

LEASE MANAGEMENT

LEASE MANAGEMENT

COLLEGE NAME:GAYATRI DEGREE COLLEGE -TIRUPATI

TEAM ID :LTVIP2025TMID29567

NAME	ROLE	E-MAIL ID
NAGARADONA HARSHITHA	TEAM LEADER	harshithan0048@gmail.com
KOTAKONDA RENUKA	TEAM MEMBER	kotakondarenukasuren05@gmail.com
CHINTAPARTHI HEMA	TEAM MEMBER	hemachintaparthi367@gmail.com
KONDETI TEJASREE	TEAM MEMEBER	tejasreekondeti25@gmail.com

LEASE MANAGEMENT

Introduction:

Are you new to Salesforce? Not sure exactly what it is, or how to use it? Don't know where you should start on your learning journey? If you've answered yes to any of these questions, then you're in the right place. This module is for you.

Welcome to Salesforce! Salesforce is game-changing technology, within a host of productivity-boosting features, that will help you sell smarter and faster. As you work toward your badge for this module, we'll take you through these features and answer the question, "What is Salesforce, anyway?"

What Is Salesforce?

Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers.

Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud.

So what does that really mean? Well, before Salesforce, your contacts, emails, follow-up tasks, and prospective deals might have been organized something like this:

<https://youtu.be/r9EX3lGde5k>

Milestone-1

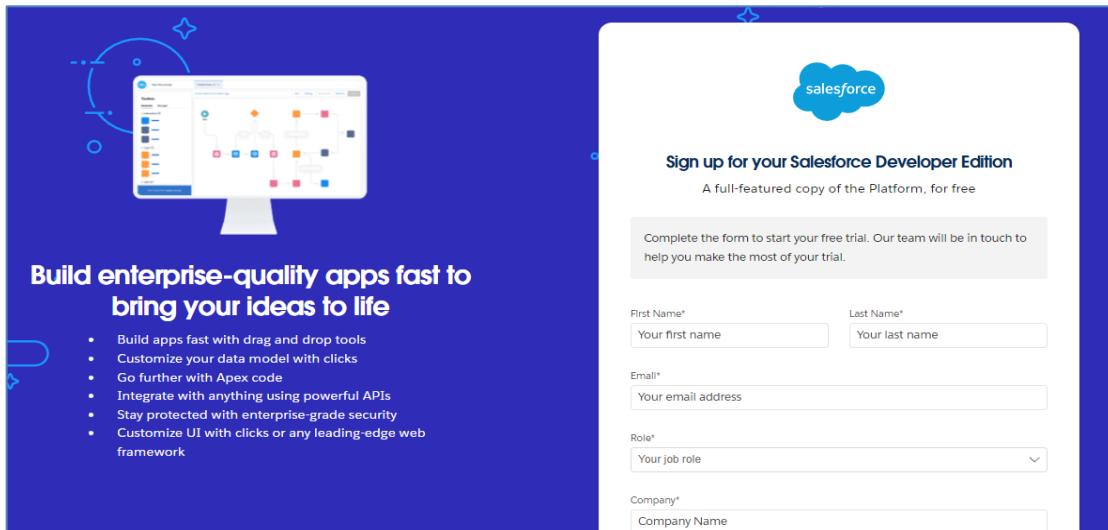
Salesforce Account

Activity – 1 Creating Developer Account:

LEASE MANAGEMENT

Creating a developer org in salesforce.

1. Go to <https://developer.salesforce.com/signup>
2. On the sign-up form, enter the following details:



First Name : NAGARADONA

Last Name : HARSHITHA

Email : harshithan0048@gmail.com

Role : Developer

Company : Gayatri Degree College Tirupati

Country : India

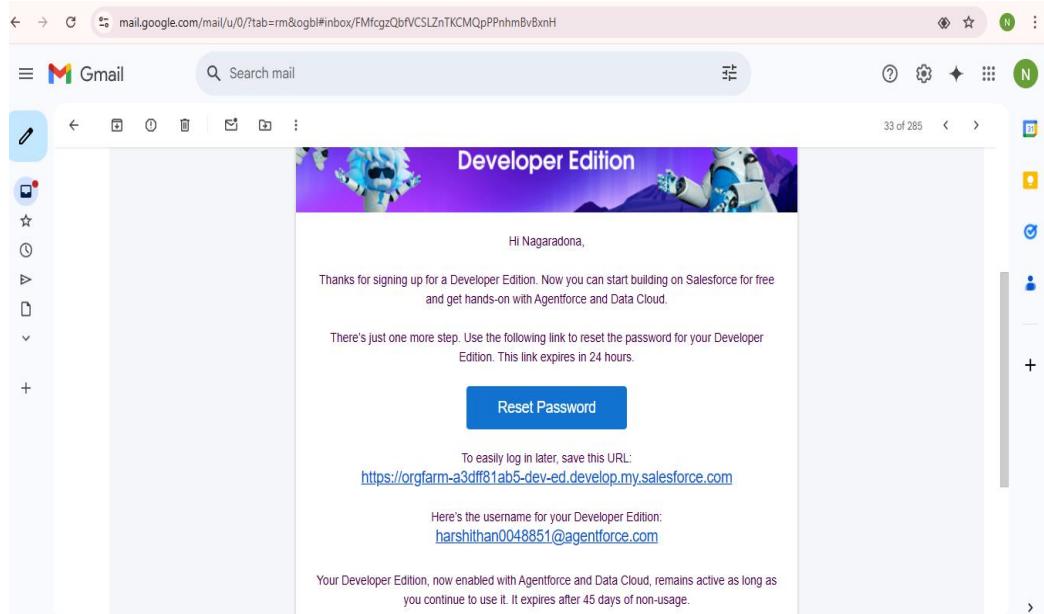
User name : harshithan0048851@agentforce.com

Activity 2: Account Activation:

Account Activation:

1. Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.

LEASE MANAGEMENT

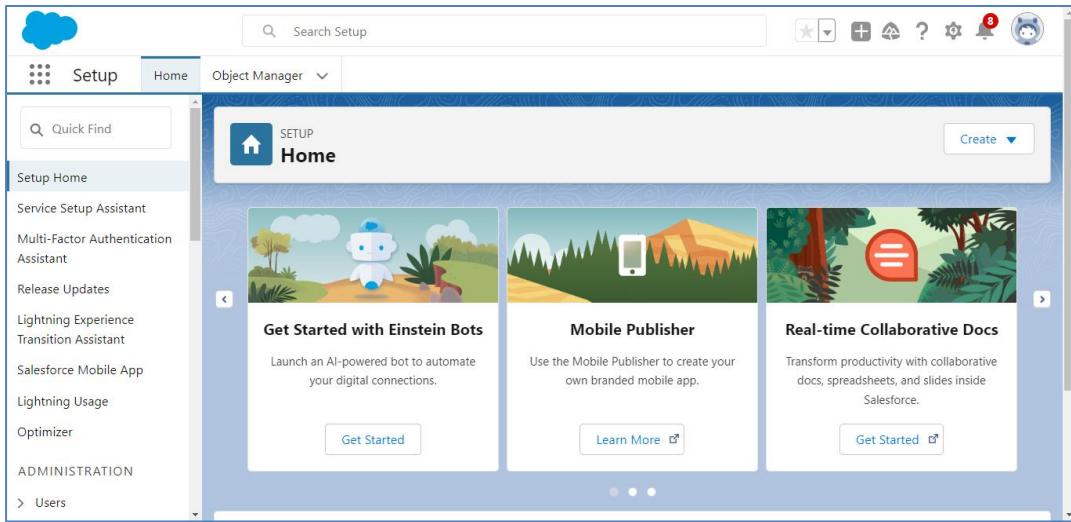


2. Click on Verify Account
3. Give a password and answer a security question and click on change password.

A screenshot of a "Change Your Password" form. The form asks for a new password that includes at least 8 characters, 1 letter, and 1 number. It has fields for "New Password" and "Confirm New Password", both of which are highlighted with a red border. Below these are fields for "Security Question" (set to "In what city were you born?") and "Answer" (containing "asdfghjkl"). The entire form is enclosed in a red border, and a blue "Change Password" button is at the bottom.

4. Then you will redirect to your salesforce setup page.

LEASE MANAGEMENT



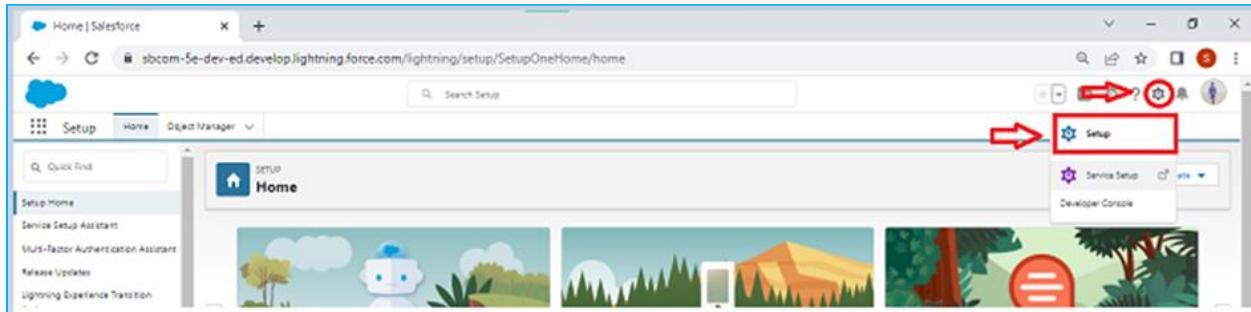
LEASE MANAGEMENT

Milestone-2

Object Creation

Activity – 1: Create Property Object:

To Navigate to Setup page:



To create an object:

1. From the setup page? Click on Object Manager? Click on Create? Click on Custom Object.



2. Enter label name as property
3. Enter plural label name as property
4. Enter Record Name as property name
5. Select data type as Text
6. Select Allow reports, Track field history, Allow activities
8. Click on Save and New

LEASE MANAGEMENT

The screenshot shows the Salesforce Setup interface under the Object Manager. A new object named 'property' is being created. The 'Record Name' field is set to 'property Name' with a 'Text' data type. Under 'Optional Features', 'Allow Reports', 'Allow Activities', and 'Track Field History' are checked. Under 'Object Classification', 'Allow Sharing', 'Allow Bulk API Access', and 'Allow Streaming API Access' are checked. The 'Deployment Status' is set to 'In Development'. The left sidebar lists various object configuration options.

This screenshot shows the same setup process for the 'property' object. The 'Deployment Status' has been changed from 'In Development' to 'Deployed'. The rest of the configuration remains the same as in the first screenshot.

Activity – 2 Create Tenant Object:

LEASE MANAGEMENT

From set up page

1. Click on Object Manager
2. Click on Create>>Click on Custom Object
3. Enter label name as Tenant
4. Enter plural label name as **Tenants**

The screenshot shows the Salesforce Setup interface for creating a new custom object. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main title is 'SETUP > OBJECT MANAGER' followed by 'Tenant'. On the left, a sidebar titled 'Details' lists various setup categories like Fields & Relationships, Page Layouts, Lightning Record Pages, etc. The main content area is titled 'Custom Object Information' and contains fields for 'Label' (set to 'Tenant') and 'Plural Label' (set to 'Tenants'). A note states: 'The singular and plural labels are used in tabs, page layouts, and reports. Be careful when changing the name or label as it may affect existing integrations and merge templates.' Below these, there's a section for 'Object Name' (set to 'Tenant') and 'Description'. At the bottom, there are 'Context-Sensitive Help Setting' options (radio buttons for 'Open the standard Salesforce.com Help & Training window' and 'Open a window using a Visualforce page'), and a 'Content Name' dropdown set to 'None...'. A note at the bottom of the form says: 'Enter Record Name Label and Format. The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name is different from the Label and Plural Label defined here.'

5. Enter record name as Tenant Name
6. Enter data type as Text
7. Click on Allow reports, Track field history, Allow activities
8. Click on Allow search

LEASE MANAGEMENT

The screenshot shows the Salesforce Setup interface under the Object Manager section. A new object is being created with the name 'Tenant'. The 'Record Name' is set to 'Tenant Name' and the 'Data Type' is 'Text'. Under 'Optional Features', 'Allow Reports', 'Allow Activities', and 'Track Field History' are checked. Under 'Object Classification', 'Allow Sharing', 'Allow Bulk API Access', and 'Allow Streaming API Access' are checked. The 'Deployment Status' is set to 'Deployed'. The left sidebar lists various object configuration options.

9. Click on Save and New

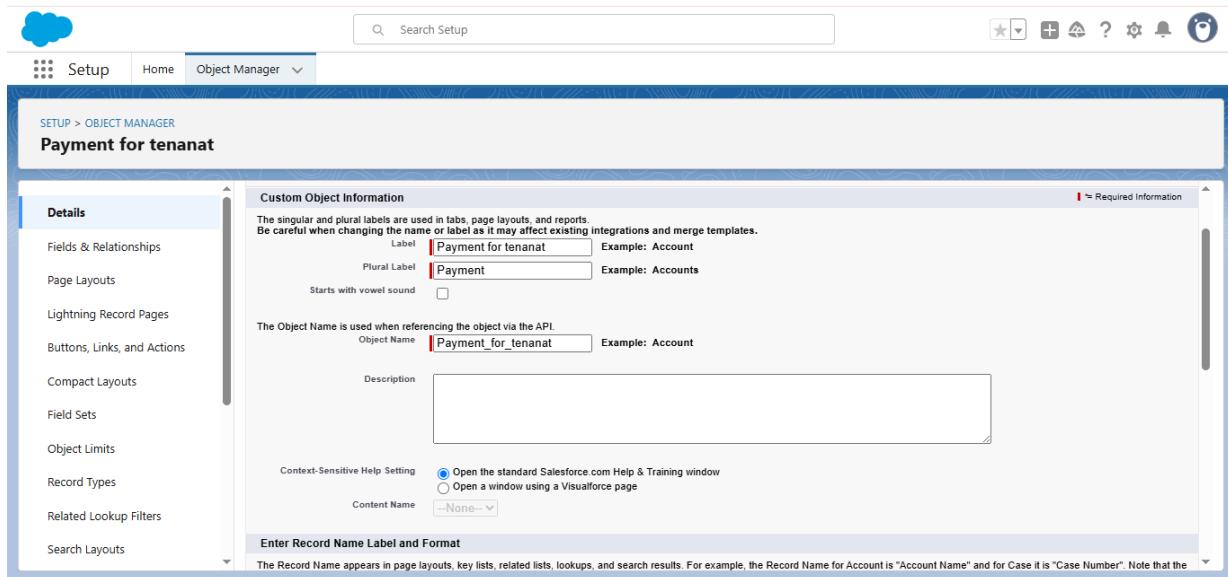
The screenshot shows the same Salesforce Setup interface after the 'Tenant' object has been saved. The 'Deployment Status' is now set to 'In Development'. The 'Search Status' is enabled with 'Allow Search' checked. The bottom of the screen features standard Salesforce navigation buttons: Save, Save & New, and Cancel.

LEASE MANAGEMENT

Activity - 3 Create Payment Object:

From set up page

1. Click on Object Manager
2. Click on Create>>Click on Custom Object
3. Enter label name as Payment for tenant
4. Enter plural label name as Payment



5. Enter record name as Payment Name
6. Enter data type as Text
7. Click on Allow reports, Track field history, Allow activities

LEASE MANAGEMENT

The screenshot shows the Salesforce Object Manager setup page for the 'Payment' object. The 'Record Name' field is set to 'Payment Name' with a data type of 'Text'. Under 'Optional Features', 'Allow Reports', 'Allow Activities', and 'Track Field History' are checked. Under 'Object Classification', 'Allow Sharing', 'Allow Bulk API Access', and 'Allow Streaming API Access' are checked. Under 'Deployment Status', 'Deployed' is selected. The left sidebar lists various configuration options: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts.

8. Click on Allow Search
9. Click on Save And New

The screenshot shows the same setup page for the 'Payment' object, but with additional configurations applied. The 'Allow Search' checkbox under 'Search Status' is checked. The 'Deployment Status' section shows 'Deployed' selected. The rest of the settings remain the same as in the previous screenshot.

Activity – 4 Create Lease Object:

From set up page

LEASE MANAGEMENT

1. Click on Object Manager
2. Click on Create>>Click on Custom Object
3. Enter label name as lease
4. Enter plural label name as lease

The screenshot shows the 'Object Manager' page in the Salesforce setup. A new object named 'lease' is being created. The 'Label' field contains 'lease' and the 'Plural Label' field also contains 'lease'. The 'Description' field is empty. Under 'Context-Sensitive Help Setting', the radio button for 'Open the standard Salesforce.com Help & Training window' is selected. The 'Content Name' dropdown is set to 'None'. The left sidebar lists various object settings like Fields & Relationships, Page Layouts, and Buttons, Links, and Actions.

