type: Form**
- The Form region contains the **Data Source**
  - As well as **WHERE & ORDER BY** clause for **Local & Remote Databases**
- All items on the page **map to a Form** region now
  - As well as the DML Initialization & Process processes
- Thus, it's possible to have more than one Form per page



137

## Items

## APEX Items

- APEX items are UI controls used to allow users to view and enter data
- APEX items can be referenced from:
  - The **same APEX page** where the item exists
  - **Any other APEX page** in the same application
  - **Any PL/SQL program** unit called from APEX
- Thus, all APEX items are **global in scope** within their application



139

## Application Items

- Applications Items are Non-UI controls used to store data at the application level
  - Scope can be set to
    - **Application** - Value is only available within the application in which it is defined.
    - **Global** - Value is available to all applications within the workspace which are sharing session information
  - From 4.2 onwards they are no longer prefaced with F####\_
    - Can be named anything you like
    - Best practice is to come up with a naming scheme for Application Items
      - For example, **G\_** or **AI\_**



140

## Page vs. Application Items

### • Page Items

- Created and associated with a page in an application
- Contain some sort of UI component
- HTML is escaped

### • Application Items

- Created and associated with an application as part of its shared components
- Contain no UI component
- HTML is not escaped



141

## Page Item Naming Prefixes

- Each Page & Application Item must have a unique name per application
- Page Items:
  - Should Start with **PXX\_**
    - Where **XX** is the Page Number that the item is on
    - Items that reference a database column should be named after that database column
      - Even if the Display Name is different



143

## Page Item Types

- Checkbox
- Color Picker
- Date Picker
- Display Image
- Display Only
- File Browser
- Hidden
- List Manager
- Number Field
- Password
- Percent Graph
- Popup LOV
- Radio Group
- Rich Text Editor
- Select List
- Shuttle
- Switch
- Text Field
- Text Field with Autocomplete
- Text Area



142

## Item Properties

- Identification
- Label
- Settings
- Layout
- Grid
- Appearance
- Validation
- List of Values
- Advanced
- Source
- Default
- Quick Picks
- Condition
- Read Only
- Security
- Configuration
- Help
- Comments



144

## Item Source

### • Form Region

- Will indicate if the item is mapped to a Form region or not
- If so, Form-region specific options will be available
  - Column & Data Type
  - Query Only
  - Primary Key
- If not, additional options will be available
  - Type
  - User



145

## Item Source

### • Source Type

- Determines where the item gets its data from
  - Static Value
  - Database Column
  - Item
  - SQL Query (return single value)
  - SQL Query (return colon separated value)
  - PL/SQL Expression
  - PL/SQL Function Body
  - Preference
  - Null



146

## Item Source

### • Source Used

#### – Only when current value in session state is null

- Typically used for items not mapped to a database column
- Will look at Source Value or Default Value for its value

#### – Always, replacing any existing value in session state

- Typically used when mapping an item to a Database Column
- Will always get the version of the data from the database



147

## Maintain Session State

### • Three options

#### – Per Request (memory only)

- Session state is only available for current process; does not save

#### – Per Session (disk)

- Stores normally per session on disk; access across requests

#### – Per User (disk)

- Stores normally per user; access across sessions



148

## Default Values

- Used to populate an item with some value
  - **Static Text with Session State Substitutions**
  - **PL/SQL Function Body**
  - **PL/SQL Expression**
- Typically used when the Item Source is mapped to a database column and you want to supply a default value if the item is NULL
  - Created On Date



149

## Form on a Table with Report Results

151

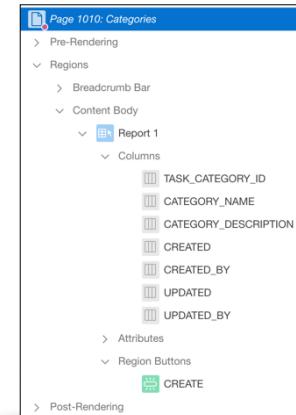
## 4.3: Form on a Table with Report

- Create a **Form on a Table with Report**
  - Create the form on the **TASK\_CATEGORIES** table



## Page 1010 (Report) - Rendering

- **Regions**
  - Task Categories
- **Buttons**
  - CREATE
- **Columns**
  - TASK\_CATEGORY\_ID
  - CATEGORY\_NAME
  - CATEGORY\_DESCRIPTION
  - CREATED
  - CREATED\_BY
  - UPDATED
  - UPDATED\_BY



152

## Page 1015 (Form) - Rendering

### • Processes

- Initialize form Manage Categories

### • Regions

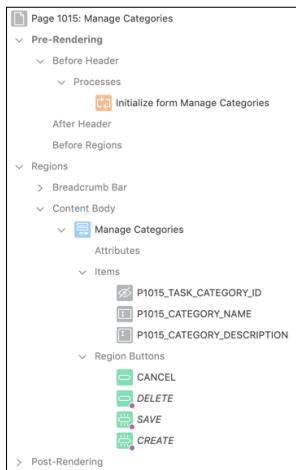
- Manage Categories (Form)

### • Items

- P1015\_TASK\_CATEGORY\_ID
- P1015\_CATEGORY\_NAME
- P1015\_CATEGORY\_DESCRIPTION

### • Buttons

- CANCEL, DELETE, SAVE & CREATE



153

## Hidden Item

### • P1015\_TASK\_CATEGORY\_ID is a Hidden Item

- Does not render on the page

### • But it is part of the underlying HTML form

- We can view the source to see it



154

## Hidden Item

APEX Project Manager

Setup \ Categories \ Manage Category

Category Name Documentation

Category Description

Styles Computed Event Listeners DOM Breakpoints Properties

```
<input type="hidden" id="P1015_TASK_CATEGORY_ID" name="P1015_TASK_CATEGORY_ID" value="R8491185531897159" />
```

155

## Buttons

### • Cancel

- Redirects back to Page 1010 - does not process Page 1015

### • Create

- Submits and processes Page 1015

APEX Project Manager

Setup \ Categories \ Manage Category

Category Name

Category Description

Cancel Create

```
<input type="button" value="Create" />
```

156

## Processing

- Computations

- None

- Validations

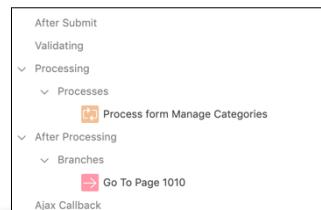
- None

- Processes

- Process form Manage Categories

- Branches

- Go to Page 1010



157

## Item Values

- During the processing phase, the **values of the APEX items** are **sent to the APEX engine** and are set in the APEX Session State

- They can then be referenced in:

- Computations
  - Validations
  - Processes
  - Branches
  - Named PL/SQL program units called by any of the above components



158

## Processing

- Computations

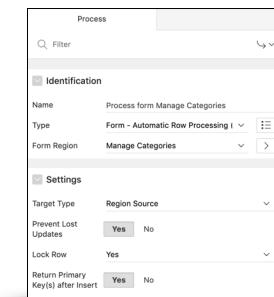
- Since there are none, nothing happens

- Validations

- APEX will use the Item Level Attribute to determine whether **P1015\_CATEGORY\_NAME** requires a value
  - Any failed validation will cause the page to stop processing and display an error message
    - All validations will always execute; a failed one will not prevent the subsequent ones from firing

## Process Form

- APEX-specific process used to insert, update and/or delete rows from a table
- Points to a Form where data source is defined
- 100% declarative
- Also automatically incorporates optimistic locking or lost update detection
- Can also return value of PK used back to an APEX item which can then be referenced in subsequent processes



159



160

# Interactive Reports

161

## Interactive Reports

- Gives end-users **powerful ad-hoc query capability with exactly zero lines of code**
- Users can customize:

- Searching
- Sort Order
- Columns
- Breaking
- Highlighting
- Computations
- Aggregations
- Charts
- Group By
- Flashback Time
- Saved Reports
- Subscription



162

## Interactive Reports

- Technically nothing more than a report type
  - **Older reports can be easily converted** to Interactive Reports
- Most features of standard reports are available with Interactive Reports:
  - Sorting
  - Column Links
  - Export to CSV
  - Pagination
  - Conditions



163

## Search Bar

- Automatically include a field that allows users to filter reports
  - **Search Field**
    - User can enter for and find text strings
  - **Finder Drop Down**
    - Allows user to select which column to filter on
  - **Reports Select List**
    - Shows Select List with Saved Reports
  - **Rows Per Page Selector**
    - Allows user to select how many rows per page



164

## Restricting Functionality - Actions

- Developer has control over what Action Menu options are shown to the end user.
  - Actions will change based on design time selections
  - Can be done on a report by report basis
  - Certain options are only available to authenticated pages
    - Save Reports
    - Save Public Report
    - Subscriptions



165

## Restricting Functionality - Columns

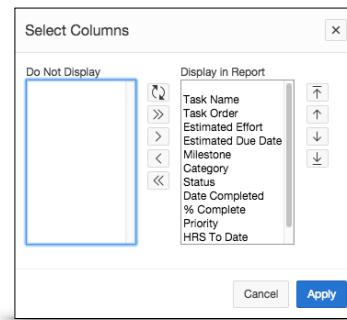
- Specific actions can also be restricted from a column
  - Allow the report to be filtered but do not allow a specific column to be filtered
  - Can also control the following on a column basis:
    - Sort, Filter, Highlight, Control Break, Aggregate, Compute, Chart, Group By, and Pivot



166

## Select Columns

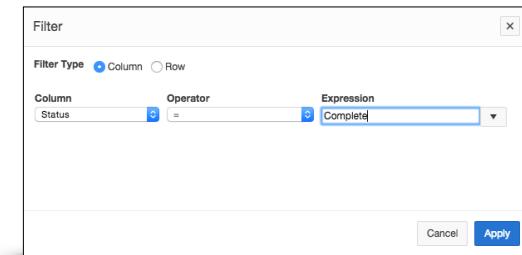
- Select or re-order columns of a report
  - Select which columns to displays
  - Re-Order Columns



167

## Filter

- Filter a column based on the result of a number of Operators
  - Conditions will be based on Data Type of Item
- Can have multiple filters per Interactive Report
  - Will be combined with ANDs



168

## Sort

- Specify Column Sort Options

- Column

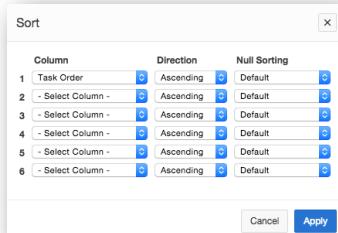
- Displayed and Not Displayed
- Computations

- Direction

- Ascending
- Descending

- Null Sorting

- Default
- Nulls Always First
- Nulls Always Last



169

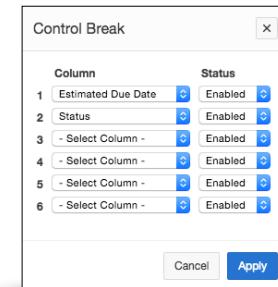
## Control Break

- User Defined Break Formatting

- Works MUCH better than standard reports break formatting does

- Break on multiple columns

- Status is default status for the break



170

## Column Headings

- Clicking on Column Headings allows the user to manipulate:

- Change Sort Order

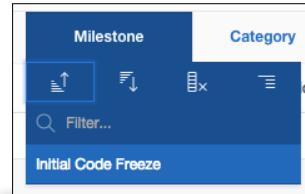
- Ascending
- Descending

- Add a Filter based on:

- User Entered String
- Actual Column Values

- Hide Column

- Add a Column Break



171

## Highlight

- Find matching data and highlight it by:

- Row

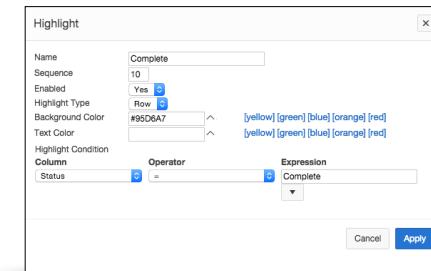
- Column

- Same operators as Filters

- Change color of:

