

SALESFORCE VIRTUAL INTERNSHIP

A CRM Application To Handle The Clients And Their Property Related Requirements

SmartInternz



Introduction

Dreams World Properties has integrated Salesforce to enhance customer interactions. An automated system captures visitor details and preferences from their website, creating records in Salesforce. Customers are categorized as approved or non-approved. Approved customers receive personalized property recommendations, while non-approved customers see a broader range of listings. This integration improves user experience and operational efficiency by automating record creation and customer categorization. As a result, customer engagement increases, supporting the company's growth in the real estate market. This integration showcases how technology can drive business success through improved customer relationship management and streamlined operations.

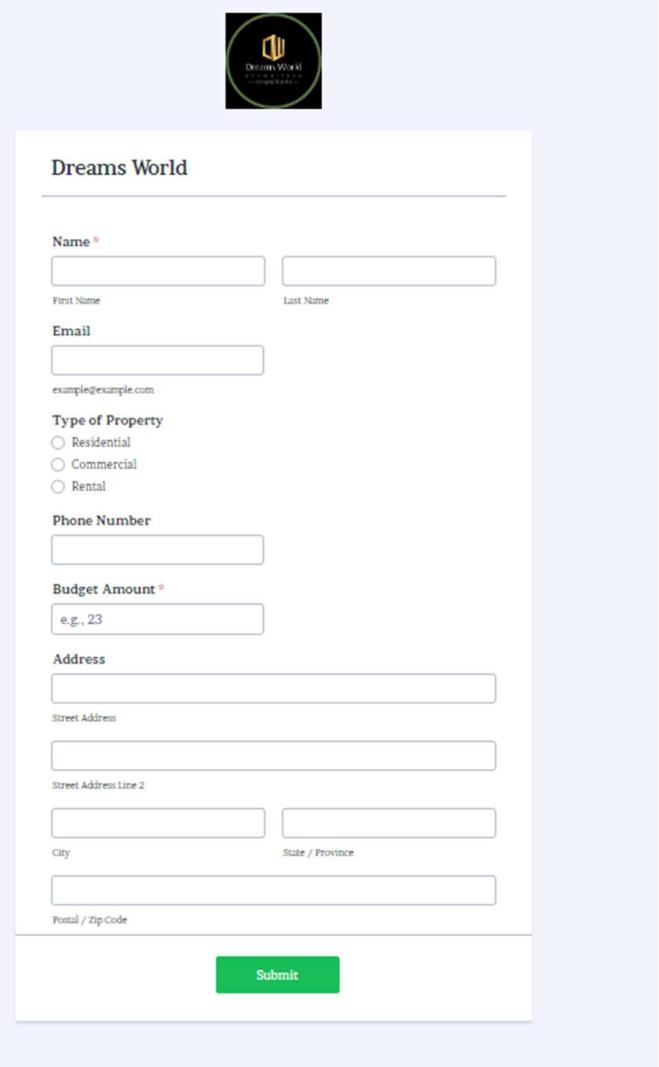
Milestone 1: - Create A JotForm and Integrate It with The Org to Create a Record of Customers Automatically

To create a form using JotForm that allows customers to input their details directly into Salesforce and enables admins to create a user in the organization, follow these steps:

Step 1: Create the Form in JotForm

1. **Sign up or log in to JotForm:**
Go to JotForm and sign in to your account.
2. **Create a new form:** Click on the "Create Form" button and choose a blank form or a template that suits your needs.
3. **Add fields to the form:** Add the necessary fields to collect customer information, such as Name, Email, Type of Property, Budget Amount, Address and Phone Number.
4. Once the form is created, publish it by clicking on publish.
5. The JotForm created link:

<https://form.jotform.com/241740916216050>



The screenshot shows a JotForm creation interface for a company named 'Dreams World'. At the top, there's a logo for 'Dreams World' and a subtext 'INTERIOR DESIGNERS'. Below the logo, the form title 'Dreams World' is displayed. The form contains the following fields:

- Name ***: Two adjacent input fields for First Name and Last Name.
- Email**: An input field with the placeholder 'example@example.com'.
- Type of Property**: Three radio button options: Residential, Commercial, and Rental.
- Phone Number**: An input field.
- Budget Amount ***: An input field with the placeholder 'e.g., 23'.
- Address**: Two stacked input fields for Street Address and Street Address Line 2.
- City** and **State / Province**: Two adjacent input fields.
- Postal / Zip Code**: An input field.

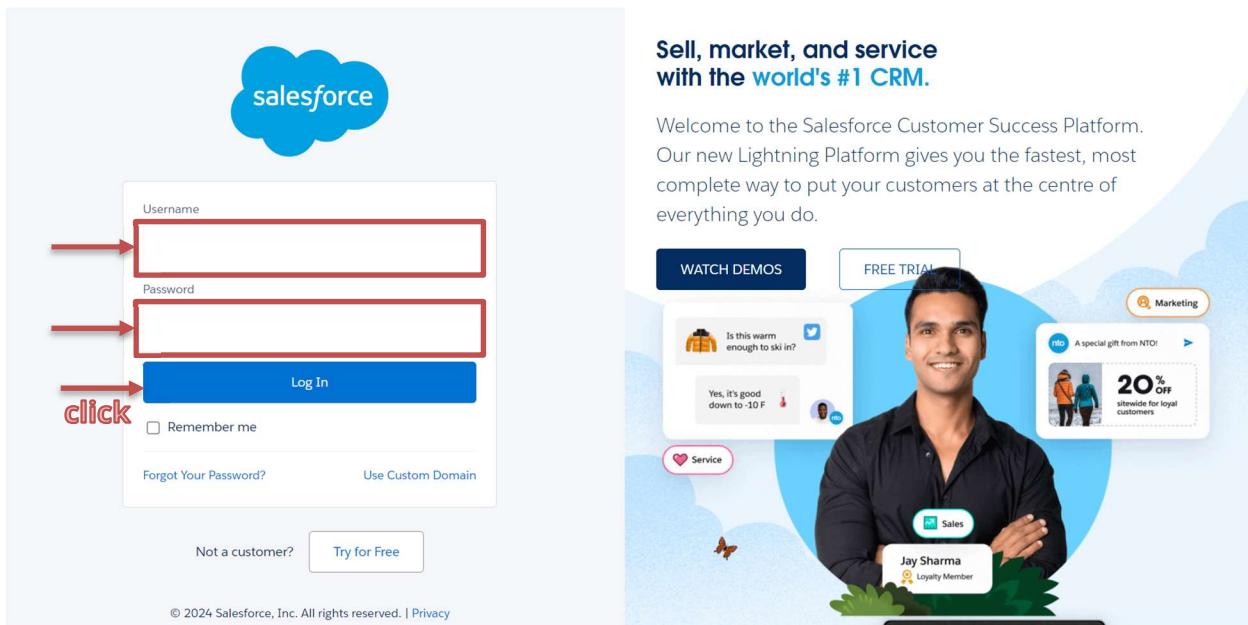
A green 'Submit' button is located at the bottom right of the form area.

Activity 1: Create Objects from Spreadsheet

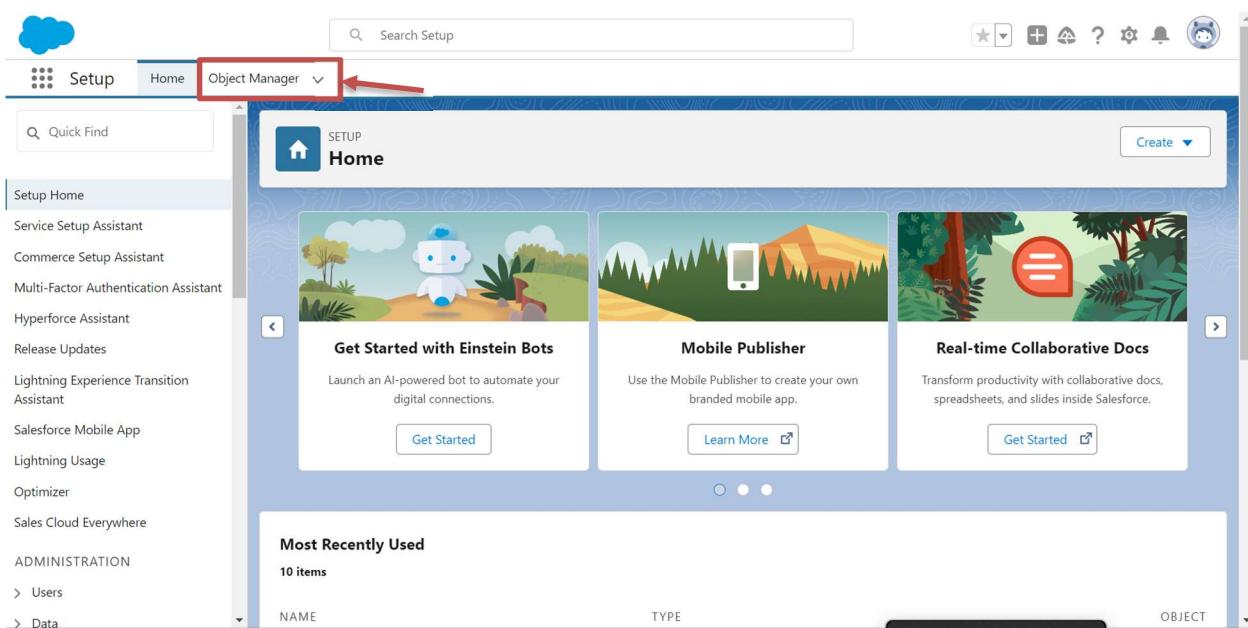
Creating Objects from Spreadsheet in Salesforce

Step 1: Login to the Salesforce account <https://login.salesforce.com/>

Credentials Required: Username and Password



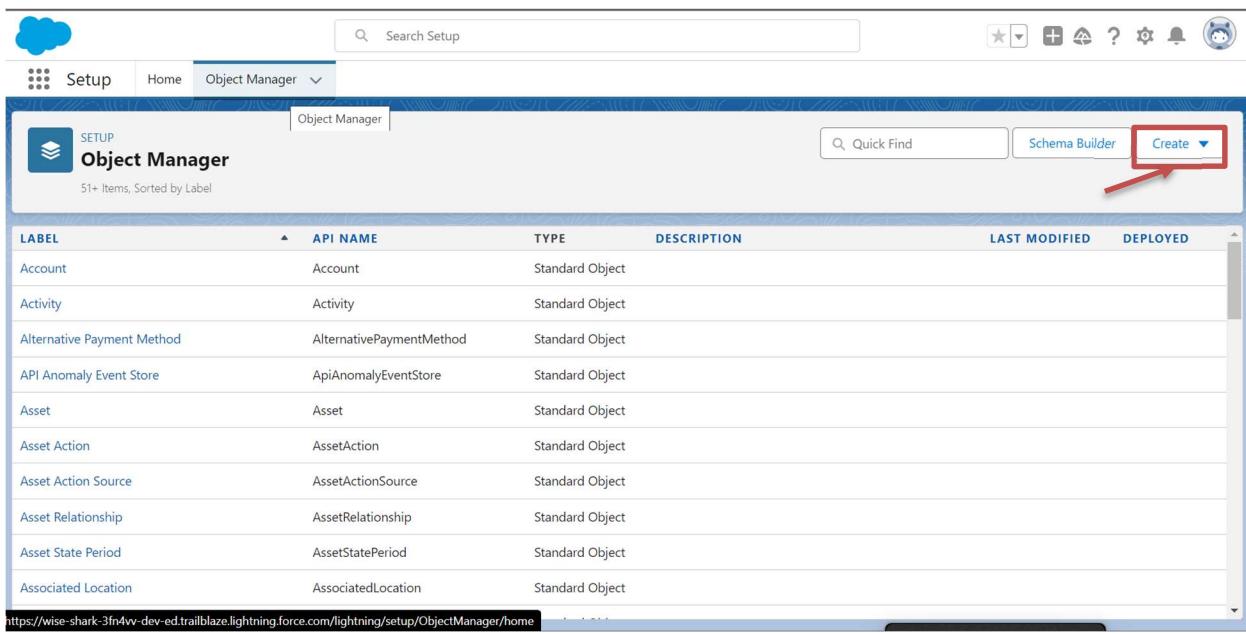
This will redirect you to the salesforce setup page.



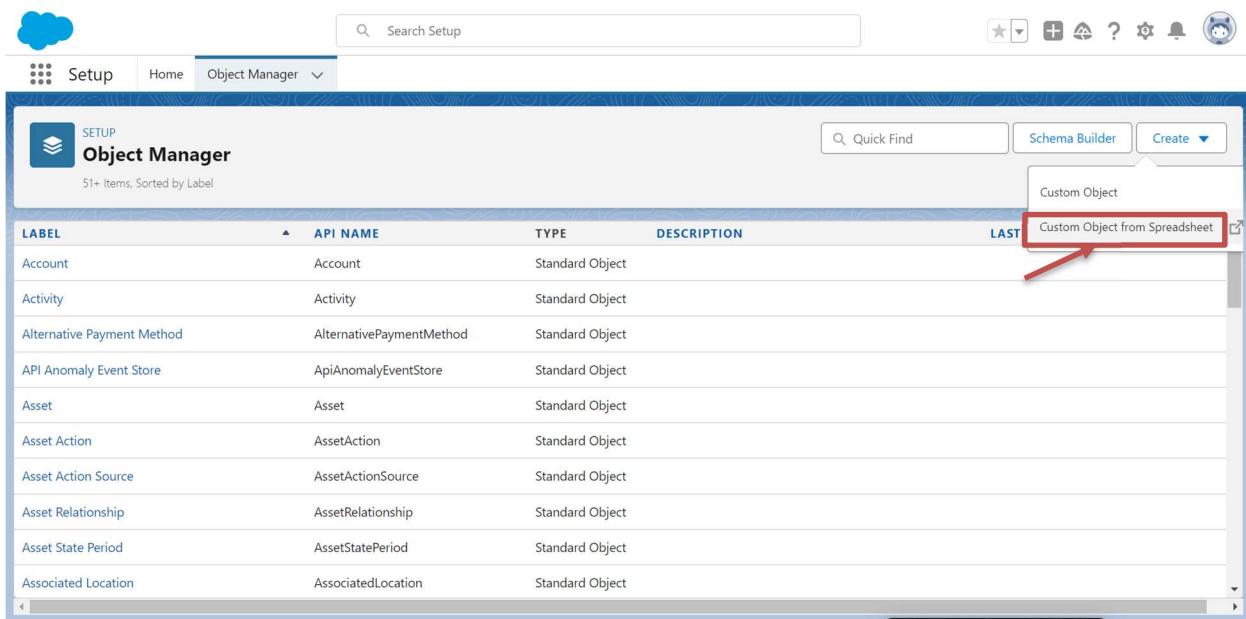
Create Customer Object

For creating the customer object follow the steps:

1. Go to the object manager and click on create object from spreadsheet.



The screenshot shows the Salesforce Object Manager page. At the top right, there is a 'Create' button with a dropdown arrow. A red box highlights this button. Below the header, there is a search bar labeled 'Search Setup' and a toolbar with various icons. The main area displays a table of objects with columns: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. The table lists standard objects like Account, Activity, Alternative Payment Method, etc. The URL in the browser address bar is <https://wise-shark-3fn4vv-ed-dev.trailblaze.lightning.force.com/lightning/setup/ObjectManager/home>.



