

**COLLEGE OF
INFORMATION**



**COMPUTING AND
SCIENCES(CoCIS)**

**COURSE
Bachelors of Information Systems and
Technology**

Recess
Term

Course
Unit
**CS 1301 Practical Skills Development (Oracle
Database)**

Lecturer
Dr. Brian Muchake

Group Members

NAME	REG. NUMBER	STUDENT NUMBER
ODOCH NELSON OCHAN	23/U/16508/EVE	2300716508
MUWANGUZI ALVIN KIGGUNDU	23/U/12372/PS	2300712372
ISSA SULAIMAN MAYOMBWE	23/U/25639/EVE	2300725639
NASSAKA BONITA POLLINE	23/U/15483/PS	2300715483
MIREMBE GRACE	23/U/11316/PS	2300711316
BWIRE MELVIN MASINDE	23/U/07831/PS	2300707831
RODNEY KAVUMA	18/U/21052/EVE	1800721052
MASEREKA BRIGHT	23/U/0738	2300700738
TUMUKUNDE SIMON PETER	23/U/18071/PS	2300718071
KASEMIIRE BRIDGET.	23/U/27044	2300727044
KUTENTSA HAPPY	23/U/0654	2300700654

INTRODUCTION TO ORACLE APEX

Oracle Application Express (APEX) is a low-code development platform that enables you to build scalable, secure enterprise apps, with world class features, that can be deployed anywhere. It's also formerly called HTML DB.

Apex is written using PL/SQL and works within a web browser completely. Oracle Apex is an IDE and runtime environment. It is used to write web applications. With APEX IDE, one can create forms, reports, and charts.

It is a very powerful development tool used to build web-based applications. The tool itself contains a schema on a website with multiple tables, views and PL/SQL Code. The techniques used with this tool are PL/SQL, HTML, CSS, and JavaScript.

HERE ARE THE DETAILED STEPS TO INSTALL ORACLE APEX:

Step 1: Download the Oracle Apex zip file

- Go to the Oracle Apex website and download the latest version of Oracle Apex.
- Choose the correct version (e.g., Windows or Linux) and download the zip file.

Step 2: Install Oracle Apex

- Extract the zip file to a directory (e.g., C:\apex or /home/apex).
- Run the installation script (apex_install.sql) using SQL*Plus or Oracle SQL Developer.
- Follow the prompts to install Oracle Apex.

Step 3: Create or update the instance administration account

- Run the script (apex_adm_pkg.sql) to create or update the instance administration account.
- Set the password for the APEX_ADMIN account.

Step 4: Restart processes

- Restart all processes that you stopped before beginning the installation (e.g., Oracle Database, listener).

Step 5: Configure the APEX_PUBLIC_USER account

- Run the script (apex_pub_user.sql) to configure the APEX_PUBLIC_USER account.
- Set the password for the APEX_PUBLIC_USER account.

Step 6: Configure RESTful Services

- Run the configuration script (apex_rest_config.sql) to configure RESTful Services.
- Follow the prompts to configure RESTful Services.

Step 7: Verify the installation

- Access Oracle Apex using a web browser (e.g., <http://localhost:8080/apex>).
- Log in with the APEX_ADMIN account to verify the installation.

ORACLE APEX WORKSPACE

Oracle Apex Workspace is a web-based development environment for building and deploying web applications. It provides a range of features and tools for designing, building and deploying applications, including;

SQL Workshop: A tool for writing and executing SQL queries.

Data Editor: A tool used for editing and managing data in your application.

Team Development: A feature for collaborating with team members on application development.

How to Create Workspace User in Oracle Apex

Steps

1. Log in: Access Oracle APEX with admin credentials.
2. Navigate: Go to "Administration" and select your workspace.
3. Create User: Click "Manage Developers and Users" or similar, then "Create User."
4. Enter Details: Provide username, password, email, and assign roles.
5. Save: Save the user details to confirm creation.
6. Access: The user can now log in and work within the assigned Oracle APEX workspace

How to Create & Add Schema in Apex Workspace Oracle Apex

Steps

1. Log in: Access Oracle APEX and log in with admin credentials.
2. Navigate to SQL Workshop: Go to "SQL Workshop" in the APEX menu.
3. Create Schema: Use "Object Browser" to create a new schema/user.
4. Enter Details: Specify schema name, password, and optionally tablespace settings.
5. Grant Privileges: Assign necessary privileges like create table, procedure, etc.
6. Add to Workspace: Associate the schema with your APEX workspace in "Manage Workspace."
7. Use in Applications: Develop database objects and applications within the schema

How to Create Application in Oracle Apex

Steps

1. Log in to Oracle APEX: Access Oracle APEX using your web browser and log in with your credentials.
2. Navigate to App Builder: Click on "App Builder" in the APEX navigation menu.
3. Create a New Application: Click on "Create" > "New Application" to start creating a new application.
4. Define Application Attributes: Enter basic information such as the application name, description, and primary schema.
5. Choose Application Type: Select the type of application you want to create (e.g., Desktop, Mobile, or Universal Theme).
6. Create Pages: Begin adding pages to your application. Pages can include forms, reports, charts, dashboards, and more.
7. Design Pages: Customize each page by adding components and defining their properties (e.g., SQL queries for reports, form attributes).
8. Define Navigation: Set up navigation menus and navigation bars to organize and access different pages within your application.
9. Configure Security: Implement security features such as authentication schemes and authorization rules to control access to your application.
10. Test and Refine: Preview and test your application to ensure functionality and usability. Make adjustments as needed.