- Background

- Text



172

## Compute

- Create **computations** based on data in columns and functions/logic
  - Columns are aliased to letters
  - Support for **Format Masks**
- Resulting column appears as a **pseudo-column** in your report
  - Can use it in filters, highlighting, etc.
  - Will be prefixed with \*\* in column selection lists



173

## Charts

- Display results as a Oracle JET Chart
  - **Horizontal Bar**
  - **Vertical Bar**
  - **Pie**
  - **Line**
- Chart Options
  - **Label**
  - **Value**
  - **Function**
  - **Sort**
  - **X & Y Axis Titles**



175

## Aggregate

- Performs aggregation function on a column
  - **Sum**
  - **Average**
  - **Count**
  - **Count Distinct**
  - **Minimum**
  - **Maximum**
  - **Median**
- Column must be of data type NUMBER
- Results displayed at the end of the report
  - Will only be displayed if corresponding column is also displayed



174

## Group By

- Allows Group functions to be applied to one column grouped buy another.
- **Group Functions**
  - Sum
  - Average
  - Min
  - Max
  - Median
  - Count
  - Count Distinct
  - Percent of Total Sum
  - Percent of Total Count



176

## Pivot

- Produces a Pivot view of data
- Users can define
  - Columns on which to pivot
  - Column to display as rows
  - Columns to aggregate
  - Function used to aggregate



177

## Save Report

- Named Report
  - Any user can name and save the current settings as a named report
  - Named Reports can be marked as Public so that all users can view them
- Default Report Settings
  - Only a user also signed on as a developer can save default settings
  - Takes current settings and applies them as defaults for all other users
  - Defaults can be saved as the Primary Report or a Named Alternative

The dialog box has fields for 'Save' (set to 'As Named Report'), 'Name', and 'Description'. It includes a note '(Only displayed for developers)'. At the bottom are 'Cancel' and 'Apply' buttons.

179



## Flashback

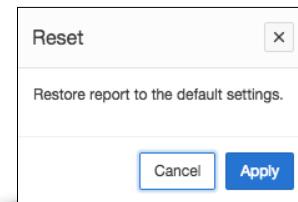
- Flashes back the database by X minutes to see what the data looked like at that point in time
  - Flashback must be enabled on the database
  - Length of time is configurable
    - Based on **UNDO\_RETENTION** parameter
    - Set to 3 hours by default



178

## Reset

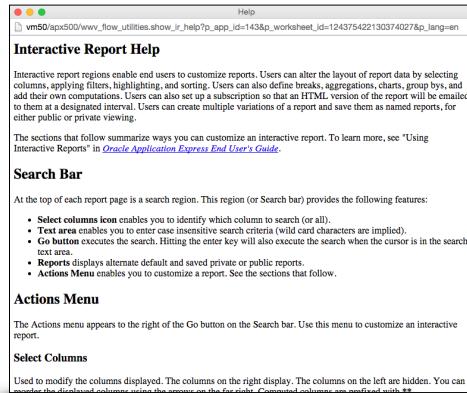
- Resets the Interactive Report to its default settings
  - All changes will be discarded



180

## Help

- Pops up a window containing Interactive Report-specific help

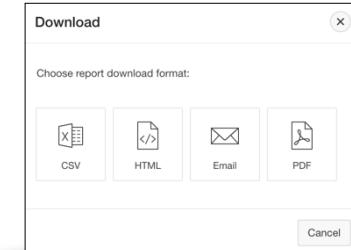


## Interactive Grids

183

## Download

- Allows user to download contents of the report in one of the following formats:
  - CSV
  - HTML
  - Email
  - XLS (Excel)
  - PDF
  - RTF (Word)
- Latter three require Oracle BI Publisher
  - Which may require a separate license from Oracle



182

## Interactive Grids

- Interactive Grids (or IGs) provide the ability to **update both rows & columns of data** in a single component
  - New in APEX 5.1, Interactive Grids are essentially the replacement for **Tabular Forms**
  - Upgrade utility included
- IGs use a **SQL statement** for their source
- Use of existing Item types
- Declarative support for **Cascading LOVs** and **Dynamic Actions**
- Master Detail / Detail support



184

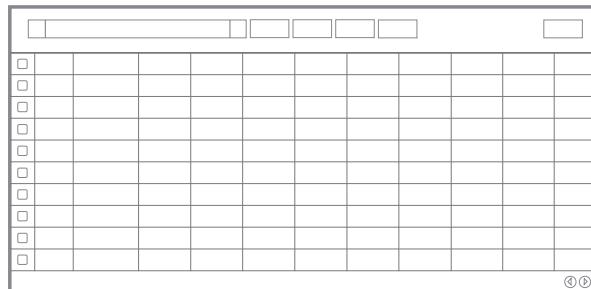
## Interactive Grids

- JSON-based
  - Faster and more lightweight Ajax communications
- Client-side and server-side validation
- Declarative validations and processes
  - Similar to APEX Items
- Support for bind variables to reference columns
- Execution scope: all rows or only rows modified and added
- Column groups, fixed headers and frozen columns
- Infinite scrolling and lazy loading
- Accessible, excellent keyboard support, right-to-left support



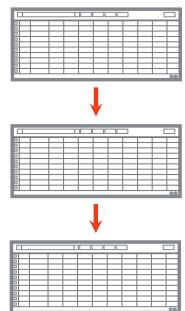
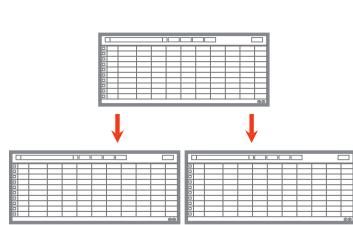
185

## Layout - Single IG



186

## Master Detail & Drill Down



187

## Multiple Drill Down



188

## 4.4 Interactive Grids

- Create an updatable Interactive Grid for **TASK\_STATUSES**

## Layout



190

## Layout

- The layout of an APEX application encompasses the **arrangement of the different components** within or on a page
  - Page > Regions > Items & Buttons
- This should be the first step when creating a page
  - Easier to arrange when style has not been applied
  - But the style should be something that you at least think about at this phase

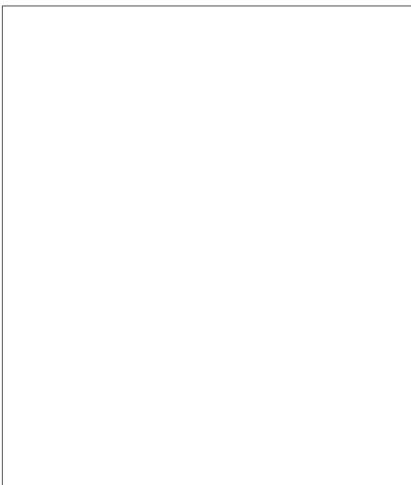
## Hierarchy



191

192

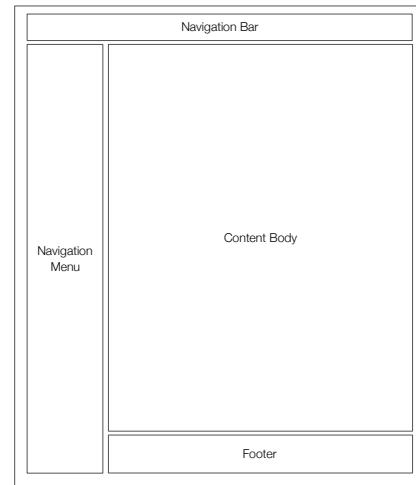
## Page



193

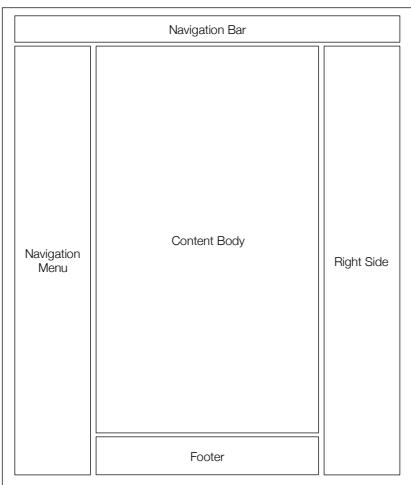


## Region Positions



194

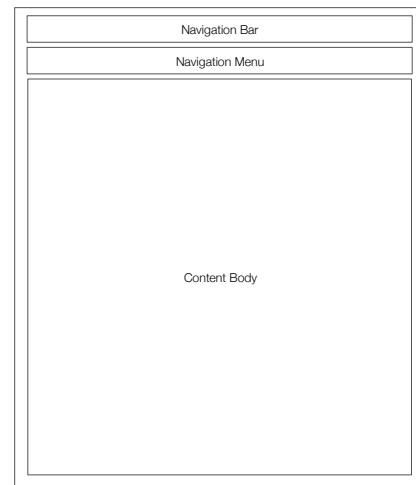
## Region Positions



195



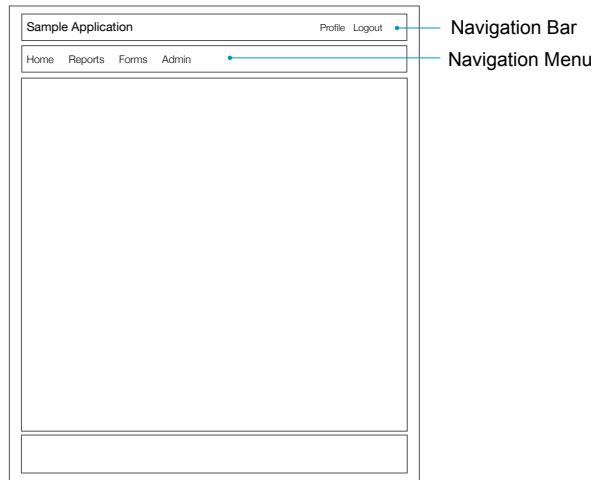
## Region Positions



196

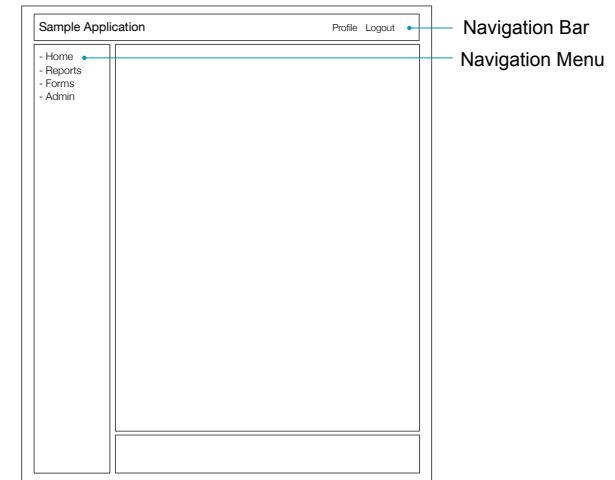


## Navigational Controls



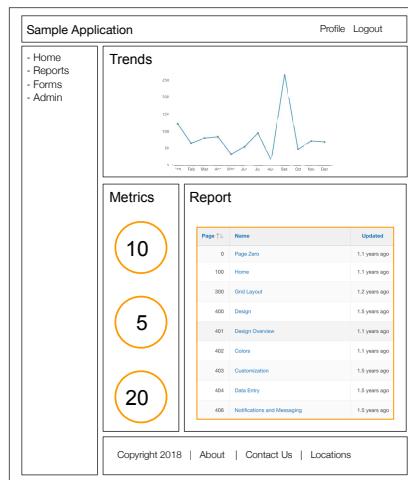
197

## Navigational Controls



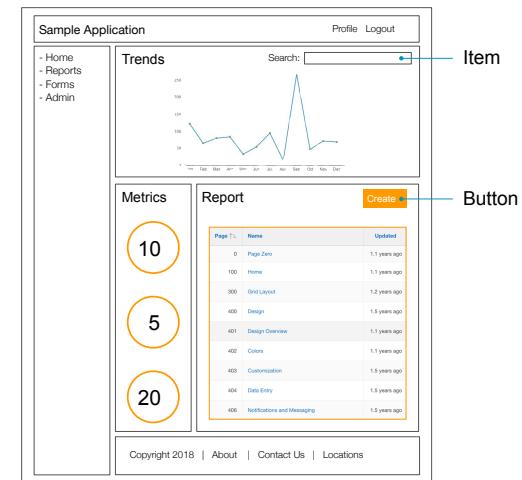
198

## Regions



199

## Items & Buttons



200

# Components

201

## Components

- The Universal Theme is **far more than just look & feel**
  - Several components are either part of it or are integrated into it
- Major components of the Universal Theme include:
  - **Page Templates & Modes**
  - **Global Page**
  - **Region Positions**
  - **Navigational Controls**
  - **User Interface Attributes**
- **Region, Item & Button Templates** are also part of the Universal Theme, but have more to do with Style vs. Layout and will be covered later