The screenshot shows the same Salesforce Object Manager page as the previous one, but with a different focus. A red box highlights a link labeled 'Custom Object from Spreadsheet' located in the top right corner of the page. The rest of the interface is identical to the first screenshot, including the table of standard objects and the URL in the browser.

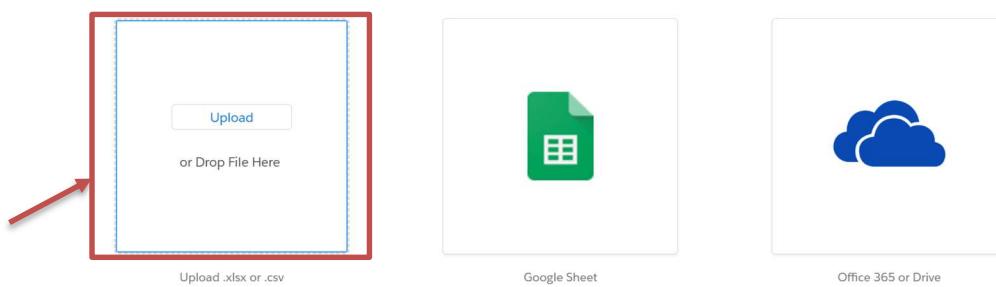
Create a custom object from a spreadsheet



Create a custom object from a spreadsheet

Select a spreadsheet

Select a source for your new object data.



2. Download the customer spreadsheet provided [Customer](#)

A screenshot of a Google Sheets document titled "Customer". A context menu is open at cell A1, with the "Download" option highlighted by a red box. The menu also includes options like "New", "Open", "Import", "Make a copy", "Share", "Email", and "Details". The main spreadsheet area shows data for properties across columns D through L.

3. After downloading, upload the file and map the fields and upload to create an object.

Create a custom object from a spreadsheet

Define object and fields

Choose the data source, map fields and their types, and import field data.

CSV File Details

Encoding Format <i>i</i>	Values Separated By	Field Label Source	* Field Labels Row	Import 3 rows of Data? <i>i</i>	Record Name Field <i>i</i>
Unicode (UTF8)	Comma	<input type="radio"/> Enter manually <input checked="" type="radio"/> Detect from row	1	<input type="radio"/> No, skip import <input checked="" type="radio"/> Yes, import data	Let Salesforce Create a Di

Fields 11 of 11 to import Hide mapped fields

IMPORT FILE FIELD NAME	SALESFORCE FIELD NAME	SALESFORCE FIELD TYPE	ADD TO LAYOUTS <i>i</i>	FIELD PREVIEW
Customer	Customer	Text	<input checked="" type="checkbox"/>	Rakesh
Phone Number	Phone Number	Integer	<input checked="" type="checkbox"/>	788797
Email	Email	Email	<input checked="" type="checkbox"/>	rakesh@gmail.com
State	State	Text	<input checked="" type="checkbox"/>	Telangana

Next

Create a custom object from a spreadsheet

Object properties

Almost finished! Time to define your object's attributes.

* Label
Customer - Sheet1

Plural Label
Customer - Sheet1

* API Name ⓘ
Customer_Sheet1

Object Description

> Advanced Settings

Back Finish

Create a custom object from a spreadsheet

Nice Work!



Now you can add your object to a Lightning app. You might need to refresh the object list to see it.

Import Overview

Object Created
Customer - Sheet1

Fields Detected
11 Fields Created
11

Rows Detected
3 Rows Imported
3

[Import Another Object](#)

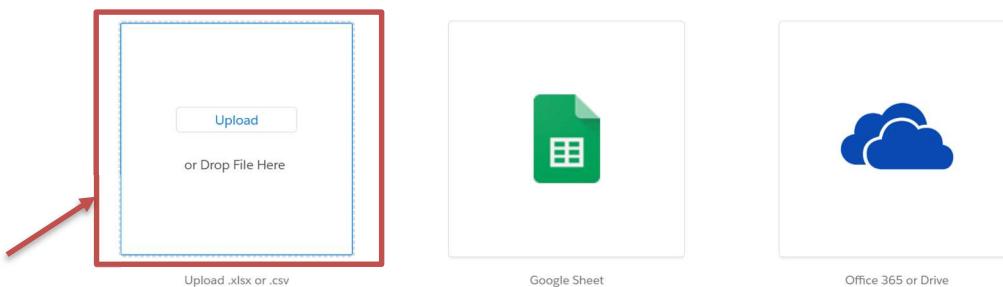
Create Property Object

1. Follow the same from creating the Property Object with the [Property](#) spreadsheet.

Create a custom object from a spreadsheet

Select a spreadsheet

Select a source for your new object data.



Property

File Edit View Insert Format Data Tools Extensions Help

A1

Ctrl+O

D E F G H I J K L

1 field
2 cked
3 ecked
4 cked

6 Download

7 Microsoft Excel (.xlsx)

OpenDocument (.ods)
PDF (.pdf)
Web page (.html)
Comma-separated values (.csv)
Tab-separated values (.tsv)

New Open Import Make a copy Share Email Add a shortcut to Drive Move Version history Make available offline Details Settings

Sheet1

Create a custom object from a spreadsheet

Define object and fields

Choose the data source, map fields and their types, and import field data.

CSV File Details

Encoding Format Values Separated By Field Label Source Enter manually Detect from row * Field Labels Row Import 3 rows of Data? No, skip import Yes, import data Record Name Field

Fields 4 of 4 to import Hide mapped fields

IMPORT FILE FIELD NAME	SALESFORCE FIELD NAME	SALESFORCE FIELD TYPE	ADD TO LAYOUTS	FIELD PREVIEW
✓ Property Name	Property Name	Text	<input checked="" type="checkbox"/>	Lotus Apartments
✓ Type	Type	Text	<input checked="" type="checkbox"/>	Residential
✓ Location	Location	Text	<input checked="" type="checkbox"/>	hydeerabad
✓ Verified	Verified	Text	<input checked="" type="checkbox"/>	checked

Back



Next

Create a custom object from a spreadsheet

Object properties

Almost finished! Time to define your object's attributes.

* Label

Property - Sheet1

Plural Label

Property - Sheet1

* API Name

Property_Sheet1

Object Description

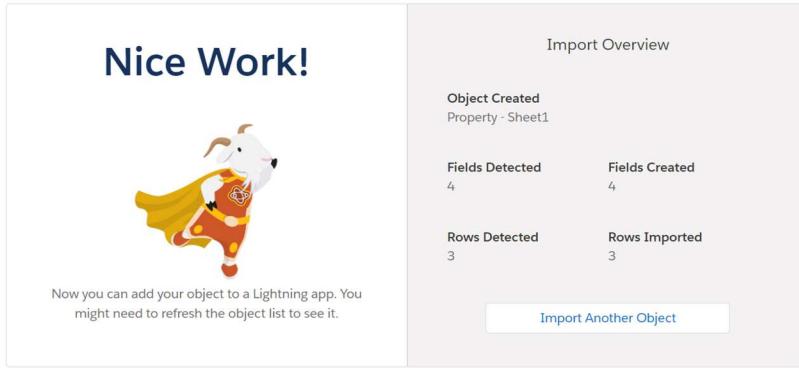
Advanced Settings

Back



Finish

Create a custom object from a spreadsheet



Activity 2: Integrate JotForm with Salesforce Platform

In this Activity we are going to integrate JotForm with Salesforce.

1. On the JotForm Platform, Click on Integration and choose Salesforce.

The screenshot shows the JotForm platform interface for a form titled 'Dreams World'. The left sidebar has a red box around the 'INTEGRATIONS' section, which is highlighted by a red arrow. The main area shows 'FORM SETTINGS' with the title 'Dreams World' and 'Form Status' set to 'ENABLED'. Other sections include 'CONDITIONS', 'THANK YOU PAGE', 'APPROVAL FLOWS', 'JOTFORM SIGN', and 'MOBILE NOTIFICATIONS'.

The screenshot shows the Jotform interface with the 'SETTINGS' tab selected. On the left sidebar, there are several options: 'CONDITIONS', 'THANK YOU PAGE' (which is checked), 'INTEGRATIONS' (highlighted with a red box), 'APPROVAL FLOWS', 'JOTFORM SIGN', and 'MOBILE NOTIFICATIONS'. The main area is titled 'INTEGRATIONS' and shows a list of available integrations. The 'Salesforce' integration is highlighted with a red box. A red arrow points from the 'INTEGRATIONS' option in the sidebar to the 'Salesforce' card. Other visible integrations include 'Square', 'Google Sheets', 'Google Drive', and 'Twilio'.

2. Click on User Integration and choose “Add to Form”.
3. Select the Org with which you want to Integrate your JotForm with.
4. Select an Action.
5. Select a Salesforce Object: - Customer
6. Map Each and every field on the Object with the fields on the form and “Save Action”.

Jotform Form Builder

Dreams World
Last edited at Mon, Jun 24, 2024

BUILD **SETTINGS** **PUBLISH** **Preview Form**

CONDITIONS
Set up conditional logic

THANK YOU PAGE
Show page after submission

INTEGRATIONS
Connect your form to other apps

APPROVAL FLOWS
Create an approval flow

JOTFORM SIGN
Power your forms with Jotform Sign

MOBILE NOTIFICATIONS
Customize mobile app notifications

Find the record that matches with the selected fields

Object Fields **Dreams World**

Name	Name
Email	Email
Property Type	Type of Property
Street Address	Address - Street Address
Street Address line 2	Address - Street Address 2
City	Address - City
State	Address - State
postal code	Address - Postal/Zip Code
	Budget Amount

Give Feedback

7. Then “Save the Integration” and “Finish”.

Jotform Form Builder

Dreams World
Last edited at Mon, Jun 24, 2024

BUILD **SETTINGS** **PUBLISH** **Preview Form**

CONDITIONS
Set up conditional logic

THANK YOU PAGE
Show page after submission

INTEGRATIONS
Connect your form to other apps

APPROVAL FLOWS
Create an approval flow

JOTFORM SIGN
Power your forms with Jotform Sign

MOBILE NOTIFICATIONS
Customize mobile app notifications

SALESFORCE
Send new leads, contacts, or accounts to your sales CRM

All Actions **See Action Logs** **+ Add New Action**

1 Find existing record
Customer

SAVE INTEGRATION

Give Feedback



Integration ready!

You have successfully created your integration.

FINISH

Activity 3: Create Roles

Creating roles as per the business requirement.

1. Go to Setup and Click on Roles, then click on Expand all and Add a Role just below the Sales Representative

Object Manager

51+ Items, Sorted by Label

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Account	Account	Standard Object			
Activity	Activity	Standard Object			
Alternative Payment Method	AlternativePaymentMethod	Standard Object			
API Anomaly Event Store	ApiAnomalyEventStore	Standard Object			
Asset	Asset	Standard Object			
Asset Action	AssetAction	Standard Object			
Asset Action Source	AssetActionSource	Standard Object			
Asset Relationship	AssetRelationship	Standard Object			
Asset State Period	AssetStatePeriod	Standard Object			
Associated Location	AssociatedLocation	Standard Object			

Home

Search Setup

roles

- Users
 - Roles **Selected**
- Feature Setting
- Sales
 - Contact Roles on Contracts
 - Contact Roles on Opportunities
- Service
 - Case Teams
 - Case Team Roles **Selected**
 - Contact Roles on Cases