How to Create Interactive Report in Oracle Apex

Steps

1. Log in to Oracle APEX: Access Oracle APEX using your web browser and log in with your credentials.
2. Navigate to Your Application: Open the APEX application where you want to create the Interactive Report.
3. Create a New Page: Click on "Create" > "Page" to start creating a new page.
4. Select Report as Page Type: Choose "Report" as the page type.
5. Choose Interactive Report: On the Report Attributes page, select "Interactive Report" as the report type. Interactive Reports offer advanced features for data interaction.

6. Define SQL Query: Write or paste your SQL query to retrieve the data for the Interactive Report. Ensure the query includes all necessary columns and conditions.
7. Configure Report Attributes: Set report name, SQL query, report template, and other attributes such as sorting, filtering, and pagination options.
8. Customize Appearance and Behavior: Customize the appearance and behavior of the Interactive Report using APEX themes, templates, and CSS as needed.
9. Save and Run: Save your Interactive Report configuration and run

How to Create Listview Report in Oracle Apex

Steps to Create a ListView Report in Oracle APEX

1. Navigate to Your Application
2. Log in to Oracle APEX and open your application in the App Builder.
3. Create a New Page: Click on "Create" > "Page" to start creating a new page for your ListView report.
4. Select "Report" as the page type.
5. Choose ListView Report: On the Report Attributes page, select "ListView" as the report type. ListView reports are ideal for displaying data in a simple list format without grid lines.
6. Define SQL Query: Write or paste your SQL query to retrieve the data you want to display in the ListView report. Ensure the query retrieves all necessary columns and filters.
7. Configure Report Attributes: Configure other report attributes such as report name, SQL query, and report template as per your application's requirements.
8. Set Display Options: Customize display options such as sorting, grouping, and pagination settings based on your data and user preferences.
9. Save and Run: Save your ListView report configuration. Run the page to preview and test the

How to Create Classic Report in Oracle Apex

Steps

1. Navigate to Your Application:
2. Log in to Oracle APEX and open your application.
3. Create a New Page:
4. Click on "Create" > "Page".
5. Select Report Type: Choose "Report" as the page type.
6. Choose Classic Report: Select "Classic Report" as the report type.
7. Define SQL Query: Write or paste your SQL query to retrieve the data.
8. Configure Report Attributes: Set report name, SQL query, template, and display options (sorting, pagination).
9. Save and Run

How to Create Toggle And Reflow Reports in Apex

Toggle Reports:

- Create a Classic Report.
- Define SQL query for data retrieval.
- Enable "Column Heading" property to toggle column visibility.
- Save and run to test toggling functionality.

Reflow Reports:

- Create an Interactive Report for responsiveness.
- Define SQL query for data retrieval.
- Choose a reflow-compatible template (e.g., "IR - Default").
- Save and run to ensure the report stacks columns vertically on smaller screens

How to Add Icon on Navigation menu in Oracle Apex

Steps

1. Access Navigation Menu: Go to "App Builder" > Your Application > "Navigation Menu" (under Shared Components).
2. Edit Menu Entries: Select the menu entry you want to add an icon to.
3. Add Icon: Enter the icon name or CSS class in the "Icon" attribute. Use Font Awesome icons like fa fa-home or specify custom CSS classes.
4. Save Changes: Save the menu entry to apply the icon.
5. Preview and Adjust: Preview your application to see the icon in the navigation menu. Adjust size and position as needed using CSS

How to Create Master-Detail Form Using Wizard in Oracle Apex

Steps to Create a Master-Detail Form Using Wizard in Oracle APEX:

1. Log in to Oracle APEX:
2. Create a New Application: Click on "App Builder" to start creating a new application.
3. Create a New Page: Within your application, create a new page where you want to add the master-detail form.
4. Add a Form Based on a Table: Use the Create Page wizard to add a form based on your master table:
5. Click on "Create" > "Form" > "Form on a Table with Report" option.
6. Select the table (master table) from your schema that you want to use for the master form.
7. Follow the wizard steps to configure the form attributes, including choosing the columns to display and edit.
8. Add a Detail Form: After creating the master form, proceed to add a detail form:
9. Click on "Create" > "Form" > "Form on a Table with Report" again.
10. Choose the detail table from your schema that is related to the master table.
11. Configure the form attributes similarly to the master form, ensuring to select the appropriate columns and relationships.
12. Define Master-Detail Relationship: Specify the relationship between the master and detail forms:
13. Configure Form Layout and Appearance: Test and Refine

How to Set Last Login in Oracle Apex

Steps involved

1. Create application item -----Global item
2. Add this query in login process
"select to_char(sysdate, 'DD-MM-YYYY HH:MI:SS') from Dual"
3. Store this SELECT Query in Global Item (:ITEM)
4. Set Last login in Navigation Bar
5. Run application and check Last Login

How to Change Login-Page Logo in Oracle Apex

Steps

1. Import logo in application
2. Run application and right click login page and inspect element
3. Copy the "class" of logo
4. Remove previous icon in "login page"
5. Go to inline CSS at Loginpage and type
background-image: url(#APP_IMAGES#iamgename);
background-size: cover;

How to Add Search Navigation Menu in Apex

The steps

1. Import "search plugin" in Apex Application
2. Go to global page
3. Create region and add page item
4. Set Type = searchNavigationMenu[Plug-in]
5. Run application

How to Create Dynamic List of Values (LOV) in Oracle Apex

The steps involved

1. Go to application
2. Shared components-----→Lists if values
3. Create "Dynamic" List of value (using database table)
4. List use in Oracle Apex form.