202

# Page Templates & Modes

203

## Page Templates

- When considering layout options, one of the first things to consider is the **Page Template** that you will use
- The Universal Theme provides **8 page template options**
  - If your requirements don't fit into one of those 8 options, change your requirements
- Unlike in the past, it is **not recommended that you change any HTML** in any of the APEX templates
  - Doing so will make it difficult, if not impossible to upgrade your application with future versions of the Universal Theme



204

## Page Templates

- Page templates in the Universal Theme:
  - Standard
  - Left Side Column
  - Right Side Column
  - Both Side Column
  - Master Detail
  - Dialog
  - Login
  - Minimal



205

## Page Mode

- A page can now be defined as **one of three modes**:
  - **Normal**
    - Standard page
  - **Modal Dialog**
    - Dialog that blocks the calling page
  - **Non-Modal Dialog**
    - Dialog that does not block the calling page
- **Easy to switch** a page from one to another
  - However, some regions only exist when the page is in dialog mode, and content will need to be moved



206

## Page Mode

- When the **Page Mode** is set to **Modal Dialog** or **Non-Modal Dialog**, there will be different **Page Template Options** available
  - Modal Dialog
  - Wizard Modal Dialog



207

## Region Positions

208

## Region Positions

- Each page in APEX contains a number of **Region Positions**
  - Position and type vary based on corresponding **Page Template**
  - Positions are fixed to a specific spot on the page
  - Many are purpose-specific: Breadcrumb, Page Navigation, etc.
- Regions on a page are mapped to a **Region Position**
  - Or to a Parent Region if they are a sub-region



209

## Region Positions

- Regions can be assigned to a **Region Position** by dragging & releasing that region onto the specific Region Position
  - However, some region positions may not be visible there
  - Alternatively, simply select the desired **Region Position** from the **Position** select list in the **Layout** region
    - Keep in mind that **Body 1**, **Body 2** & **Body 3** are **essentially deprecated** and should not be used
    - Most content should be mapped to **Content Body**



210

## Region Positions

- While this is a relatively simple concept, it is **critical to standardize** on which **page templates** & **region positions** will be used
  - Consistency should be enforced not just within an application, but also across applications
- Specific **page templates** should also only be used for **specific purposes**
  - Example: Restrict the right-side region for less important content, such as small charts and/or metrics



211

## Navigational Controls

212

## Navigation Menu

- The **Navigation Menu** drives the core navigation of an APEX application
  - Replaces legacy tabs from previous versions
  - Is nothing more than a list
- Easy to change where it is displayed:
  - Side
  - Top



213

## Navigation Menu

- The **Navigation Menu** is nothing more than a **List** that will be used by the Universal Theme for navigation
  - Can have one per User Interface (Desktop & Mobile)
- By default, one is **added automatically** with the Universal Theme
  - Any List can be used as the **Navigation Menu**
  - Defined at **Shared Components > User Interfaces > Desktop** or **Mobile**



214

## Navigation Bar List

- The **Navigation Bar List** can either be a **List** or use the legacy **Navigation Bar Entries**
  - Defined in the **User Interface** section
  - Defaulted to a **List** when creating new applications
  - **Classic Navigation Bar Entries** when importing a pre-5.0 application
- Same **List Templates** available as for Navigation Menu
  - As long as implementation is set to List



215

## User Interface Attributes

- In APEX, a **User Interface** can be defined as either **Desktop** or **Mobile**
  - Designed to allow a single application have different settings for different devices
- **Note:** As of APEX 18.1, support for **jQuery Mobile** in APEX will be **discontinued**
  - Not sure how User Interfaces will be impacted



216

## User Interface Attributes

- A User Interface contains a **number of settings** that work with the Universal Theme
  - **General Attributes**
    - Home & Logout URLs
  - **Navigation List/Bar Options**
    - Location, Templates, Template Options
  - **JavaScript Options**
    - CDN, Files, Options
  - **CSS**
    - Files



217

## Theme Defaults

## Theme Defaults

- There are a number of defaults that can be set at the Theme level
- Defaults serve **one of two purposes**
  - Value will be used when new components are created
    - More common use
  - Value will be used when component is set to use Default
- Broken into three sections:
  - **Component**
  - **Region**
  - **Dialog**



219

## Theme Defaults

- Best Practice:
  - **Standardize** on a set of defaults
  - Ensure that **all new applications** have those options selected
    - Achieved by starting with a “template” application - which is just a copy of an application with the proper defaults set
  - Retrofit existing applications that do not adhere to the standard



220

# Regions

221

## Regions

- Regions come in **many different variations**
  - Reports, Charts, Static, Calendar, Plugins, etc.
- Regardless of the type of region, **they are all treated equally** when it comes to layout
  - In fact, all regions share about 90% of the same core attributes
  - It's the source that typically changes from region to region



222

# Grid

- The Universal Theme has made vast strides in improving how APEX regions & items are laid out on the page
- All regions & items are placed on a **12 column grid**
  - Why 12? Allows for the even distribution of 2, 3, 4, 6 or 12 columns
  - Using 10 would only get 2, 5 or 10
- As a developer, you can click "**Show Layout Columns**" to see a "candy-striped" view of these columns beneath the regions on the page



223

## Layout - Columns

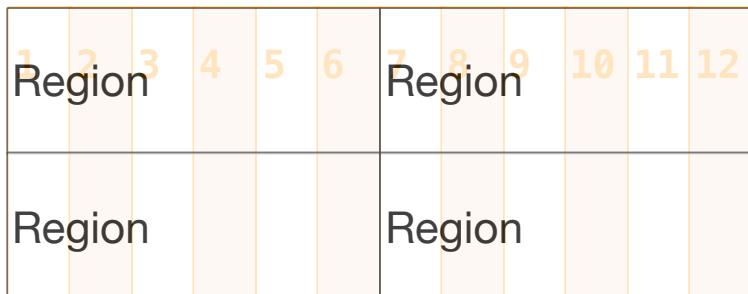
- Like many popular frameworks, the Universal Theme uses a **12 column grid** for region & item layout



224

## Layout - Rows

- **Multiple rows** in a grid can be added to a page
  - Typically done when a new group of items needs to be horizontally aligned by their top edge



225

## WYSIWYG (sometimes)

- APEX 5 introduced a new layout editor for regions, items & buttons
  - Visible in the **Page Designer** view only
- With this editor, it is possible to arrange regions & items by dragging & dropping them
  - But it can take some getting used to and can be a bit frustrating
- Good approach is to use the **Layout Editor** to create and roughly arrange regions/items
  - Then use the declarative attributes to fine-tune/correct



227

## Layout Attributes

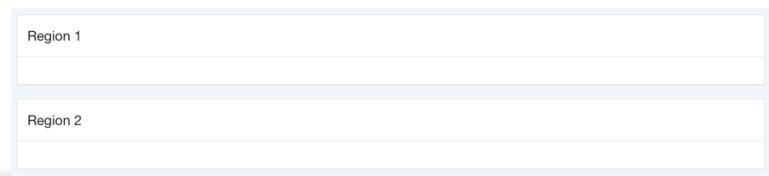
- Rows and Columns are managed via attributes in the **Layout** section
  - **Start New Row**
    - Creates content on a new row
  - **Column**
    - Determines which column to start the content in
  - **New Column**
    - Creates content in a new column
  - **Column Span**
    - Determines how many columns to use for the content



226

## Region Layout

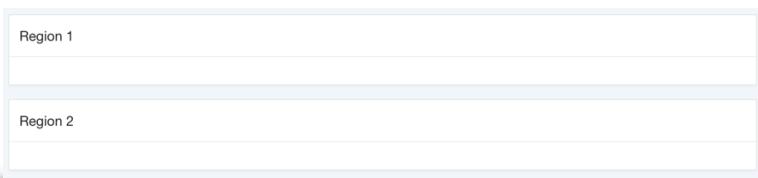
- Region on same row and column in the grid
  - Start New Row: **No**
  - New Column: **No**



228

## Region Layout

- Region in new row in the grid
  - Start New Row: **Yes**



229

## Region Layout

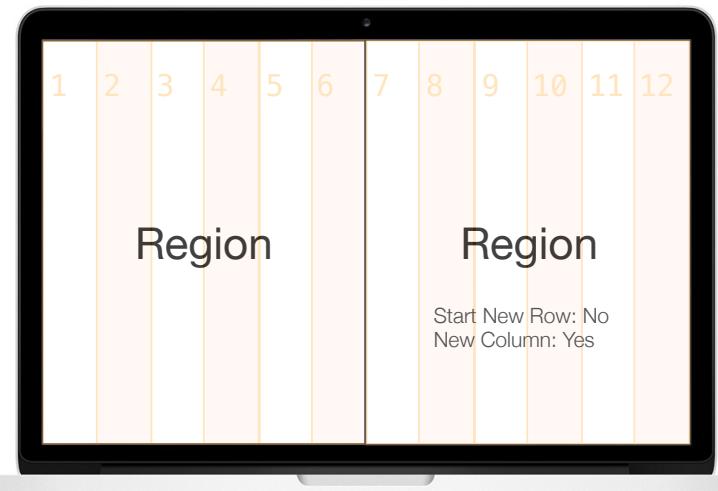
- Region on same row in the grid but in a new column
  - Start New Row: **No**
  - New Column: **Yes**