Get Started with Einstein Bots

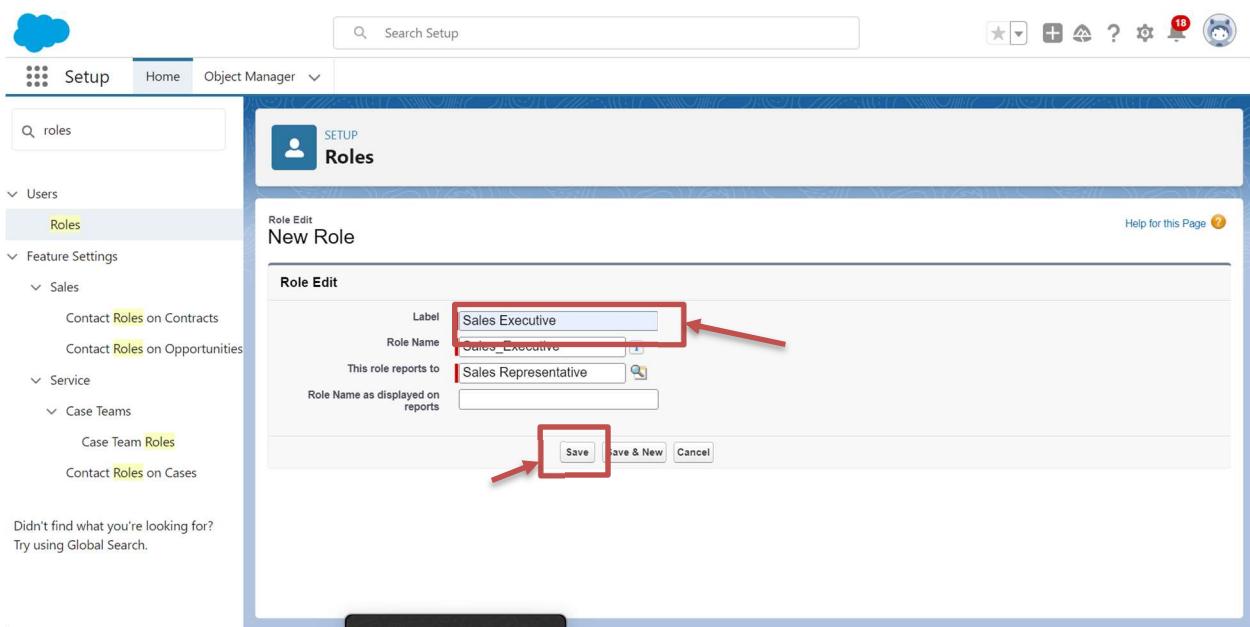
Mobile Publisher

Real-time Collaborative Docs

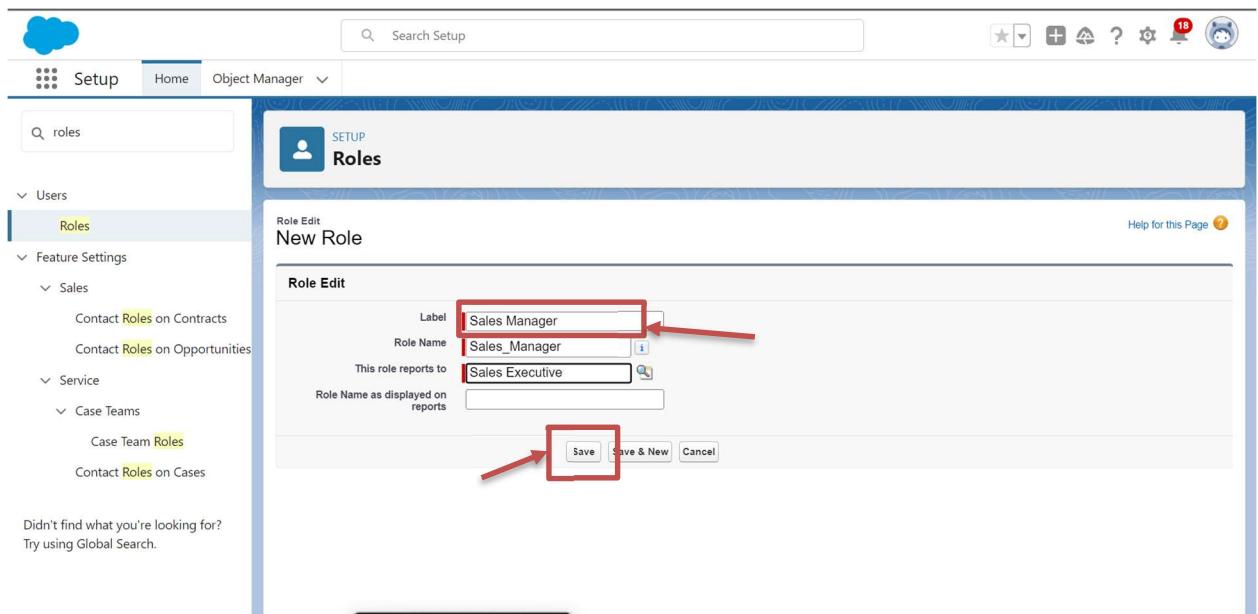
Most Recently Used

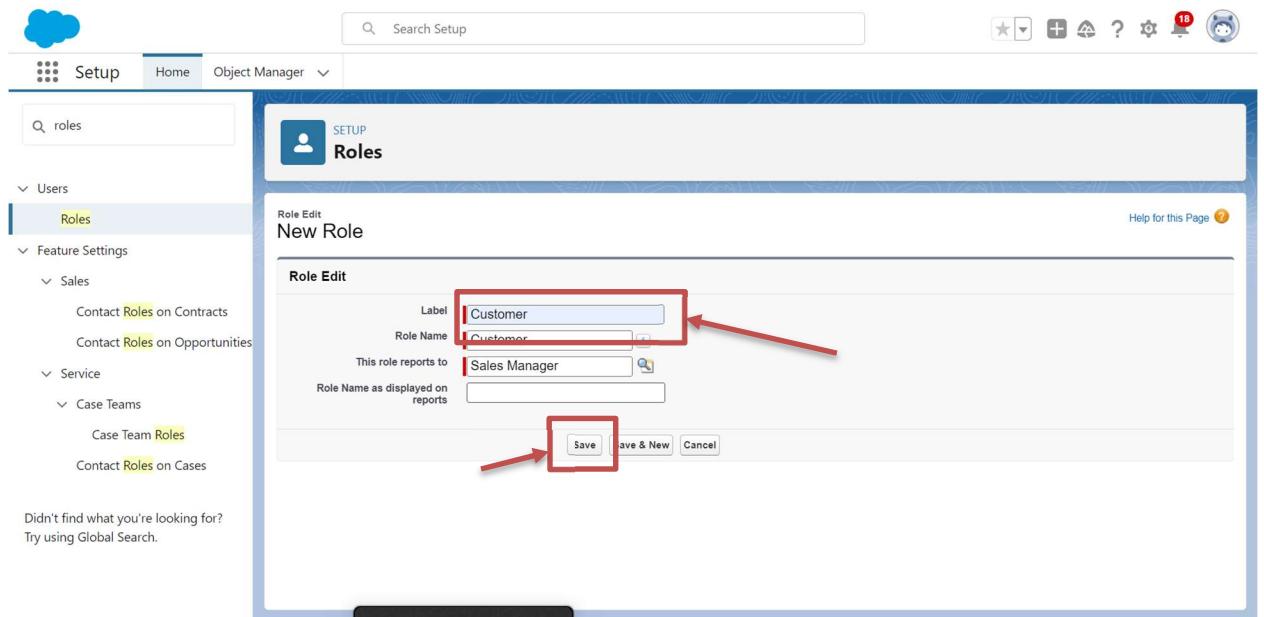
2. Label - Sales Executive

Reports to - Sales Representative



3. Similarly Create a Role Name “Sales Manager” below Sales Executive which reports to Sales Executive, Also Add a Role below Sales Manager labeled as “Customer” which reports to Sales Manager.

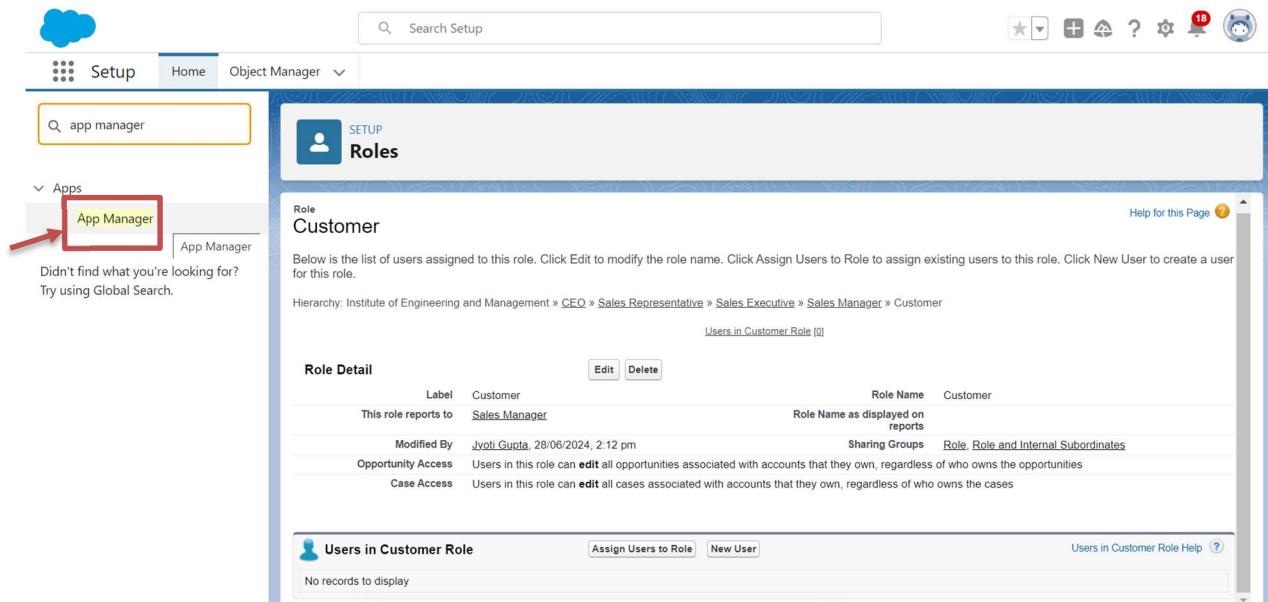




Activity 4: Create A Property Details App

Creating an app where the objects will be displayed.

1. From Setup, Go to App Manager and click on New Lightning App and Name it as “Property Details” and add “Customer” and “Property” Object.



The screenshot shows the 'Lightning Experience App Manager' page. At the top, there's a search bar labeled 'Search Setup' and various navigation icons. Below the header, a sidebar on the left lists 'Apps' and 'App Manager'. The main area displays a table of 28 items, sorted by App Name. The columns include App Name, Developer Name, Description, Last Modified, App Type, and Vi... (View). A red arrow points to the 'New Lightning App' button in the top right corner of the page.

This screenshot shows the 'New Lightning App' configuration page. It has two main sections: 'App Details' and 'App Branding'. In the 'App Details' section, the 'App Name' field is filled with 'Property Details' and has a red box around it. The 'Developer Name' field is also visible. In the 'App Branding' section, there's a placeholder image for 'Dreams World Properties' with a green border, and a color hex code '#050505' is selected. At the bottom, there's an 'Org Theme Options' checkbox and a 'Next' button, which is highlighted with a red box and an arrow pointing to it.

New Lightning App

App Options

Navigation and Form Factor i

* Navigation Style

- Standard navigation
 Console navigation

* Supported Form Factors

- Desktop and phone
 Desktop
 Phone

Setup and Personalization i

Setup Experience

- Setup (full set of Setup options)
 Service Setup

App Personalization Settings

- Disable end user personalization of nav items in this app
 Disable temporary tabs for items outside of this app
 Use Omni-Channel sidebar

Back



Next

New Lightning App

Utility Items (Desktop Only)

Give your users quick access to productivity tools and add background utility items to your app.

Add Utility Item

Utility Bar Alignment i

Default

The utility bar is a fixed footer that opens components in docked panels. Available only when the app is viewed in Lightning Experience on a desktop.

Back



Next

New Lightning App

Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation to add or move items, but users can't remove or rename the items that you add. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is viewed in a format that the item doesn't support.

Available Items



grope

Property - Sheet1

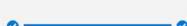
Selected Items



Customer - Sheet1



Back



Next

2. Click Next and Next then add "System Admin" Profile and Save.

New Lightning App

Navigation Items

Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation to add or move items, but users can't remove or rename the items that you add. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is viewed in a format that the item doesn't support.

Available Items Selected Items

Customer - Sheet1
Property - Sheet1

Back Next

New Lightning App

User Profiles

Choose the user profiles that can access this app.

Available Profiles Selected Profiles

System Administrator

Back Save & Finish

Activity 5: Create Profiles

Customer

1. From Setup, Go to Profiles and Clone Salesforce Platform User and Name it "Customer".

The screenshot shows the Salesforce Setup Home page. In the left sidebar, under the 'Users' section, there is a 'Profiles' item which is highlighted with a red box and has a red arrow pointing to it. Below this, a message says 'Didn't find what you're looking for? Try using Global Search.' The main content area features a 'Home' card with three sections: 'Get Started with Einstein Bots', 'Mobile Publisher', and 'Real-time Collaborative Docs'. Below these cards is a section titled 'Most Recently Used' with 10 items. At the bottom of the page, there are navigation tabs for 'NAME', 'TYPE', and 'OBJECT'.

New Profile			A B C D E F G H I J
<input type="checkbox"/>	Action	Profile Name	User License
<input type="checkbox"/>	Edit Clone	Silver Partner User	Silver Partner
<input type="checkbox"/>	Edit Clone	Solution Manager	Salesforce
<input type="checkbox"/>	Edit Clone	Standard Platform User	Salesforce Platform
<input type="checkbox"/>	Edit Clone	Standard User	Salesforce
<input type="checkbox"/>	Edit Clone	System Administrator	Salesforce

Clone Profile

Enter the name of the new profile.

You must select an existing profile to clone from.

Existing Profile	Standard Platform User
User License	Salesforce Platform
Profile Name	<input type="text" value="Customer"/>

2. Uncheck all the Custom Objects and Check only Property Details from Custom App Settings.

Custom App Settings				Custom App Settings	
	Visible	Default		Visible	Default
Analytics Studio (standard_Insights)	<input type="checkbox"/>	<input checked="" type="radio"/>	(trihdhtips_	Playground Starter (Playground_Starter)	<input type="checkbox"/>
App Launcher (standard_AppLauncher)	<input type="checkbox"/>	<input checked="" type="radio"/>	Property Details (Property_Details)	<input checked="" type="checkbox"/>	<input checked="" type="radio"/>
Platform (standard_Platform)	<input type="checkbox"/>	<input checked="" type="radio"/>	WDC (standard_Work)	<input type="checkbox"/>	<input checked="" type="radio"/>

3. Also Remove all the Standard Object Permissions.

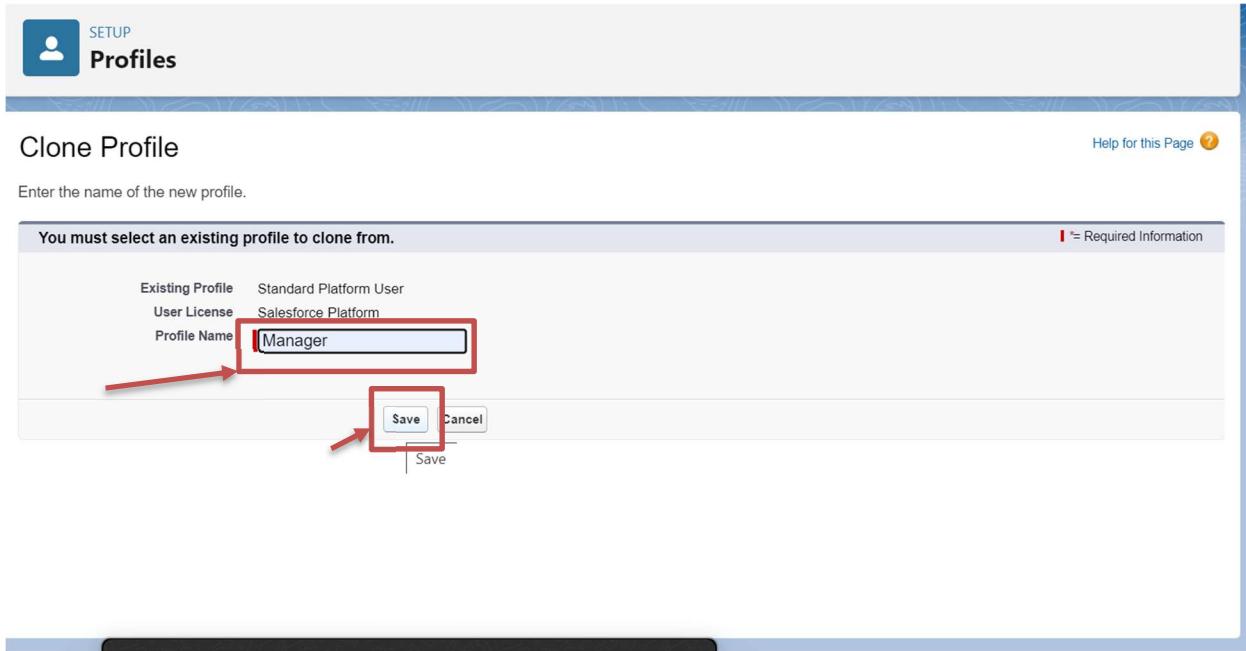
Standard Object Permissions													
	Basic Access			Data Administration				Basic Access			Data Administration		
	Read	Create	Edit	Delete	View All	Modify All		Read	Create	Edit	Delete	View All	Modify All
Accounts	<input type="checkbox"/>	Contact Point Phones	<input type="checkbox"/>										
Assets	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			Contact Point Type Consents	<input type="checkbox"/>					
Authorization Forms	<input type="checkbox"/>	Customers	<input type="checkbox"/>										
Authorization Form Consents	<input type="checkbox"/>	D&B Companies											
Authorization Form Data Uses	<input type="checkbox"/>	Data Use Legal Bases	<input type="checkbox"/>										
Authorization Form Texts	<input type="checkbox"/>	Data Use Purposes	<input type="checkbox"/>										
Background Operations							Documents	<input type="checkbox"/>					
Business Brands	<input type="checkbox"/>	Engagement Channel Types	<input type="checkbox"/>										
Communication Subscriptions							Ideas						