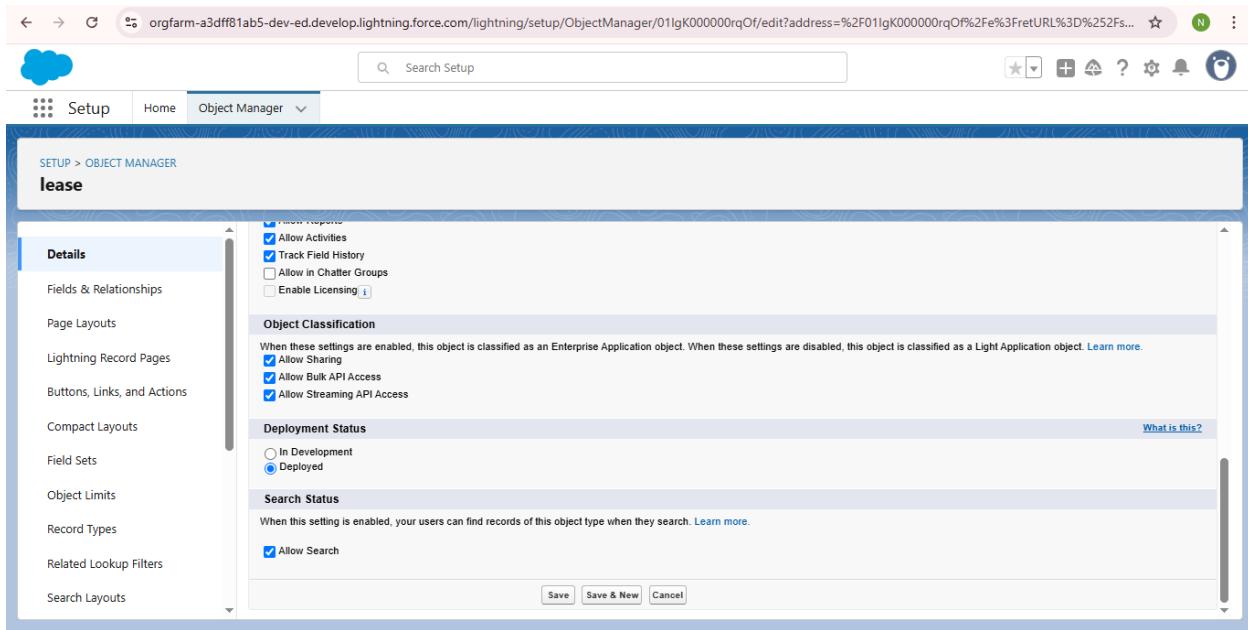
5. Enter record name as lease Name
6. Enter data type as Text
7. Click on Allow reports, Track field history, Allow activities

The screenshot shows the 'Object Manager' page with the 'lease' object selected. In the 'Optional Features' section, three checkboxes are checked: 'Allow Reports', 'Allow Activities', and 'Track Field History'. Below this, under 'Object Classification', three checkboxes are checked: 'Allow Sharing', 'Allow Bulk API Access', and 'Allow Streaming API Access'. At the bottom, the 'Deployment Status' is set to 'In Development'. The left sidebar shows other object settings.

8. Click on Allow search

LEASE MANAGEMENT

9. Click on Save and New



Milestone-3

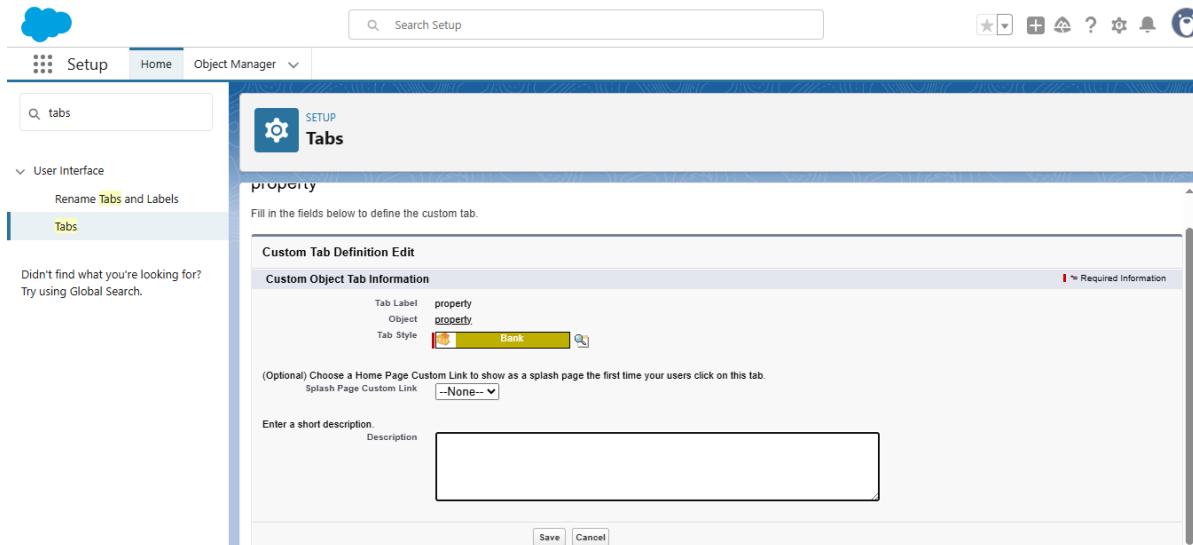
Tab Creation

Activity - 1 To create a Tab:(Property)

From set up page

- 1.Search for Tabs in the Quick find box
- 2.Click on tabs
- 3.Click on New under custom object tab
- 4.Select object Property >>Select the tab style

LEASE MANAGEMENT



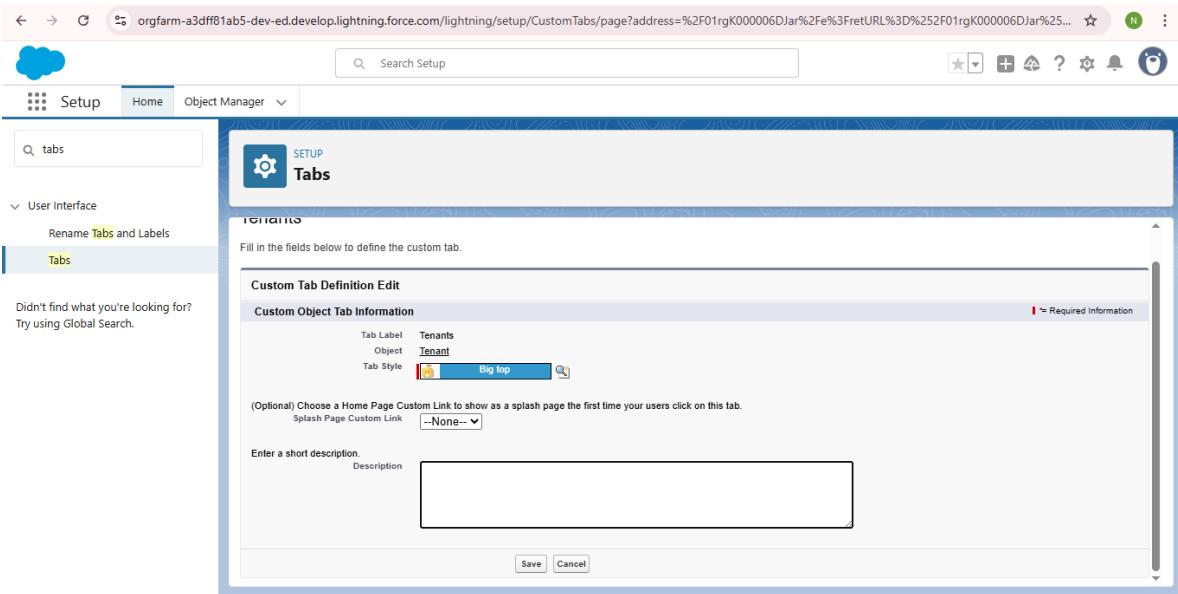
5. Click on next>> (Add to profile page) keep it as default>>Click on Next (Add to custom app)
Uncheck the include tab.
6. Make sure that the Append to tab user's existing personal customizations is checked
7. Click on Save

Activity – 2 To create a Tab:(Tenant)

From set up page

1. Search for Tabs in the Quick find box
2. Click on tabs
3. Click on New under custom object tab
4. Select object Tenant>>Select the tab style

LEASE MANAGEMENT



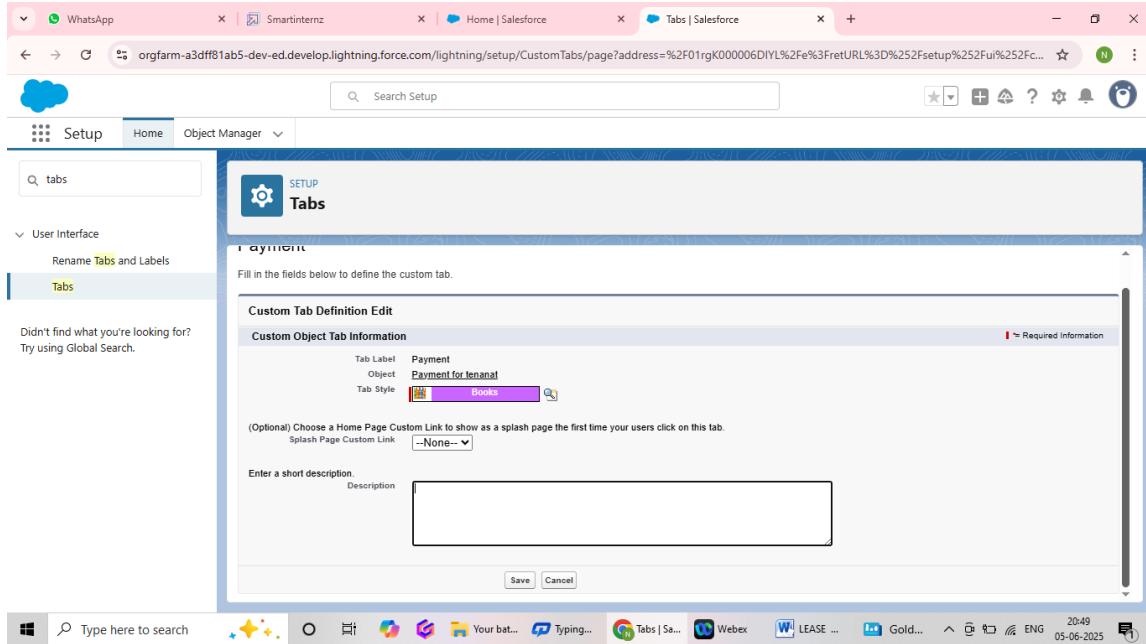
5. Click on next>> (Add to profile page) keep it as default>>Click on Next (Add to custom app)
Uncheck the include tab.
6. Make sure that the Append to tab user's existing personal customizations is checked
7. Click on Save

Activity – 3 To create a Tab:(Payment for tenants)

From set up page

1. Search for Tabs in the Quick find box
2. Click on tabs
3. Click on New under custom object tab
4. Select object Payment for tenant>>Select the tab style

LEASE MANAGEMENT



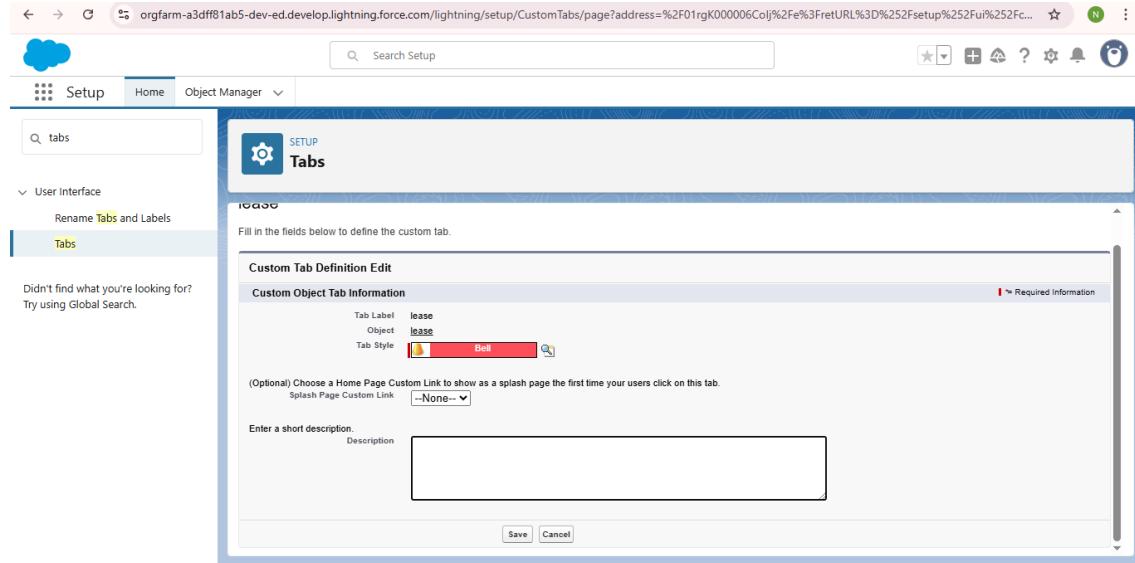
5. Click on next>> (Add to profile page) keep it as default>>Click on Next (Add to custom app)
Uncheck the include tab.
6. Make sure that the Append to tab user's existing personal customizations is checked
7. Click on Save

Activity – 4 To create a Tab:(lease)

From set up page

1. Search for Tabs in the Quick find box
2. Click on tabs
3. Click on New under custom object tab
4. Select object lease>>Select the tab style

LEASE MANAGEMENT



5. Click on next>> (Add to profile page) keep it as default>>Click on Next (Add to custom app)
Uncheck the include tab.
6. Make sure that the Append to tab user's existing personal customizations is checked
7. Click on Save

Milestone – 4

The Lightning App

Activity – 1 To create a lightning app page:

From the set up

1. Search AppManager in the Quick find box>>Select App Manager>>Click on New Lightning App

LEASE MANAGEMENT

The screenshot shows the Salesforce App Manager interface. A red box highlights the search bar with 'app manager'. Another red box highlights the 'Cloud Apps(Beta)' link. A third red box highlights the 'New Lightning App' button in the top right corner.

2. Fill the app name in app details and branding as follow

App Name: Lease Management

Developer Name: This will auto populated

Image: optional (if you want to give any image you can otherwise not)

(mandatory)Primary colour hex value: keep this default.

3. Then click Next >> (App option page) Set Navigation Style as Standard Navigation >>

Next.

(Utility Items) keep it as default >> Next.

The screenshot shows the 'New Lightning App' configuration page. The left sidebar shows a tree view with 'Salesforce' expanded, 'Data' selected, and 'Apps' expanded. Under 'Apps', 'Lease Management' is selected. The main area is titled 'App Options' with a progress bar at the bottom. The 'Navigation and Form Factor' section has 'Standard navigation' selected. The 'Setup and Personalization' section has 'Setup Experience' set to 'Setup (full set of Setup options)'. The 'App Personalization Settings' section has two checkboxes: 'Disable end user personalization of nav items in this app' and 'Disable temporary tabs for items outside of this app', both are unchecked. At the bottom, there are 'Back' and 'Next' buttons, with 'Next' being highlighted.

LEASE MANAGEMENT

4. To Add Navigation Items:

The screenshot shows the 'Navigation Items' section of the Lightning App Builder. On the left, a sidebar lists 'App Settings' (App Details & Branding, App Options, Utility Items (Desktop Only)), 'Navigation Items' (selected), and 'User Profiles'. The main area has a title 'Navigation Items' with a note about personalizing the navigation. It shows two lists: 'Available Items' (Accounts, Activation Targets, Activations, All Sites, Alternative Payment Methods, Analytics, App Launcher, Appointment Categories) and 'Selected Items' (Payment, property, lease). Buttons for 'Create' and arrows for moving items between lists are visible.

Search for the item in the (Payment for tenant, Tenants, property, lease) from the search bar and move it using the arrow button? Next? Next.

5. To Add User Profiles:

The screenshot shows the 'User Profiles' section of the Lightning App Builder. On the left, a sidebar lists 'App Settings' (App Details & Branding, App Options, Utility Items (Desktop Only)), 'Navigation Items' (selected), and 'User Profiles' (selected). The main area has a title 'User Profiles' with a note about choosing user profiles. It shows two lists: 'Available Profiles' (Analytics Cloud Integration User, Analytics Cloud Security User, Anypoint Integration, Authenticated Website, Authenticated Website, B2B Reordering Portal Buyer Profile, Contract Manager, Custom: Marketing Profile, Custom: Sales Profile) and 'Selected Profiles' (System Administrator). A search bar and arrows for moving profiles are visible.

Search profiles (System administrator) in the search bar >>click on the arrow button
>> save & finish.