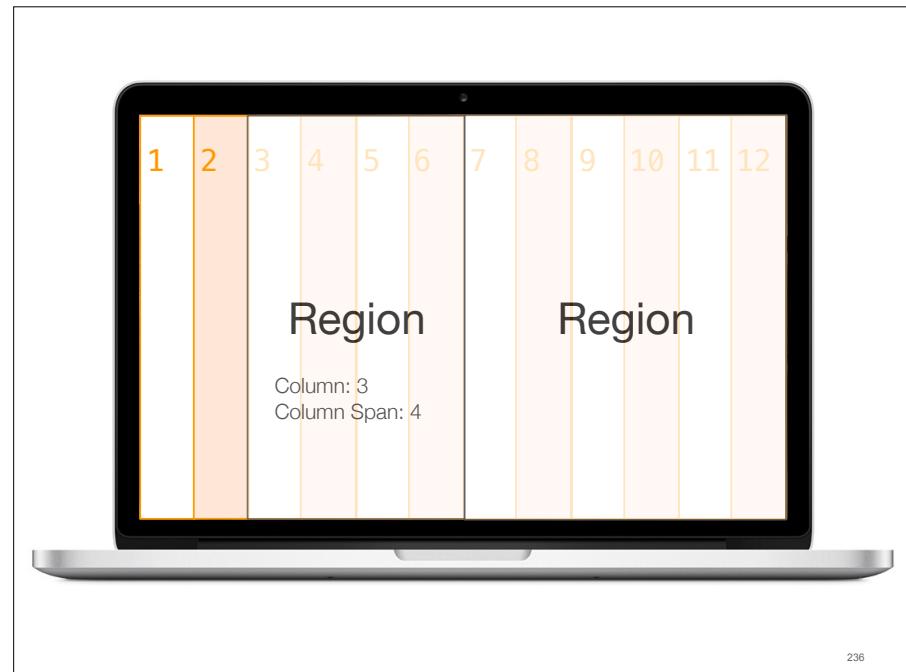
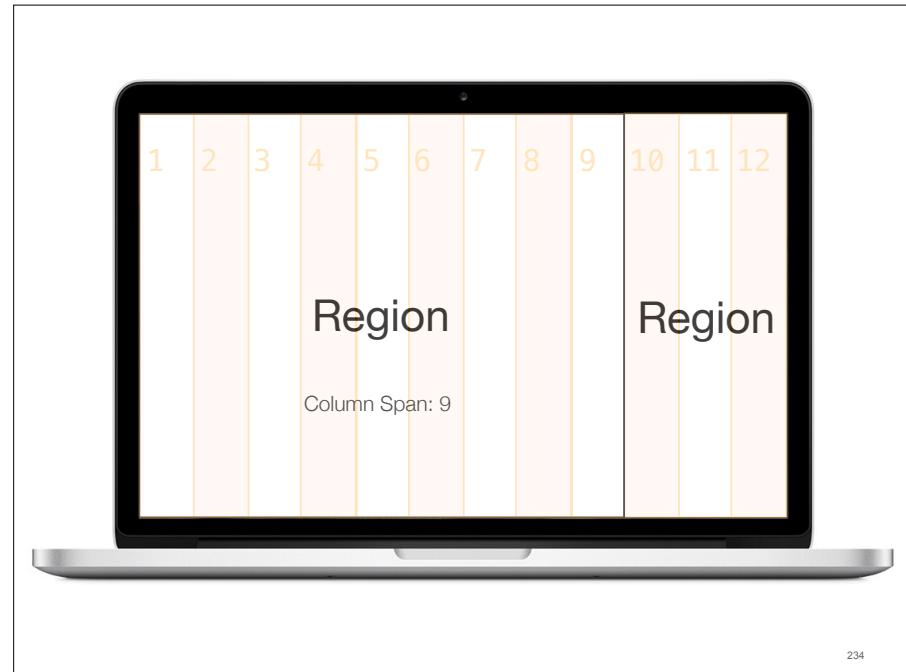
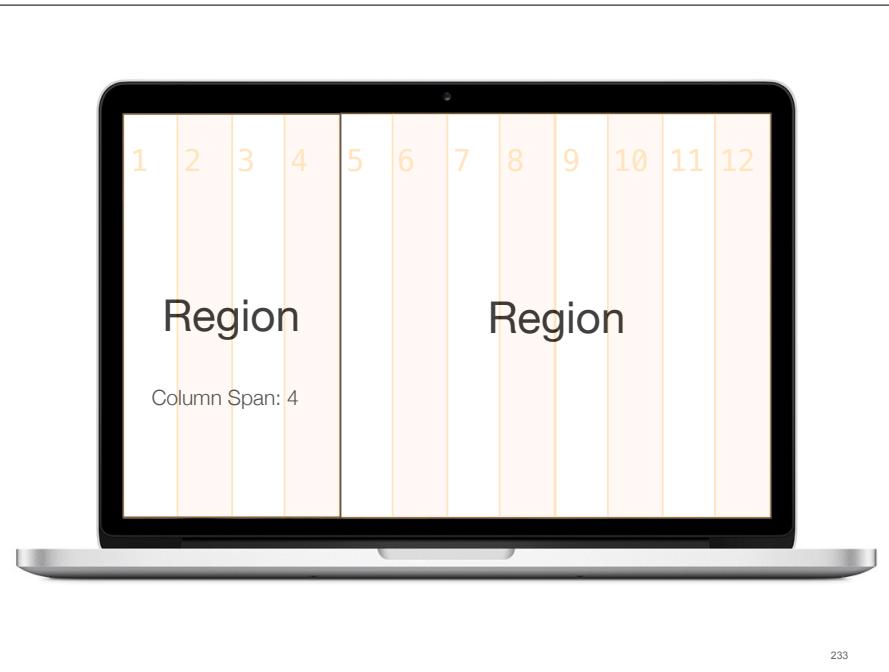
230



231



232





237



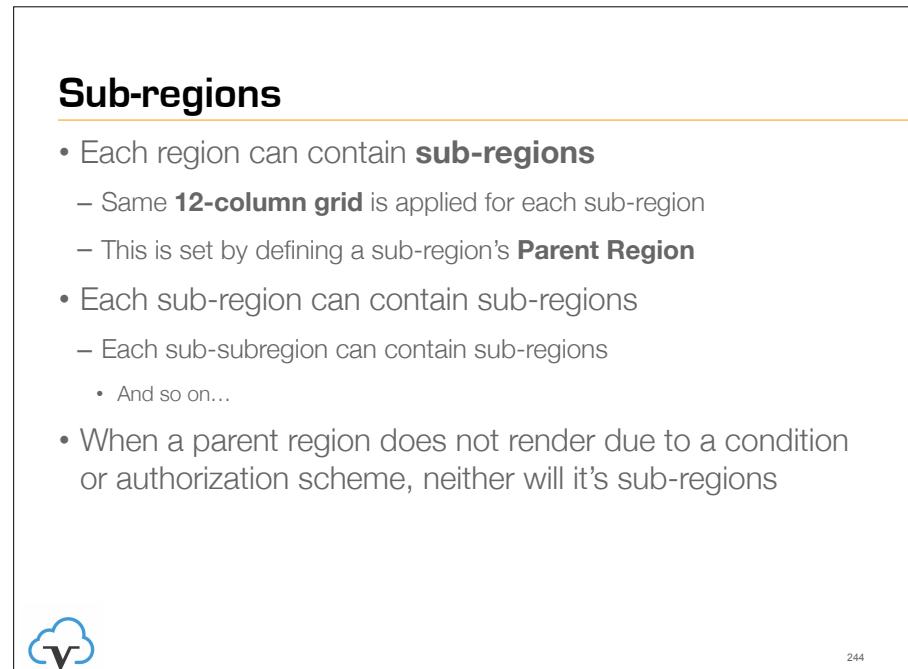
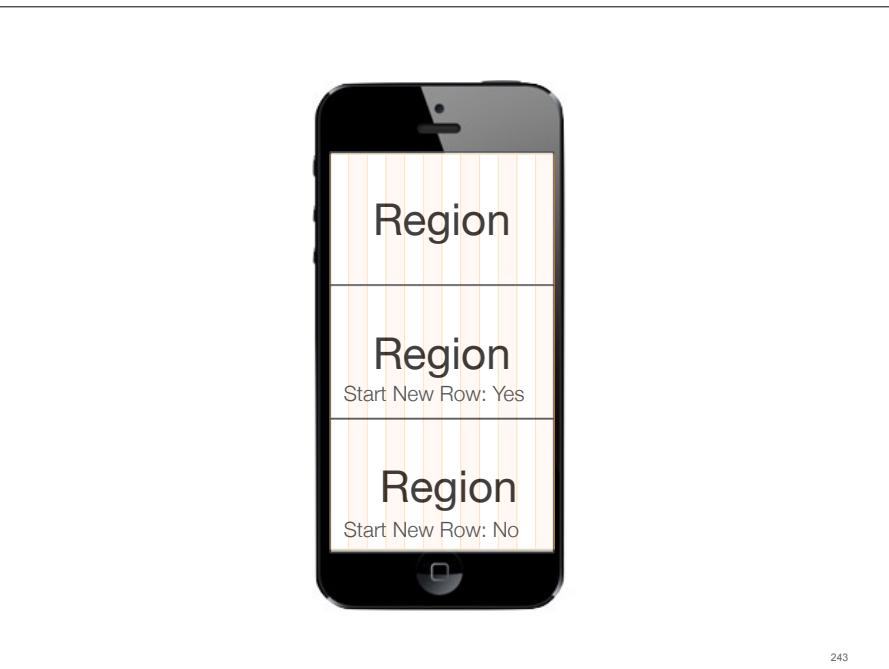
238



239



240



## Sub-regions

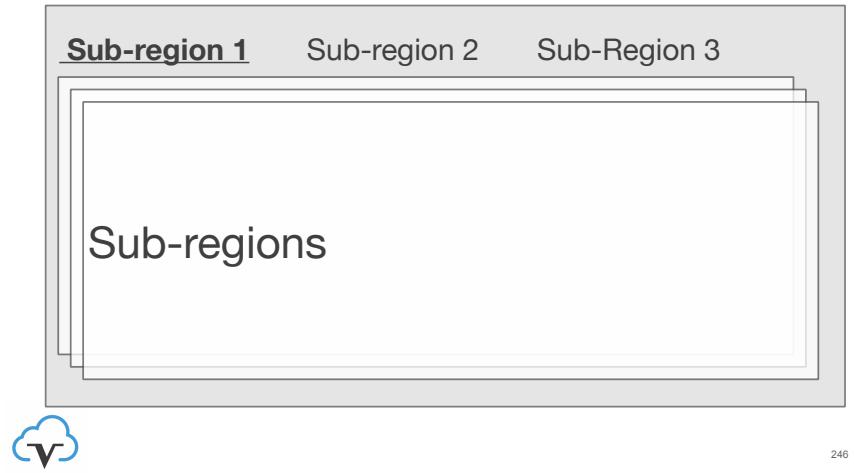
- Useful for grouping content that you want to stay aligned or flow better



245

## Sub-regions

- Also useful for the **Region Display Selector**



246

## Items

247

## Items

- Items are how **users interact with the database**
- Several **different types**
  - Text field, Text Area, Radio Group, Select List, etc.
- Regardless of the type, **all items are laid out the same way**
  - Keep in mind certain types will look better with more columns allocated to them
  - Also, Item Templates needs to be accounted for, as they will also take up columns



248

## Item Layout

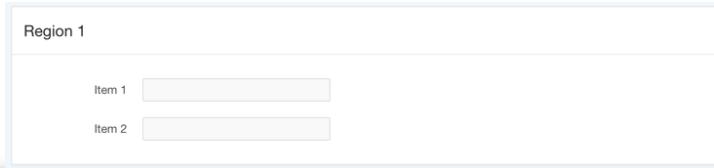
- The 12-column grid also extends to within a region
  - Each region has 12 columns to use when adding items or sub-regions
  - Functions the same way for items within a region as it does for regions on the page
- Click “**Show Grid**” in the developer’s toolbar to display the grid
  - Makes it a lot easier to see how it works and what is where



249

## Item Layout

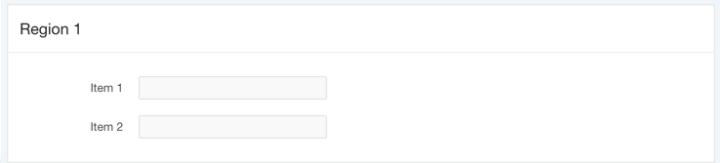
- Item on same row and column in the grid
  - Start New Row: **No**
  - New Column: **No**



250

## Item Layout

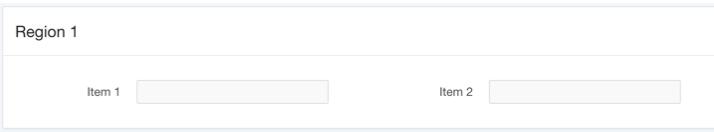
- Item in new row in the grid
  - Start New Row: **Yes**