4. Uncheck all the Custom Object Permissions and check read and view all in "Property"

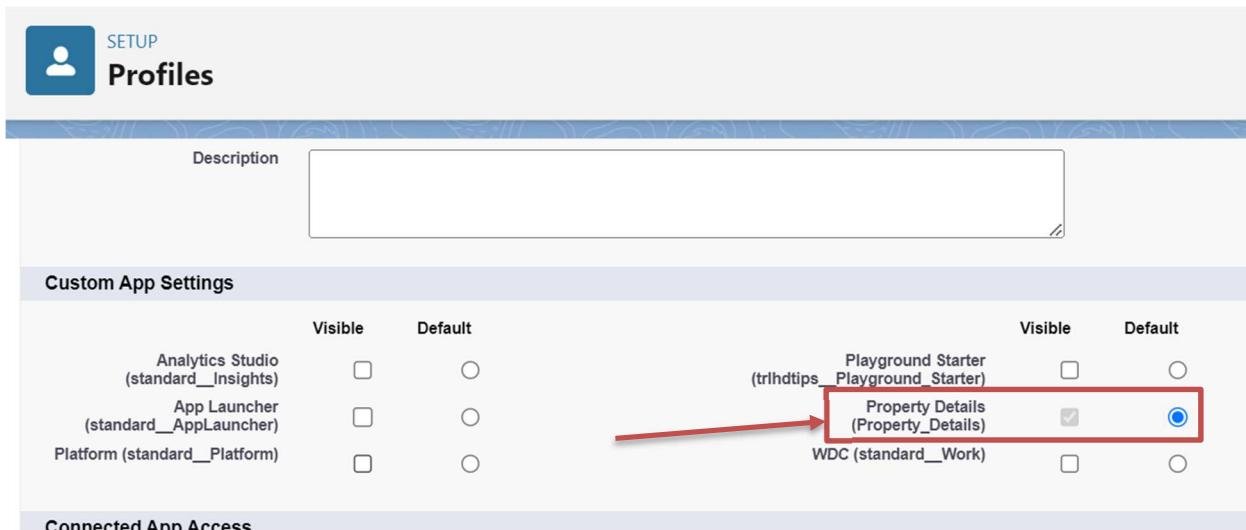
Custom Object Permissions													
	Basic Access			Data Administration				Basic Access			Data Administration		
	Read	Create	Edit	Delete	View All	Modify All		Read	Create	Edit	Delete	View All	Modify All
Contact Point Consents	<input type="checkbox"/>	Streaming Channels	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>					
Contact Point Emails	<input type="checkbox"/>	User External Credentials	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>					
Customer - Sheet1	<input type="checkbox"/>	Property - Sheet1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>					

Manager

1. From Setup, Go to Profiles and Clone Salesforce Platform User and Name it "Manager".



2. Uncheck all the Custom Objects and Check only Property Details from Custom App Settings.



3. Also Remove all the Standard Object Permissions.

SETUP Profiles

Standard Object Permissions

The permissions defined here control access at the object level. Access to individual records within that object type is controlled by the sharing model. Set access levels based on the functional requirements for the profile. For example, create different groups of permissions for individual contributors, managers, and administrators. [How do I choose?](#) [?](#)

	Basic Access						Data Administration		Basic Access						Data Administration				
	Read	Create	Edit	Delete	View All <small>i</small>	Modify All <small>i</small>	Read	Create	Edit	Delete	View All <small>i</small>	Modify All <small>i</small>	Read	Create	Edit	Delete	View All <small>i</small>	Modify All <small>i</small>	
Accounts	<input type="checkbox"/>	<input type="checkbox"/>							Contact Point Phones	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
Assets	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>									Contact Point Type Consents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Authorization Forms	<input type="checkbox"/>	<input type="checkbox"/>							Customers	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
Authorization Form Consents	<input type="checkbox"/>	<input type="checkbox"/>							D&B Companies	<input type="checkbox"/>									
Authorization Form Data Uses	<input type="checkbox"/>	<input type="checkbox"/>							Data Use Legal Bases	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
Authorization Form Texts	<input type="checkbox"/>	<input type="checkbox"/>							Data Use Purposes	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
Background Operations	<input type="checkbox"/>												Documents	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Business Brands	<input type="checkbox"/>	<input type="checkbox"/>							Engagement Channel Types	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				
Communication Subscriptions	<input type="checkbox"/>	<input type="checkbox"/>							Ideas	<input type="checkbox"/>	<input type="checkbox"/>								

- Uncheck all the Custom Object Permissions and check only "modify all" from "Property" and "Customer".

SETUP Profiles

Contact Point Emails	<input type="checkbox"/>	<input type="checkbox"/>	User External Credentials	<input type="checkbox"/>															
Custom Object Permissions																			
	Basic Access						Data Administration		Basic Access						Data Administration				
	Read	Create	Edit	Delete	View All <small>i</small>	Modify All <small>i</small>			Read	Create	Edit	Delete	View All <small>i</small>	Modify All <small>i</small>					
Customer - Sheet1														Property - Sheet1					
														<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Session Settings														<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Activity 6: Create A Check Box Field on User

- Setup go to Object Manager then go to Search for User select Fields and Relationships

Setup Home Object Manager ▾

SETUP Object Manager

51+ Items, Sorted by Label

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Account	Account	Standard Object			
Activity	Activity	Standard Object			
Alternative Payment Method	AlternativePaymentMethod	Standard Object			
API Anomaly Event Store	ApiAnomalyEventStore	Standard Object			
Asset	Asset	Standard Object			
Asset Action	AssetAction	Standard Object			
Asset Action Source	AssetActionSource	Standard Object			
Asset Relationship	AssetRelationship	Standard Object			
Asset State Period	AssetStatePeriod	Standard Object			
Associated Location	AssociatedLocation	Standard Object			

Setup Home Object Manager ▾

SETUP > OBJECT MANAGER User

Details

Fields & Relationships

User Page Layouts

User Profile Page Layouts

Lightning Record Pages

Buttons and Links

Compact Layouts

Field Sets

Object Limits

Related Lookup Filters

Search Layouts

Details

Description

API Name

User

Custom

Singular Label

User

Plural Label

Users

Enable Reports

Track Activities

Track Field History

Deployment Status

Help Settings

Standard salesforce.com Help Window

SETUP > OBJECT MANAGER

User

Fields & Relationships

31+ Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
About Me	AboutMe	Text Area(1000)		
Active	IsActive	Checkbox		✓
Address	Address	Address		
Admin Info Emails	ReceivesAdminInfoEmails	Checkbox		
Alias	Alias	Text(8)		✓
Allow Forecasting	ForecastEnabled	Checkbox		
Banner Photo	BannerPhotoId	Lookup(Photo)		
Call Center	CallCenterId	Lookup(Call Center)		✓
Chatter Email Highlights Frequency	DigestFrequency	Picklist		

SETUP > OBJECT MANAGER

User

Fields & Relationships

User Page Layouts

Field Types

- Formula
- Roll-Up Summary
- Checkbox
- Hierarchical Relationship
- Currency
- Date
- Date/Time
- Email
- Geolocation
- Number
- Percent
- Phone

User
New Custom Field

Step 1. Choose the field type

Data Type

- None Selected
- Auto Number
- Formula
- Roll-Up Summary
- Hierarchical Relationship

Step 1

Next Cancel

2. Create new Field Named as “Verified” as Data type “Check Box”

Step 2. Enter the details

Step 2 of 4

Field Label

Default Value Checked Unchecked

Field Name

Description

Help Text

Auto add to custom report type Add this field to existing custom report types that contain this entity

Previous Next Cancel

SETUP > OBJECT MANAGER

User

Details

Fields & Relationships

User Page Layouts

User Profile Page Layouts

Lightning Record Pages

Buttons and Links

Compact Layouts

Field Sets

Object Limits

Related Lookup Filters

Search Layouts

Field-Level Security for Profile	<input type="checkbox"/> Visible	<input type="checkbox"/> Read-Only
Analytics Cloud Integration User	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Analytics Cloud Security User	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Authenticated Website	<input type="checkbox"/>	<input type="checkbox"/>
Authenticated Website	<input type="checkbox"/>	<input type="checkbox"/>
B2B Reordering Portal Buyer Profile	<input type="checkbox"/>	<input type="checkbox"/>
Contract Manager	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Cross Org Data Proxy User	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Custom: Marketing Profile	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Custom: Sales Profile	<input checked="" type="checkbox"/>	<input type="checkbox"/>

SETUP > OBJECT MANAGER

User

Details

Fields & Relationships

User Page Layouts

User Profile Page Layouts

Lightning Record Pages

Buttons and Links

Compact Layouts

Field Sets

Object Limits

Related Lookup Filters

Search Layouts

Step 4. Add to page layouts

Step 4 of 4

Previous Save & New **Save** Cancel

Field Label	Verified
Data Type	Checkbox
Field Name	Verified
Description	

Select the page layouts that should include this field. The field will be added as the last field in the first 2-column section of these page layouts. The field will not appear on any pages if you do not select a layout.

To change the location of this field on the page, you will need to customize the page layout.

Add Field Page Layout Name

Community Member Layout

User Layout

User Profile Layout

When finished, click Save & New to create more custom fields, or click Save if you are done.

Previous Save & New Save Cancel

Activity7: Create Users

User1

1. Go to Setup then from Administration select Users then New User.

The screenshot shows the Salesforce Setup interface under the 'Users' section. On the left, there's a sidebar with options like 'Permission Set Groups', 'Permission Sets', 'Profiles', 'Public Groups', 'Queues', 'Roles', 'User Management Settings', and 'Users'. The 'Users' option is selected. The main area is titled 'All Users' and contains a table of users with columns for Action, Full Name, Alias, Username, Role, Active, and Profile. At the bottom of the table, there are buttons for 'New User', 'Reset Password(s)', and 'Add Multiple Users'. A red arrow points to the 'New User' button.

2. Last Name - Executive

3. Role - Sales Executive

4. License - Salesforce

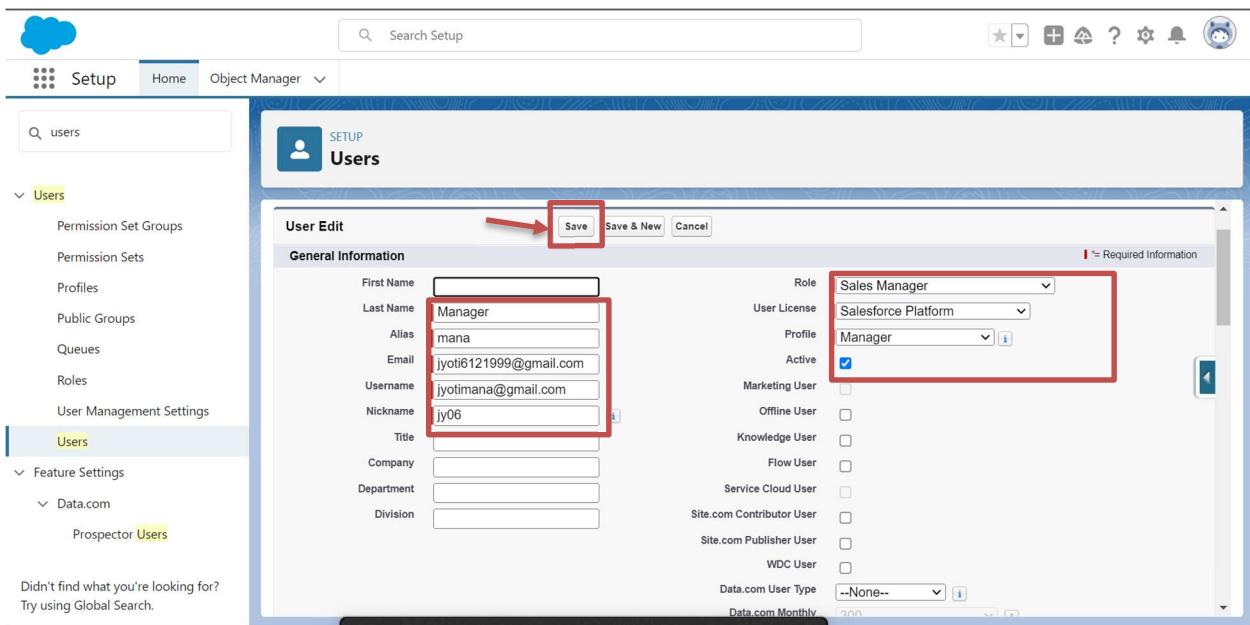
5. Profile - System Administrator

6. Save

The screenshot shows the 'User Edit' form for a user named 'Executive'. The 'General Information' section is highlighted with a red box. The 'Last Name' field is set to 'Executive', and the 'Role' dropdown is set to 'Sales Executive'. The 'Save' button at the top right is also highlighted with a red box. A red arrow points from the 'General Information' section towards the 'Save' button.

User2

1. Go to Setup then from Administration select Users then New User.
2. Last Name - Manager
3. Role - Sales Manager
4. License - Salesforce Platform
5. Profile - Manager
6. Save



User3

1. Go to Setup then from Administration select Users then New User.
2. Last Name - Customer
3. Role - Customer
4. License - Salesforce Platform

5. Profile - Customer
6. Make Sure the verified check box is "Unchecked"
7. Save

The screenshot shows the 'User Edit' page in the Salesforce Setup interface. The 'General Information' section contains fields for First Name, Last Name, Alias, Email, Username, Nickname, Title, Company, Department, and Division. The 'Advanced Settings' section includes Role (set to Customer), User License (set to Salesforce Platform), Profile (set to Customer), Active (checkbox checked), Marketing User, Offline User, Knowledge User, Flow User, Service Cloud User, Site.com Contributor User, Site.com Publisher User, WDC User, Data.com User Type (dropdown set to --None--), and Data.com Monthly (dropdown set to 300). A red box highlights the 'Last Name' field and the 'Role', 'User License', and 'Profile' sections.