LEASE MANAGEMENT

Milestone – 5

Fields

Activity – 1 To create fields in an object:

From set up

1. Click on Object Manager>>type Object name (property)in search bar>>Click on the object

The screenshot shows the Salesforce Object Manager interface. At the top, there's a navigation bar with 'SETUP' and 'Object Manager'. Below it is a search bar containing 'property'. The main area displays a table with one row of data:

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
property	property_c	Custom Object		6/3/2025	✓

LEASE MANAGEMENT

2. Click on Fields and Relationships>>New

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes a cloud icon, 'Search Setup', and various global buttons. The main header reads 'SETUP > OBJECT MANAGER' and 'property'. On the left, a sidebar lists options like Details, Fields & Relationships (which is selected), Page Layouts, Lightning Record Pages, etc. The right pane is titled 'Fields & Relationships' and displays a table of existing fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address_c	Long Text Area(32768)		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Name	Name_c	Text(20)		
Owner	OwnerId	Lookup(User,Group)		✓
property Name	Name	Text(80)		✓
sfqt	sfqt_c	Text(18)		
Type	Type_c	Picklist		

3. Select data type as Text

The screenshot shows the 'Fields & Relationships' page for the 'property' object. The sidebar lists various field types: Date, Date/Time, Email, Geolocation, Number, Percent, Phone, Picklist, Picklist (Multi-Select), Text (selected), Text Area, Text Area (Long), Text Area (Rich), Text (Encrypted), and Time. The right pane provides a detailed description for each type, such as 'Allows users to enter a date or pick a date from a popup calendar.' for Date.

4. Click on next

LEASE MANAGEMENT

The screenshot shows the 'Object Manager' section of the Salesforce Setup. A new object named 'property' is being created. The 'Label' field is set to 'property'. The 'Plural Label' field is also set to 'property'. The 'Object Name' field is set to 'property'. The 'Description' field is empty. Under 'Context-Sensitive Help Setting', the radio button for 'Open the standard Salesforce.com Help & Training window' is selected. The 'Content Name' dropdown is set to 'None'. A note at the bottom states: 'The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name is different from the Label.' A sidebar on the left lists various setup options like Fields & Relationships, Page Layouts, and Buttons, Links, and Actions.

- Enter field label as Name
- Field name will be auto generated
- Length: 25
- Required: Check box

The screenshot shows the 'Custom Field Definition Edit' screen for a field named 'Name'. The 'Field Label' is 'Name' and the 'Field Name' is also 'Name'. The 'Data Type' is 'Text'. Under 'Field Information', there are fields for 'Description', 'Help Text', 'Data Owner' (set to 'User'), 'Field Usage' (set to 'None'), 'Data Sensitivity Level' (set to 'None'), and 'Compliance Categorization' which includes categories like PII, HIPAA, GDPR, and PCI. A note at the bottom right says 'Activate Windows Go to Settings to activate Windows.' A sidebar on the left lists various setup options like Fields & Relationships, Page Layouts, and Buttons, Links, and Actions.

5. Click on Next>>Next>>Save and New

LEASE MANAGEMENT

To create another fields in an object:

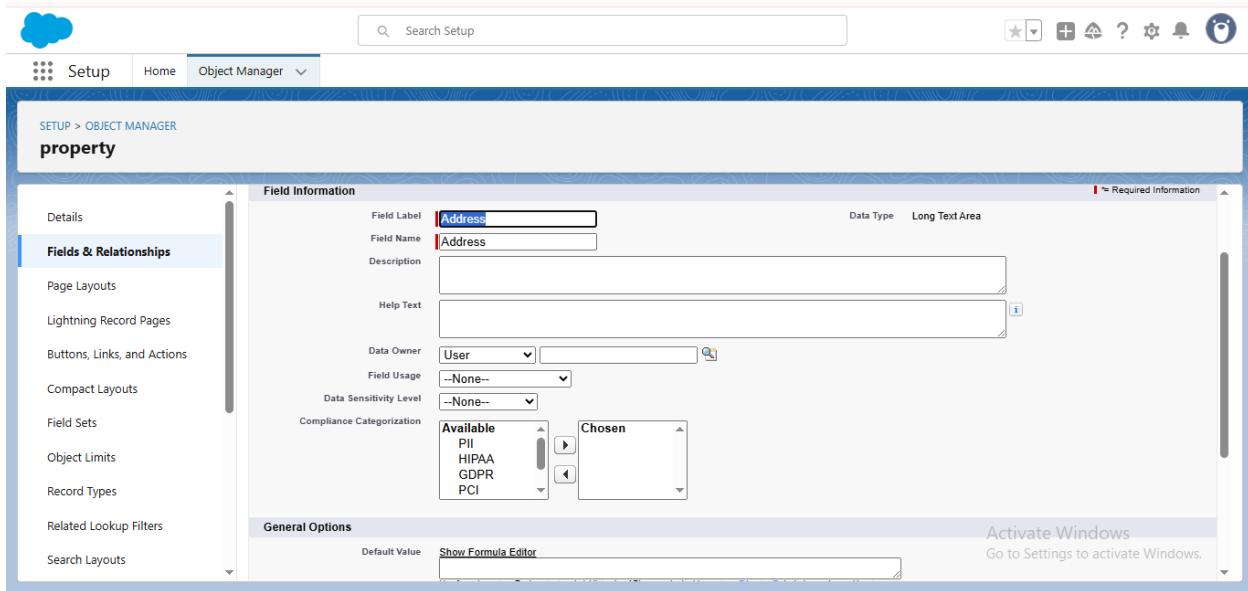
From the set up

- 1.Click on ObjectManager>>type object name(property)in search bar>>Click on the object
- 2.Click on Fields and Relationships>>New
- 3.Select Data type as a"Long Text" and click on Next
- 4.Fill the Above as following

Field Label:Address

Field Name:gets auto generated

LEASE MANAGEMENT



Click on Next>>Next>>Save and New

To create another fields in an object

From the set up

- 5.Click on ObjectManager>>type object name(property)in search bar>>Click on the object
- 6.Click on Fields and Relationships>>New
- 7.Select data type as “picklist” and click on Next
- 8.Fill the above as following

Field label:Type

Field Name:gets auto generated

Enter values with each value separated by a new line

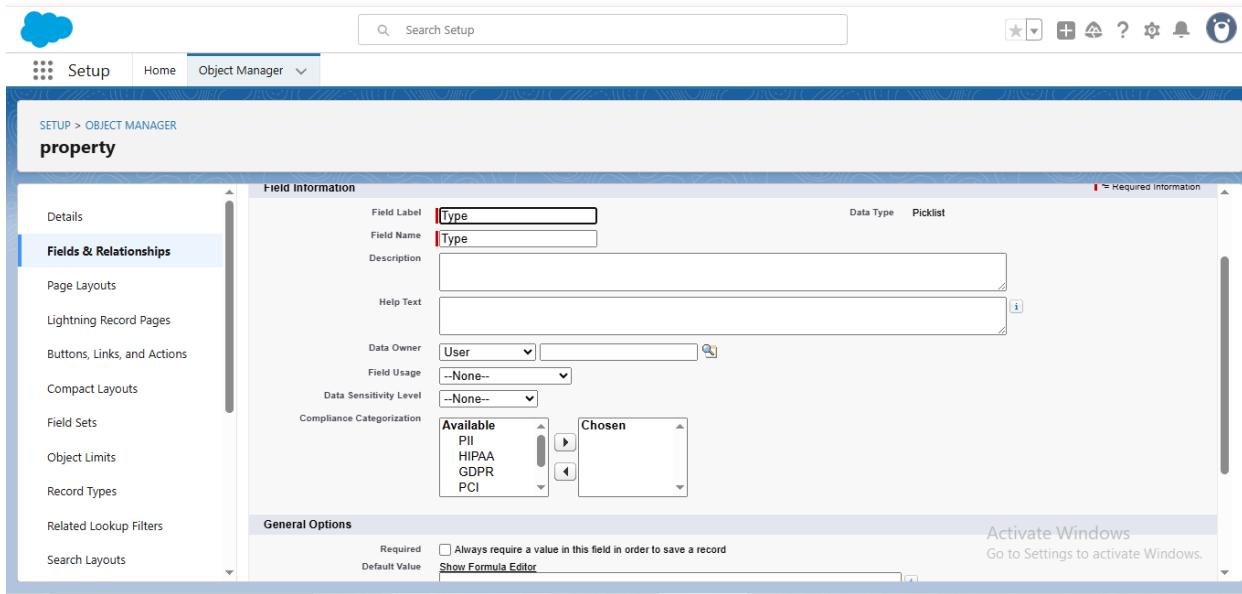
Enter these values

1BHK

2BHK

LEASE MANAGEMENT

3BHK



9.Click on Next>>Next>>Save and New

To create other fields in an object

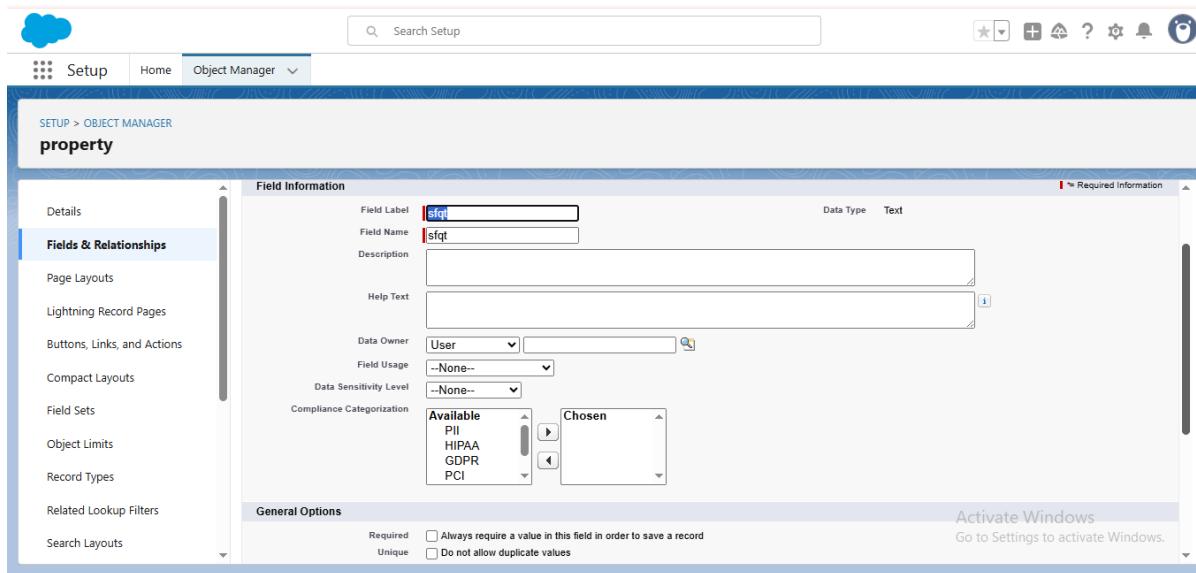
From the set up

1. Click on Object Manager>>type object name(property)in search bar>>Click on the object
2. Click on Fields and Relationships>>New
3. Select data type as “Text” and click on Next

Fill the above as following

- Field label: sfqt
- Field Name: gets auto generated
- Length:18

LEASE MANAGEMENT



4.Click on Next>>Next>>Save

Activity – 2 Creation of fields for the Tenant object

To create another fields in an object

From the set up

1.Click on Object Manager>>type object name(property)in search bar>>Click on the object

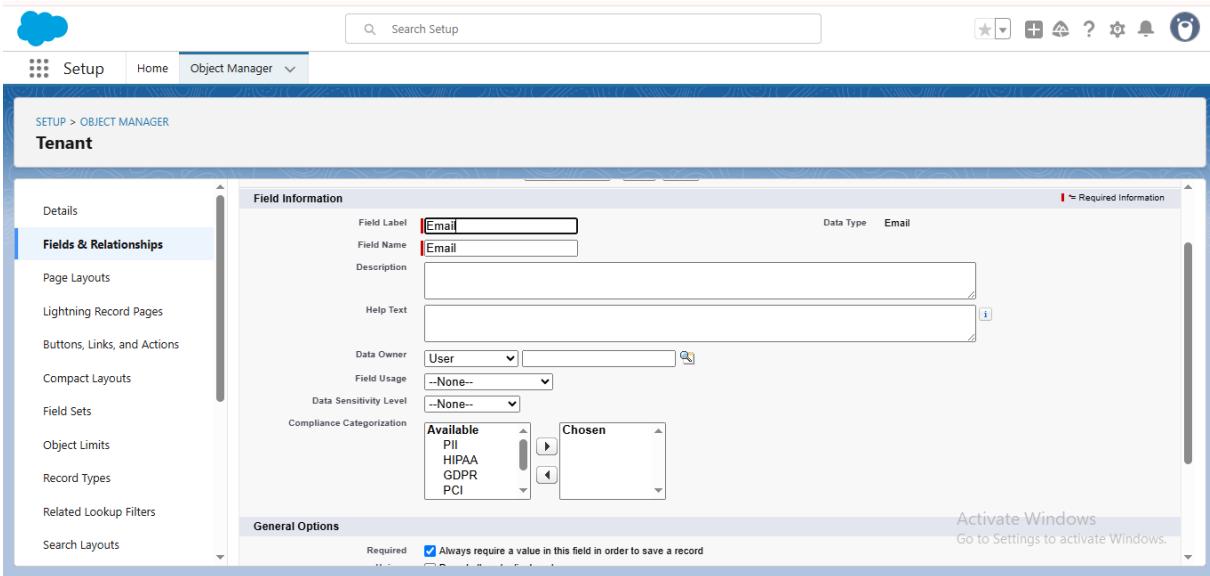
2.Click on Fields and Relationships>>New

3.Select data type as a “Email” and click on next

4.Fill the Above as following:

- Field label: Email
- Field Name: gets auto generated
- Click on required check box

LEASE MANAGEMENT



6.Click on Next>>Next>>Save and New

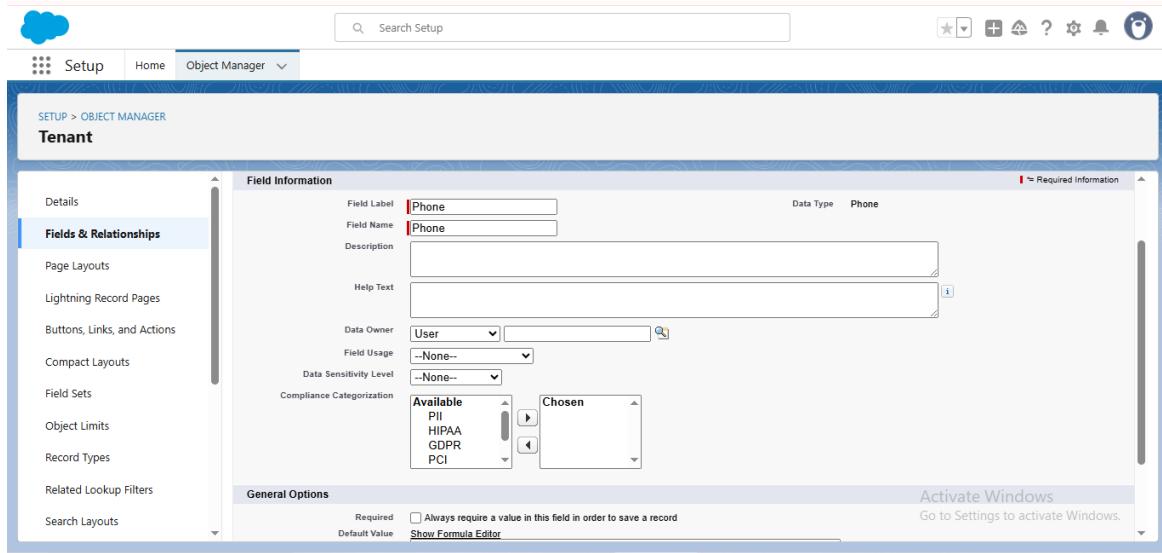
To create another fields in an object

From the set up

- 1.Click on Object Manager>>type object name(property)in search bar>>Click on the object
- 2.Click on Fields and Relationships>>New
- 3.Select data type as a “phone” and click on next
- 4.Fill the Above as Following:

- Field label: Phone
- Field Name: gets auto generated

LEASE MANAGEMENT



Click on Next>>Next>>Save and New

To create another fields in an object

From the set up

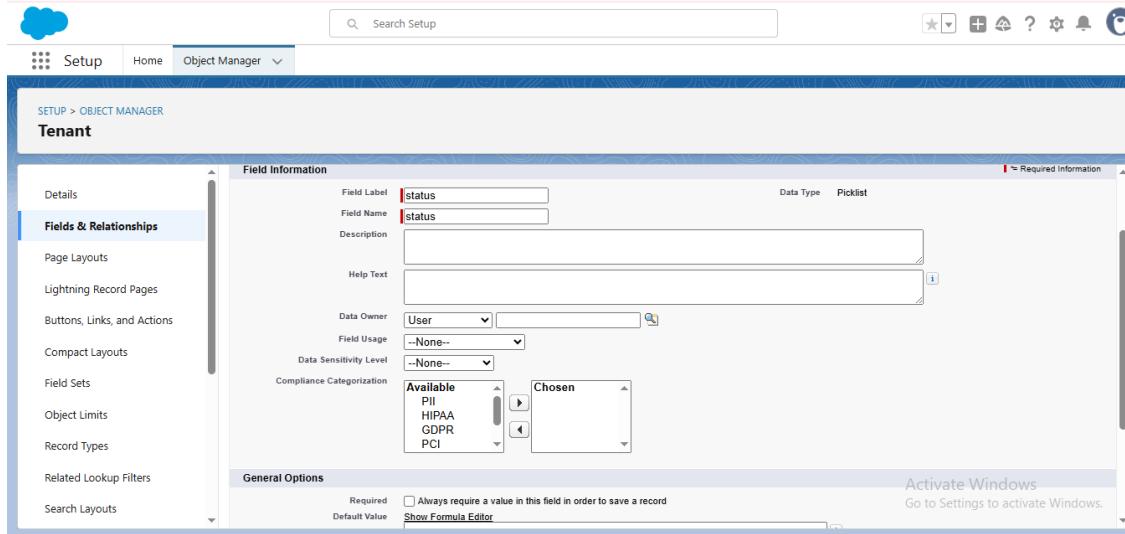
- 1.Click on Object Manager>>type object name(property)in search bar>>Click on the object
- 2.Click on Fields and Relationships>>New
- 3.Select data type as “Picklist” and click on next
- 4.Fill the Above as Following:

- Field label: status
- Field Name: gets auto generated
- Enter values with each value separated by a new line
- Enter these values:

➤ Stay

LEASE MANAGEMENT

➤ Leaving



Click on Next>>Next>>Save and New

Activity – 3 Creation of fields for the Lease object

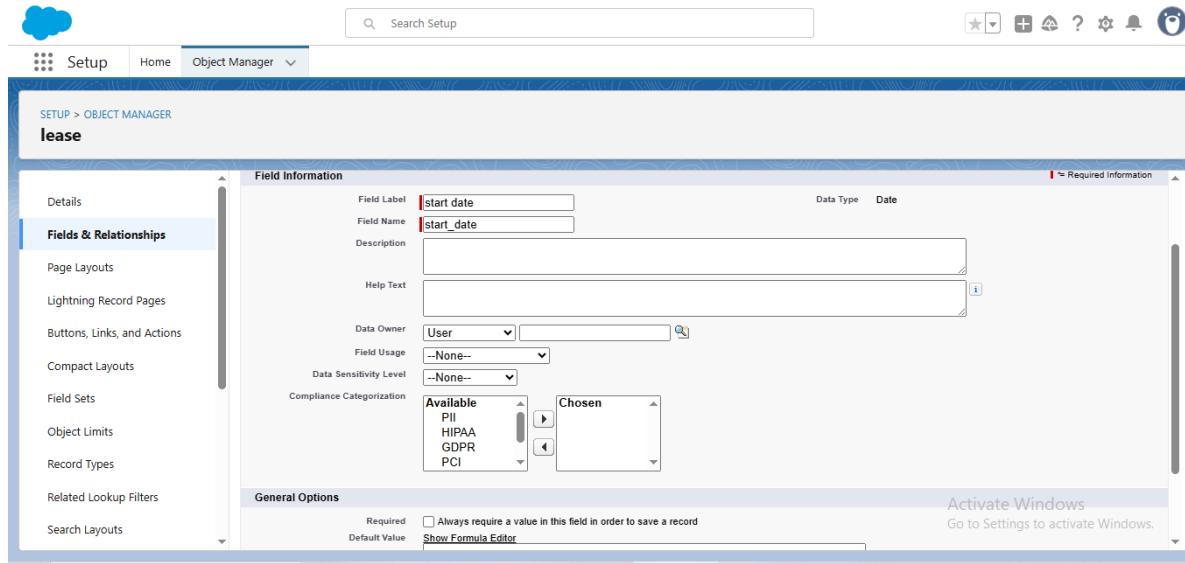
To create another fields in an object

From the set up

- 1.Click on Object Manager>>type object name(property)in search bar>>Click on the object
- 2.Click on Fields and Relationships>>New
- 3.Select data type as “Date” and click on Next
- 4.Fill the Above as Following:

- Field label: start date
- Field Name: gets auto generated

LEASE MANAGEMENT



Click on Next>>Next>>Save and New

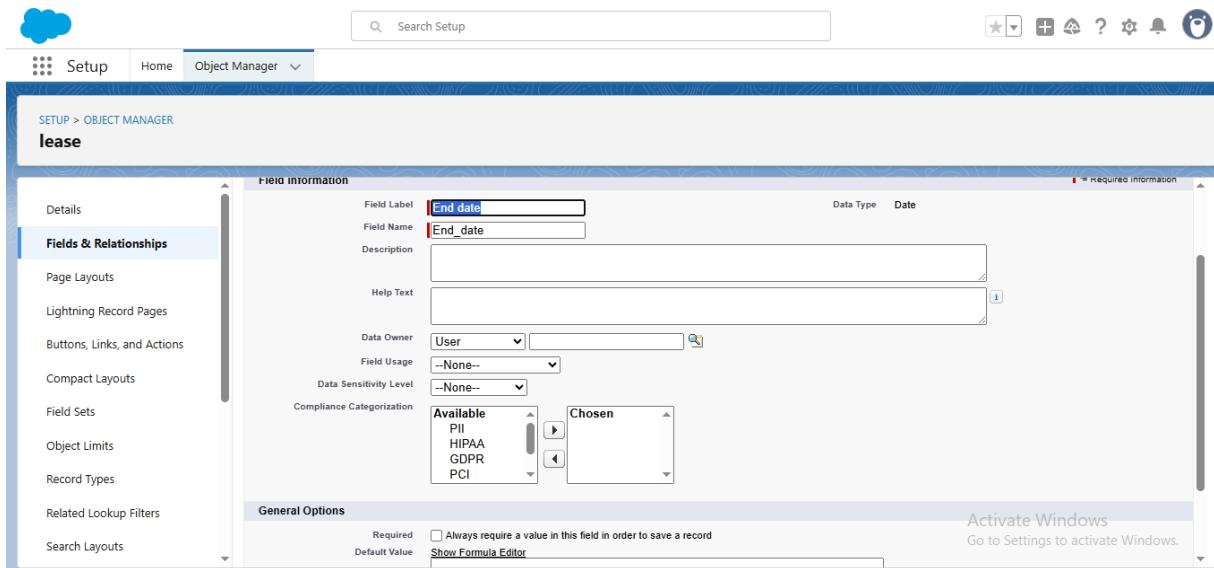
To create another fields in an object

From the set up

- 1.Click on Object Manager>>type object name(property)in search bar>>Click on the object
- 2.Click on Fields and Relationships>>New
- 3.Select data type as “Date” and click on Next
- 4.Fill the Above as Following:

- Field label: end date
- Field Name: gets auto generated

LEASE MANAGEMENT



Click on Next>>Next>>Save and New

Activity – 4 Creation of fields for the Payment for tenant object:

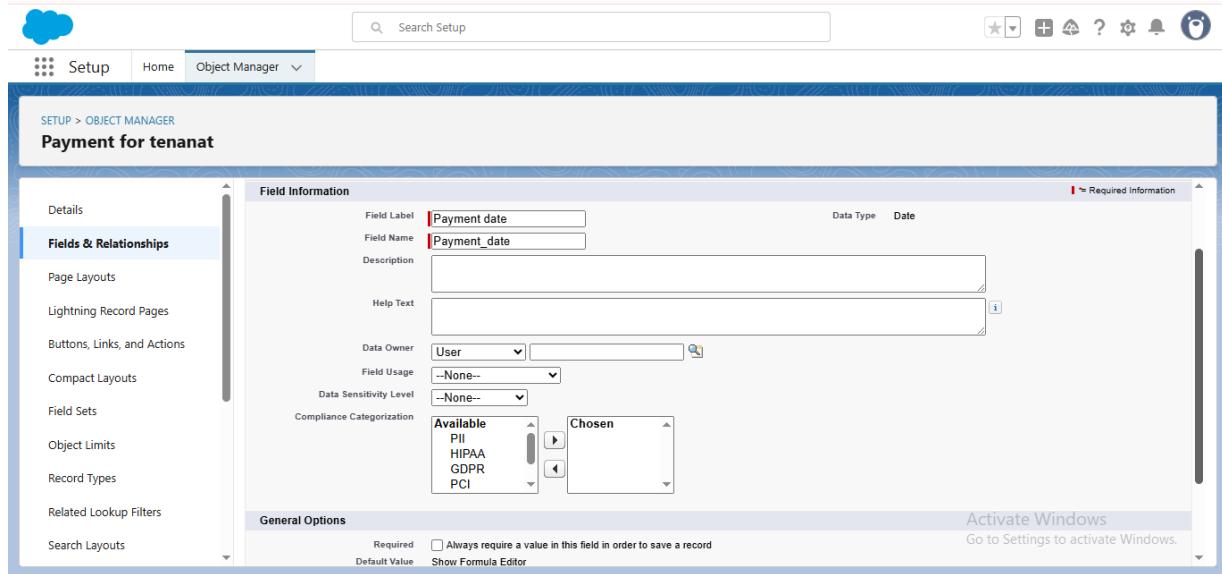
To create another fields in an object

From the set up

- 1.Click on Object Manager>>type object name(property)in search bar>>Click on the object
- 2.Click on Fields and Relationships>>New
- 3.Select data type as “Date” and click on Next
- 4.Fill the Above as Following:

- Field label: Payment date
- Field Name: gets auto generated

LEASE MANAGEMENT



Click on Next>>Next>>Save and New

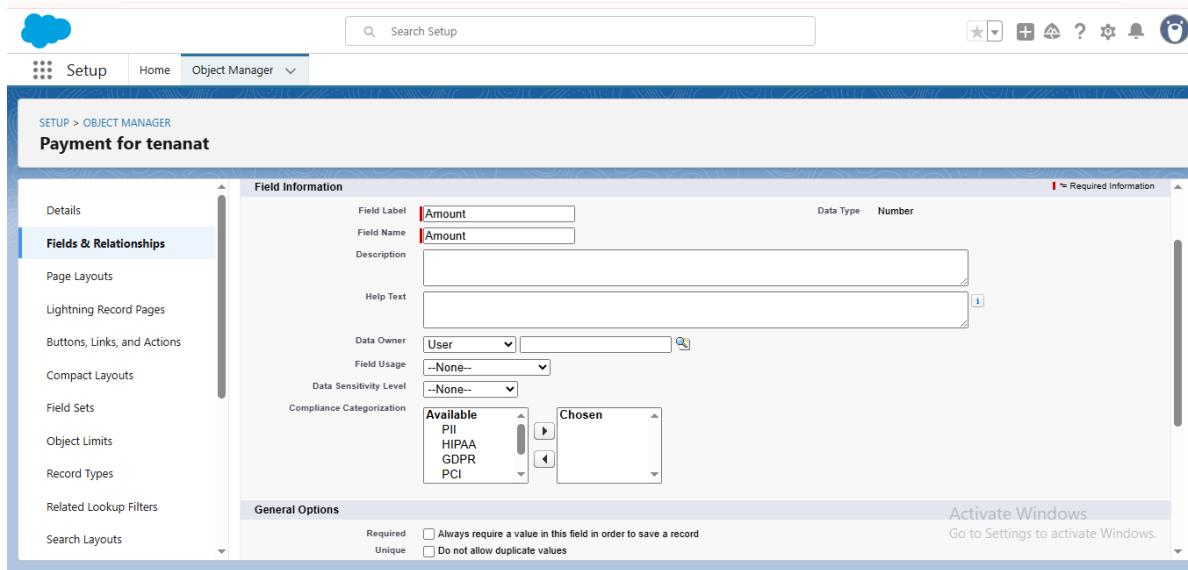
To create another fields in an object

From the set up

- 1.Click on Object Manager>>type object name(property)in search bar>>Click on the object
- 2.Click on Fields and Relationships>>New
- 3.Select data type as “Number” and click on Next
- 4.Fill the Above as Following:

- Field label: Amount
- Length:18
- Field Name: gets auto generated

LEASE MANAGEMENT



5. Click on Next>>Next>>Save and New

To create another fields in an object

From the set up

- 1.Click on Object Manager>>type object name(property)in search bar>>Click on the object
- 2.Click on Fields and Relationships>>New
- 3.Select data type as “Picklist” and click on Next

Field label: check for payment

Field Name: gets auto generated

LEASE MANAGEMENT

The screenshot shows the Salesforce Setup interface under the Object Manager. A new field is being created for the 'tenantat' object. The field is named 'check_for_payment' with a label 'check for payment'. It is a picklist type. The 'Available' section of the picklist includes 'PII', 'HIPAA', 'GDPR', and 'PCI'. The 'Chosen' section is currently empty. Other tabs like Details, Page Layouts, and Fields & Relationships are visible on the left.

Enter values with each value separated by a new line

- Paid
- Not paid

Click on Next>>Next>>Save and New

Activity – 5 Creation of Lookup fields:

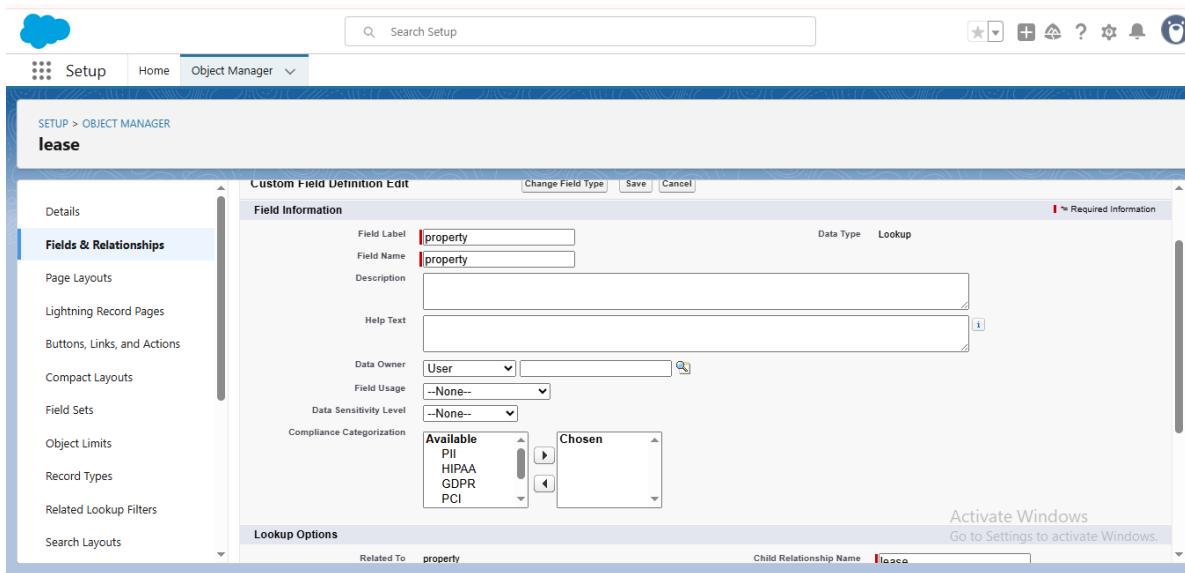
Creation of Look up field on lease object:

From the set up

1. Click on Object Manager>>type object name(property)in search bar>>Click on the object
2. Click on Fields and Relationships>>New
3. Select look up relationship
4. Select the related object “property” and click on next

- Field label: property
- Field Name: gets auto generated

LEASE MANAGEMENT



5. Click on Next>>Next>>Save and New

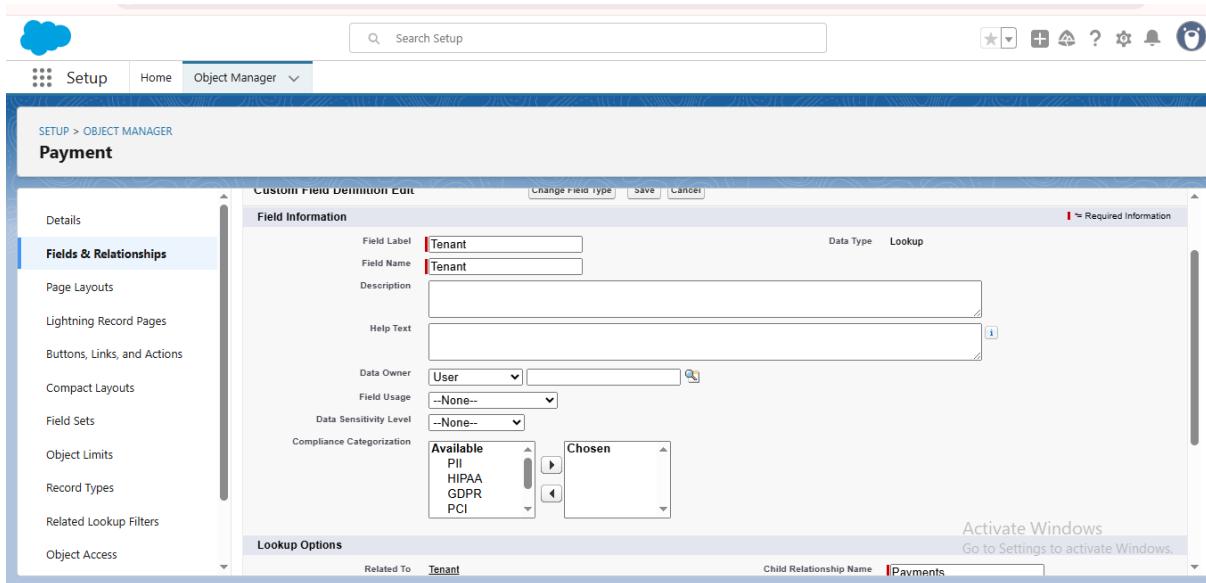
Creation of Look up field on Payment object:

From the set up

1. Click on Object Manager>>type object name(property)in search bar>>Click on the object
2. Click on Fields and Relationships>>New
3. Select look up relationship
4. Select the related object “Tenant” and click on next

- Field Name: Tenant
- Field label: gets auto generated

LEASE MANAGEMENT



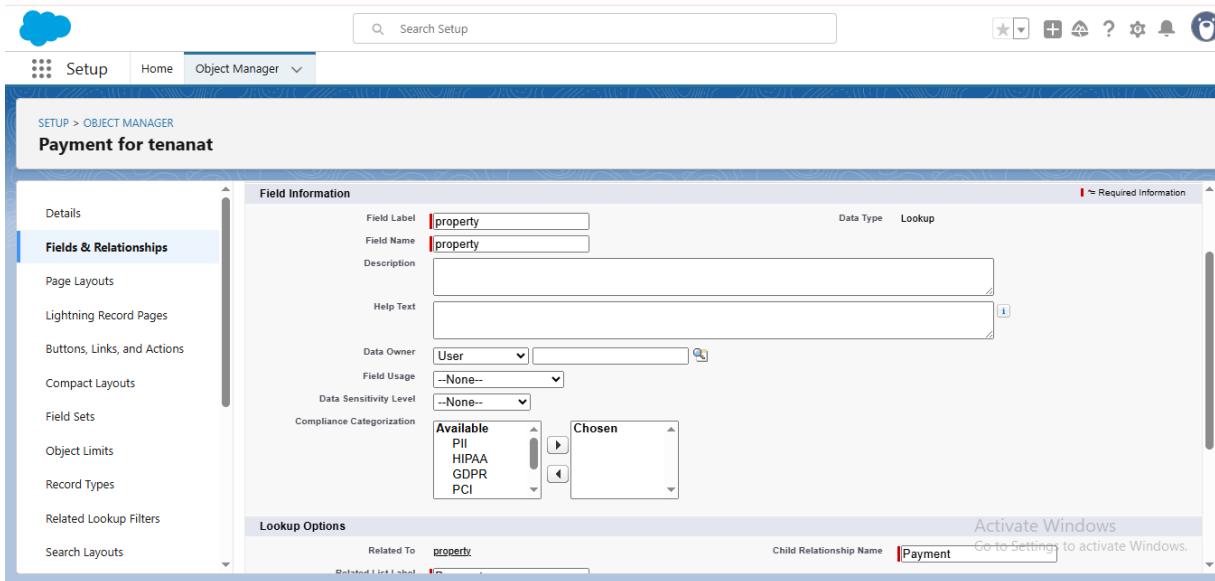
Click on Next>>Next>>Save and New

Creation of Look up field on Payment for tenant object

From the set up

1. Click on Object Manager>>type object name(property)in search bar>>Click on the object
2. Click on Fields and Relationships>>New
3. Select Master detail relationship
4. Select the related object “property” and click on next
 - Field Name: property
 - Field label: auto generated

LEASE MANAGEMENT



5. Click on Next>>Next>>Save and New

Milestone – 6

Validations Rules

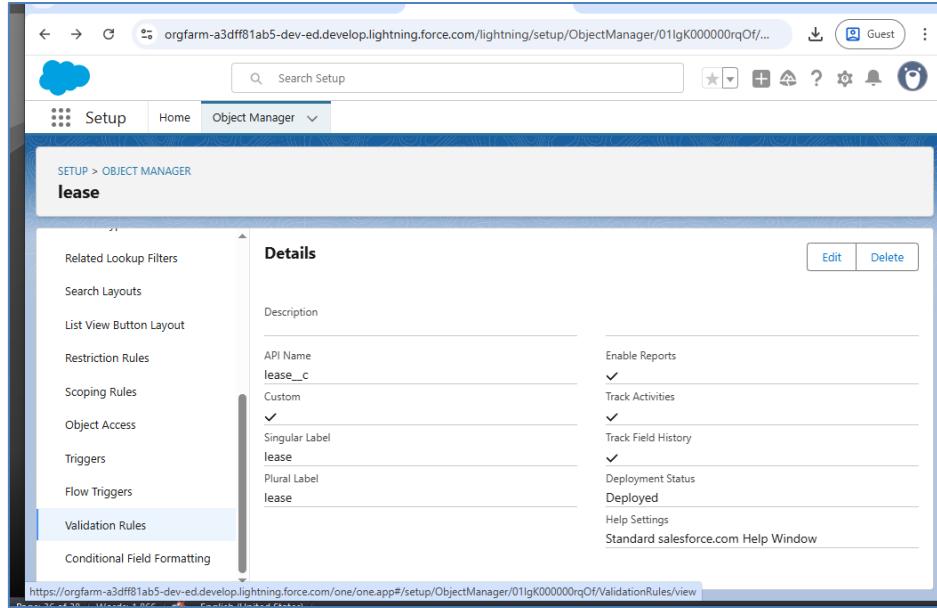
Activity – 1 To create a validation rule to an Lease Object:

From the set up

1.Click on Object Manager>>From drop down menu click edit for lease object

2.Click on Validation rule>>Click on New

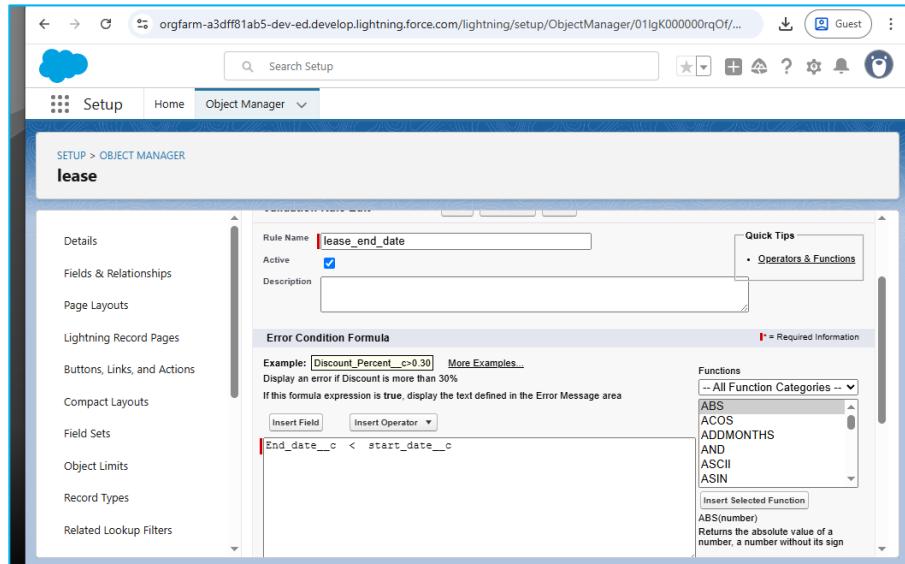
LEASE MANAGEMENT



3. Enter the rule name as “lease_end_date”

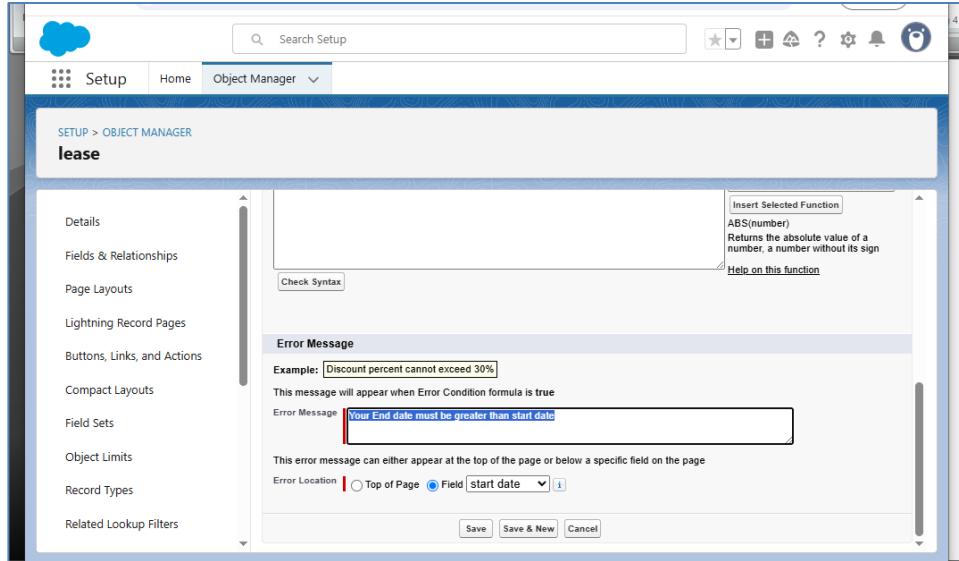
4. Insert the error condition formula as:

End_date__c < start_date__c



5. Enter the error message as your end date must be greater than “start date”, Select the error location as field and select the field as “start date” and click save

LEASE MANAGEMENT



Milestone-7

Email Templates

Activity – 1 Create Email Template for Tenant Leaving:

To create Email Templates:

From the set up

1. In the quick find box >>Enter Email templates>>Click on classic email template
2. Click on >>New Email
 - Template==>Choose Text
 - Folder: Unified public Classic Email templates
 - Click on available for use
3. Email Template Name is “tenant leaving”
4. Template Unique name: auto populated
5. Subject: “request for approve the leave”

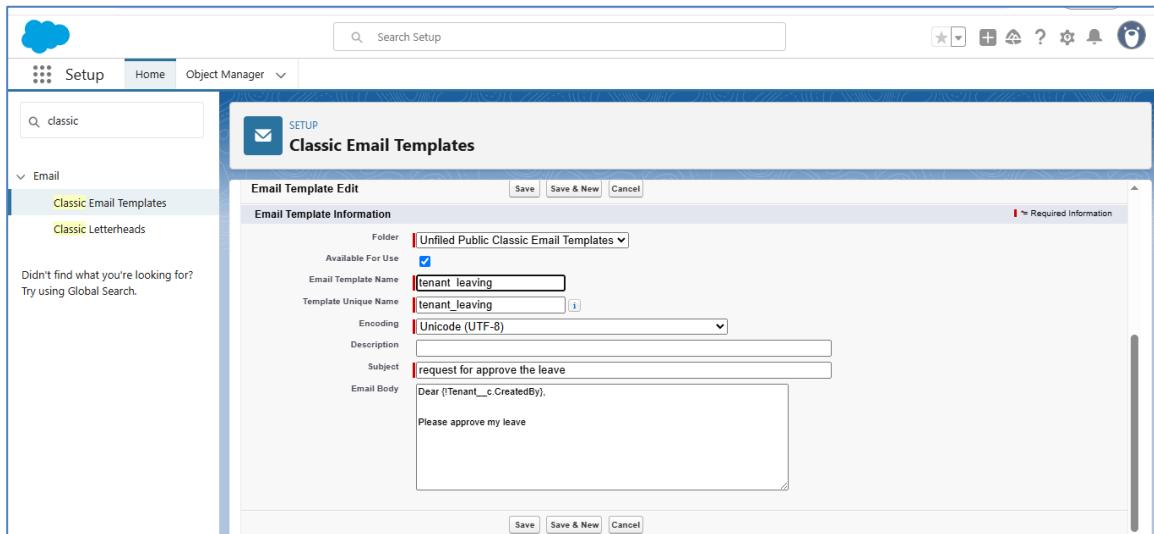
LEASE MANAGEMENT

6. Email body:

Dear {! Tenant__c. CreatedBy},

Please approve my leave

7. Save



Activity – 2 Create Email Template for Leave Approved

To create Email Templates:

From the set up

1. In the quick find box >>Enter Email templates>>Click on classic email template

2. Click on >>New Email

Template==>Choose Text

Folder: Unified public Classic Email templates

Click on available for use

3. Email Template Name is “Leave approved”

4. Template Unique Name: auto populated

5. Subject: “leave approved”

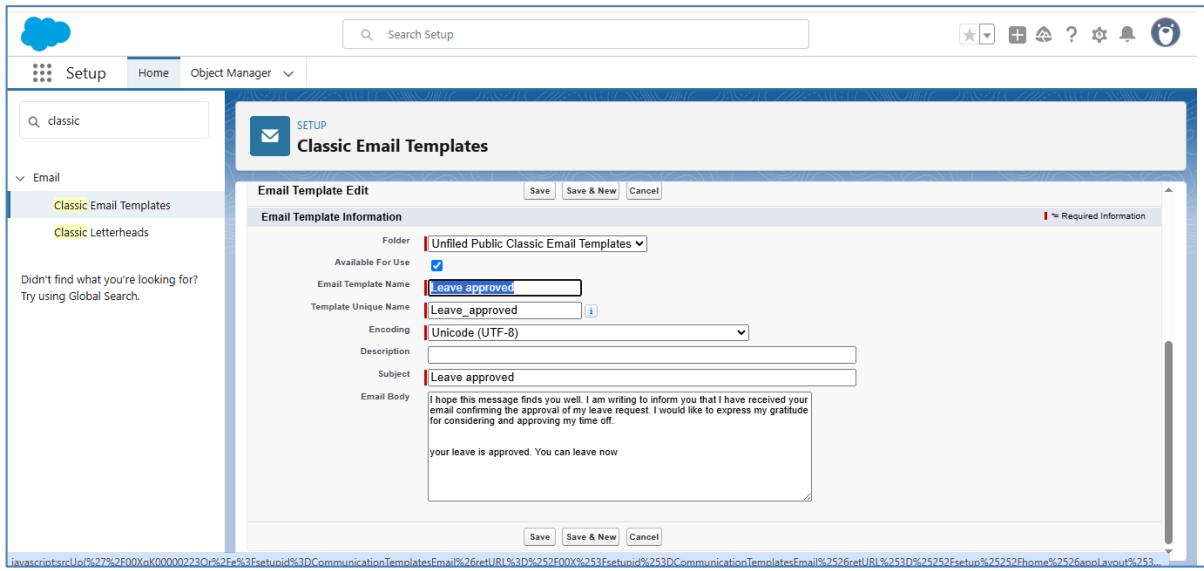
6. Email body:

Dear {! Tenant__c.Name.},

I hope this message finds you well. I am writing to inform you that I have received your email confirming the approval of my leave request. I would like to express my gratitude for considering and approving my time off.
your leave is approved. You can leave now

LEASE MANAGEMENT

7. Save



Activity – 3 Create Email Template For rejection for leave

LEASE MANAGEMENT

To create Email Templates:

From the set up

1. In the quick find box >>Enter Email templates>>Click on classic email template
2. Click on >>New Email

Template====>Choose Text

Folder: Unified public Classic Email templates

Click on available for use

3. Email Template Name is “Leave rejected”

4. Template Unique Name: auto populated

5. Subject: “Leave rejected”

6. Email body:

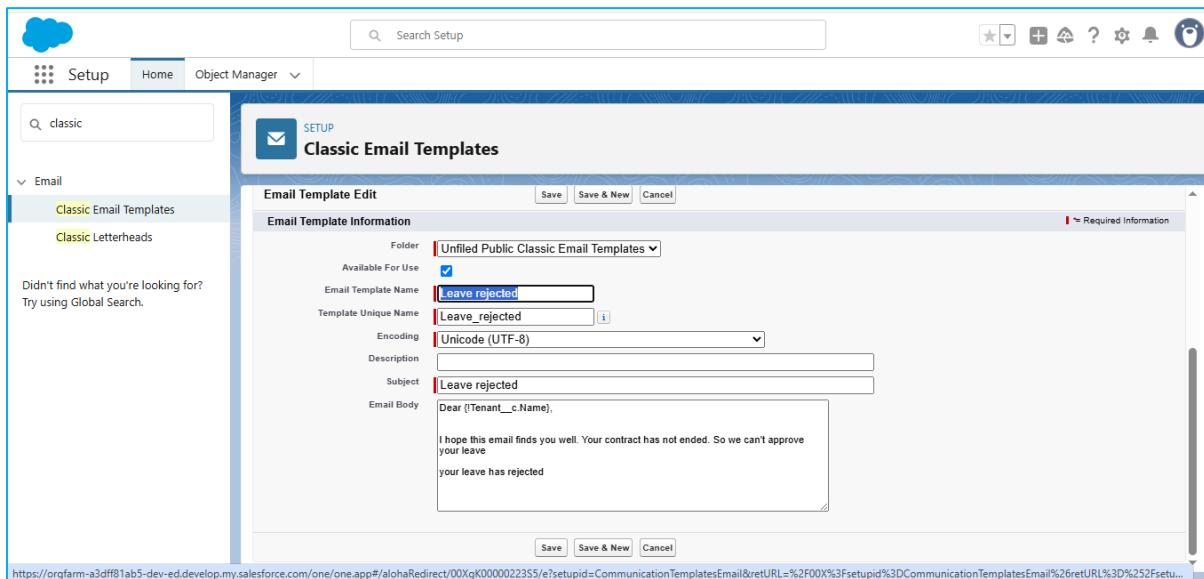
Dear {! Tenant__c. Name},

hope this email finds you well. Your contract has not ended.