251

## Item Layout

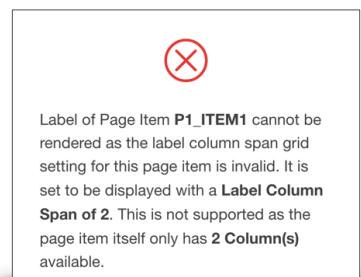
- Items on same row in the grid but in a new column
  - Start New Row: **No**
  - New Column: **Yes**



252

## Label Column Span

- By default, APEX will **distribute labels evenly** across a row
  - This can optionally be overridden using the **Label Column Span** attribute
  - Careful when doing this; you may run into an error denoting that you ran out of columns
  - Simply reset the **Label Column Span** attribute to fix this or use “above” option of the label
  - **Label Column Span** only available in 5.1.3+



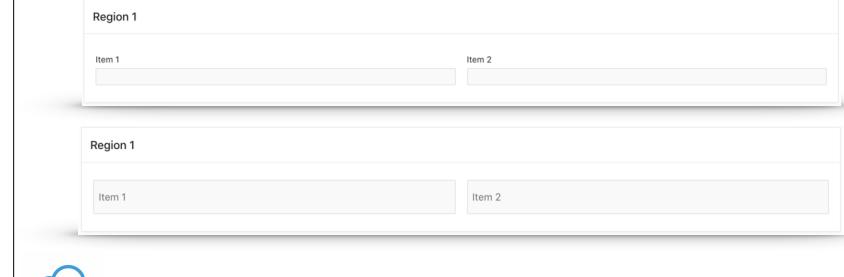
253

## Buttons

255

## Labels Above & Floating Labels

- Another option for labels is to set the template to **Optional/Required - Above** or **Optional/Required - Floating**
  - This will render the labels either above the item or within in the items itself



254

## Buttons

- Buttons are used to **initiate an action**
  - Save, Delete, Cancel, etc.
- Typically placed in a **Button Position**
  - But can also be placed **inline with items** as well as **above/below a region**
- Buttons can contain
  - **Text**
  - **Icon**
  - **Icon & Text**



256

## Button Positions

- Different region templates will contain **different button positions**

A screenshot of a standard region template showing buttons for COPY, EDIT, PREVIOUS, and NEXT at the top, and CLOSE, HELP, DELETE, CHANGE, and CREATE at the bottom.

Standard

A screenshot of a Buttons Container region template showing buttons for PREVIOUS, CLOSE, and DELETE at the top, and ITEMS, REGION CONTENT, and SUB REGIONS below. At the bottom, there are buttons for CHANGE, EDIT, CREATE, and NEXT.

Buttons Container

A screenshot of an Inline Dialog region template showing buttons for PREVIOUS, DELETE, CLOSE, EDIT, CREATE, and NEXT at the bottom.

Inline Dialog

A screenshot of a Wizard Container region template showing buttons for PREVIOUS, CLOSE, and NEXT at the bottom, along with WIZARD SUB REGIONS.

Wizard Container



257

## Button Positions

- Caution:** not all button positions render where you think they will
  - Previews in the page designer are not entirely accurate
- Universal Theme Sample Application** will show where all the named buttons positions are
  - Components > Region Name

A screenshot of a page titled "Button Positions" which contains the text "The following region shows all possible button positions." Below this is a form with fields for "Region Title" (set to "Previous"), "Copy" (button), "Edit" (button), "Next" (button), "Close" (button), "Help" (button), and "Delete" (button).

258

## Modal Form on a Table with Report

259

## Dialog Style Pages

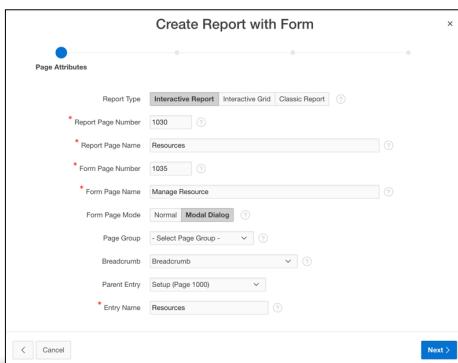
- When using the Universal Theme, APEX provides the ability to present nearly any page as a **Dialog**
  - The calling page stays visible to the user
  - The called page is presented as a dialog and floats above the calling page
  - The resulting dialog may be
    - Modal** - the user is not allowed to interact with the calling page
    - Non-Modal** - the user is allowed to interact with the calling page



260

## Dialog Style Pages

- Dialog pages are created using the same wizards
  - Simply choose an alternate Page Mode



261

## BLOBs

- Files such as Word Documents, PDF Files and Images can easily be stored in the Oracle Database as a **BLOB** (Binary Large OBject) data type
- Benefits:
  - DB gets backed up, files get backed up
  - Oracle Text can index & search the contents of almost any file type
  - Simple to implement in APEX



262

## BLOBs

- Allows you to **seamlessly integrate BLOB data types** into your APEX forms & reports
  - In the “old days”, this had to be done by hand
    - Code to move item from shared table to your schema
    - Code to securely download item



263

## BLOBs

- Feature is **seamlessly integrated** into the APEX wizards
  - When you create a form on a table with a BLOB column, APEX will automatically implement the Declarative BLOB feature
  - No additional options to specify or code to write
- Result will be a form that uploads files directly into the specified database table
  - Also works for Reports; will provide download link as well



264

## Formatting BLOBs

- You may want to plan ahead and include the following additional columns in your tables that will use the Declarative BLOB feature:

### - FILE\_NAME

- Stores the actual file name, which is used when a user downloads the file

### - MIME\_TYPE

- Stores the type of the file so that browsers know which application to launch (Word for .DOC for example)

### - LAST\_UPDATED

### - CHARACTER\_SET



265

## Formatting BLOBs - Forms

- At the Item level
  - set the Type to **File Browse**
  - Set Storage Type to **Blob Column Specified in Item Source**

The screenshot shows the 'Identification' section with Name set to P230\_DOCUMENT and Type set to File Browse. In the 'Label' section, the Label is set to Document. Under 'Settings', the Storage Type is set to BLOB column specified in item S. The 'MIME Type Column' is set to MIME\_TYPE, 'Filename Column' to FILE\_NAME, and 'Character Set Column' to null. The 'BLOB Last Updated Column' is set to null. The 'Display Download Link' checkbox is checked ('Yes'). The 'Download Link Text' field contains 'Attachment'. The 'Content Disposition' dropdown is set to Attachment.



267

## Formatting BLOBs - Reports

- At the Column level set the Type to Download BLOB
  - Shows Blob attributes section where attributes can be set
    - Table Name
    - Blob Column
    - Primary Key Column(s)
    - Mime Type Column
    - File Name Column
    - Last Updated Column
    - Character Set Column



266

The screenshot shows the 'Identification' section with Column Name set to DOCUMENT and Type set to Download BLOB. In the 'Heading' section, the Heading is set to Document and Alignment is set to center. Under 'Layout', the Sequence is set to 5 and Column Alignment is set to left. The 'BLOB Attributes' section includes settings for Table Owner (PARSING\_SCHEMA), Table Name (TASK\_DETAILS), BLOB Column (DOCUMENT), Primary Key Column 1 (TASK\_DETAIL\_ID), Primary Key Column 2 (null), Mime Type Column (MIME\_TYPE), Filename Column (FILE\_NAME), Last Updated Column (null), and Character Set Column (null).

## Formatting BLOBs - Forms

- At the Item level
  - Set Storage Type to **Blob Column Specified in Item Source**
- Set Attributes
  - Mime Type Column
  - File Name Column
  - BLOB Last Updated Column
  - Character Set Column
  - Display Download Link
  - Download Link Text

The screenshot shows the 'Identification' section with Name set to P230\_DOCUMENT and Type set to File Browse. In the 'Label' section, the Label is set to Document. Under 'Settings', the Storage Type is set to BLOB column specified in item S. The 'MIME Type Column' is set to MIME\_TYPE, 'Filename Column' to FILE\_NAME, and 'Character Set Column' to null. The 'BLOB Last Updated Column' is set to null. The 'Display Download Link' checkbox is checked ('Yes'). The 'Download Link Text' field contains 'Attachment'. The 'Content Disposition' dropdown is set to Attachment.



267

## Dialog Style Pages

- Our new pages will provide
  - A report listing the available Project Resources
  - A Form allowing the creation or edit of Project Resources
  - Creating or Editing will happen in a Modal Dialog

The screenshot shows two views of a modal dialog for 'Project Resources'. The left view is the 'Project Resources' page with a table of resources (e.g., 'Dusty', 'Lee', 'Ava', 'Hal') and a 'New' button. The right view is the 'Edit Project Resource' dialog, showing fields for First Name (Dusty), Last Name (Lee), Username (DUSTY), Active Flag (V), Created (30-MAR-2019), Updated (31-MAR-2019), Photo (image of Dusty Lee), and Password (REDACTED). Buttons for 'Cancel' and 'Delete' are at the bottom.



268

## 4.5 Modal Form on a Table with Report

- Create a Form on a Table with Report on **RESOURCES**
  - Make the Form a Modal Dialog
  - Adjust the page to use the correct LOV's



270

## Lists

### Lists

- Lists are, well, Lists...
- There are two types of lists
  - **STATIC**
    - Made up of **predefined** display and return values
  - **DYNAMIC**
    - Based on a **SQL Query** or **PL/SQL Function** returning a SQL Query
    - Executed at runtime
- Lists can be Flat or Hierarchical
  - If hierachal, specific List templates must be used



271

### Lists Repurposed

- In APEX 5+, lists play a more predominant role when it comes to navigation
- There are two new “types” of Lists:
  - **Navigation Menu List**
    - Which is another way of saying “Tabs”
  - **Navigation Bar List**
    - We already worked with this - Login & Logout
- Any list can be either of these
  - Defined at **Shared Components > User Interfaces**



272

## Traditional List Usage

- Standard Lists regions can and are also still widely used in APEX
  - “Tasks” Region
    - List of tasks that a user can perform
  - Sub Navigation
    - List of pages that a user can navigate to



273

## Static List Entries

- Each **List Entry** can:
  - Optionally contain a Parent List Entry
  - Contain an image or text
  - Link to either an APEX page or External URL
  - Be current for one or more pages
  - Automatically record each time a user clicks it
  - Be associated with:
    - An APEX condition
    - An Authorization Scheme
    - A Build Option



275

## Static Lists

- Each **Static List** contains one or more **List Entries**
  - Each entry is a link to either another APEX page or an external URL



274

## 4.6: Lists

- Create an **Setup Tasks** list to be displayed on the Setup Home Page
  - Assign icons to each list
  - Add the list to Page 1000



# Roles for Setup Pages

277

## Roles

- New to APEX 18, you can create **Roles** within your application
  - Roles can then be mapped to users via the **Access Control** application feature
  - Role and role mapping data stored in **APEX\_APPL\_ACL\_ROLES** & **APEX\_APPL\_ACL\_USER\_ROLES**
- When installed, there will be three roles created by default
  - **Admin**, **Contributor**, **Reader**
- Each role will have a corresponding **Authorization Scheme**



278

# Authorization Schemes

- What do you have access to?
- Can be associated with almost every APEX Component
  - Application, Page, Region, Item, Process, Validation, etc.
- When scheme evaluates to TRUE, item renders or process executes



279

# Authorization Scheme Types

- **Authorization Scheme Types**
  - Is/Is Not Member of Role
  - Exists/Not Exists SQL Query
  - Item is NULL/NOT NULL
  - Item Comparison
  - PL/SQL Function
  - User Preference
- **Evaluation Point**
  - Per Page View
  - Per Session
  - Per Component
  - Always



280

## 4.7: Roles for Setup Pages

- Create a new **Role**
- Create a new **Authorization Scheme** and associate it with the role
- Associate the **Authorization Scheme** with all of the **Setup** pages



## Advanced Forms & Reports

282

## Advanced Forms, Reports and Pages

- Thus far you've created straight forward reports and forms
  - APEX provided wizards created each on its own page
  - Reports were generated in a standard tabular format
  - Forms were fairly uncomplicated in their layout needs
- There are times when the application requires you to go beyond the basics
  - More complex Form and Report layouts
  - Multi Functional Pages
  - Use of Plug-in Region and Item Types