User4

1. Go to Setup then from Administration select Users then New User.
2. Last Name - Customer2
3. Role - Customer
4. License - Salesforce Platform
5. Profile - Customer

User Edit

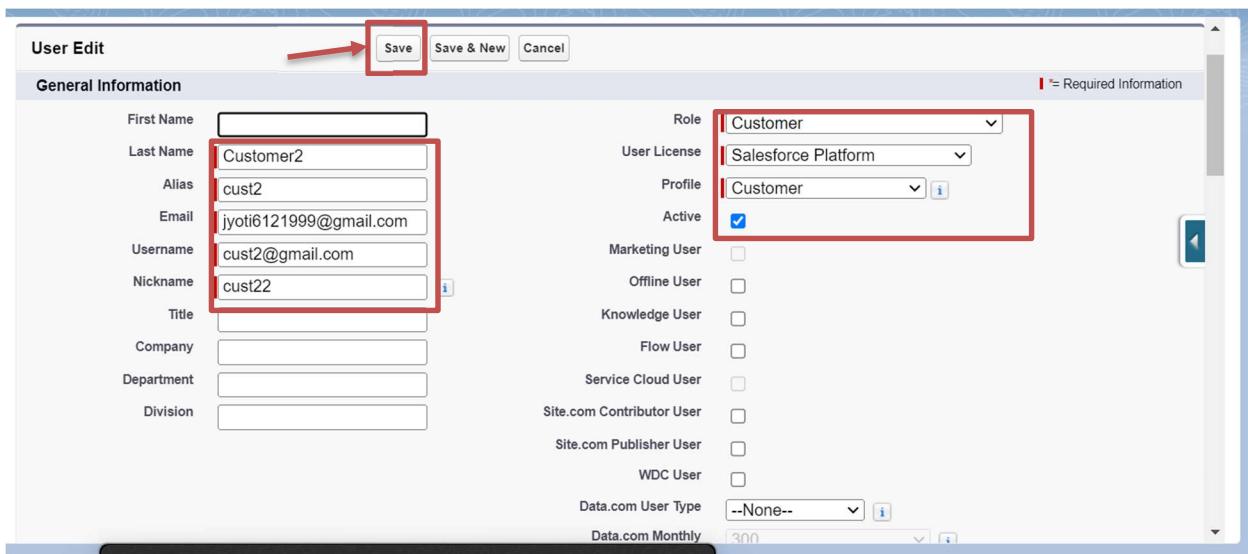
General Information

First Name: Customer2
Last Name: cust2
Alias: cust2
Email: jyoti6121999@gmail.com
Username: cust2@gmail.com
Nickname: cust22
Title:
Company:
Department:
Division:

Role: Customer
User License: Salesforce Platform
Profile: Customer
Active: Marketing User:
Offline User:
Knowledge User:
Flow User:
Service Cloud User:
Site.com Contributor User:
Site.com Publisher User:
WDC User:
Data.com User Type: --None--
Data.com Monthly: 300

Save Save & New Cancel

! = Required Information



6. Make Sure the verified check box is “checked”.

SETUP

Users

Street:
City:
Zip/Postal Code:
State/Province:
Country:

Single Sign On Information

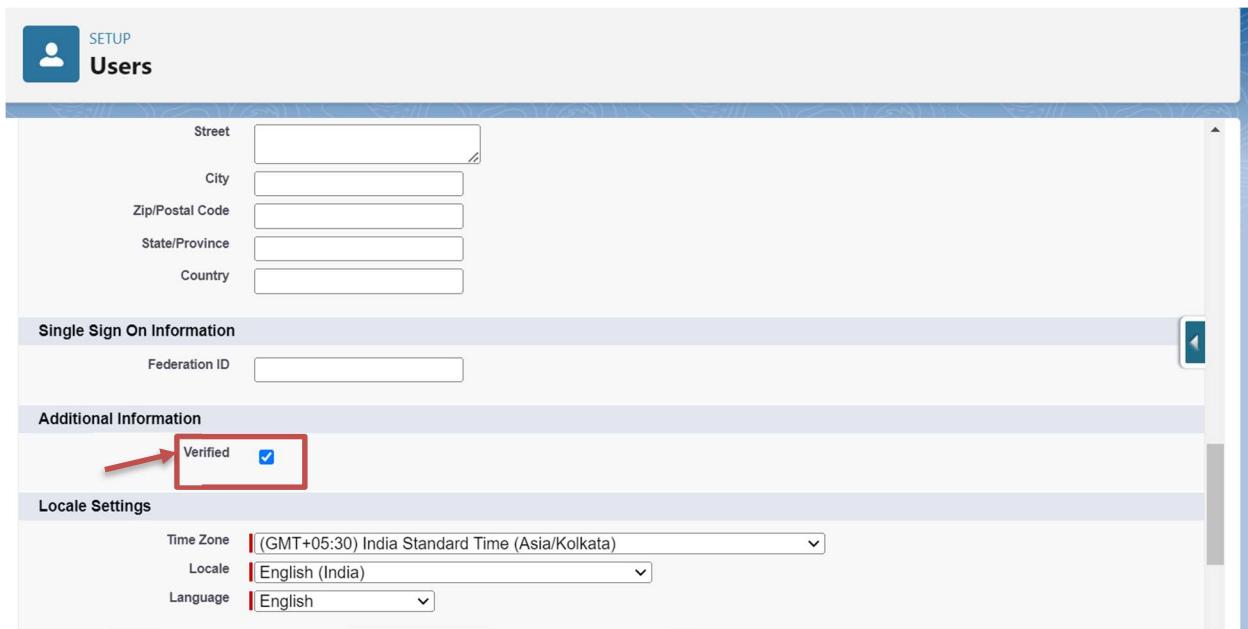
Federation ID:

Additional Information

Verified:

Locale Settings

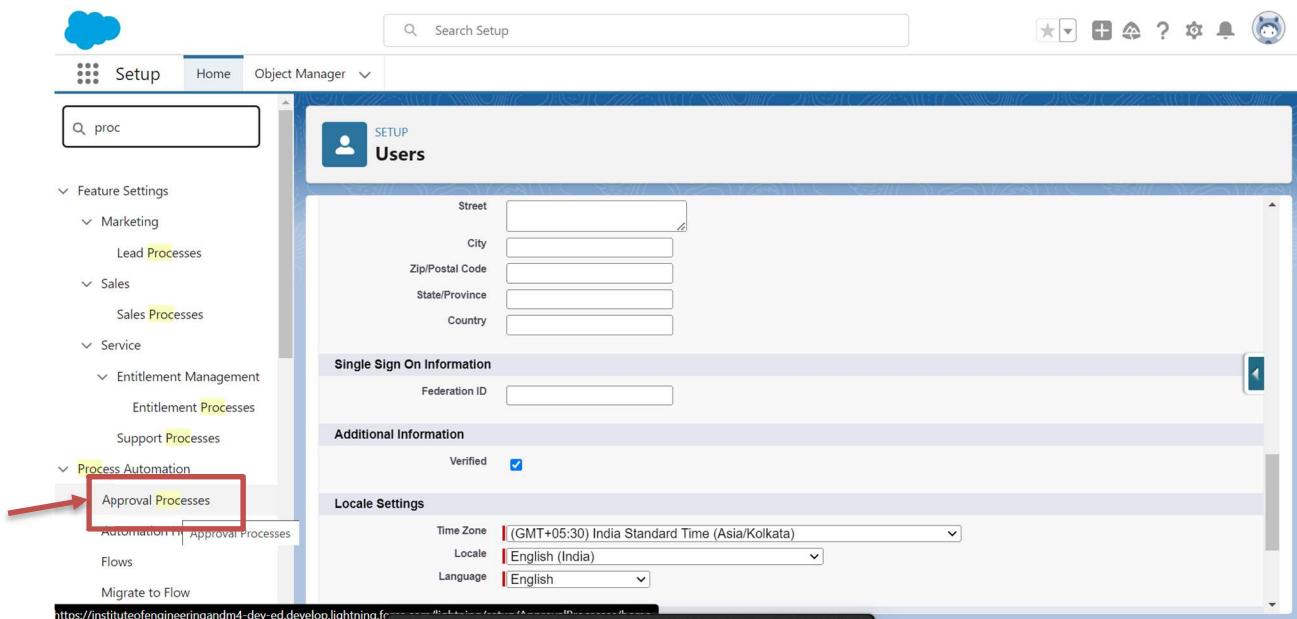
Time Zone: (GMT+05:30) India Standard Time (Asia/Kolkata)
Locale: English (India)
Language: English



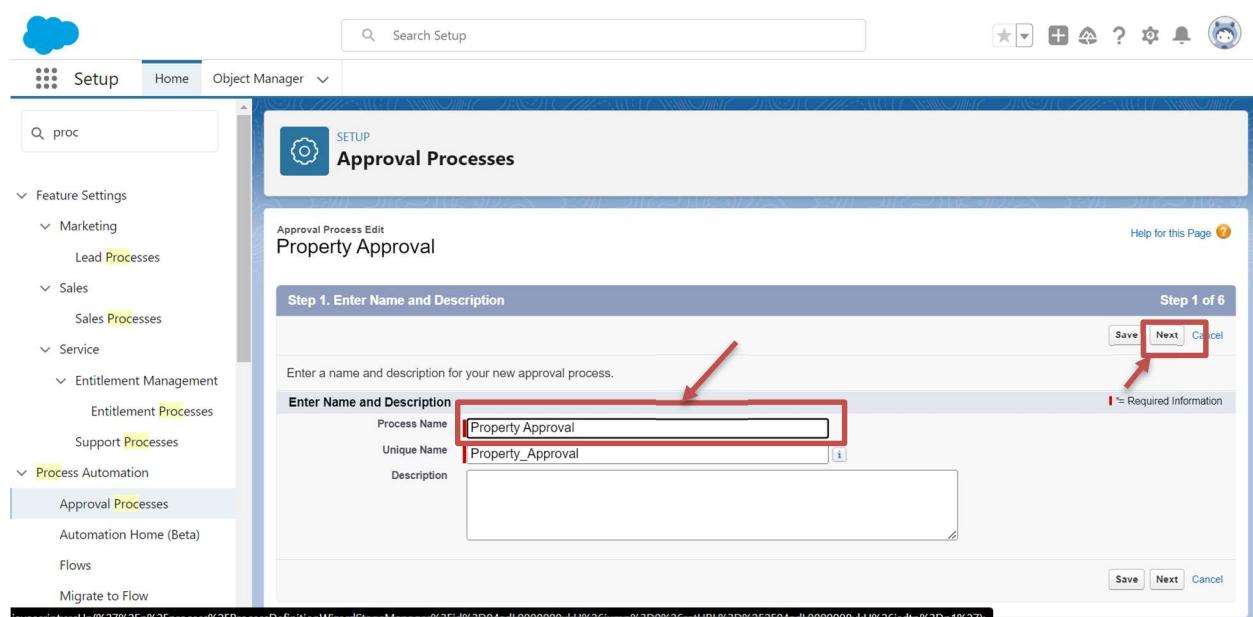
7. Save

Activity 8: Create An Approval Process for Property Object

- From Setup search Process Automation and select Approval Process



- Process Name - Property Approval



- Give 2 criteria -

- a. Location is not equal to blank,
- b. Verified Equals false.

Step 2. Specify Entry Criteria

If only certain types of records should enter this approval process, enter that criteria below. For example, only expense reports from employees at headquarters should use this approval process.

Specify Entry Criteria

Use this approval process if the following criteria are met :

Field	Operator	Value	Logic
Property: Location	not equal to	Blank	AND
Property: Verified	equals	False	AND
--None--	--None--		AND
--None--	--None--		AND
--None--	--None--		AND

[Add Filter Logic...](#)

Step 2 of 6

Previous Save **Next** Cancel

4. Click next and “Next Automated Approver Determined By” Select Manager
5. From Record Editability Properties click on Administrators OR the currently assigned approver can edit records during the approval process.

Property Approval

Step 3. Specify Approver Field and Record Editability Properties

When you define approval steps, you can assign approval requests to different users. One of your options is to use a user field to automatically route these requests. If you want to use this option for any of your approval steps, select a field from the picklist below. Also, when a record is in the approval process, it will always be locked-- only an administrator will be able to edit it. However, you may choose to also allow the currently assigned approver to edit the record.

Select Field Used for Automated Approval Routing

Next Automated Approver Determined By **Manager**

Use Approver Field of Property Owner

Record Editability Properties

Administrators ONLY can edit records during the approval process.
 Administrators OR the currently assigned approver can edit records during the approval process.

Step 3 of 6

Previous Save **Next** Cancel

6. From Step 5. Select Fields to Display on Approval Page Layout select Property, Owner, Location, Type.

The screenshot shows the 'Approval Processes' setup page under the 'SETUP' tab. On the left, a list of 'Available Fields' includes 'Created By', 'Last Modified By', 'Property Name', and 'Verified'. On the right, a list of 'Selected Fields' contains 'Property Owner', 'Location', and 'Type', which are highlighted with a red box. Below these lists are 'Add' and 'Remove' buttons, and 'Up' and 'Down' navigation buttons. To the right of the lists is a preview window showing a report titled 'Executive Report EX001-0001' with various data tables. At the bottom right of the main area is a link: 'Click here to view an example'.

7. Click Next and Select the Initial Submitters
 - a. Owner - Property Owner
 - b. Roles - Sales Manager

The screenshot shows the 'Initial Submitters' selection screen. At the top, there is a search bar with 'Submitter Type: Search: Owner' and a 'Find' button. Below the search bar are two sections: 'Available Submitters' and 'Allowed Submitters'. The 'Available Submitters' section shows a dropdown menu with '--None--'. The 'Allowed Submitters' section shows two items: 'Role: Sales Manager' and 'Property Owner', which are highlighted with a red box. An arrow points from the 'Property Owner' item towards the 'Allowed Submitters' section.

8. Save.
9. Add an approval step name "Executive Approval"

Executive Approval

Step 1. Enter Name and Description Step 1 of 3

Enter a name, description, and step number for your new approval step.

Enter Name and Description

Approval Process Name: Property_Approval

Name: Executive Approval

Unique Name: Executive_Appr

Description: Executive Approval

Save Next Cancel

NodeWizardSt... [redacted]

10. Specify the Criteria as All record should enter

Executive Approval

Step 2. Specify Step Criteria Step 2 of 3

Specify whether a record must meet certain criteria before entering this approval step. If these criteria are not met, the approval process can skip to the next step, if one exists.