How to Make Parameterized Report in Oracle Apex

Steps

1. Create Blank Page.
2. Create Region for "Parameters" & Set Name.
3. Add two Page Items.
4. Add two Buttons "Submit" and "AllData".
5. Create Another Region for "Classic Report" OR Another Reports.
6. Set Report Query with Parameters.
7. Add Dynamic Action to "AllData" Button:
 "Clear" Action
 "Submit Page" Action
8. Now Run Application and Search Data with Parameters.

How to Make Custom Role based Authentication in Oracle Apex

1. Create "auth_role" Table: "USER_KEY" field on Users table is used for holding the status of existing user, whether the user is active or not.
2. Insert Data in Users Table
3. Go to Shared Components -> Authentication Schema
4. Create new Authentication Schema
 schema type = Custom
5. Type PL/SQL Code

```
FUNCTION custom_auth (p_username IN VARCHAR2, p_password IN
VARCHAR2)
RETURN BOOLEAN
AS
my_user NUMBER := 0;
BEGIN
SELECT 1 INTO my_user FROM auth_rol
```

How to Set Role based Access Control in Oracle Apex

We have two users ("ADMIN" & "EMPLOYEE")

Both Users Access All Menus

So we set the "EMPLOYEE" user to have only access to 'Employees Menu' and not access to 'Add Employee Information'

Steps

1. Go to Shared Components -> Navigation Menu
2. Click Parameters Report Menu
3. Go to Conditions & choose PL/SQL Expression
 Type v('APP_USER') <> 'EMPLOYEE'
4. Set this condition to 'AddEmployee' Button
5. Go to Employees Page & click 'AddEmployee' Button
6. Go to Server-side Condition -> choose PL/SQL Expression
 Again type v('APP_USER') <> 'EMPLOYEE'

How to Make User Registration Form with (Pic) in Oracle Apex

Steps

1. Create a table of your choice for example "user profile"
2.

```
CREATE TABLE USER_PROFILE  
  (USER_ID NUMBER(20,0),  
   USER_NAME VARCHAR2(50),  
   ADDRESS VARCHAR2(100),  
   PHONE VARCHAR2(15),  
   EMAIL VARCHAR(20),  
   ROLE VARCHAR(50),  
   USER_PHOTO BLOB,  
   PRIMARY KEY (USER_ID) ENABLE  
  );
```
3. Create form (Report with form) in application
4. Add "user_profile" Table and set Primary key
5. Check report query
6. Set "user_photo" type

How to Set Auto Increment in Oracle Apex Form

1. Create table for example
2. Create table student

```
( student_id number(5)  
  Student_name varchar2(20)  
  Father_name varchar2(20)  
 );
```
3. Create sequence for student table

```
CREATE SEQUENCE "seq_name"  
  MINVALUE 1  
  MAXVALUE 10  
  INCREMENT BY 1  
  START WITH 3  
  CACHE 20
```
4. Use sequence to "Student_id" column for Auto increment

How to Make Slider in Oracle Apex

1. Import images in application
2. Create region then type = "static content"
3. Position = "content body"
4. Choose template = carousel container
5. Set template option
6. Create sub region then type = "static content"
Source = ``
7. Set Parent region & Template = "Standard"
8. Set Template option
9. Create sub region and same work to previous region.

How to Customized Application (Name & icon) in Oracle Apex

1. Customize the application icon
2. Open application>>>>>static application file
3. Import png icon in application
4. Go to share component>application definition attribute
5. Change "Application Name" & change icon"
6. Customize Application name in "Header section"
7. Go to share component>user Interface attributes
8. Got to logo section and change the text with CSS code
9. How to Set Favicon Icon in Oracle Apex

Import icon in Application

1. Go to Application → Shared Components
2. Go to User Interface Attributes → Favicon HTML
3. Set this code in Favicon Box

```
<link rel="shortcut icon" href="#APP_IMAGES#">
<link rel="icon" sizes="16x16" href="#APP_IMAGES#">
<link rel="icon" sizes="32x32" href="#APP_IMAGES#">
<link rel="apple-touch-icon" sizes="180x180" href="#APP_IMAGES#">
```

How to Set Background Image in Oracle Apex

1. Change login page background
2. Import image/wallpaper in apex application
3. Run the login page
4. Right click to inspect
5. Find the class/id for changing background

How to Set Show/Hide Password Mask in Apex

1. Go to login page
2. Set the TYPE "password" for password Item
3. Go to "Post Text Section" and add this code

```
<i id="pass-status" class="fa fa-eye field-icon" aria-hidden="true"
onClick="viewPassword()"></i>
Add this function in "GLOBAL VARIABLE DECLARATION SECTION"
```

```
function viewPassword() {
var passwordInput = document.getElementById('ITEM_NAME');
var passStatus = document.getElementById('pass-status');

if (passwordInput.type == 'password') {
passwordInput.type = 'text';
passStatus.className = 'fa fa-eye-slash field-icon';
} else {
passwordInput.type = 'password';
passStatus.className = 'fa fa-eye field-icon';
}
};
```

How to Add Select All / Un-Select All option in Oracle Apex

1. Create Interactive Report for any table
2. Add this query in query table

```
apex_item.hidden(01, table_id) || apex_item.CHECKBOX2(02, table_id) CHECK1
Set static ID in interactive report such as "myreport"
```