So, we can't approve your leave

your leave has rejected

7. Save



LEASE MANAGEMENT

Activity – 4 Create Email Template For Monthly payment

To create Email Templates:

From the set up

1. In the quick find box >>Enter Email templates>>Click on classic email template

2. Click on >>New Email

Template==>Choose Text

Folder: Unified public Classic Email templates

Click on available for use

3. Email Template Name is “Tenant Email”

4. Template Unique Name: auto populated

5. Subject:” Urgent: Monthly Rent Payment Reminder”

6. Email body:

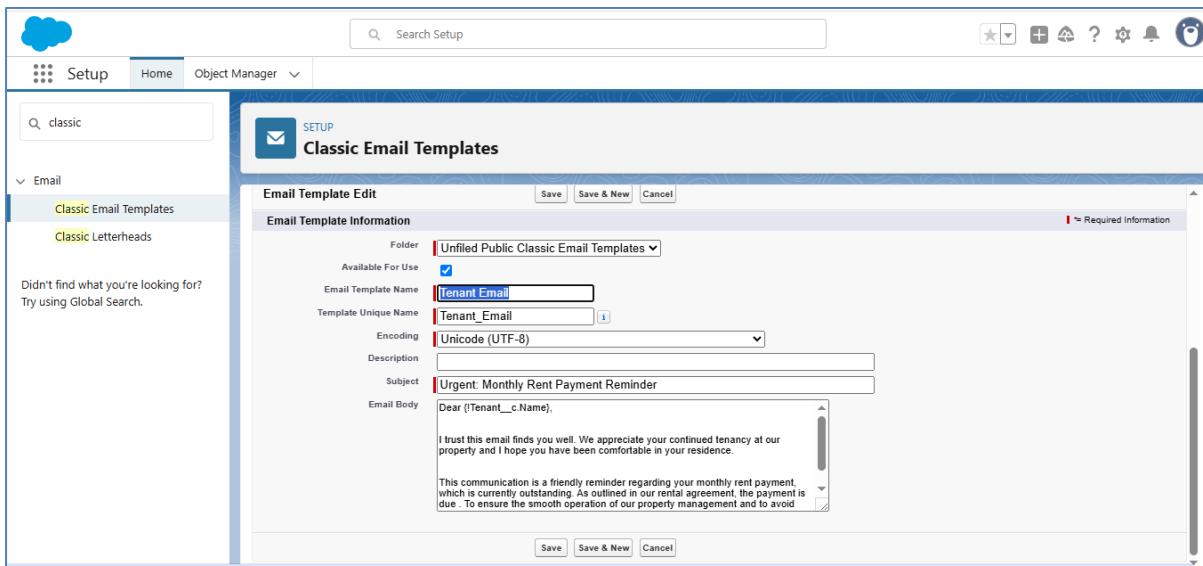
Dear {! Tenant__c. Name},

I trust this email finds you well. We appreciate your continued tenancy at our property and I hope you have been comfortable in your residence.

LEASE MANAGEMENT

This communication is a friendly reminder regarding your monthly rent payment, which is currently outstanding. As outlined in our rental agreement, the payment is due . To ensure the smooth operation of our property management and to avoid any inconvenience, we kindly request you to settle the payment at your earliest convenience.

7. Save



LEASE MANAGEMENT

Activity – 5 Create Email Template For successful payment:

To create Email Templates:

From the set up

1. In the quick find box >>Enter Email templates>>Click on classic email template

2. Click on >>New Email

Template==>Choose Text

Folder: Unified public Classic Email templates

Click on available for use

3. Email Template Name is “tenant payment”

4. Template Unique Name: auto populated

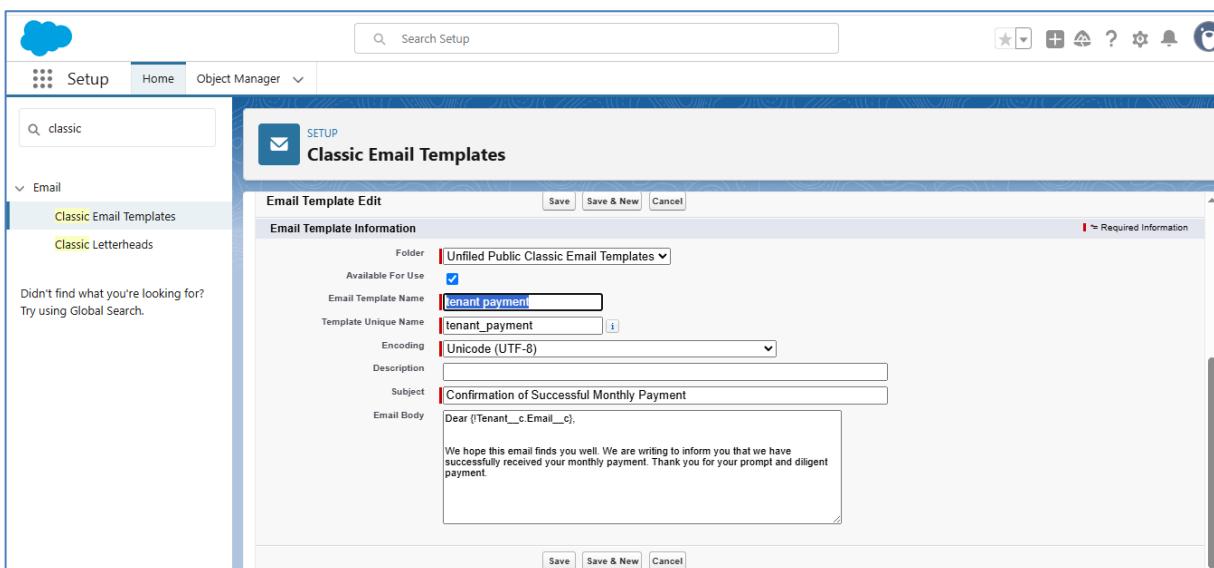
5. Subject: “Confirmation of Successful Monthly Payment”

6. Email body:

Dear {! Tenant__c. Email__c},

We hope this email finds you well. We are writing to inform you that we have successfully received your monthly payment. Thank you for your prompt and diligent payment.

7. Save



LEASE MANAGEMENT

Milestone – 8

Approval Process

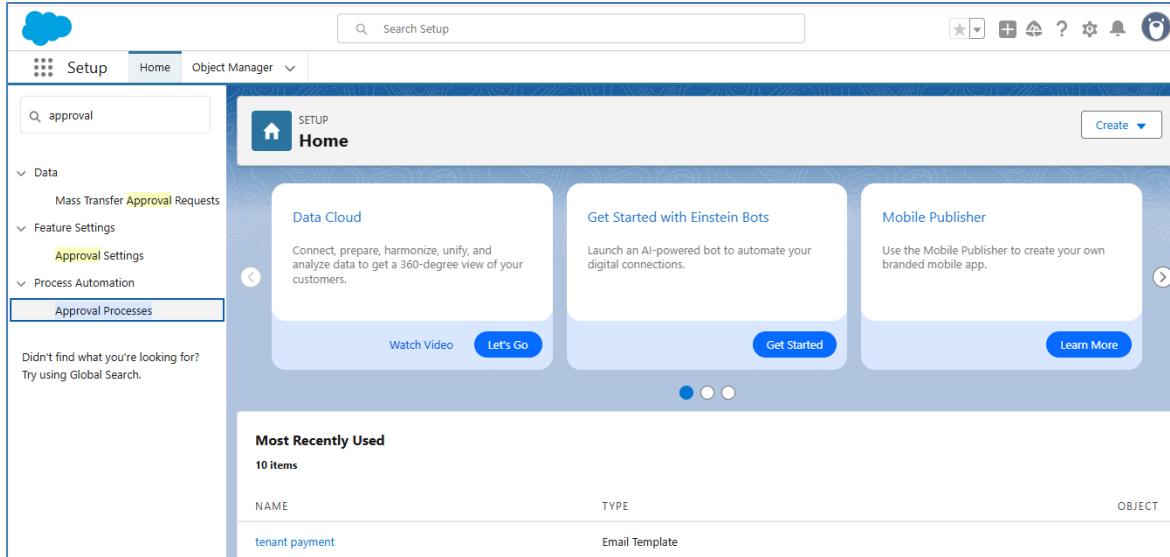
Activity – 1 Create Approval Process for check for vacant:

To create fields in an object

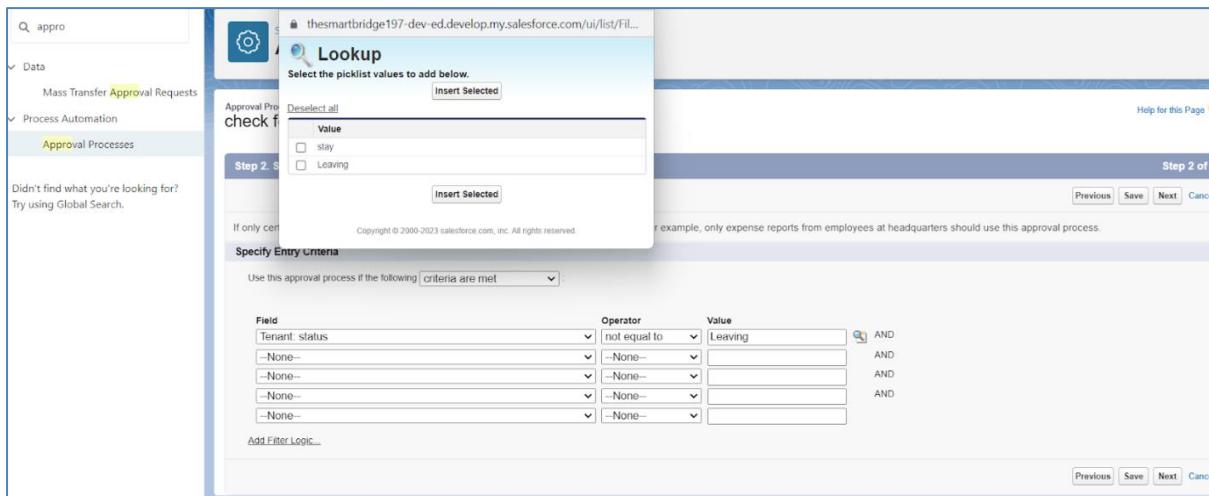
From the set up

1. In the quick find box >>Search for Approval process>>Click on it

LEASE MANAGEMENT



2. Manage approval process for “Tenant” from the drop down
3. Click on “create new approval process”>>use standard set up wizard
4. Process name “check for vacant”>>Click on next
5. Field >>tenant: status>>Operator: Not equals>>value>>Click on the look up filter icon and select “Leaving”
6. Click on insert field, click on next



7. Next Automated Approver determined by “None” from the drop down.
8. Select the “Administrators ONLY can edit records during the approval process”.
9. Then click Next.

LEASE MANAGEMENT

Approval Process Edit
check for vacant

Help for this Page ?

Step 3 of 6

Previous Save Next Cancel

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 ⓘ
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.

Previous Save Next Cancel

10. From the available fields select >> Tenant Name, and then add >>Add it to the selected. Then Next

Make sure Display approver history is checked.

And under security settings check the “Allow approvers to access the approval page only from within the Salesforce application. (Recommended)” option.

11. Submitter type Search>>Owner, Allowed Submitters>>Property Owner. Then Next. 12. Then click save.

SETUP Approval Processes

Help for this Page ?

What Would You Like To Do Now?

You have just created an approval process. However, you cannot activate this process until you define at least one approval step. Would you like to do that now?

Yes, I'd like to create an approval step now.
 I'll do this later. Take me to the approval detail page to review what I've just created.
 I'll do this later. Take me back to the listing of all approval processes for this object.

Go!

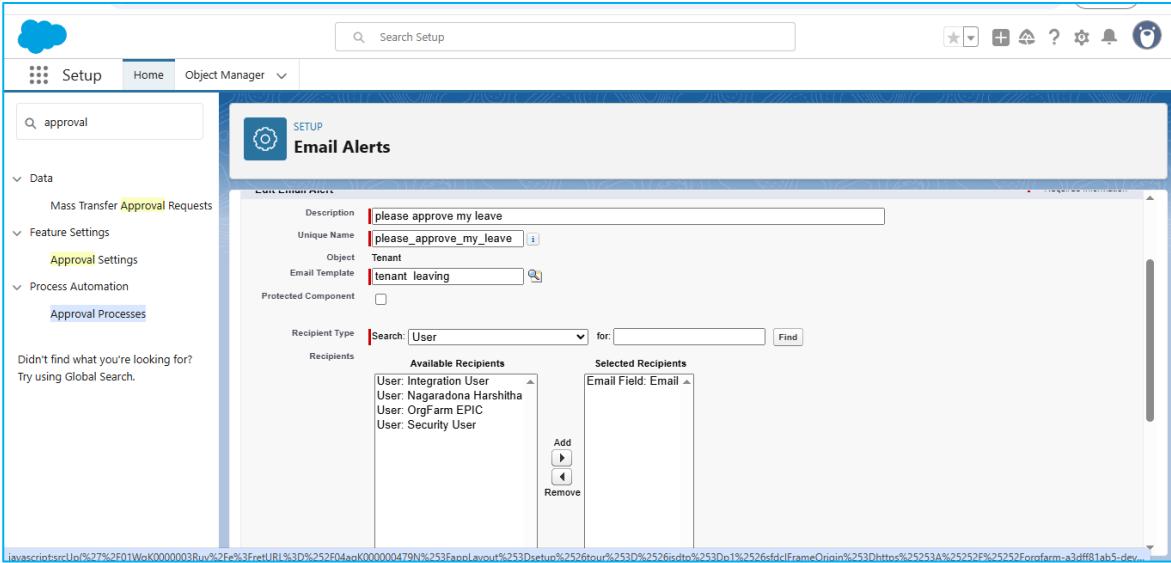
LEASE MANAGEMENT

- Click on “I'll do this later. Take me back to the listing of all approval process for this object
- Click go

Activity – 2 Initial Submission Action:

1. Under initial submission action click on add new and then select email alert.
2. Description: “please approve my leave”
3. unique name: auto populated
4. Email template: tenant leaving
5. Recipient type: Email field
6. Available Recipients: Email field: Email
7. From Email address: Current user’s email
8. Click save.

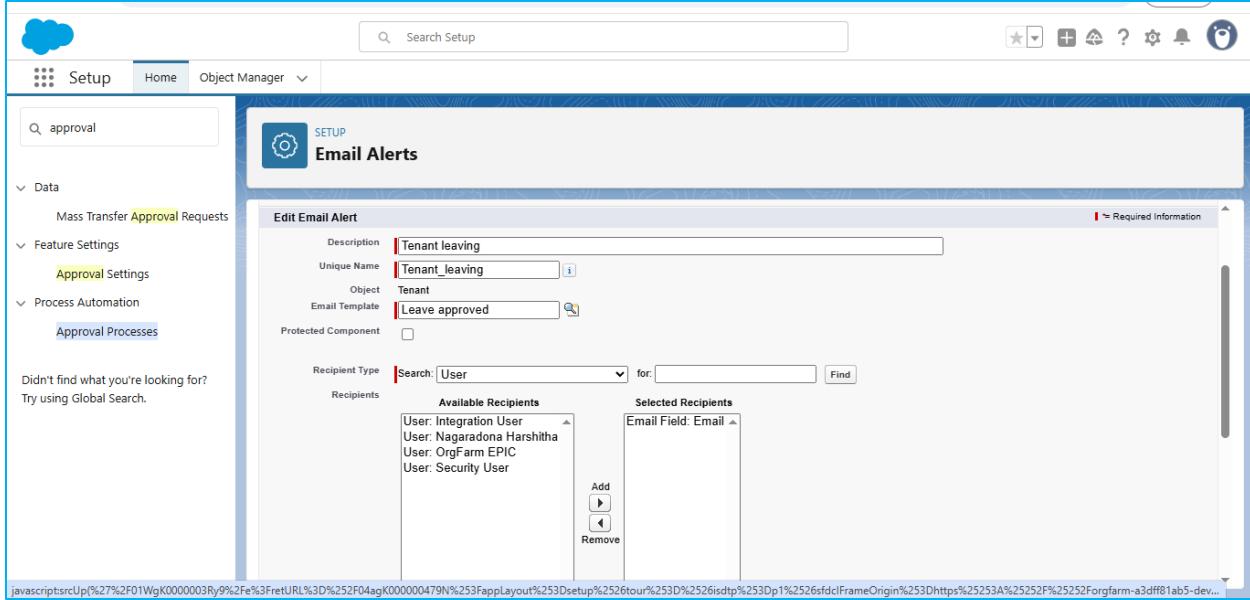
LEASE MANAGEMENT



Activity – 3 Final Approval Action:

1. Under Final approval action click on new and then select email alert.
2. Description: “Tenant leaving” .
3. unique name: auto populated
4. Email template: Leave approved
5. Recipient type: Email field
6. Available Recipients: Email field: Email
7. From Email address: Current user’s email
8. Click save.