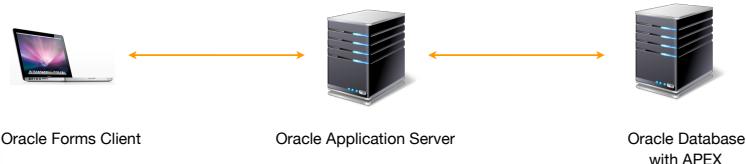
283

## Session State

284

## Database Session

- Similar to a Phone Call
  - Both parties have to invest resources to carry on a conversation
    - Even if no one is talking
  - Each additional conversation puts some additional strain on the host party



285

## APEX Session

- smlr 2 a txt msg 😊
  - Both parties need few resources to carry on a conversation
    - Regardless of whether someone is talking or not
  - Each additional conversation puts little additional strain on the host party



286

## Session State

- APEX sessions are **logically and physically distinct** from **underlying database sessions**
  - **ORDS** on the Middle Tier will connect to the database via the **APEX\_PUBLIC\_USER**
  - Multiple APEX users can **share the same database connection**
  - Thus, there is a **one-to-many relationship** between APEX users and database sessions
    - Main reason APEX can scale as well as it does

## Session State

- Since HTTP is a stateless protocol, APEX must rely on an external mechanism to manage session state
  - Thus, the **APEX engine has a built-in session state management component**
    - So integrated that you can't disable it
    - Unique Session IDs are assigned to each user of an application
  - APEX by default will purge sessions older than 24 hours every hour
    - Job can be changed to alter the interval



288

## Session State

- Values of all APEX items - both page and application - are tied to a unique Session ID
  - Commonly referred to **APP\_SESSION\_ID**
- You can see the Session ID in the URL of most pages in an APEX application:



https://apex.oracle.com/pls/apex/f?p=4550:1200143790664

## Session State

- APEX Session State values are **stored in a table** in the database
  - **More secure**
    - Nothing sensitive is in a cookie
  - **More efficient**
    - Data is where it gets processed
- Session State management functions the same regardless of how users authenticate
  - Even unauthenticated users will get a Session ID



290

## Session State – SET

- User Input via Items
- Computations
- Processes
- PL/SQL Code
  - Inside anonymous PL/SQL blocks in APEX:
    - `:P1_ITEM_NAME := 'some_value';`
  - Inside PL/SQL program units called from APEX:
    - `apex_util.set_session_state('P1_ITEM_NAME', 'some_value');`

## Session State – GET

- SQL
  - `:P1_ITEM_NAME`
- PL/SQL
  - Inside anonymous PL/SQL blocks in APEX:
    - `:P1_ITEM_NAME`
  - Inside PL/SQL program units called from APEX:
    - `v('P1_ITEM_NAME')`
- Templates or Regions/Tabs/Menus/Lists
  - `&P1_ITEM_NAME.`
- Conditions
  - `P1_ITEM_NAME`



291

## Viewing Session State

- As an APEX developer, you can examine the current values of items in your session via the Session link in the Developer's Toolbar
  - Page Items
  - Application Items
  - Session State
  - Collections
  - All of the Above

The screenshot shows the 'Session State' page of an APEX application. At the top, it displays the URL: vnd01px000f7p=402034-8789180419472/PAGE/NO.2474000\_P04\_SESSION/4000\_P04\_FLOW/F400\_P04\_PAGE/F0\_P0\_FLOW\_ID.317754838124... and the session ID: 217754838124. Below this, it shows the application ID: 143 Project Tracking, user: SCOTT, and workspace: 12345678901234567890. The main content area is titled 'Page Items' and contains a grid of session items. The columns are: Application ID, Page, Item Name, Display, Item Value, Status, and Encrypted. The grid lists various items such as P020\_TASK\_ID, P020\_MILESTONE\_ID, P020\_PROJECT\_ID, P020\_RESOURCE\_ID, P020\_TASK\_CATEGORY\_ID, P020\_TASK\_NAME, P020\_TASK\_DESCR, P020\_TASK\_ORDER, P020\_ESTIMATED EFFORT, P020\_DATE\_COMPLETED, P020\_TASK\_STATUS\_ID, P020\_TASK\_PRIORITY\_ID, P020\_TASK\_ID\_NEXT, P020\_TASK\_ID\_PREV, and P020\_TASK\_ID\_COUNT. Most items have a status of 'Inserted' and are not encrypted. At the bottom of the grid, it says '1 - 18'. Below the grid is a toolbar with links: Home, Application 143, Edit Page 50, Session (highlighted), View Debug, Debug, Show Grid, Quick Edit, Theme Roller, and a help icon.

293

## Built-in Items

- APP\_ID**
  - Application ID of currently running Application
- APP\_ALIAS**
  - Alias of currently running Application
- APP\_USER**
  - Currently signed on user
- APP\_SESSION**
  - Session ID of currently signed on user
- APP\_PAGE\_ID**
  - Currently running page ID



294

## APEX URL Syntax

f?p=	URL
APP_ID:	Application ID
APP_PAGE_ID:	Application Page
APP_SESSION:	Session ID
REQUEST:	Request (Button)
DEBUG:	Debug Mode
Clear Cache:	Clear Cache/Pagination
item1, item2:	Item Names
itemValue1, itemValue2:	Item Values
printerFriendly	Printer Friendly

295

## Using the URL Syntax

- Examples of URL Syntax:
  - Clear the Session Cache for Page 10:
    - f?p=&APP\_ID.:10:&APP\_SESSION.:::10
  - Setting the value of P2\_ID to 1234:
    - f?p=&APP\_ID.:5:&APP\_SESSION.::NO::P2\_ID:1234
  - Running Page 5 in Debug Mode:
    - f?p=&APP\_ID.:5:&APP\_SESSION.::YES



296

# SQL Injection

297

## SQL Injection

- Code injection technique
  - Presented where user input isn't escaped or strongly typed and is then used in the formation of a SQL statement.
  - Effectively allows the user to alter the core functionality of the SQL that is executed
- 3 potential areas of concern
  - Use of **&ITEM.** notation within SQL or PL/SQL
  - Calls to **DBMS\_SQL**
  - Calls to **EXECUTE IMMEDIATE**



298

## Flawed SQL

- All it takes is a single SQL injection flaw to open the flood gates which allows any SQL to be run
- Our example contains a report with the following SQL:

```
SELECT empno, ename, job  
      FROM emp WHERE ename LIKE '%&P1_ITEM.%'
```

- Using the **&ITEM.** Syntax will allow a user to re-write the SQL statement

299

## Flawed SQL

- Thus, if the user enters a malicious string as a filter, the SQL will be re-written:

```
SELECT empno, ename, job  
      FROM emp WHERE ename LIKE '%' UNION  
SELECT empno, ename, to_char(sal) job FROM emp  
WHERE '%' LIKE '%'
```

- Now, the SQL will return the **SAL** of each employee - something that was not part of the intended functionality of the application



300

## Flawed SQL

- Or:

```
SELECT empno, ename, job  
  FROM emp WHERE ename LIKE '%ABC' UNION ALL SELECT  
NULL,TO_CHAR(CREATED),USERNAME FROM SYS.ALL_USERS --%
```

- Now, the SQL will return the **CREATED, USERNAME** and **USER\_ID** from **SYS.ALL\_USERS**
- Essentially, it's trivial to neuter the original query and introduce any new query we want via a simple UNION



301

## Card Report

303

## Resolution

- Use a bind variable to eliminate this threat
- Thus, the new SQL should read:

```
SELECT empno, ename, job  
  FROM emp  
 WHERE ename LIKE '%' || :P1_ITEM || '%'
```

- Using the **:ITEM** Syntax will eliminate the possibility of a malicious string of altering the syntax of our query



302

## Universal Theme Report Templates

- The Universal Theme provides a number of new and interesting report templates
  - Provide information in more consumable formats
  - Can provide a better overview than a standard tabular report
- Templates include
  - **Alerts**
  - Badge List
  - **Cards**
  - Comments
  - Search Results
  - Time Line
  - **Value Attribute Pairs**



304

## Card Report

- The Card Report template presents data in the form of virtual index cards.
  - Title
  - Text
  - SubText
  - Link
  - Initials
  - Icon
  - Modifiers



305

### 5.1: Card Report

- Create a new page (200) to display projects using the Card Report Template
- Modify the Template Options of the report



## Card Report - SQL Syntax

- The Card Report requires certain columns to be generated by the underlying query

```
SELECT
  card_title,
  card_text,
  card_subtext,
  card_link,
  card_initials,
  card_icon,
  card_modifiers
FROM
  table
```



306

## Multi-Functional Page

308

# Building Complex Pages

309

## Building Complex Pages

- Often you'll need to create more complex pages to provide a more complete overview of data
  - Master report with multiple Detail reports
  - Links to various edit forms
  - Conditional regions
  - Plug-in Region Types
  - Region Display Selectors



310

# Building Complex Pages

- The best approach is to
  - Map the page requirements ahead of time
    - Know what data/components will be required
    - Identify need for plug-ins or extended functionality
    - Identify if any components will interfere with others
    - Insure layout of all data will be possible and readable
  - Build the page up one component at a time
    - Start with the lowest level components built on the Master data
    - Make sure each component works before moving on
    - Move on to each component in the order they are required

311

## Project Overview Mockup

A screenshot of a web-based project management application interface. The top navigation bar includes links for Home, Projects, Task Calendar, Setup, and Administration. The main content area is titled 'Project > Project Name' and shows tabs for 'Show off!', 'Milestones & Resources', and 'Requirements & Tasks'. The 'Milestones' section displays two items: 'October' (with a 'MILESTONE NAME' field) and 'November' (with a 'Milestone Details' link). The 'Resource Assignments' section shows 'GEDDY' assigned to 'Oskar Lee' (1 Task) and 'NEIL' assigned to 'Neil Pearl' (0 Tasks). The 'Requirements' section contains a 'Requirement' card with a plus sign and a 'Sub-Requirement' link. The 'Tasks' section lists three tasks: 'Task 1' (Due Date: 30-SEP-2015, Username: Geddy, Status: Pending), 'Task 2' (Due Date: 30-SEP-2015, Username: Neil, Status: Functionality Complete), and 'Task 3' (Due Date: 30-SEP-2015, Username: Alex, Status: Complete). A sidebar on the right contains a 'Edit Project' button and icons for Requirements, Tasks, Resources, and Milestones, along with a large 'X' button.



312

## Creating the Overview Page

313

## Overview Page

- The Project Overview page will serve as the hub for all components related to a project
  - Reports
  - Metrics
  - Buttons
    - That link to other pages



314

### 5.2.1: Creating the Overview Page

- Create a blank page you'll use to build up the complex components
  - Alter the Page Template
- Add the Metrics report to the Project Overview page
  - Apply the corresponding Template Options



## Page Layout

316

## Region Display Selectors

- Special region type that enables the end-user to select which participating region to show (focus on).
  - Regions on the page must have their **Region Display Selector** attribute set to **YES** to be controlled
  - Those regions set to **NO** will not be affected by the Region Display Selector
  - Selecting a **Region Name** in the Region Display Selector will hide all other participating regions and show only the selected one
  - Selecting **Show All** will show all participating regions



317

## Milestones

319

## 5.2.2: Adding a Region Display Selector

- Create a Region Display Selector for the page
- Create two placeholder regions



## Milestones

- Milestones represent important dates within the life of a project
  - There may be many milestones within a project
  - Tasks are assigned to a specific milestone



320

### 5.2.3: Milestones

- Create a Classic Report for Milestones
  - Change the template to Timeline
- Create a Form for Milestones
  - Link it to the Report



322

## Resources

### Resource Assignment

- You can already maintain Resources in the administration section
- However you also need to be able to add and remove Resources to and from a project
  - If a resource has been assigned tasks in a project, they should not be able to be removed until the tasks are reassigned
- In this exercise, you'll:
  - Create a report to show the assigned resources along with the number of assigned tasks
  - Create a form from scratch (not using a wizard) to add or remove resources from the project.