3. Go to CHECK1 column set property

```
Type = plaintext
Heading = <input type="checkbox" id="selectunselectall" title="Selection">
Disable all function switches
In the page definition, add a Dynamic Action:
```

```
Event = Change
Selection Type = jQuery Selector
jQuery Selector = #selectunselectall
```


Event Scope = Dynamic
Static Container (jQuery Selector) = #myreport
True Action = Execute JavaScript Code

```
if ($('#selectunselectall').is(':checked')) {  
    $('input[type=checkbox][name=f02]').attr('checked', true);  
} else {  
    $('input[type=checkbox][name=f02]').attr('checked', false);  
}
```

How to Make Card List & link with Pages in Oracle Apex

4. Create 3 pages in application
5. Create 3 different table reports in the 3 pages
6. Go to share components/navigation box
7. Click list and create static list
8. Add pages link for each list
9. Go to home page and create region
10. Set type list and select list
11. Go to attribute and set list template

How to add delete option in Oracle Apex Report

Steps

1. Create report in application with any table
2. Add “Null” column in report query
3. Create a blank page for “Delete transaction”
Set page mode = “Model Dialog”
4. Create region in the “Delete Transaction” page
Set type = static content
5. Set text in source area such as “are you sure you want to delete”
Set template “Blank with Attributes (No Grid)”
Add page item in “Delete Transaction” region
Set type = Hidden
6. Add 2 buttons in the “Delete Transaction region” (“cancel”, “Ok”)
7. Set action “defined by dynamic action” in “cancel” Button

How to Add Edit option in Oracle Apex Report

1. Using the last departments Report in application
2. Add “Null” for Edit option column in report query
3. Create blank page for form & set page mode = “modal Dialog”
4. Add region in the form page and set type = form
5. Form table = Departments
6. Add 2 buttons in region form (“cancel”, “save”)
7. Go to save button and set properties
Behaviour
Action = submit page
Database action = SQL UPDATE

Server side condition:
Type = item is not null
Item = page_ID (primary_key)

How to Add Checked icon in Login Page Oracle Apex

1. Go to login page
2. Set CSS classes to username and password
3. Run application
4. Enable value required option “yes” on both fields
5. Now paste this code on username and password “post text” box

How to Change Value Color Using HTML Expression in Oracle Apex

1. Login
2. Use any table to create a report
3. Add case statement for change column value color in query
4. Add code in html expression

How to Calculate Age Using Date in Oracle Apex Form

Using Existing Table Column (DOB, AGE)

1. Go to DOB Page Item & Set (Change) dynamic action

Event: Change
Selection Type: Item(s)
Item(s): DOB_PAGE_ITEM
True Action: Set Value
Set Type: PL/SQL Function Body

```
DECLARE
    vSAGE NUMBER;
BEGIN
    vSAGE := TRUNC((TO_NUMBER(SYSDATE -
    TO_DATE(:DOB_PAGE_ITEM)) / 365.25));
    RETURN vSAGE;
END;
Items to Submit: DOB_PAGE_ITEM
Selection Type: AGE_PAGE_ITEM
```

2. Go to AGE Page Item & Set (Lose Focus) dynamic action

Event: Change
Selection Type: Item(s)
Item(s): AGE_PAGE_ITEM
True Action: Set Value
Set Type: PL/SQL Function Body

```
DECLARE
    vSDOB DATE;
BEGIN
    vSDOB := ADD_MONTHS(TRUNC(SYSDATE, 'MM'), -
    TO_NUMBER(:AGE_PAGE_ITEM) * 12);
    RETURN vSDOB;
END;
```

How to Expand /Collapse the navigation sub-menu in Oracle Apex

3. Create new application
4. Open shared components >>user interfaces
5. Go to navigation menu box & Set “Collapse Mode : Icon(Default)” and click only collapse by Default (Default)”
6. Run and check the navigation
7. Create sub-menu and set parent Home page
8. Then set collapse submenu

How to create a Master-Detail Form Using Manual Process in Apex

Steps to Create a Master-Detail Form

1. Using Existing Table for Master Entry ("USER_FORM")
Change Type to "Form" and define the source column.
Declare the primary key column in the insert statement process for returning the ID purpose.

```

DECLARE
  USER_ID USER_FORM.USER_ID%TYPE;
BEGIN
  -- Your insert query here
  RETURNING USER_ID INTO :P2_USER_ID;
END;

```

2. Create New Detail Table or Use Existing Table
Assume creating a new table named "USER_DEPT":

```

CREATE TABLE USER_DEPT
(
  DEPT_ID NUMBER PRIMARY KEY NOT NULL,
  DEPT_NAME VARCHAR2(20),
  LOCATION VARCHAR2(100),
  USER_ID NUMBER -- MASTER TABLE PRIMARY KEY USED AS A
FOREIGN KEY
);

```

3. Set Foreign Key in "USER_DEPT" Table
4. Alter Table to Add Foreign Key Constraint

```

ALTER TABLE USER_DEPT
ADD CONSTRAINT USER_FK FOREIGN KEY (USER_ID)
REFERENCES USER_FORM (USER_ID);

```

5. Set Trigger on Primary Key Column in "USER_DEPT" Table for Auto-Generating ID
Create or replace the trigger:

```

CREATE OR REPLACE TRIGGER TRG_USER_DEPT_PK
BEFORE INSERT ON USER_DEPT
REFERENCING NEW AS NEW OLD AS OLD
FOR EACH ROW
DECLARE
BEGIN
  SELECT NVL(MAX(DEPT_ID), 0) + 10
  INTO :NEW.DEPT_ID
  FROM USER_DEPT;
EXCEPTION
  WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20001, 'Error in
TRG_USER_DEPT_PK');
END;

```

How to use custom cascading style (CSS) in Oracle Apex

Steps to Apply Custom CSS:

1. Create Application or Use Existing App
2. Start with an application in Oracle APEX.
3. Change Header Background Color using CSS
4. Assume the task is to change the header background color using CSS.
5. Right-click on the page and select "Inspect" to open the browser's developer tools.
6. Find the Class or ID Name for Header
7. Identify the class or ID name for the header element.
8. Apply CSS on Header
9. When applying CSS, use class or ID selectors:
Class: Use a dot . before the class name.
ID: Use a hash # before the ID name.