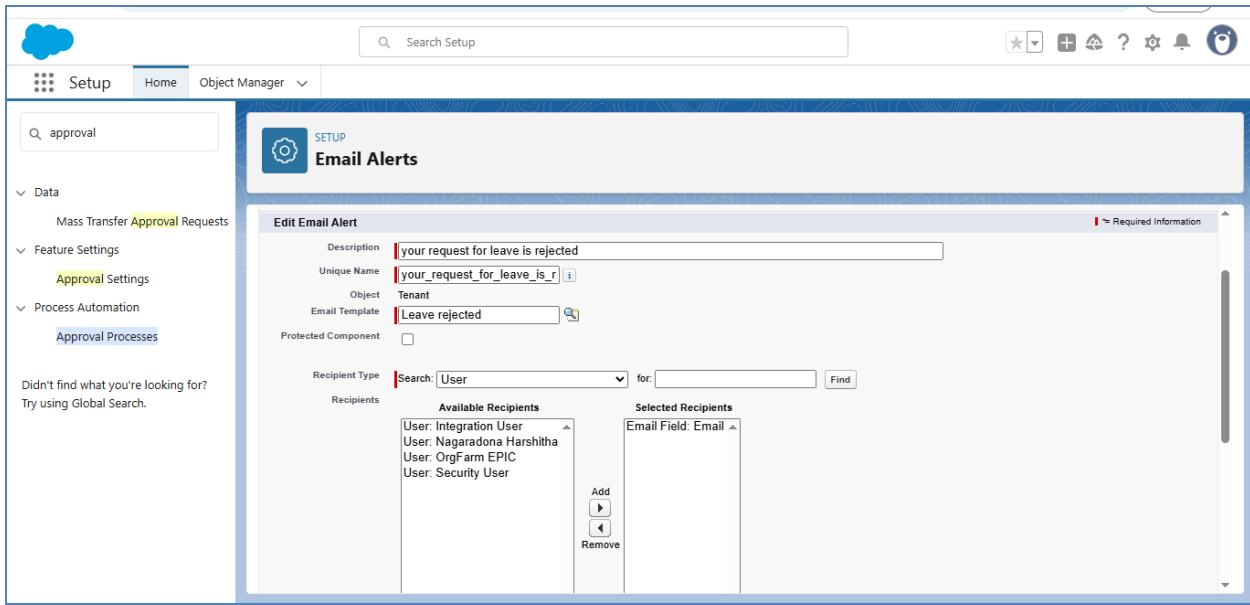
LEASE MANAGEMENT



Activity – 4 Final Rejection Action:

1. Under Final rejection action click on new and then select email alert.
2. Description: “Your request for leave is rejected”.
3. unique name: auto populated
4. Email template: leave rejected
5. Recipient type: Email field
6. Available Recipients: Email field: Email
7. From Email address: Current user’s email
8. Click save.

LEASE MANAGEMENT



Milestone – 9

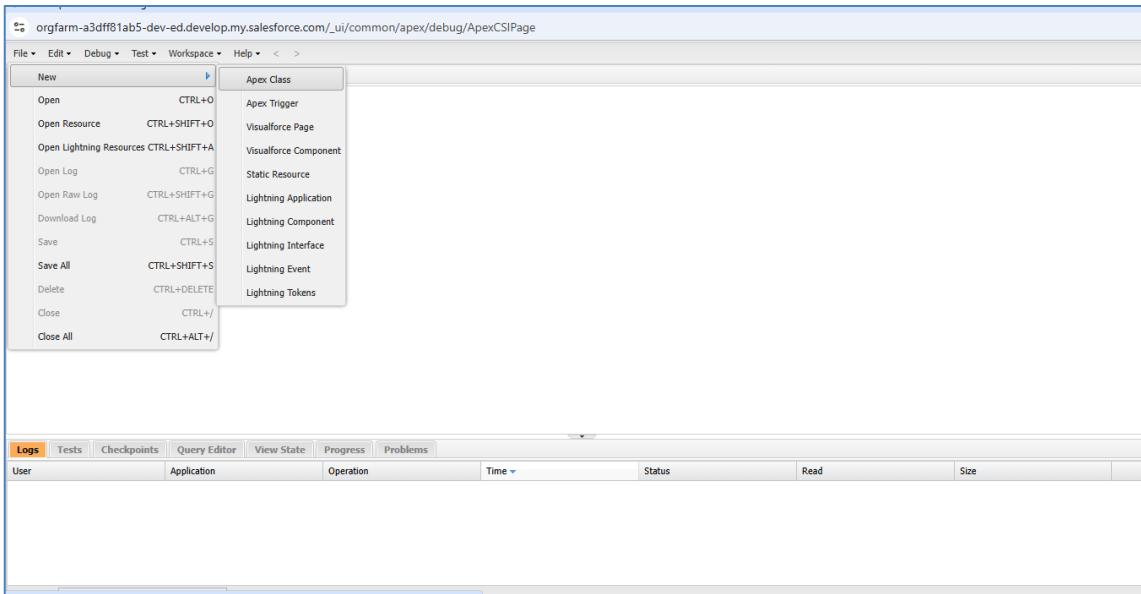
Apex Trigger

Activity – 1 Create an Apex Trigger:

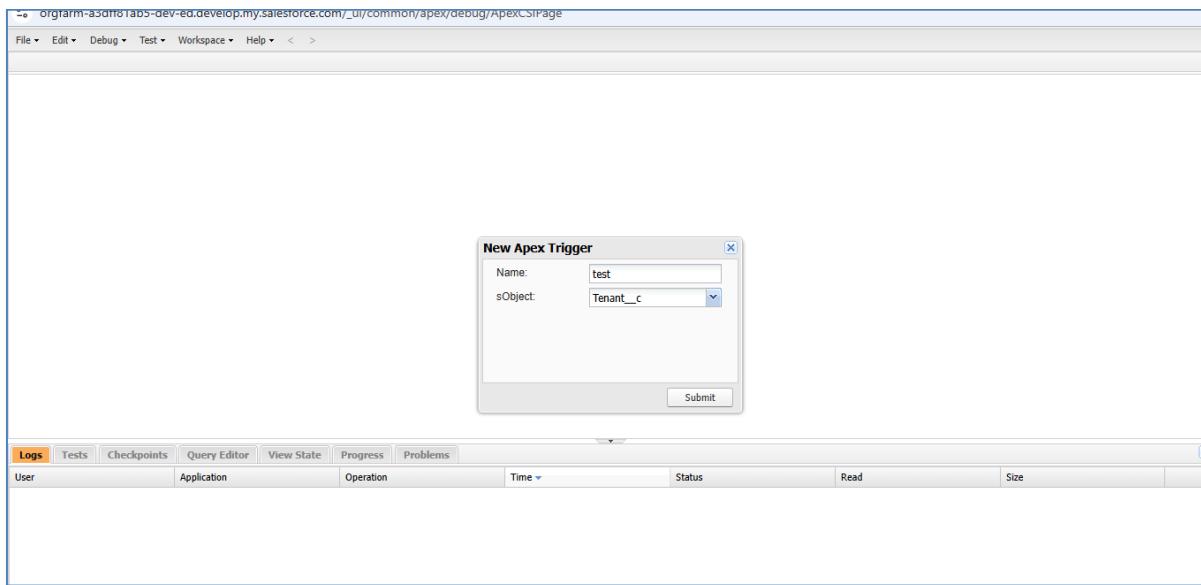
To create a new Apex Trigger, follow the below steps

1. Click on file>>New>> Apex Trigger

LEASE MANAGEMENT

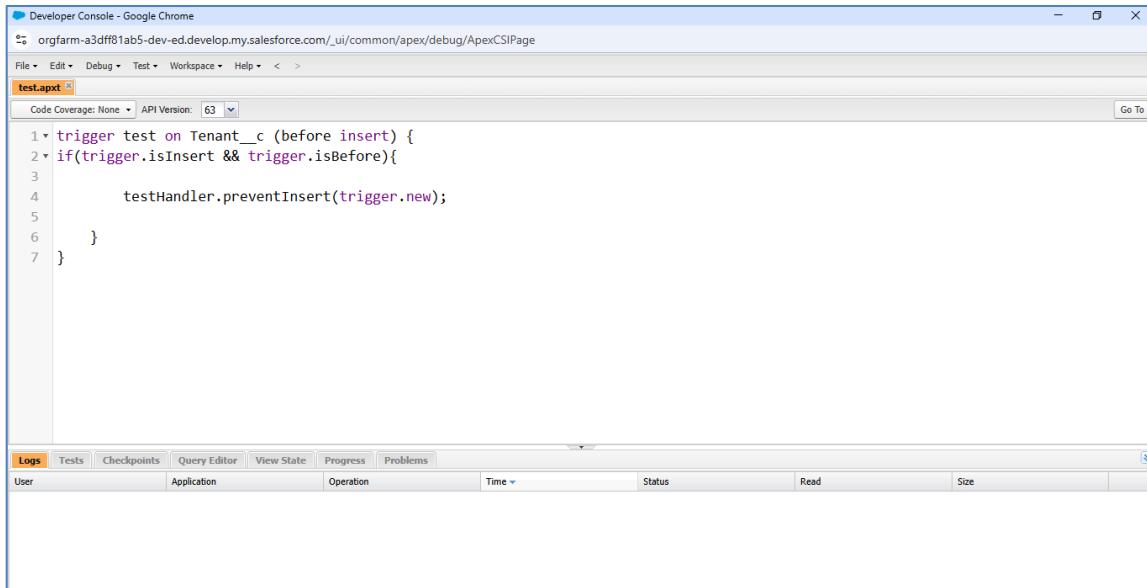


2. Give the Apex trigger name as Test>>select Tenant__c from the drop down sObject



3. Click on submit
4. Now write the code logic here

LEASE MANAGEMENT



The screenshot shows the Salesforce Developer Console in Google Chrome. The URL is orgfarm-a3dff01ab5-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage. The tab is titled 'test.apxt'. The code editor contains the following Apex trigger code:

```
trigger test on Tenant__c (before insert) {
    if(trigger.isInsert && trigger.isBefore){
        testHandler.preventInsert(trigger.new);
    }
}
```

Below the code editor is a table titled 'Logs' with columns: User, Application, Operation, Time, Status, Read, and Size. The table is currently empty.

Trigger Code:

trigger test on Tenant__c (before insert)

```
{  
    if (trigger. is Insert && trigger.isBefore){  
        testHandler.preventInsert(trigger.new);  
    }  
}
```

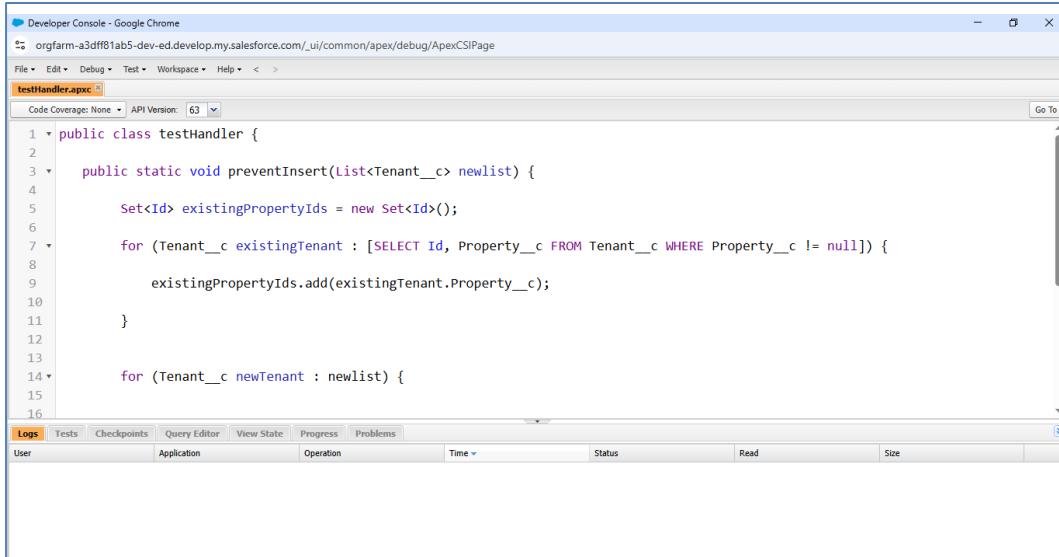
5.Click on Save

Create an Apex Handler class:

To create a new Apex class follow the below steps

1. Click on file>>New>> Apex class
2. Enter the trigger name as testHandler

LEASE MANAGEMENT



The screenshot shows the Salesforce Developer Console in Google Chrome. The title bar says "Developer Console - Google Chrome". The URL is "orgfarm-a3df81ab5-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage". The tab is "testHandler.apex". The code editor contains the following Apex class:

```
1 public class testHandler {
2
3     public static void preventInsert(List<Tenant__c> newList) {
4
5         Set<Id> existingPropertyIds = new Set<Id>();
6
7         for (Tenant__c existingTenant : [SELECT Id, Property__c FROM Tenant__c WHERE Property__c != null]) {
8
9             existingPropertyIds.add(existingTenant.Property__c);
10
11        }
12
13        for (Tenant__c newTenant : newList) {
14
15    }
```

Below the code editor is a tabs bar with "Logs" selected, followed by Tests, Checkpoints, Query Editor, View State, Progress, and Problems. At the bottom is a status bar with tabs for User, Application, Operation, Time, Status, Read, and Size.

Apex logic:

```
public class testHandler {
    public static void preventInsert(List<Tenant__c> newList) {
        Set<Id> existingPropertyIds = new Set<Id>();
        for (Tenant__c existingTenant : [SELECT Id, Property__c FROM Tenant__c WHERE Property__c != null]) {
            existingPropertyIds.add(existingTenant.Property__c);
        }
        for (Tenant__c newTenant : newList) {
            if (newTenant.Property__c != null &&
existingPropertyIds.contains(newTenant.Property__c)) {
                newTenantaddError('A tenant can have only one property');
            }
        }
    }
}
```

3. Click on Save

Activity – 3 Testing the Trigger:

Go to AppLauncher search for Lease Management>>Select Tenant>>

Try to create new tenant with the existing property

LEASE MANAGEMENT

1. Go to AppLauncher search for Lease Management>>Select Tenant>>Click on New
2. Enter the details of User

The screenshot shows the 'Edit Madhu' page in the 'Lease Management' application. The page includes fields for Tenant Name (Madhu), Email (madhu@gmail.com), Phone (6281292720), status (Stay), and property (House - 2). The Owner is listed as Nagaradona Harshitha. At the bottom, there are buttons for Cancel, Save & New, and Save.

3. Select property as House-2
4. Click on Save

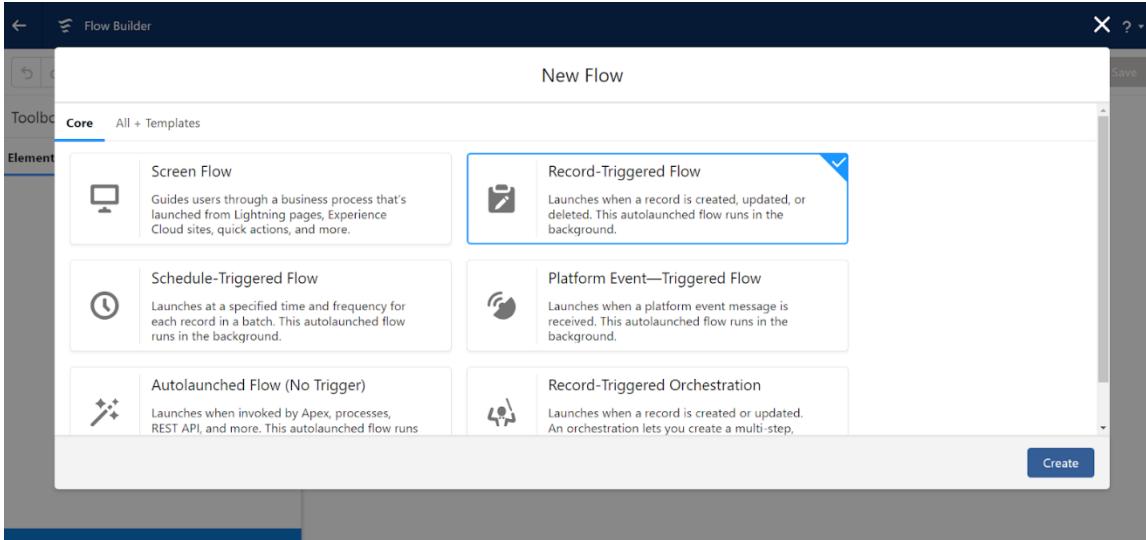
Milestone – 10

Flows

Activity – 1 Create Flow for monthly payment

1. Go to setup >> type Flow in quick find box >> Click on the Flow and Select the New Flow.
2. Select the record Triggered flow. Click on create.

LEASE MANAGEMENT



3. Under Object select "Payment for tenant". Click on A record is updated.

Start
Record-Triggered Flow

Object: **Payment for tenant** Edit
Trigger: **A record is updated**
Optimize for: **Actions and Related Records**

+ Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Payment for ...

End

Configure Start

Select Object
Select the object whose records trigger the flow when they're created, updated, or deleted.
* Object: Payment for tenant

Configure Trigger
* Trigger the Flow When:
 A record is created
 A record is updated
 A record is created or updated
 A record is deleted

Set Entry Conditions
Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.
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.

Set Entry Conditions

Under Condition Requirements

All Conditions are met

Field: check_for_payment__c operator: Equals value: Paid

5. Click on: Every time a record is updated and meets the condition requirements

LEASE MANAGEMENT

6. Click on: Actions and related records, done

Set Entry Conditions

Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

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.