Learn more

Specify Step Criteria

All records should enter this step.

Enter this step if the following criteria are met [dropdown], else [dropdown]:

Previous Save Next Cancel

11. Click next and select the Approver as "Sales Executive "and "Save"

Specify the user who should approve records that enter this step. Optionally, choose whether the approver's delegate is also allowed to approve these requests.

Select Approver

Let the submitter choose the approver manually.

Automatically assign using the user field selected earlier. (Manager)

Automatically assign to queue: [dropdown] [button]

Automatically assign to approver(s).

User: Executive

When multiple approvers are selected:

Approve or reject based on the FIRST response.

Require UNANIMOUS approval from all selected approvers.

The approver's delegate may also approve this request. [button]

Previous Save Cancel

12. Add One field Update as "Verified Property"

Search Setup

Home Object Manager

Process Automation

Workflow Actions

Field Updates

Didn't find what you're looking for?
Try using Global Search.

Approval Processes

Property: Property Approval

« Back to Approval Process List

Process Definition Detail

Process Name	Property Approval	Active	✓
Unique Name	Property_Approval	Next Automated Approver	Determined By Manager of Record Submitter
Description			
Entry Criteria	(Property: Location NOTEQUAL TO Blank) AND (Property: Verified EQUALS False)		
Record Editability	Administrator OR Current Approver	Allow Submitters to Recall Approval Requests	<input type="checkbox"/>
Approval Assignment Email Template			
Initial Submitters	Role_Sales Manager, Property Owner		
Created By	Jyoti Gupta	Created On	24/06/2024, 2:27 pm
Modified By	Jyoti Gupta	Modified On	28/06/2024, 3:14 pm

Initial Submission Actions

Add Existing Add New

Action Type	Description
-------------	-------------

- a. Select Object - Property
- b. Field to Update - Verified
- c. Field Data Type - Checkbox
- d. Select Checkbox Option as "True"
- e. Save.

Field Updates

Identification

Name	Verified Property
Unique Name	Verified_Property
Description	
Object	Property
Field to Update	Property: Verified
Field Data Type	Checkbox
Re-evaluate Workflow Rules after Field Change	<input type="checkbox"/>

Specify New Field Value

Checkbox Options
<input checked="" type="radio"/> True
<input type="radio"/> False

Save Save & New Cancel

13. Add One field Update as "UnVerified Property"

- a. Select Object - Property

- b. Field to Update - Verified
- c. Field Data Type - Checkbox
- d. Select Checkbox Option as "False"
- e. Save.

SETUP

Field Updates

Identification

Name	UnVerified Property
Unique Name	UnVerified_Property
Description	
Object	Property
Field to Update	Property: Verified
Field Data Type	Checkbox
Re-evaluate Workflow Rules after Field Change	<input type="checkbox"/> i

Specify New Field Value

Checkbox Options	
<input type="radio"/> True	<input checked="" type="radio"/> False

Buttons: Save [Save & New](#) [Cancel](#)

14. Activate the Approval Process.

SETUP

Approval Processes

Property: Property Approval

Process Definition Detail

Process Name	Property Approval
Unique Name	Property_Approval
Description	
Entry Criteria	(Property: Location NOTEQUALTO Blank) AND (Property: Verified EQUALS False)
Record Editability	Administrator OR Current Approver
Approval Assignment Email Template	
Initial Submitters	Role_Sales Manager, Property Owner
Created By	Jyoti_Gupta, 24/06/2024, 2:27 pm
Modified By	Jyoti_Gupta, 28/06/2024, 3:18 pm

Initial Submission Actions

Action	Type	Description
Add Existing		
Add New		

Activity 9: Create A Record Trigger Flow to Submit the Approval Process Automatically

1. From Setup Search for Flows then Click on New and Select “Record Trigger Flow”.

The screenshot shows the Salesforce Setup interface. On the left, the navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar at the top says 'Search Setup'. Below it, a sidebar shows categories like 'Process Automation' (with 'Flows' highlighted), 'Identity', and 'Login Flows'. A red arrow points to the 'Flows' link in the sidebar. The main content area is titled 'SETUP Approval Processes' for 'Property: Property Approval'. It displays process details such as 'Process Name: Property Approval', 'Unique Name: Property_Approval', and 'Active: ✓'. Under 'Initial Submission Actions', there are buttons for 'Add Existing' and 'Add New'. A red box highlights the 'New' button.

The screenshot shows the Salesforce Setup interface. The navigation bar includes 'Setup', 'Home', and 'Object Manager'. A search bar at the top says 'Search Setup'. Below it, a sidebar shows categories like 'Process Automation' (with 'Flows' highlighted), 'Identity', and 'Login Flows'. A red arrow points to the 'Flows' link in the sidebar. The main content area is titled 'SETUP Flows'. It features a section 'Try the Automation Lightning App!' with a cartoon squirrel icon. A red box highlights the 'New Flow' button. Below it, a table titled 'Flow Definitions' lists items such as 'Add or Modify Service Appoi...', 'Basic Approval Request', and 'Book Appointment from Inv...'. The table has columns for 'Flow Label', 'Process Type', 'Package State', and 'Last Modified'.

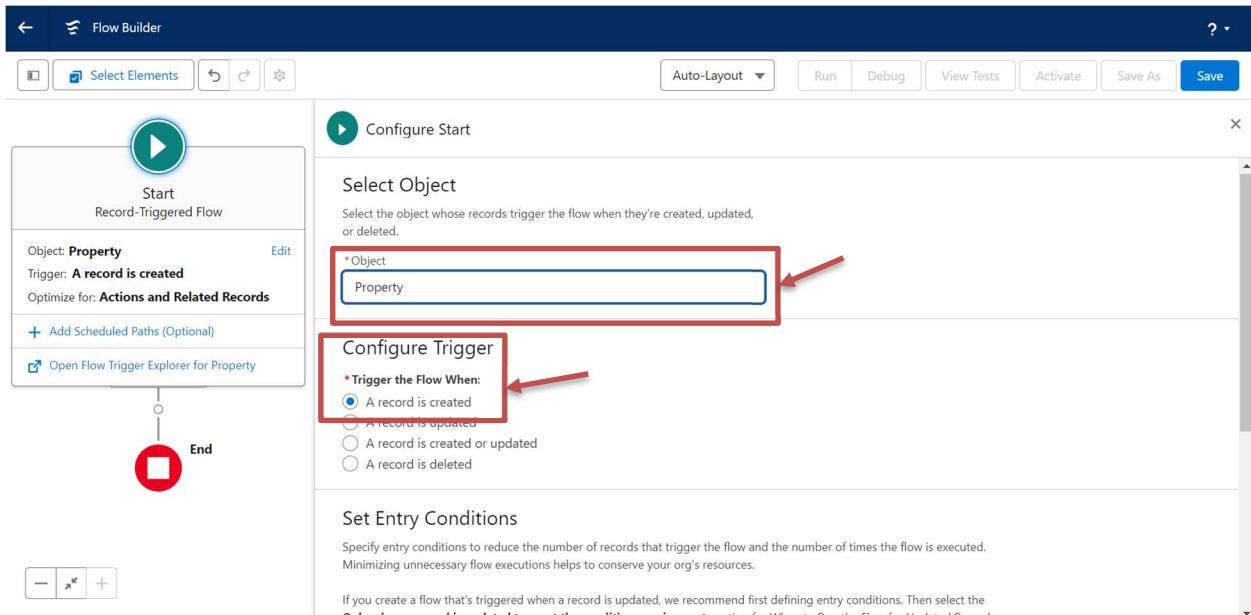
Select Type

Recommended

 Screen Flow Guides users through a business process that's launched from Lightning pages, Experience Cloud sites, quick actions, and more.	 Record-Triggered Flow Launches when a record is created, updated, or deleted. This autolaunched flow runs in the background.
 Schedule-Triggered Flow Launches at a specified time and frequency for each record in a batch. This autolaunched flow runs in the background.	 Platform Event-triggered Flow Launches when a platform event message is received. This autolaunched flow runs in the background.
 Autolaunched Flow (No Trigger) Launches when invoked by Apex, processes, REST API, and more. This autolaunched flow runs in the background.	 Record-Triggered Orchestration Launches when a record is created or updated. An orchestration lets you create a multi-step, multi-user process.

[Back](#) [Create](#)

2. Select Object as Property
3. Select "Trigger the flow when" - "A record is created"
4. Set Entry Conditions - "None"
5. Add a "Action" - "Submit for Approval"



The screenshot shows the Flow Builder interface with a modal window open for "Configure Start".

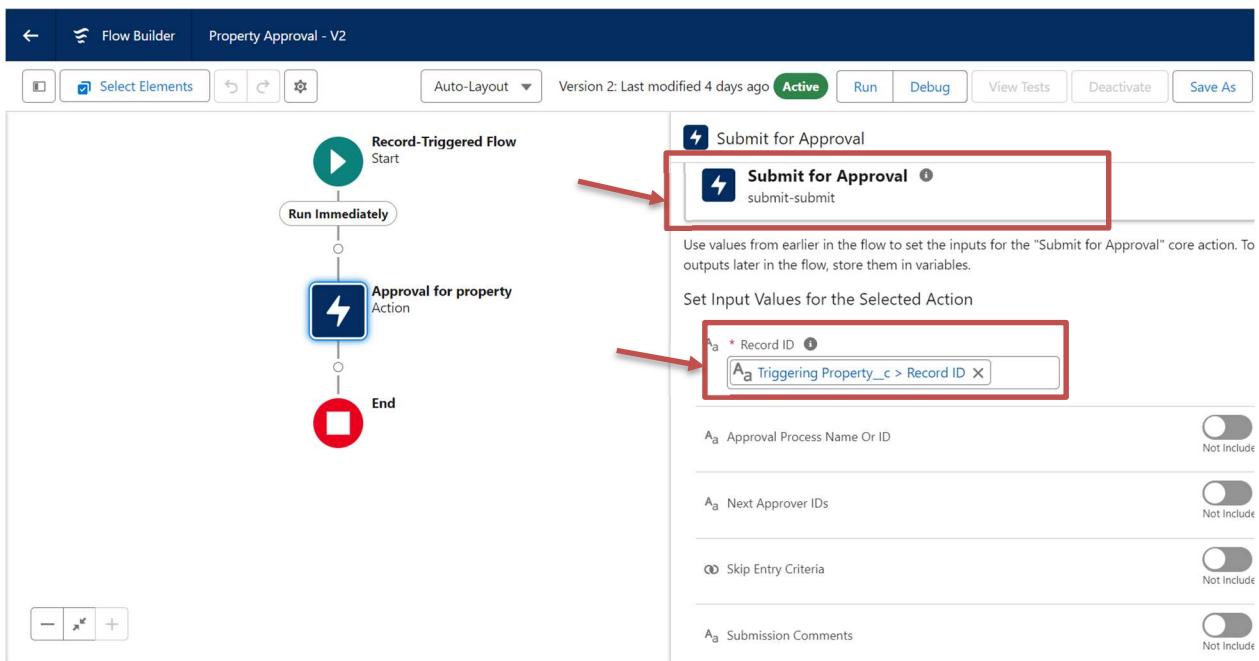
Flow Details:

- Object: **Property**
- Trigger: **A record is created**
- Optimize for: **Actions and Related Records**

Configure Start Modal:

- Select Object:** The input field contains "Property", which is highlighted with a red box and has a red arrow pointing to it.
- Configure Trigger:** The "Trigger the Flow When:" section is expanded, showing the radio button for "A record is created" selected, which is highlighted with a red box and has a red arrow pointing to it.
- Set Entry Conditions:** A note states: "If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the Only when a record is updated to meet the condition requirements option for When to Run the Flow for Updated Records."