10. Write the CSS code and copy it into a Notepad file.
11. Save File with .css Extension
12. Save the file in any directory with a .css extension.
13. Import CSS File into Application
14. Go to "Static Application Files" in Oracle APEX and import the CSS file.
15. Copy Reference Name
16. Copy the reference name of the CSS file.
17. Add Reference to User Interface

18. Navigate to "User Interfaces" and paste the reference name into the Cascading Style Sheets box.
19. Run and Check

How to Set Auto Page Refresh In Oracle Apex

Add the following HTML code to the HTML header section of your page:

```
<meta http-equiv="refresh" content="3">
```

This will automatically refresh the page every 3 seconds.

Second Way:

Add the following JavaScript code to the Function and Global Variable Declaration section:

```
setTimeout("location.reload(true);", 3000);
```

This will refresh the page after 3 seconds.

Third Way:

Create a dynamic action on page load and paste the following code:

```
$(document).ready(function() {
    setInterval('loadPage()', 3000);
});
```

Add the following code to the Function and Global Variable Declaration section:

```
javascript
Copy code
function loadPage() {
    location.reload();
}
```

How to create link column in Interactive Report in Apex

Steps

1. Create interactive report in any table
2. Create normal table and add page item(hidden)
3. Now to interactive report page & choose any column for making link
Change type :LINK
4. Set items and value
Link test = column name
5. Set link attribute as a button type
6. Write this code
"class =t-Button-hot t:Button-simple t:Button-small"

How to add custom footer with image/logo in oracle apex

Steps

1. First go to 0 (global page)
2. Add region and set position :Footer
3. Template :blank with attributes
4. Import image png/jpg
5. Create page item for year
6. Go to default window and type

SELECT TO_CHAR(SYSDATE, 'YYYY') FROM DUAL

7. Now run

How to add Max/Min record functionality in Apex

Steps

1. Login, Use last department form.
2. Add 2 buttons for Max and Min record
3. Create a process for the Max button

```
Begin
select Max(primarykeydefine)
into :P2_PRIMARY
from tbl;
```

```
exception when others then
null;
end;
```

4. Create a process for the MIN button

```
Begin

select MIN(primarykeydefine)
into :P2_PRIMARY
from tbl;
```

```
exception when others then
null;
end;
```

How to create Star Rating page item in Oracle Apex

1. Add number column (for rating) in department table
2. Add column in department form page
3. Add item in department form and set type = star rating
4. Set number of stars and enable (user default)
5. Run page
6. If you change rating color so disable (user default) and set color
7. After submit form it will store rating value in the department table

How to create Multi-column Popup LOV in Oracle Apex

Steps

1. Go to shared component and create lov (Dynamic)
2. Using for example employees table
Create lov set (display & return column)
3. Open existing lov and scroll down (additional display columns)
4. Add column in LOV
5. Go to page item and set type :POPUP LOV
6. Now run

How to create a cascading list of values in Oracle Apex

1. Create a blank page
2. Add 2 page items (for employee name and department)
3. Write query in department page_item(show department name)
4. Write query in employees page_item(show employees name)
5. Add (clause in employees page_item)
"where department-ID = : department page_item"
Set :department page_item in (cascading list of values option)
6. Run

How to download Report to PDF using plugin in Oracle Apex

Steps

1. Choose the page to work and add Button for PDF
2. Import client side PDF plugin (<https://drive.google.com/file/d/1qz-j...>)
3. Add dynamic action on pdf button
event: click & action: client side pdf plugin
4. Copy "pdf-body" in report CSS class
5. Now run and download pdf

How to download IR(Interactive Reports)/Grid to Excel in Oracle Apex

Steps

1. Download Plugin in system (example from this link <https://glebovpavel.github.io/Descrip...>)
2. Extract file
3. Import script and run in Apex (install_all_packages.sql)
4. Look at database installed in database
5. Install plug in Apex
6. Now add to Report add Button (for download excel)
7. Add Dynamic action in button
Event: click
action: (GPV Interactive Report/Grid to MSEExcel)
8. Now run and download file in Excel

How to show messages using Dynamic Action in Oracle Apex

Steps

1. Create a bank page and add button for alert
2. Add dynamic event click
Action "ALERT" and add message, title
3. Run it

How to create automated Application backup in Oracle Apex

Backups are performed as part of daily maintenance

Steps

1. Go to application and click utilities
2. Go to manage backups
3. Create a button to create an automatic Backup for your current application
4. If you create a backup again, the window will show
5. Then do some modification on the page
6. Create a backup after modifying the page