323

### Icon Buttons

- You also need a button to call the assignment form
  - Certain button types can show images instead of text
    - The Icon Button template can use Font Awesome glyphs
  - Icon buttons
    - Take less space
    - Can be more recognizable
    - Can be used across multiple languages without translation



324

## Creating Forms from Scratch

- Most times it is best to create forms using the wizards
- When specialized requirements dictate, you may need to create forms from scratch
  - Query the data from the database
  - Present the data to the user
  - Validate user input
  - Process any changes
- Caution: When you build forms from scratch you must take control of **all Processing & Lost Update Detection**
  - This can be quite complex and involved
  - Where ever possible, use built-in logic



325

## Requirements

327

## 5.2.4: Resource Assignments

- Create a Classic Report to display Resource Assignments
  - Change the report template to Cards
- Add an icon button to allow new assignments
- Create the Form to assign resources to the project



## Tree

- The **Tree** region in APEX displays data in a tree
  - Just as the name implies
- Specify a table or enter SQL
  - Then use the declarative options to specify how the tree
- **CONNECT BY** relationship must be in the data
  - ID & PARENT\_ID
  - Can select which column is which in the attributes
- Can be further modified with JavaScript API calls



328

## 5.2.5: Requirements

- Create a Tree region to display requirements
- Add the ability to edit a requirement

## Tasks



330

## Tasks

- Lastly, we'll use a simple Report & Form to manage project tasks

## 5.2.6: Tasks

- Create a report & form to manage tasks
- Use the drag & drop interface to polish the form



331

# Calendars, Charts & Faceted Search

333

# Calendars

334

## Calendars

- Calendars are a type of **APEX report**
  - Data is **rendered on a calendar** vs. traditional row/column format
  - Enables to see your data in a different way
  - Spot trends such as events that occur in the beginning of the week or month
- One requirement:
  - Table or View used must have at least one **DATE** column



335

## Calendars

- To help visualize the Tasks and their due dates you'll use a calendar to plot them
  - You'll give the end user the ability to re-assign the due dates by using the calendar's Drag & Drop Features
  - You'll also need to allow the end user to filter the tasks by Project



336

## 6.1 Calendars

- Create a Tasks calendar showing the due date of each task
  - Modify the calendar to include Supplemental Information in the hover-hint
  - Link each even to the Task edit page
  - Create a filter that uses Session State to limit the Project shown in the calendar



## Charts

338

## Charts

- APEX provides a number of **built-in chart** options
- In APEX 5.1, charts are based on the Oracle JET chart library
  - Open Source Javascript library that Oracle is using in SaaS products and elsewhere
  - Older charts can easily be upgraded to Oracle JET versions



339

## Chart Types

- Area
- Bar
- Bubble
- Combination
- Dial Gauge
- Donut
- Funnel
- Line
- Line with Area
- Pie
- Polar
- Radar
- Range
- Scatter
- Stock



340

## Using Charts to Create a Dashboard

- Charts are a good way to represent high level data
  - Can give an overview across a large group of data
  - Can provide relevant information quickly without needing to search
- You'll use charts to create a dashboard on Page 1 of your application
  - Project Status of all Projects
  - Projects with Past Due Tasks
- You'll also create a gauge chart on the main project page



341

## Faceted Search

343

## 6.2 Charts

- Create two charts using APEX's built-in charting engine
  - Bar
  - Line with Area
- Create a chart on the Project Details page
  - Dial Gauge



## Faceted Search

- New in APEX 19.2, there is a new report type: **faceted search**
- You've likely seen & used this before
  - List of categories on the left
  - Selecting values narrows result set

USGS Earthquake			
Q: Search...	Or	M: Tl:	
<input checked="" type="checkbox"/> Magnitude	<input type="checkbox"/> Clear	#37427733	1.0 4 weeks ago M 1.0 - 15km ESE of Bakersfield, CA
+ 1.0 (33)		#37427777	1.1 3 weeks ago M 1.1 - 1km ENE of Coarse Junction, CA
+ 1.3 (338)		#37427468	1.0 3 weeks ago M 1.0 - 8km SSW of Modocville, CA
+ 1.3 (338)	<input checked="" type="checkbox"/> Or	#37427988	1.0 3 weeks ago M 1.0 - 8km SSW of Modocville, CA
		#37427987	1.0 3 weeks ago M 1.0 - 8km NE of Caliente, CA
		#37428006	1.0 3 weeks ago M 1.0 - 12km NW of Caliente, CA
		#37428021	1.0 3 weeks ago M 1.2 - 9km ENE of Coarse Junction, CA
		#37428059	1.0 3 weeks ago M 1.0 - 10km ENE of Coarse Junction, CA
		#37428056	1.0 3 weeks ago M 1.0 - 8km ENE of Coarse Junction, CA
		#37428033	1.0 3 weeks ago M 1.1 - 1km SSE of Seaside Valley, CA
		#37428033	1.0 3 weeks ago M 1.1 - 1km SSE of Seaside Valley, CA
		#37428049	1.0 2 weeks ago M 1.2 - 13km WNW of Anza, CA
		#37428049	1.0 2 weeks ago M 1.0 - 22km NW of Ridgecrest, CA
		#37428049	1.0 2 weeks ago M 1.2 - 10km N of Pueblo, B.C., MX
		#37428257	1.0 2 weeks ago M 1.1 - 8km NE of Heber, CA
		#37428257	1.0 2 weeks ago M 1.1 - 8km NE of Heber, CA
		#37428507	1.0 10 days ago M 1.0 - 8km NW of Seaside Valley, CA
		#37428505	1.0 11 days ago M 1.0 - 7km NW of Anza, CA
		#37428603	1.0 10 days ago M 1.0 - 12km SSE of Ridgecrest, CA



344

## Faceted Search

- Create a new **Page** and select **Reports > Faceted Search**
- Select your **table/view** for the data source
- Select **columns** to include as **facets**
  - If you can't select a column, you can easily add it later



345

## Security

347

## 6.3 Faceted Search

- Create a new page w/a Faceted Search region
- Modify the Faceted Search report and add a new facet



## Security

- **Security is Hard**
  - If it's **easy**, then it's **wrong**
- It should not be an afterthought; you need to design your application with security in mind from day 1
  - Failure to do so will likely produce an insecure application or at least one that is hard to manage from a security point of view



348

## APEX Security

### • User Access

- Conditions
- Authentication
- Authorization
- Read-Only

### • Data Access

- Session State Protection
- Virtual Private Database
- Oracle Label Security
- Views



349

## How Secure is Secure Enough?

- Modest House
- Good Neighborhood
- Small, Rural Town
- Low to No Crime
- Before Break-In:
  - Left Front Door Open
- After Break-In:
  - Keep shotgun by the bed



351

## How Secure is Secure Enough?

- It depends on:
  - **What** you're protecting
  - **Who** you are protecting it from
  - The **likelihood** of someone wanting to steal what you are protecting
  - The **repercussions** you would face if someone were to successfully steal it
- Unfortunately, adding security is **typically event-driven**
  - Best example: Install a car alarm after the car is broken into



350

## How Secure is Secure Enough?

- Not so Modest House
- Good Neighborhood
- Big City
- Some Crime
- Before 9/11:
  - Occasionally see SWAT agents hiding in the bushes
- After 9/11:
  - They see you first



352

# Authentication

353

## Authentication

- APEX contains a built-in Authentication engine
  - Very extensible & configurable
- Can easily switch between different Authentication Schemes
  - But only at design time
- Out of the box, APEX supports a number of different Authentication Themes



354

# Authentication

## • LDAP

- Use LDAP credentials
- Any LDAP-compliant server should work
  - Oracle Internet Directory, Active Directory, etc.

## • Oracle Application Server Single Sign On

- Integrate with the Oracle SSO server
- Authenticate once to SSO and be authenticated to all APEX applications

## • Database Accounts

- Provide a database schema username and password



355

## Authentication

## • Custom

- Useful for Internet-facing applications where you have no control over who is a user
  - Also used for obscure LDAP servers which do not work with APEX's LDAP scheme
- In < 4.1 you can clone the Sample Application's functions and use them as a baseline for your applications
- Authentication Function must have the following signature and return a **BOOLEAN**:

```
p_username varchar2(255)  
p_password varchar2(255)
```



356

## Authentication

- **APEX Credentials**

- Not very scalable, as users are mapped to an APEX workspace
- Useful for small, one-off situations

- **Open Door**

- Use Open Door to simulate logging on as a specific user when testing your application
- Reset to actual Authentication Scheme before deploying to production

- **No Authentication**

- Allow any user to access your application without providing any credentials



357

## Authentication

- **Social Sign On**

- Re-use credentials from popular social media sites
- Google, Facebook, LinkedIn, Office 365

- **HTTP Header Variable**

- User provided by variable in the HTTP Header call to APEX



358

### 7.1 Authentication

- Create an Authentication Scheme that uses the **RESOURCES** table as well as the functions loaded in section 2.2



### Conditional Security

360

## Conditional Security

- Some APEX conditions can aid in implementing security
  - **User is Authenticated**
    - Any user at all has successfully authenticated
  - **User is the Public User**
    - User has not successfully authenticated



361

## Conditional Security

- Don't fall victim to **Security by Obscurity!**
  - Hiding or obstructing something does not necessarily make it secure
  - In the APEX world:
    - Just because you remove a link on page 1 that goes to page 2 with a condition doesn't mean that a malicious yet savvy user cannot directly navigate to page 2...



362

## 7.2 Conditional Security

- Set the condition to User is Authenticated for the Navigation Menu Items that are only accessible to authenticated users



## Read-Only Items

364

## Read Only Items

- You may want to prevent certain users from editing data based on a business rule
- APEX Items can be set to Read Only
  - Item renders but cannot be edited
  - Similar to, yet independent of Conditions
  - Can be applied to Entire Regions or Individual Items
- One Drawback:
  - If you want to use Authorization Schemes to determine whether an item is read only or not, you will have to copy & paste the SQL from the Authorization Scheme



365

## Read Only Items

- There is a way to use Authorization Schemes for Read Only logic
  - **APEX\_AUTHORIZATION.IS\_AUTHORIZED**
- Checks the value of the named Authorization scheme for the current user
  - Name must match exactly, including case
  - Returns TRUE or FALSE based on the Authorization Scheme
  - May need to Negate the return value

```
RETURN NOT APEX_AUTHORIZATION.IS_AUTHORIZED
(
  p_authorization_name => 'Contribution Rights'
);
```



366

### 7.3 Read Only Items

- Make use of the Read Only attribute and the aforementioned APEX API to make sure view only users are only able to view data.