6. Give Label - Approval for property
7. Record Id - `{!$Record.Id}`



8. Done.

9. Save the Flow and Give label as "Property Approval" and "Activate"

Save as

A New Version A New Flow

* Flow Label: Property Approval

* Flow API Name: Approval_for_property

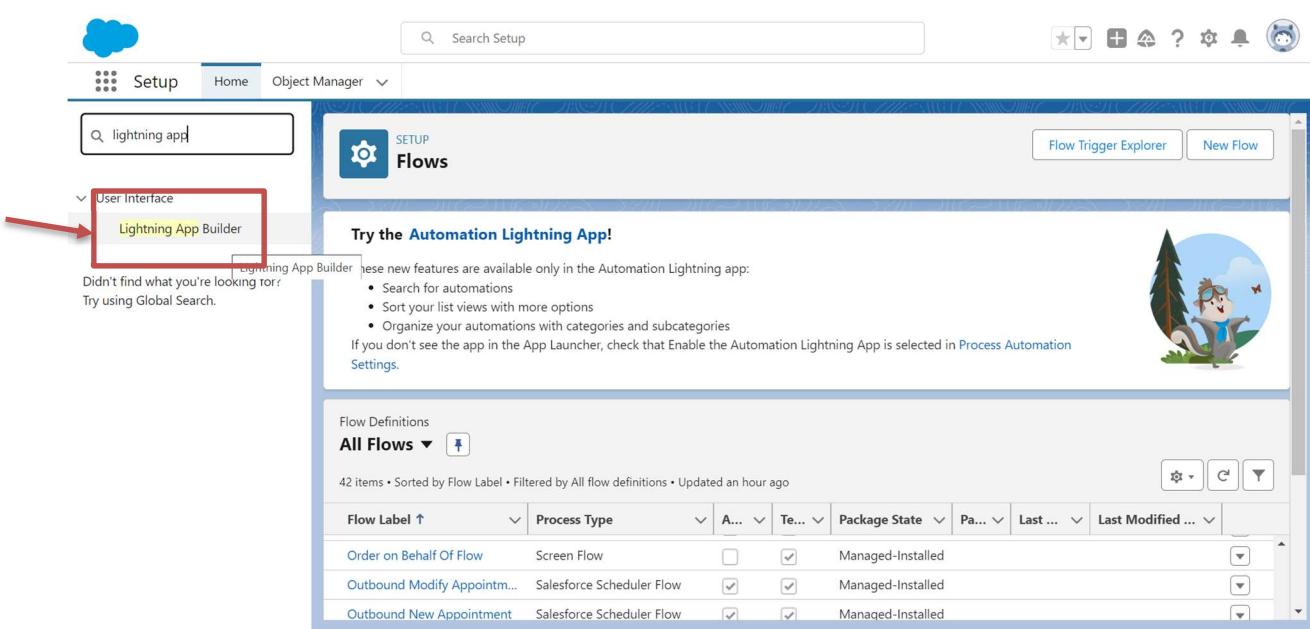
Description: Approval for property

Show Advanced

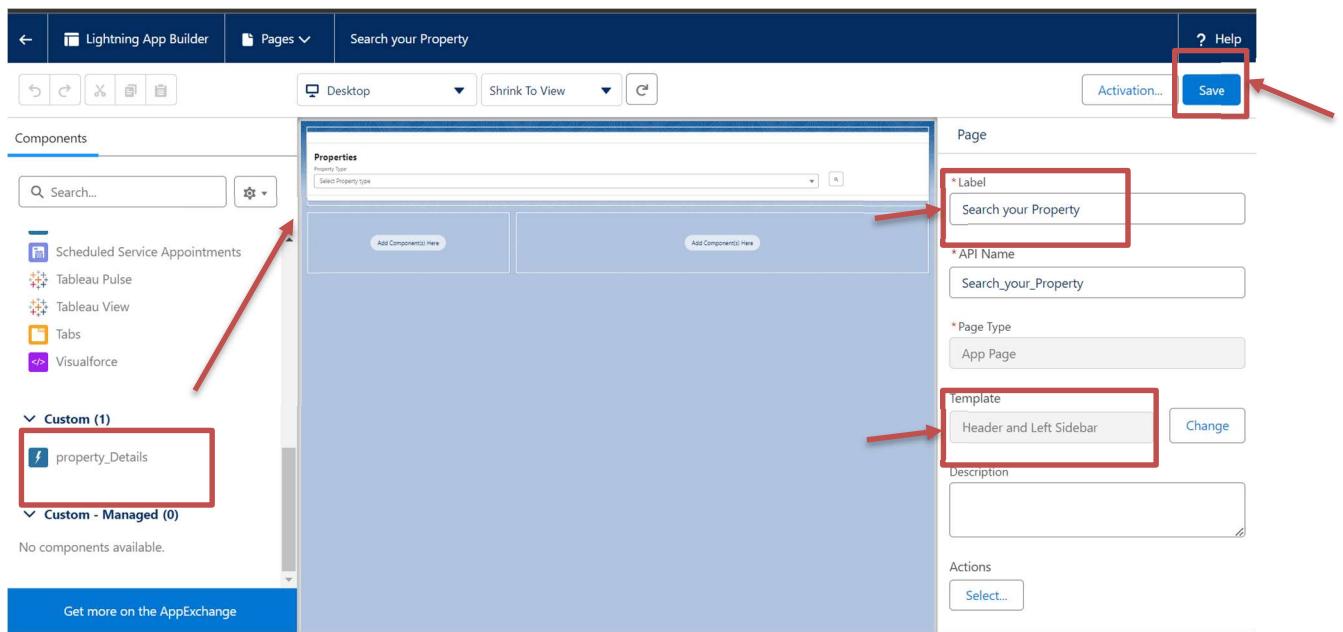
Cancel Save

Activity 10: Create An App Page

- From Setup go to Lightning App Builder Then Click on New and Select App Page then Click on Next.



2. Give Label as “Search your Property” click “Next”.
3. Click “Header and Left Sidebar” and click on “Done”
4. Click on “Save” and then click on “Activate”.
5. From Page Setting select page activation as “Activate for all Users”.
6. From Lightning Experience Click on “Property Details” and click on Add Page “.”
7. Then Click on “Save”



Activity 11: Create A LWC Component

Create an LWC Component for the customers so that only verified customers can access the verified properties and non-Verified customers can access non verified properties, and deploy it on “Search your Property Page”.

1. Create an Apex Class and make it aura enabled and name it “PropertHandler_LWC”

Code: -

```
public class PropertHandler_LWC{
    @AuraEnabled(cacheable=true)
    public static list<Property__c> getProperty(string type , boolean verified){
        return [SELECT Id, Location__c, Property_Name__c, Type__c, Verified__c FROM
Property__c Where Type__c =: type AND Verified__c =: verified];
    }
}
```

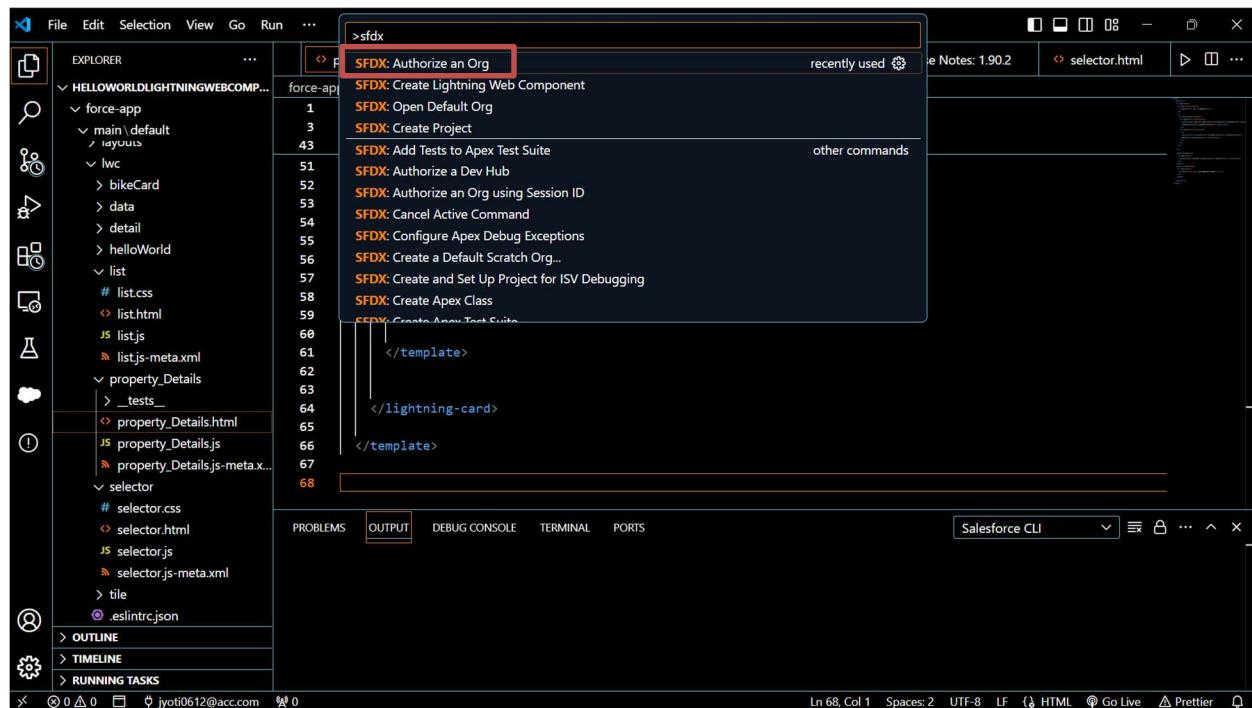
```

public class PropertHandler_LWC{
    @AuraEnabled(cacheable=true)
    public static list<Property__c> getProperty(string type , boolean verified){
        return [SELECT Id, Location__c, Property_Name__c, Type__c, Verified__c FROM Property__c
                Where Type__c =: type AND Verified__c =: verified];
    }
}

```

The screenshot shows the Salesforce Dev Console interface. At the top, there's a header bar with tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. Below the header is a toolbar with buttons for User, Application, Operation, Time, Status, Read, and Size. The main area displays the code for the PropertHandler_LWC class.

2. Create a Lightning Web Component in your VsCode, and (ctrl+shift +P) and click on authorize an org.



3. Enter your login id and password to authorize your org.
4. Now (ctrl+shift +P) and Create a lightning Web Component and Name it Anything you want to.
5. In your Html File Write this code :-

Code :-

```
<template>
  <lightning-card>
    <div class="slds-box">
      <div class="slds-text-align_left">
        <h1 style="font-size: 20px;"><b>Properties</b></h1>
      </div>
      <div>
        <div class="slds-grid slds-gutters">
          <div class="slds-col slds-size_5-of-6">
            <lightning-combobox name="Type" label="Property Type" value={typevar}
placeholder="Select Property type"
              options={propertyoptions}          onchange={changehandler}></lightning-
combobox>
          </div>
          <div class="slds-col slds-size_1-of-6">
            <br>
            <lightning-button-icon    variant="neutral"    icon-name="standard:search"
alternative-text="Search"
              label="Search" onclick={handleClick}></lightning-button-icon>
          </div>
        </div>
      </div>
    </div>
    <template if:true={istrue}>
      <div class="slds-box">
        <lightning-datable key-field="id" data={propertylist} columns={columns}>
        </lightning-datable>
      </div>
    </template>
    <template if:false={isfalse}>
      <div class="slds-box">
        <div style="font-size: 15px;"><b>No properties Are Found !!</b></div>
      </div>
    </template>
  </lightning-card>
</template>
```

```

1 <template>
2   <lightning-card>
3     <div class="slds-box">
4       <div class="slds-text-align_left">
5         <h1 style="font-size: 20px;"><b>Properties</b></h1>
6       </div>
7     </div>
8     <div class="slds-grid slds-gutters">
9       <div class="slds-col slds-size_5-of-6">
10      <lightning-combobox name="Type" label="Property Type" value={typevar} placeholder="Select Property type"
11        options={propertyoptions} onchange={changehandler}></lightning-combobox>
12      </div>
13      <div class="slds-col slds-size_1-of-6">
14        <br>
15        <lightning-button-icon variant="neutral" icon-name="standard:search" alternative-text="Search"
16          label="Search" onclick={handleClick}></lightning-button-icon>
17      </div>
18    </div>
19  </lightning-card>
20  </div>
21 </div>
22 </div>
23 </div>
24 </div>
25 </div>
26 </div>
27 </div>
28 </div>
29 </div>
30 </div>
31 </div>
32 </div>

```

6. In Your Js File Write this code :-

Code :-

```

import { LightningElement, api, track, wire } from 'lwc';
import getProperty from "@salesforce/apex/PropertHandler_LWC.getProperty"
import { getRecord } from 'lightning/uiRecordApi';
import USER_ID from '@salesforce/user/Id';
export default class C_01_Property_Management extends LightningElement {
  @api recordId
  userId = USER_ID;
  verifiedvar
  typevar
  isfalse = true;
  istrue = false;
  @track propertylist = [];
  columns = [
    { label: 'Property Name', fieldName: 'Property_Name__c' },
    { label: 'Property Type', fieldName: 'Type__c' },
    { label: 'Property Location', fieldName: 'Location__c' },
    { label: "Property link", fieldName: "Property_link__c" }
  ]
  propertyoptions = [
    { label: "Commercial", value: "Commercial" },
    { label: "Residential", value: "Residential" },
    { label: "rental", value: "rental" }
  ]
}

```

```
]

@wire(getRecord, { recordId: "$userId", fields: ['User.Verified__c'] })
recordFunction({ data, error }) {
    if (data) {
        console.log(data)
        console.log("This is the User Id ---> "+this.userId);
        this.verifiedvar = data.fields.Verified__c.value;
    } else {
        console.error(error)
        console.log('this is error')
    }
}
changehandler(event) {
    console.log(event.target.value);
    this.typevar = event.target.value;
}
handleClick() {
    getProperty({ type: this.typevar, verified: this.verifiedvar })
        .then((result) => {
            this.isfalse = true;
            console.log(result)
            console.log('This is the User id ---> ' + this.userId);
            console.log('This is the verified values ---> ' + this.verifiedvar);
            if (result != null && result.length != 0) {
                this.isTrue = true;
                this.propertylist = result;
                console.log(this.verifiedvar);
                console.log(this.typevar)
            } else {
                this.isfalse = false;
                this.isTrue = false;
            }
        })
        .catch((error) => {
            console.log(error)
        })
}
```

```

    }
}

1 import { LightningElement, api, track, wire } from 'lwc';
2
3 import getProperty from "@salesforce/apex/PropertyHandler_LWC.getProperty"
4
5 import { getRecord } from 'lightning/uiRecordApi';
6
7 import USER_ID from '@salesforce/user/Id';
8
9 export default class C_01_Property_Management extends LightningElement {
10
11     @api recordId
12
13     userId = USER_ID;
14
15     verifiedvar
16
17     typevar
18
19     isfalse = true;
20
21     istrue = false;
22
23     @track propertylist = [];
24
25     columns = [
26
27         { label: 'Property Name', fieldName: 'Property_Name__c' },
28
29         { label: 'Property Type', fieldName: 'Type__c' },
30
31         { label: 'Property Location', fieldName: 'Location__c' },
32

```

7. In Your metafile give your targets to deploy the component.

Code :-

```

<?xml version="1.0" encoding="UTF-8"?>
<LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">
    <apiVersion>59.0</apiVersion>
    <isExposed>true</isExposed>
    <targets>
        <target>lightning__RecordPage</target>
        <target>lightning__AppPage</target>
        <target>lightning__HomePage</target>
    </targets>
</LightningComponentBundle>

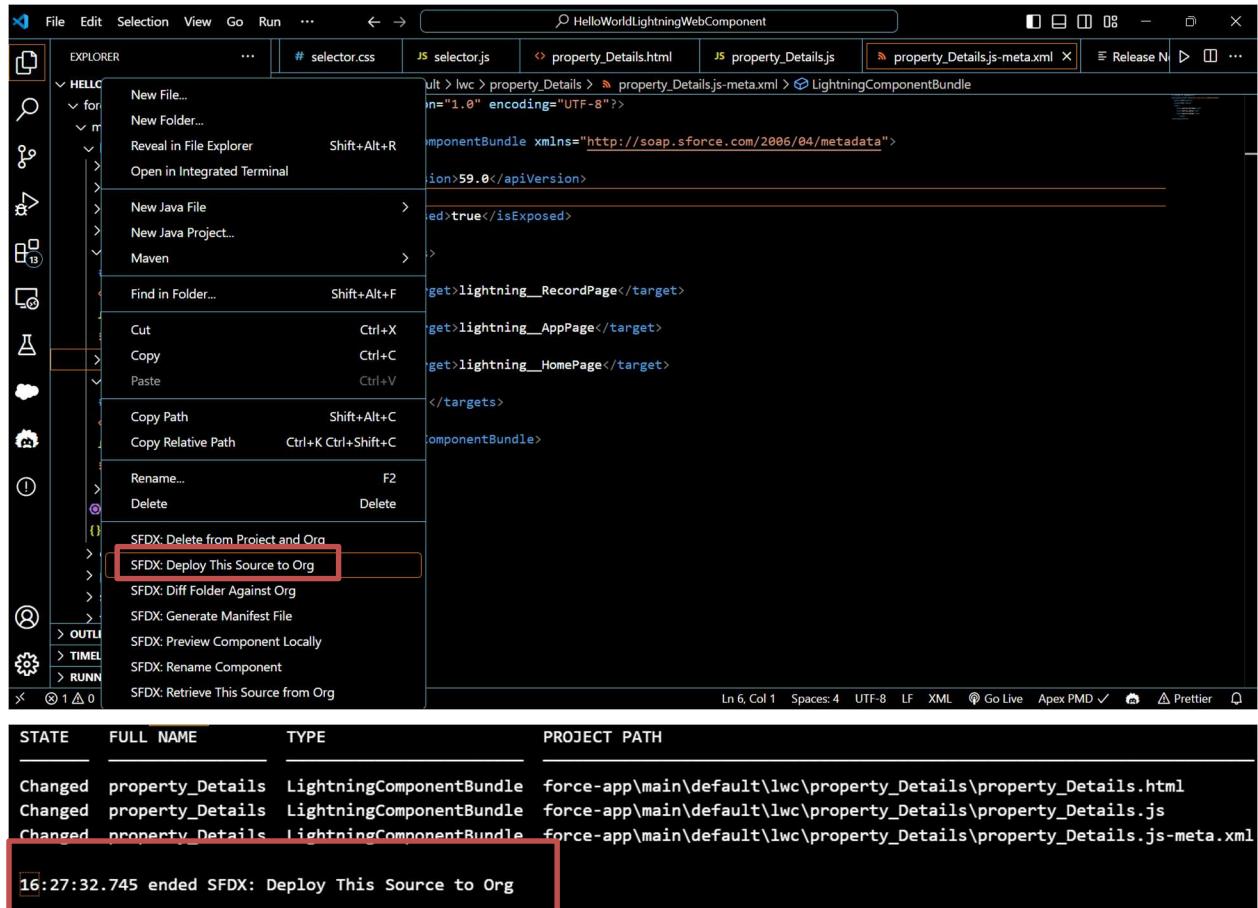
```

```

1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">
4
5     <apiVersion>59.0</apiVersion>
6
7     <isExposed>true</isExposed>
8
9     <targets>
10
11         <target>lightning__RecordPage</target>
12
13         <target>lightning__AppPage</target>
14
15         <target>lightning__HomePage</target>
16
17     </targets>
18
19 </LightningComponentBundle>