How to Add Strong Password functionality using Plugin in Oracle Apex

1. Choose and Install Plugin:
2. Select a password strength checker plugin from Oracle Apex Plugin Catalog or a third-party source.
3. Install the plugin in your Apex application.
4. Configure Plugin:
5. Define password complexity rules like minimum length, required characters (uppercase, lowercase, digits, special characters).
6. Customize error messages for validation.
7. Integrate with Password Fields:
8. Replace standard password fields with the plugin's password item on your Apex pages.
9. Set validation rules to enforce strong password criteria.
10. Test and Implement:
11. Validate that the plugin accurately checks password strength and displays appropriate error messages.
12. Ensure it enhances security by guiding users to create strong passwords

How to Encrypt & Decrypt Password in Oracle Apex

1. To encrypt passwords in Oracle Apex, you can use the DBMS_CRYPTO package, which provides cryptographic functions.
2. Encrypting the Password:

```

DECLARE
    l_password VARCHAR2(4000) := 'your_password_here';
    l_encrypted_password RAW(2000);
    l_key RAW(2000) := UTL_RAW.cast_to_raw('your_secret_key'); --
    Replace with your secret key
    l_encryption_type PLS_INTEGER :=
    DBMS_CRYPTO.ENCRYPT_AES128 + DBMS_CRYPTO.CHAIN_CBC +
    DBMS_CRYPTO.PAD_PKCS5;

BEGIN
    l_encrypted_password :=
    DBMS_CRYPTO.ENCRYPT(UTL_RAW.cast_to_raw(l_password),
    l_encryption_type, l_key);
    DBMS_OUTPUT.PUT_LINE('Encrypted Password: ' ||
    l_encrypted_password);
END;

```

3. Replace 'your_password_here' with the actual password you want to encrypt.
4. Replace 'your_secret_key' with a secure secret key for encryption.
5. Decrypting the Password (if needed):

```

DECLARE
    l_encrypted_password RAW(2000); -- Replace with your encrypted
    password
    l_decrypted_password VARCHAR2(4000);
    l_key RAW(2000) := UTL_RAW.cast_to_raw('your_secret_key'); -- Same
    secret key used for encryption
    l_encryption_type PLS_INTEGER :=
    DBMS_CRYPTO.ENCRYPT_AES128 + DBMS_CRYPTO.CHAIN_CBC +
    DBMS_CRYPTO.PAD_PKCS5;

BEGIN
    l_decrypted_password :=
    UTL_RAW.cast_to_varchar2(DBMS_CRYPTO.DECRYPT(l_encrypted_pass
    word, l_encryption_type, l_key));
    DBMS_OUTPUT.PUT_LINE('Decrypted Password: ' ||
    l_decrypted_password);
END;

```

6. Replace l_encrypted_password with the actual encrypted password you want to decrypt.
7. Use the same l_key and l_encryption_type as used during encryption.

How to Load Data into table using "Data Load Definitions" In Apex

1. Access Data Load Definitions: Navigate to "SQL Workshop" -> "Data Load Definitions" in Oracle Apex.
2. Create New Definition: Click "Create" to start defining a new data load process.
3. Specify Source and Target: Choose the data source (e.g., CSV file upload, SQL query).
4. Select the target database table where data will be loaded.
5. Map Columns (if needed): Map source data columns to target table columns.
6. Configure Options: Set options like handling duplicates, truncating table before load, and error handling preferences.

7. Execute Data Load: Run the data load definition to initiate the data loading process.
8. Monitor and Verify.

How to Call Certain (LOV) on a specific Condition in Oracle APEX

1. Set Up LOV: Define your LOV in Oracle Apex using SQL queries, static lists, or PL/SQL functions.
2. Create Dynamic Action: Navigate to your page in Oracle Apex.
3. Go to "App Builder" -> "Your Page" -> "Dynamic Actions".
4. Create a new Dynamic Action.
5. Configure Dynamic Action:
 - Event: Choose an event trigger such as "Page Load" or "Change".
 - Condition: Define the condition under which the LOV behavior should change (e.g., based on a checkbox (P1_CHECKBOX) value).
6. Define Action: Add an action like "Execute JavaScript Code".
7. Use JavaScript to enable or disable the LOV item (P1_MY_LOV) based on the condition.

Example JavaScript Code:

```
var lovId = "P1_MY_LOV"; // Replace with your LOV item name
var conditionMet = $v('P1_CHECKBOX'); // Replace with your condition item

if (conditionMet) {
    apex.item(lovId).enable(); // Enable LOV
} else {
    apex.item(lovId).disable(); // Disable LOV
}
```

Testing:

8. Save and test your Dynamic Action.

How to Refresh Report after closing Modal Page in Apex

1. Set Up Dynamic Action: Navigate to your Modal Page in Oracle Apex.
2. Ensure you have a button or process that closes the modal dialog (e.g., a "Close" button with action "Close Dialog").
3. Create a Dynamic Action: Go to "App Builder" -> "Your Page" -> "Dynamic Actions".
4. Click on "Create" to define a new Dynamic Action.
5. Configure Dynamic Action:
 - Event: Choose "Dialog Closed".
 - Selection Type: Choose "Region".
6. Region: Select the report region you want to refresh.
7. Define Action: Click on "Add Action".
8. Choose "Refresh" as the action.
9. Select the report region you want to refresh from the "Affected Elements" section.
10. Save and Test: Save the Dynamic Action.
11. Test your application by opening the modal dialog page, making changes if needed, and then closing the dialog.