## Data Upload

368

## Data Upload

- New wizard that can be called from the Create App page or SQL Workshop
- Ability to **drag & drop files** for **multiple popular data formats** now possible
  - CSV, TXT, XLSX, XML or JSON supported
- Wizard will auto-detect the file format, settings, data types & format masks when & where possible



369

## Data Upload

- File size is unlimited
  - Can be uploaded via background process if file is large
- Wizard includes:
  - Preview
  - Error Table
  - Ability to select sheet from XLSX when multiple are present



370

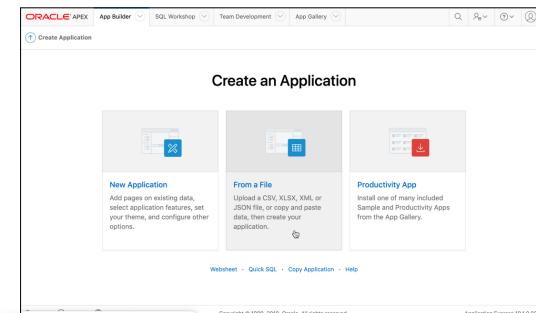
## Data Upload

- New Public API - **APEX\_DATA\_PARSER** - allows **developers to implement their own data loading process**
  - Support for CSV, XLSX, XML & JSON
- Parse implemented as a table function
  - Detects headers & displays columns
- Create custom lookup tables or use SQL and/or PL/SQL to process & refine data
- Updated Sample Data Loading packaged application



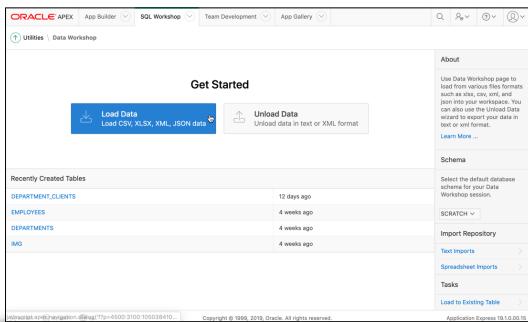
371

## Create an App from a File



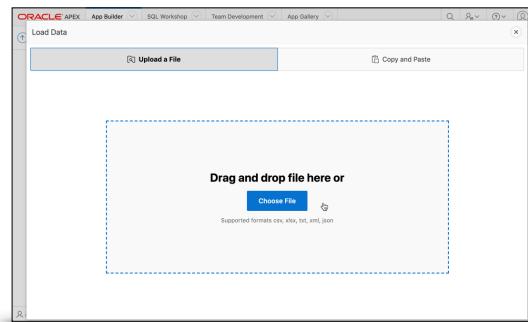
372

## Upload a File from SQL Workshop



373

## Drag & Drop File



374

## 8.1: Data Upload

- Create a page that allows data upload via the **APEX\_DATA\_PARSER API**

## Themes

376

# Universal Theme

377

## The Universal Theme

- Starting with APEX 5, managing the user interface and associated styles has been migrated to a declarative framework called the **Universal Theme**
  - This means no more messy CSS - inline or otherwise
  - Developers can now select options from menus to change the look & feel of their applications without having to know the first thing about CSS or even design principles
  - While there are limits to what is possible without CSS, most applications will not even come close to them



378



## More Than a Theme

- The **Universal Theme** is much, much more than a new theme
- It's really more of a "declarative user interface"
  - Specify options vs. writing code
  - Very extensible and can meet most UI needs
  - Like APEX, can be extended when the bound of the declarative framework are reached



379

## Templates

- The Universal Theme is made up of **templates**, just like other themes
- However, all Universal Theme templates are **locked and cannot be edited**
  - This is so that Oracle can guarantee that when patches and new version of APEX are deployed, the template code is unchanged
  - Unwise to make changes to them - similar to how you shouldn't change any of the Oracle provided CSS or JS files
- Worst case, if you need to make a change, **clone it and add change the cloned copy**
  - Keeping in mind that any changes will have to be applied manually



380

## Design Goals

- **Responsive**
  - Work across multiple devices - big and small
- **Versatile**
  - Contains all building blocks necessary to build enterprise-grade web applications
- **Customizable**
  - Designed to have the look & feel customized via CSS



381

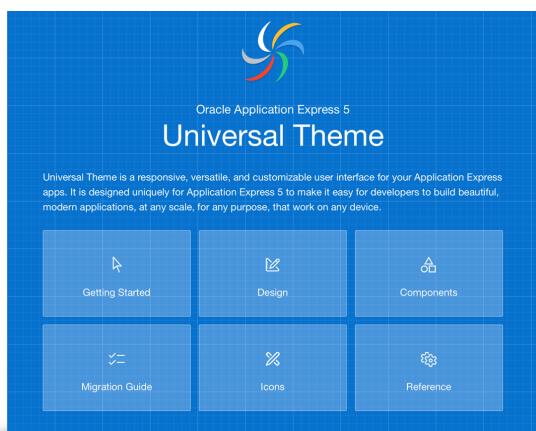
## Universal Theme Sample Application

- A good place to get started with the Universal Theme is the **Universal Theme Sample Application**
  - Install **locally in your workspace** via Packaged Applications
  - Run from here: <https://apex.oracle.com/ut>
    - This link will reflect the most current version of APEX
- Also acts as a **good on-line reference** for the Universal Theme



382

## Universal Theme Sample Application



383

## Font APEX

- Font APEX is an open source font library created exclusively for APEX
  - Used throughout the Universal Theme
  - Variation of Font Awesome
  - Can be used in any project - Oracle or otherwise
- Can choose between Font Awesome and Font APEX per application
  - Defined in Theme Settings
- Universal Theme Sample Application contains a searchable page of the full icon library
  - Also includes a tool to customize each icon



384

## 9.1 Universal Theme

- Use the Theme Roller to customize the style of your application
  - When finished, save the changes



386

## Deployment

### Deploying an Application

- An application is made up of more than just the APEX application
  - APEX Application Export
  - APEX Based Files
  - Database Objects
  - External Files
- You need to understand how all of these fit together



387

## Exporting

388

## Application Export

- Exports are created from within the Application Builder
  - Application Home > Export/Import > Export
- APEX will generate a SQL script which contains API calls to re-instantiate our application
  - Either in the same or another workspace
- But it will NOT include
  - DDL or DML!
  - External Files
  - APEX Based Workspace Files
- You have to handle these separately



389

## Application Export Options

- **File Format**
  - UNIX vs. DOS
  - Designates how the CR/LFs within the file are generated
- **Owner Override**
  - Associates a different Owner or “Parse As” schema with the application
  - Can easily be changed when importing, so not a necessary step
- **Build Status Override**
  - Run and Build Application
    - Developers can modify the application
  - Run Application Only
    - Application is read-only; only an APEX administrator can change the status



390

## Application Export Options

- **Debugging**
  - Set to No to disable Debug mode
  - Necessary for Production deployment
- **As Of X Minutes Ago**
  - If Flashback is enabled, this will roll the database back X minutes and perform the export at that point in time
- **Export Supporting Object Definitions**
  - If you have a Packaged Application, this option will determine whether or not the supporting object scripts are included in your export file
- **Export Public Interactive Reports**
  - Set to Yes to include any saved Interactive Reports marked as PUBLIC into the export file



391

## Application Export Options

- **Export Private Interactive Reports**
  - Set to Yes to include any saved Interactive Reports marked as PRIVATE into the export file
- **Export Interactive Report Subscriptions**
  - Set to Yes to include subscription information for the Interactive Reports being exported
- **Export Developer Comments**
  - Set to Yes to include any Developer Comments in the export file
- **Export Translations**
  - Set to Yes to include the translation mappings and all text from the translation repository.
- **Export with Original IDs**
  - Designates whether the export file should use current component IDs or the IDs as of the last import



392

## APEX Based Files

- APEX provides the ability to upload files into the metadata repository
  - Static **Application** Files
    - Tied to a specific application in the workspace using **#APP\_IMAGES#** prefix
    - ARE INCLUDED with an application export
  - Static **Workspace** Files
    - Available to all applications in the workspace using **#WORKSPACE\_IMAGES#** prefix
    - ARE NOT INCLUDED with an application export
- Can be any type of file
  - Cascading Style Sheets
  - Images
  - Static Files



393

## APEX Based Files

- **Application** and **Workspace** static files can be exported from their respective management screens
  - Individually using the **Download** link
  - As a group by clicking the **Download as Zip** button
- Workspace files will need to be migrated to the target workspace **separately** from the Application Export



394

## Database Objects

- You need to consider
  - **DDL** - All objects that your application needs to run
    - Tables, Views, Procedures, Packages, Functions, Triggers, Sequences
  - **DML** - Any Seed data required for the application
    - Statuses, Lookups, Dynamic Lists of Value data
- Normally everything you need will be in your “Parse As” schema
  - Don’t forget any grants or synonyms pointing to data or objects outside your “Parse As” schema



395

## Database Objects

- You also need to consider the differences between deploying a **Brand New** application and upgrading and **Existing** one
  - New applications likely don’t have any existing database objects in the target system
  - With existing applications, you need to worry about upgrading objects and migrating and preserving user data



396

## Database Objects

- Database objects for **New Applications**

- If there are no pre-existing database object, simply generate the DDL and DML for the required objects
  - APEX Generate DDL Wizard
  - Oracle SQL Developer's Database Export Wizard
  - DBMS\_SQL database package
  - TOAD For Oracle
  - Other 3rd Party Tools
- You may also use a feature of Supporting Objects to synchronize database objects
  - This is covered a bit later



397

## Database Objects

- Database objects for **Existing Applications**

- Need to produce scripts that **upgrade** the existing database structure to new structure
- There are a number of tools that can produce upgrade scripts
  - Oracle SQL Developer
  - Oracle Enterprise Manager (must license Change Management Pack)
  - RedGate Schema Compare for Oracle (Windows Only)
  - Dell's TOAD for Oracle
- Scripts generated will do the work necessary to alter the table structure but **will not take into consideration any data migration that may need to be done**
  - Unfortunately this is a **manual process** and needs to be heavily tested.



398

## Database Objects

- Install/Upgrade Scripts will need to be run in the appropriate “Parse As” Schema
- Should be tested to make sure
  - The scripts run without error
  - Any dependencies are accounted for
  - No data is lost or mangled during upgrade process
- May need help from the DBA Team
  - Take into account any change management processes your company may have



399

## External Files

- Files that reside outside of APEX or the database
  - Shared **CSS** Files
  - Shared **Javascript** Files
  - **Image** files used in Themes
  - System Help **Documents**
- Most likely reside on the Web Server
  - Under the **docroot** for the specific domain



400

## External Files

- Need to keep track of
  - Which files your application uses
  - Which files have changes from previous versions
  - Whether these files are used by any other systems
- Make sure your changes don't affect other systems inadvertently
- Migration of these files may require help from Web Server Admins



401

## Importing

## Importing an Application

- Once you export an Application, it's just as easy to import it into a new instance
  - Application Home > Import/Export > Import
- When importing, you can specify:
  - **Parsing Schema**
    - Limited to those associated with your workspace
  - **Build Status**
    - Run Only vs. Run & Build
  - **Application ID**
    - Reuse, Specify or Auto-Assign



403

## Importing an Application

- If there are supporting objects, APEX will ask if you wish to install them
  - Choosing **YES** will run through the suite of supporting objects scripts
    - Pre requisites, Install, Upgrade, etc.
  - Choosing **NO** will skip the installation of supporting objects
    - Application is not likely to work unless the required objects already exists in the target Parse As schema



404

## Deployment

- As easy as giving out the URL:
  - <http://servername.com/apex/f?p=100>
- You do not need anything past the Application ID
- APEX will resolve the home page and create a new Session ID automatically



405

## Deployment

- Alternatively, you can create a more “friendly” URL
  - <http://servername.com/survey>
- Easier for users to remember
- Easier for you to market
  - Even if it’s not a sales-driven application
- Will most likely need to solicit the help of the web system administrator
  - They can create the redirection for you



406

## Deployment

- When promoting to Production, be sure to set:
  - **Build Status:** Run Only
- This will:
  - Remove the Application from the Builder – even for Developers!
    - APEX Administrator can change the status back to Run & Build, if needed



407

## Deployment

- APEX is Forward Compatible
  - 3.2 Applications can be Imported into 5.0+
  - But **not the other way around**
- Best to **keep up with the current** release of APEX
  - The **further behind you get**, the **more potential issues** you may have when upgrading
  - There have been minor issues when going from 3.2 to 5.0, for instance
    - Which is really: 3.2 > 4.0 > 4.1 > 4.2 > 5.0



408

## 10.1 Exporting & Importing Applications

- 10.1.1 Export your application
- 10.1.2 Import your application



VISCOSITY  
NORTH AMERICA