```

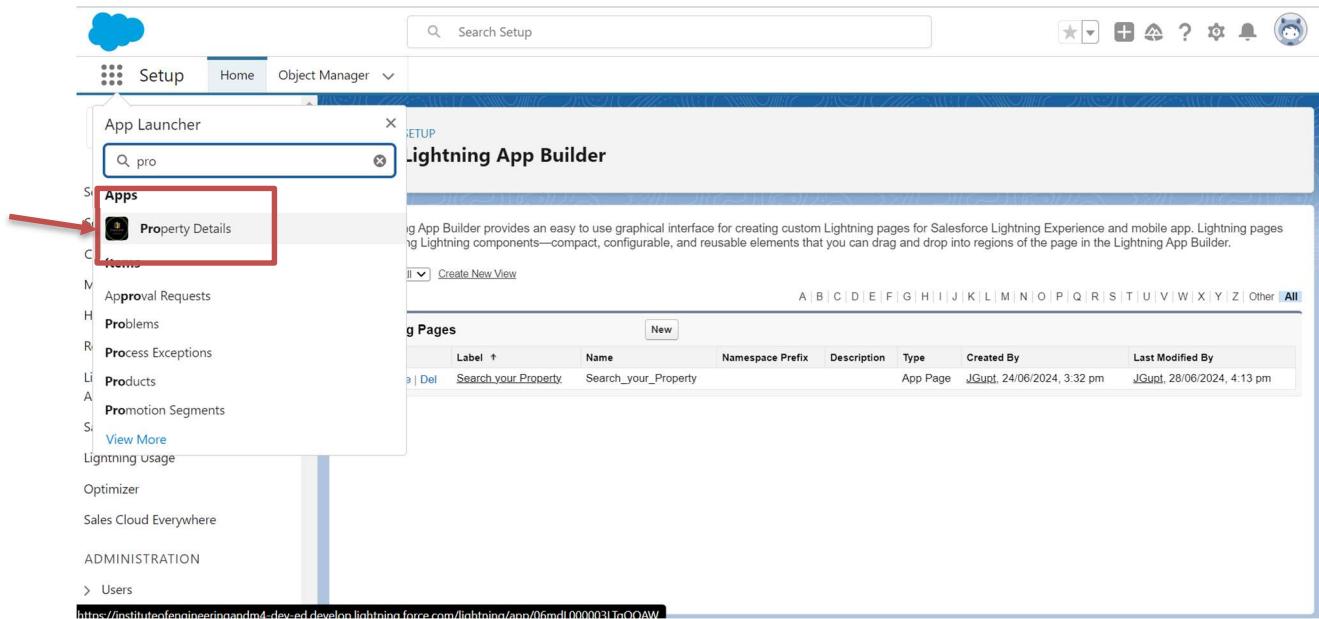
8. After Saving all the three Codes , Right Click and deploy this component to the org.



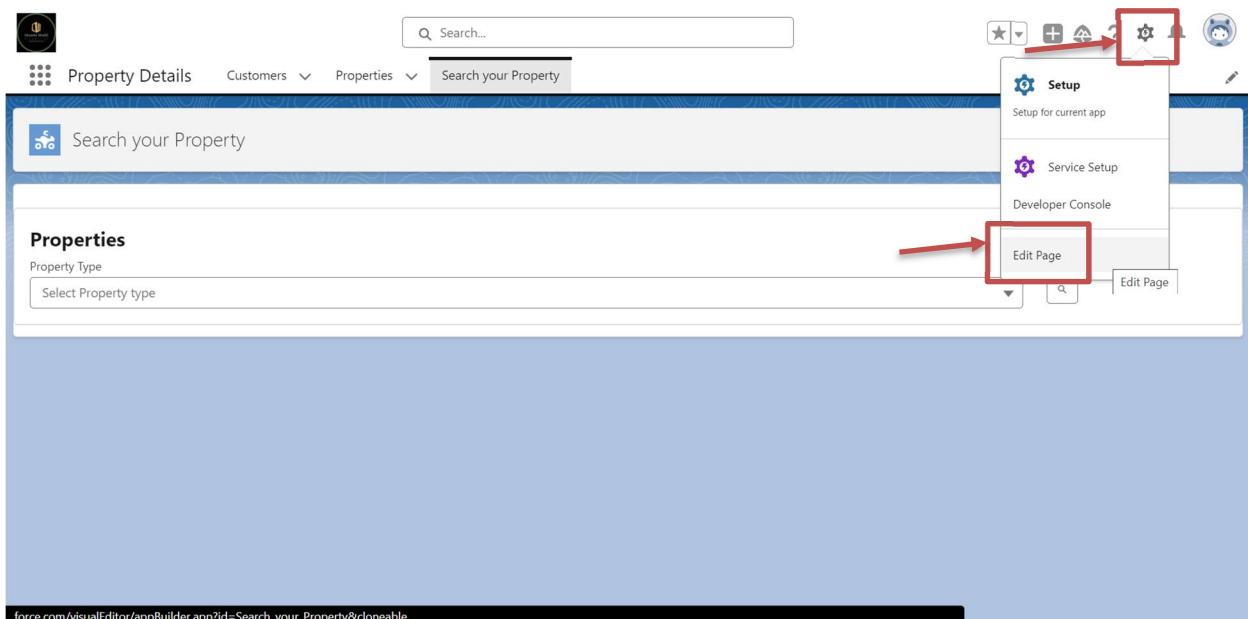
Activity 12: Drag This Component To Your App Page

Adding the Component to your Page

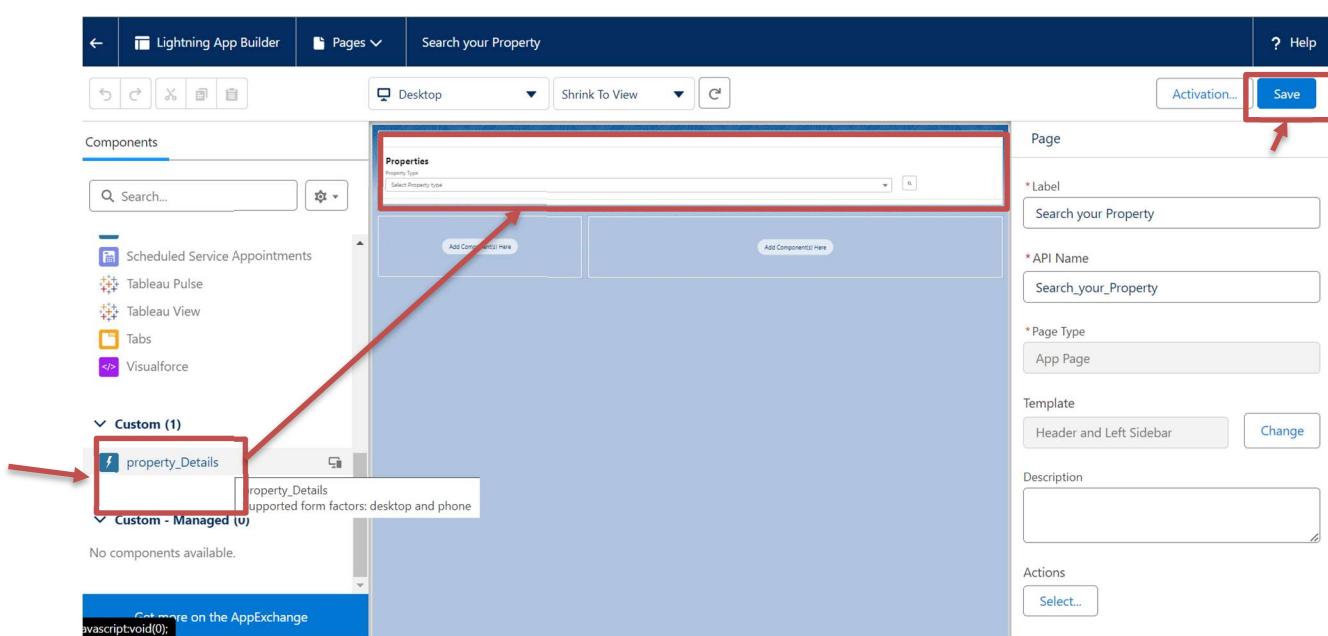
1. From Setup go to App Launcher and select Search for Property Details



2. On this Page click on gear icon and click on Edit Page

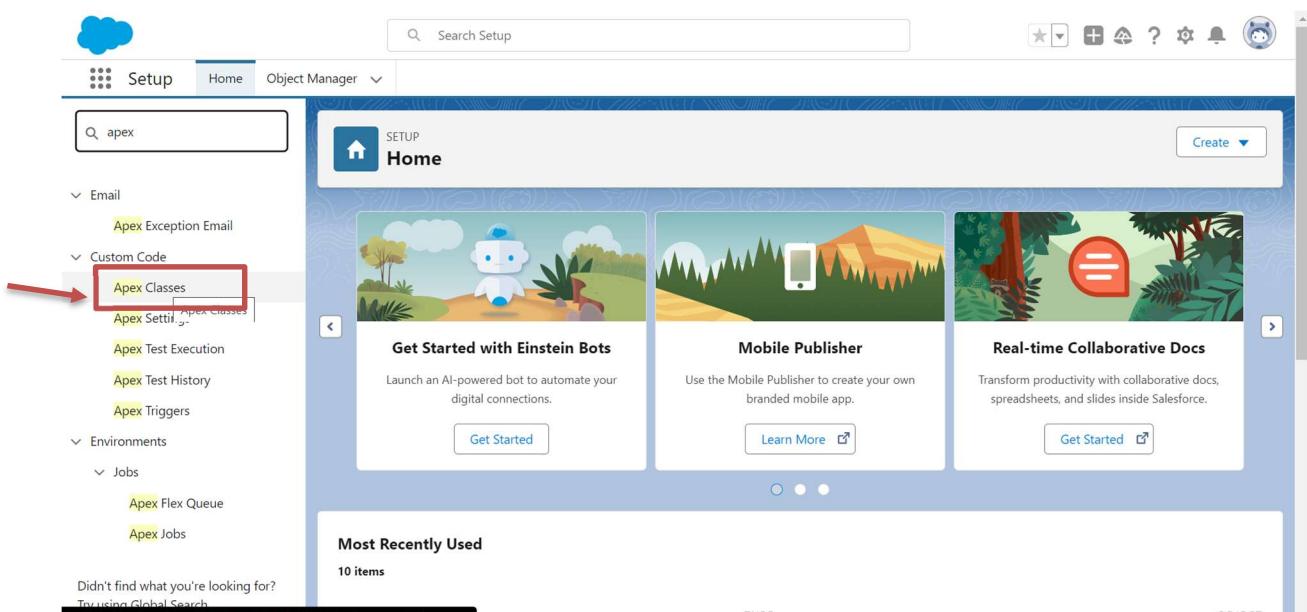


3. Drag the Component to your App Page and Save the Page.



Activity 13: Give Access of Apex Classes to Profiles

- From Setup search for Apex Classes and click on "Security" behind "PropertyHandler__LWC".



Percent of Apex Used: 0.01%
You are currently using 303 characters of Apex Code (excluding comments and @isTest annotated classes) in your organization, out of an allowed limit of 6,000,000 characters. Note that the amount in use includes both Apex Classes and Triggers defined in your organization.

Estimate your organization's code coverage [i](#)
Compile all classes [i](#)

View: All [Create New View](#)

Action	Name ↑	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By	Has Trace Flags
Edit Delete Security	PropertHandler_LWC		61.0	Active	303	Jyoti Gupta, 24/06/2024, 3:37 pm	<input type="checkbox"/>

Security - Record 1 - PropertHandler_LWC

Dynamic Apex Classes

Dynamic Apex extends your programming reach by interacting with Lightning Platform components.

View: All [Create New View](#)

Class Name ↑	Namespace Prefix	Api Version	Created By	Last Modified By
No records to display.				

profile%2FApexClassProfilePermissionEdit%2Fe%3Fapex_id%3D01pdl000003dbpC%26apex_name%3DPropertHandler_LWC%26apex_namespace%3D%26save_button%3Dhas_save_but...

2. From Profiles Add “Manager” and “Customer” and “Save”.

Enable Profile Access for Apex Class
PropertHandler_LWC

Available Profiles		Enabled Profiles	
Analytics Cloud Integration User	Add	Customer	Remove
Analytics Cloud Security User		Manager	
Authenticated Website		System Administrator	
B2B Reordering Portal Buyer Profile			
Contract Manager			
Cross Org Data Proxy User			
Custom: Marketing Profile			
Custom: Sales Profile			
Custom: Support Profile			
Customer Community Login User			
Customer Community Plus Login User			
Customer Community Plus User			
Customer Community User			
Customer Portal Manager Custom			