What is Collections & How Insert Each Record Using Collections in Apex

1. Define Collection Type: Create a collection type (my_data_table) to hold structured data (my_data_type).
2. Populate the Collection: Initialize and extend the collection.
3. Assign values to each element (l_data(1), l_data(2), etc.) programmatically or through SQL queries.
4. Insert Records Using Collection: Use a FOR LOOP to iterate over the collection.
5. Insert each record into the target table (your_target_table) using PL/SQL.

6. Commit Changes: Ensure to commit changes (COMMIT;) after completing the insertion process to persist data.

How to Insert Multiple Records Using "FOR LOOP COLLECTION" in Oracle Apex

1. Define a Collection: Create a collection type (my_data_table) to hold data (my_data_type).
2. Write PL/SQL Block:
3. Populate the collection (l_data) with data.
4. Use a FOR LOOP to iterate through the collection.
5. Insert each item into the target table (your_table_name).

Example PL/SQL Block:

```
DECLARE
    l_data my_data_table := my_data_table();
BEGIN
    -- Populate and extend collection
    -- Use FOR LOOP to insert records
    FOR i IN 1..l_data.count LOOP
        INSERT INTO your_table_name (id, name)
        VALUES (l_data(i).id, l_data(i).name);
    END LOOP;
    COMMIT;
END;
```

Execution:

6. Run the PL/SQL block to insert records.
7. Verify data insertion by querying the target table

How to Add Change Password Option in Oracle Apex Application

1. Create a Page: Create a new form page in Oracle Apex for changing passwords.
2. Design the Form: Include fields for Username, Old Password, New Password, and Confirm New Password.
3. Set Up Processing: Implement a PL/SQL process to verify the old password, update the password if valid, and commit changes securely.
4. Ensure Security: Use parameterized queries, hash passwords securely, and implement HTTPS for secure data transmission.
5. Test and Deploy: Thoroughly test the functionality and deploy the application after successful testing

How to create "Dynamic Navigation Menu" Using DB Table in Oracle Apex

1. Database Table Setup: Create a table (menu_items) in your Oracle database to store menu items with columns like menu_id, menu_name, menu_link, and optionally parent_menu_id for hierarchical menus.
2. SQL Query for Menu Items: Write a SQL query to fetch menu items from the menu_items table based on your menu structure (flat or hierarchical).

```
SELECT menu_id, menu_name, menu_link
FROM menu_items
WHERE parent_menu_id IS NULL -- Adjust for hierarchical menus
ORDER BY menu_id; -- Or another suitable order
```

3. Configure Navigation Menu List: go to "App Builder" -> "Create" -> "List".
4. Select "Navigation Menu List" and specify the SQL query to fetch menu items.
5. Customize the list template and appearance to match your design.
Optional: Add Dynamic Actions:
6. Configure dynamic actions to handle menu item clicks.
7. Define actions such as navigating to specific pages or executing JavaScript code based on user interactions.

8. Testing and Deployment: Test the dynamic navigation menu in your Apex application to ensure menu items are displayed correctly and actions perform as expected.
9. Deploy the application after verifying functionality.

How to Login App Using DB Table Users "Custom Auth" in Apex

1. Create a Users Table: Design a table (users) in your Oracle database with columns like username, password (hashed), email, and role.
2. Define Authentication Function: Create a PL/SQL function (validate_user) to check credentials against the users table. example

```
CREATE OR REPLACE FUNCTION validate_user(
    p_username IN VARCHAR2,
    p_password IN VARCHAR2
) RETURN BOOLEAN
IS
    l_count NUMBER;
BEGIN
    SELECT COUNT(*)
    INTO l_count
    FROM users
    WHERE username = p_username
    AND password = p_password; -- Ensure password is hashed and matched
    securely

    RETURN l_count = 1;
END;
```

Set Up Authentication Scheme:

1. In Oracle Apex, go to "App Builder" -> "Shared Components" -> "Authentication Schemes".
2. Create a new scheme with type "Custom" and link it to your validate_user function.
3. Create Login Page: Create a new page in your Apex application of type "Login".
4. Customize the page layout and appearance.
5. Configure Page Authentication: Set the authentication scheme for the login page to your custom scheme.
6. Adjust session settings and error handling as needed.
7. Test and Validate: Run the Apex application and test the login page with credentials stored in the users table.
8. Ensure authentication and role-based access work correctly.

note

Password Security: Always hash passwords securely (e.g., using SHA-256) before storing them in the database.

Session Management: Configure session timeout and expiration settings to enhance security.

Error Handling: Provide clear error messages for users during login failures due to incorrect credentials.

How to Use "Authorization Schemes" Security in Oracle Apex Pages

Navigate to Application Builder: Log in to Oracle Apex.

Go to "App Builder".

Create or Edit an Application: Select the application where you want to apply Authorization Schemes.

Click on "Shared Components" in the left panel.

Manage Authorization Schemes: Under "Security", click on "Authorization Schemes".

Create a New Authorization Scheme: Click on "Create" to define a new Authorization Scheme.

Provide a name and description for the scheme.

Define the conditions that will determine whether a user is authorized.

Example: You can base authorization on user roles

(`APEX_UTIL.IS_USER_IN_ROLE('ADMIN')`), application items (`:APP_USER = 'admin'`), or other conditions relevant to your application.

Configure Authorization Scheme Conditions: Use PL/SQL expressions or predefined conditions to set up rules.

Define when the scheme should evaluate to true (user is authorized) or false (user is not authorized).

Associate Authorization Scheme with Pages: After defining the Authorization Scheme, associate it with specific pages in your application.

Go to "App Builder" -> "Pages" and select the page you want to secure.

Set Authorization Scheme for Page Access:

In the "Security" section of the page attributes, find the "Authorization" option.

Choose the Authorization Scheme you created from the dropdown list.

Save your changes.

Test the Authorization Scheme:

Log in with different user roles or conditions to test access to the secured page.

Ensure users who meet the Authorization Scheme criteria can access the page, while others are denied access

Steps to Create Daily Expenses Management Application in Oracle Apex

Create a New Application:

Log in to Oracle Apex.

Click on "App Builder" -> "Create" -> "From scratch".

Enter an application name and a workspace.

Create a Table for Expenses:

Go to "SQL Workshop" -> "Object Browser".

Right-click on "Tables" -> "Create Table".

Define table columns: `expense_id` (auto-generated), `expense_date`, `description`, `amount`, `category`.

Set `expense_id` as the primary key.

Create Pages for Managing Expenses:

Form Page for Data Entry:

Go to "App Builder" -> "Create" -> "Form".

Select "Form on a Table" and choose the expenses table.

Include fields: `expense_date`, `description`, `amount`, `category`.

Customize the form layout and save.

Interactive Report Page for Viewing Expenses:

Go to "App Builder" -> "Create" -> "Report".

Select "Interactive Report" and choose the expenses table.

Include columns: `expense_date`, `description`, `amount`, `category`.

Configure filters, sorting, and grouping as needed.

Optional: Chart Page for Expense Analysis:

Go to "App Builder" -> "Create" -> "Chart".

Choose "SQL Query" and write a query to summarize expenses.

Configure chart type (e.g., bar chart, pie chart) and display options.

Enhance Application Features:

Validation and Error Handling:

Implement data validation rules (e.g., amount should be numeric, date format valid).

Add error handling for data entry errors.

Authentication and Authorization:

Set up authentication schemes (e.g., login credentials).

Define roles to restrict access based on user permissions.

Responsive Design:

Ensure application design is responsive for desktop and mobile devices.

Testing and Deployment:

Test the application thoroughly for functionality and user experience.

Deploy the application to a suitable environment for user access.

How to Calculate Days / Leave Days using PL/SQL in Oracle Apex

Define the start and end dates (e.g., `start_date` and `end_date`)

Calculate the difference between the two dates using the `DATE_SUB` or `DATE_DIFF` function

Extract the number of days from the result

example of a PL/SQL function that calculates the number of days between two dates:

```
CREATE FUNCTION leave_days (  
    start_date IN DATE,  
    end_date IN DATE  
) RETURN NUMBER AS  
BEGIN  
    RETURN DATEDIFF(end_date, start_date);  
END;
```

call this function in your Oracle Apex application using the following syntax:

```
DECLARE  
    start_date DATE := SYSTIMESTAMP;  
    end_date DATE := SYSTIMESTAMP + 10;  
    leave_days NUMBER;  
BEGIN  
    leave_days := leave_days(start_date, end_date);  
    DBMS_OUTPUT.PUT_LINE('Leave days: ' || leave_days);  
END;
```

This will output the number of days between the `start_date` and `end_date`.

How to Page Access Control With DB Users Using Authorization Scheme in Apex

Create Authorization Scheme:

Navigate to your APEX application.

Go to Shared Components > Authorization Schemes.

Create a new authorization scheme.

Define Authorization Conditions:

Specify conditions under which access should be granted.

Use SQL queries to check user roles or specific attributes.

For example, to allow access only to users with a specific role:

sql

Copy code

```
SELECT NULL  
FROM dual  
WHERE EXISTS (  
    SELECT 1  
    FROM user_roles  
    WHERE username = :APP_USER  
    AND role_name = 'Admin'  
)
```

Replace user_roles and role_name with your actual tables and role criteria.

Assign Authorization Scheme:

Assign the authorization scheme to specific pages or components where access control is needed.

Go to the page properties or component properties.

Under Security, select your authorization scheme in the Authorization section.

Test Access Control:

Test the application with different users to ensure proper access control.

Verify that users without appropriate roles are denied access to restricted pages.

Additional Considerations:

Multiple Schemes: Create multiple authorization schemes for different access levels (e.g., admin, user).

Dynamic Conditions: Use bind variables like :APP_USER to dynamically evaluate user-specific conditions.

How to Use Application Computations Component in Oracle Apex

Access Computations:

Navigate to your application in Oracle APEX.

Go to Shared Components > Computations.

Create a Computation:

Click Create.

Name your computation and select a computation point (e.g., before header, after submit processing).

Define Conditions (Optional):

Specify conditions under which the computation should run (e.g., based on item values).

Write PL/SQL Code:

Enter your PL/SQL code to perform actions like setting default values or calculations:

sql

Copy code

```
:P1_ITEM := 'Default Value';
```

Save and Apply:

Save your computation.

Test to ensure it executes correctly under specified conditions.

How to Adding Image/Logo to Side Navigation in Oracle Apex

Prepare Your Image:

Ensure your logo is in PNG or JPG format and appropriately sized for the sidebar.

Upload Image:

Go to Shared Components > Static Application Files.

Upload your image (e.g., side_nav_logo.png).

Edit Navigation:

Navigate to the page with your side navigation.

Edit the navigation region (e.g., "Navigation Menu").

Insert Image:

Add a new HTML or Static Content region.

Use .

Adjust Styling (Optional):

Use CSS for sizing and alignment adjustments if needed.

Save and Test:

Save changes and preview your application.

Ensure the logo displays correctly in the side navigation.