Condition Requirements

All Conditions Are Met (AND)

Field: check_for_payment_c Operator: Equals Value: paid

+ Add Condition

When to Run the Flow for Updated Records ⓘ

Every time a record is updated and meets the condition requirements
 Only when a record is updated to meet the condition requirements

* Optimize the Flow for:

Fast Field Updates

Update fields on the record that triggers the flow to run. This high-performance flow runs *before* the record is saved to the database.

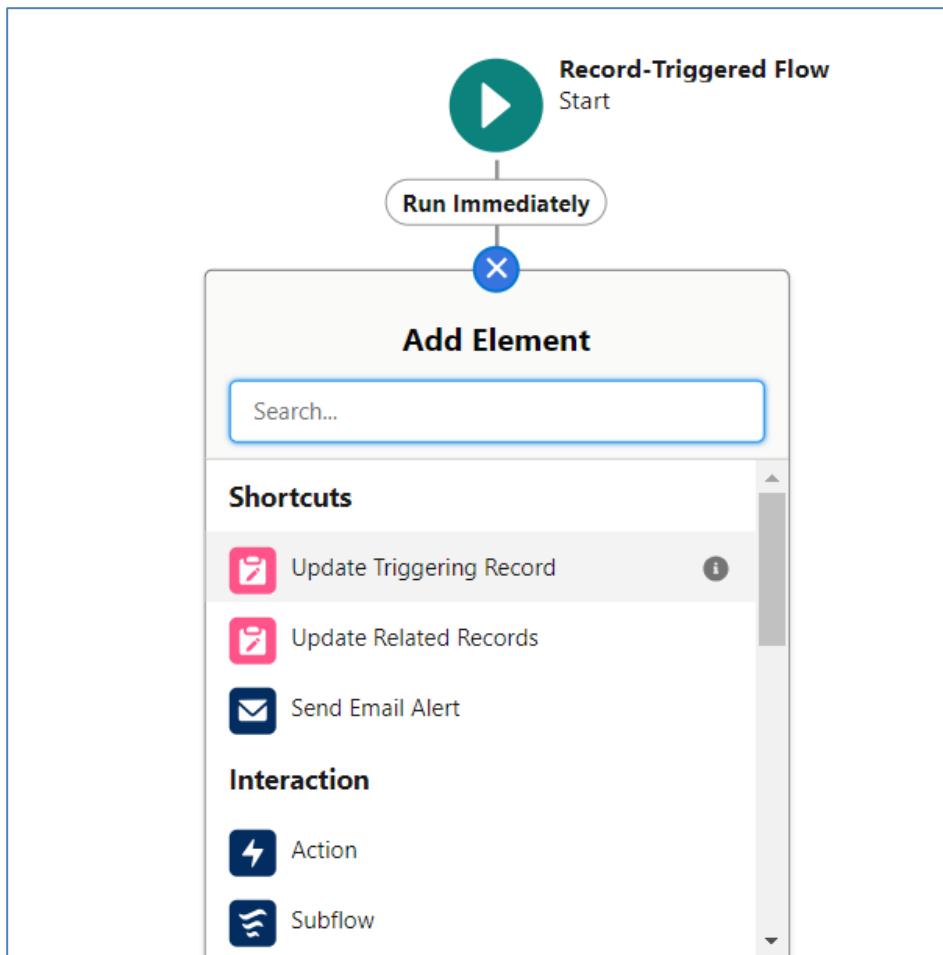
Actions and Related Records

Update any record and perform actions, like send an email. This more flexible flow runs *after* the record is saved to the database.

7.

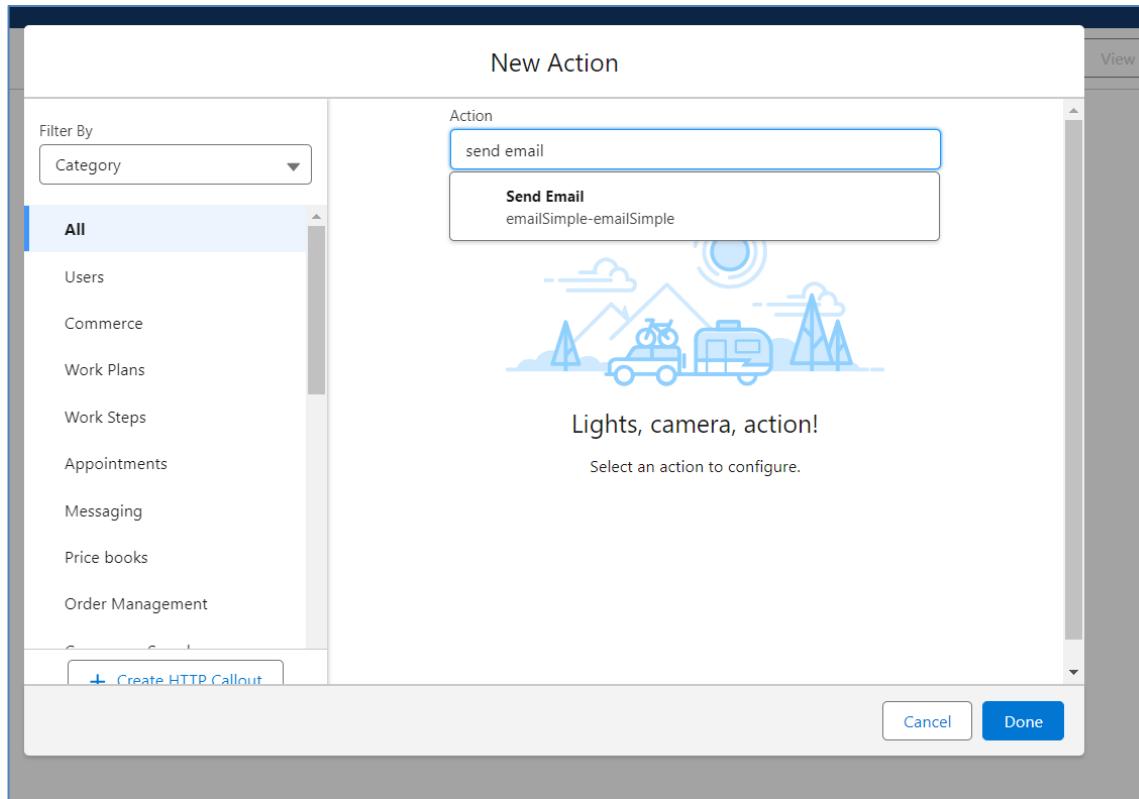
Under record trigger flow click on “+” icon and select action

LEASE MANAGEMENT



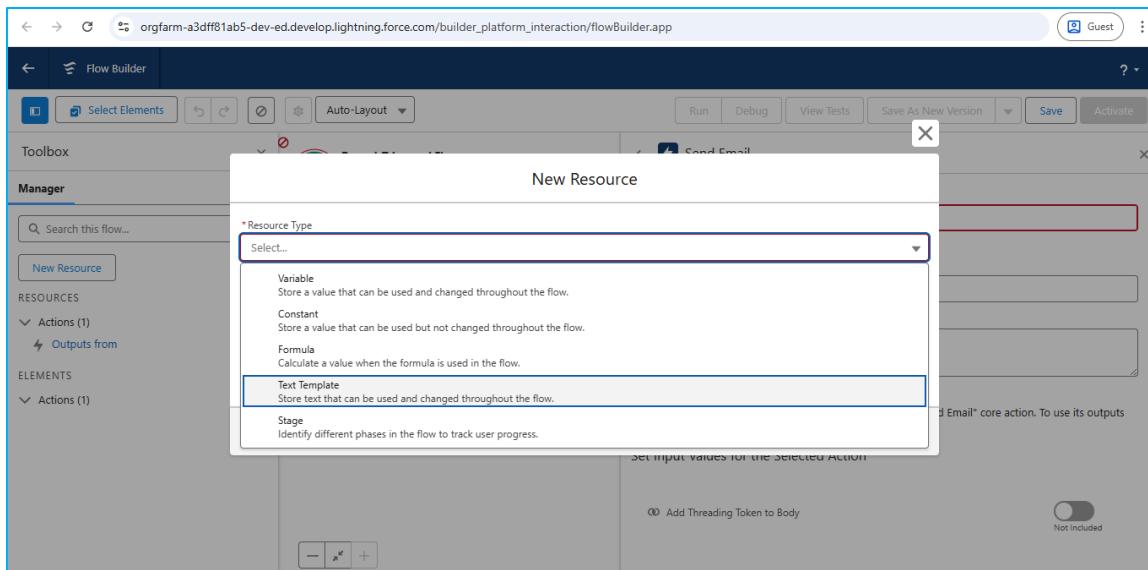
7. In action search for send email then click on send email (check below image)
8. Label: send email
9. API Name: send email

LEASE MANAGEMENT



10. Enable Body

11. Click on new resource



LEASE MANAGEMENT

Under resource type select “Text Template”

API Name : emailbody

Under body:(paste the below text)

Dear {! \$Record.Tenant__r. Name},

We hope this email finds you well. We are writing to inform you that we have successfully received your monthly payment. Thank you for your prompt and diligent payment.

12. Click Done.

13. Enable recipient Address List

Paste this? {! \$Record.Tenant__r. Email__c}

14. Click Done

15. Enable subject

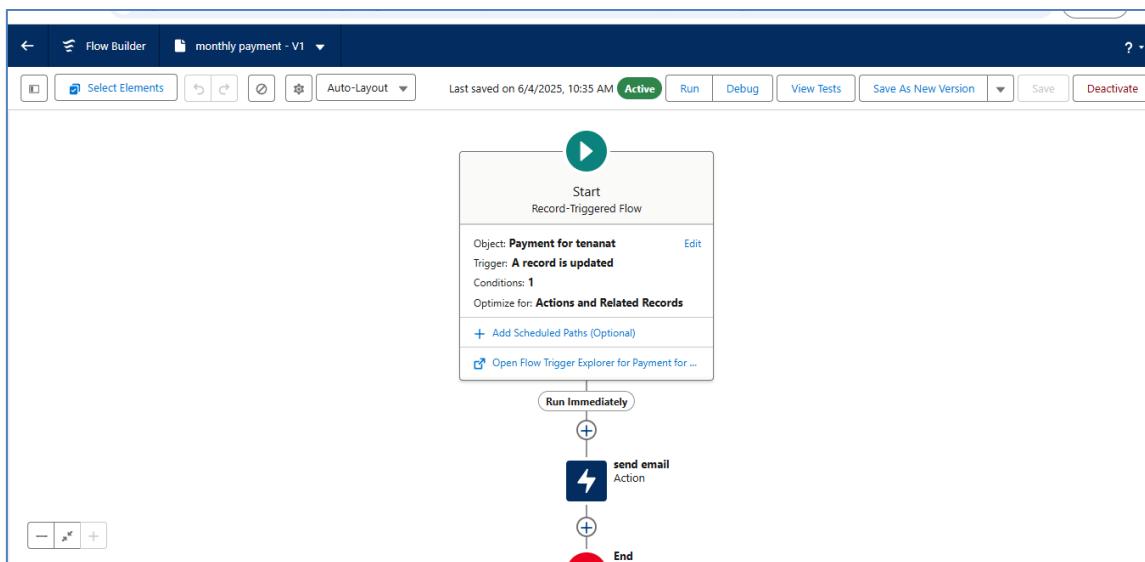
Paste this >> Confirmation of Successful Monthly Payment

16. Click on save

Flow label: monthly payment

Flow API Name: monthly payment

Click on activate



Milestone – 11

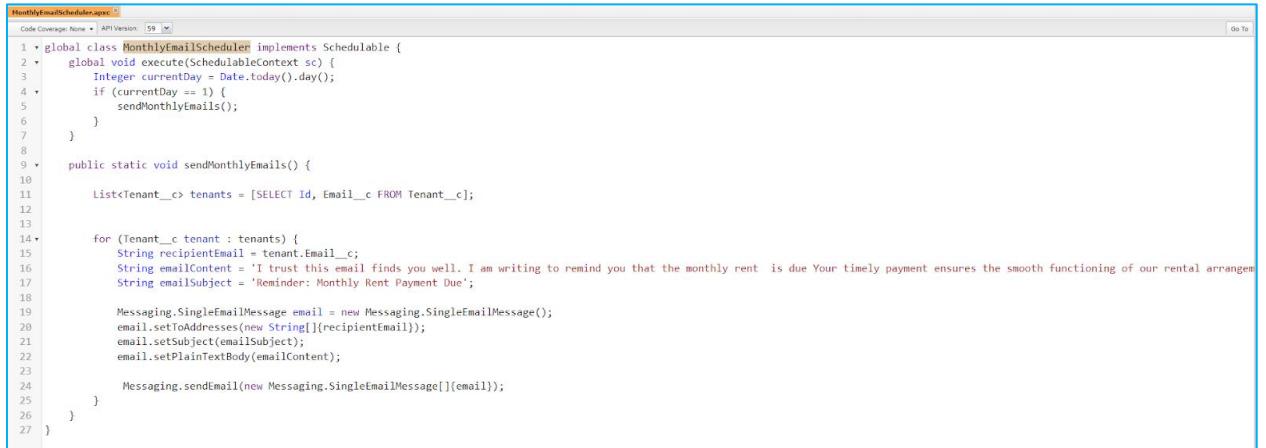
Schedule class

Create an Apex Class

To create a new Apex Class, follow the below steps:

LEASE MANAGEMENT

- 1.Click on the file >> New >> Apex Class.
- 2.Enter class name as MonthlyEmailScheduler.



The screenshot shows the code editor for the MonthlyEmailScheduler.apex class. The code implements the Schedulable interface and contains logic to send monthly emails to tenants if the current day is 1. It uses the Messaging.SingleEmailMessage class to construct and send the emails.

```
1 global class MonthlyEmailScheduler implements Schedulable {
2     global void execute(SchedulableContext sc) {
3         Integer currentDay = Date.today().day();
4         if (currentDay == 1) {
5             sendMonthlyEmails();
6         }
7     }
8
9     public static void sendMonthlyEmails() {
10
11         List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
12
13
14         for (Tenant__c tenant : tenants) {
15             String recipientEmail = tenant.Email__c;
16             String emailContent = 'I trust this email finds you well. I am writing to remind you that the monthly rent is due. Your timely payment ensures the smooth functioning of our rental arrangement and helps maintain a positive living environment for all.';
17             String emailSubject = 'Reminder: Monthly Rent Payment Due';
18
19             Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
20             email.setToAddresses(new String[]{recipientEmail});
21             email.setSubject(emailSubject);
22             email.setPlainTextBody(emailContent);
23
24             Messaging.sendEmail(new Messaging.SingleEmailMessage[]{email});
25         }
26     }
27 }
```

Apex logic:

```
global class MonthlyEmailScheduler implements Schedulable {
    global void execute(SchedulableContext sc) {
        Integer currentDay = Date.today().day();
        if (currentDay == 1) {
            sendMonthlyEmails();
        }
    }

    public static void sendMonthlyEmails() {
        List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
        for (Tenant__c tenant : tenants) {
            String recipientEmail = tenant.Email__c;
            String emailContent = 'I trust this email finds you well. I am writing to remind you that the monthly rent is due. Your timely payment ensures the smooth functioning of our rental arrangement and helps maintain a positive living environment for all.';
            String emailSubject = 'Reminder: Monthly Rent Payment Due';
            Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
            email.setToAddresses(new String[]{recipientEmail});
            email.setSubject(emailSubject);
            email.setPlainTextBody(emailContent);
            Messaging.sendEmail(new Messaging.SingleEmailMessage[]{email});
        }
    }
}
```

LEASE MANAGEMENT

```
}
```

```
}
```

```
}
```

3. Save the code

Activity – 2 Schedule Apex class:

1. Enter Apex class in quick find box
2. Select schedule Apex

The screenshot shows the Salesforce Setup Apex Classes page. The left sidebar has a search bar with 'apex' typed in. Under 'Custom Code', 'Apex Classes' is selected, showing two entries:

Action	Name	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By	Has Trace Flags
Edit Del Security	MonthlyEmailScheduler		63.0	Active	1,128	Nagaradona Harshitha, 6/3/2025, 10:17 PM	<input type="checkbox"/>
Edit Del Security	testHandler		63.0	Active	584	Nagaradona Harshitha, 6/2/2025, 11:39 PM	<input type="checkbox"/>

Below the table, there's a section titled 'Dynamic Apex Classes' with the following text: 'Dynamic Apex extends your programming reach by interacting with Lightning Platform components.' A table below it shows no records.

LEASE MANAGEMENT

3. Enter job Name: MonthlyEmailScheduler
4. Apex class: MonthlyEmailScheduler
5. Frequency: Monthly====>select on day 1
6. Start date: 04/12/2023
7. End date: 04/01/2024
8. Preferred start time: 09:00 am
9. save

Schedule Apex

Schedule an Apex class that implements the 'Schedulable' interface to be automatically executed on a weekly or monthly interval.

Save Cancel

Job Name: MonthlyEmailScheduler

Apex Class: MonthlyEmailScheduler

Schedule Apex Execution

Frequency: Weekly Monthly On day of every month On Sunday of every month

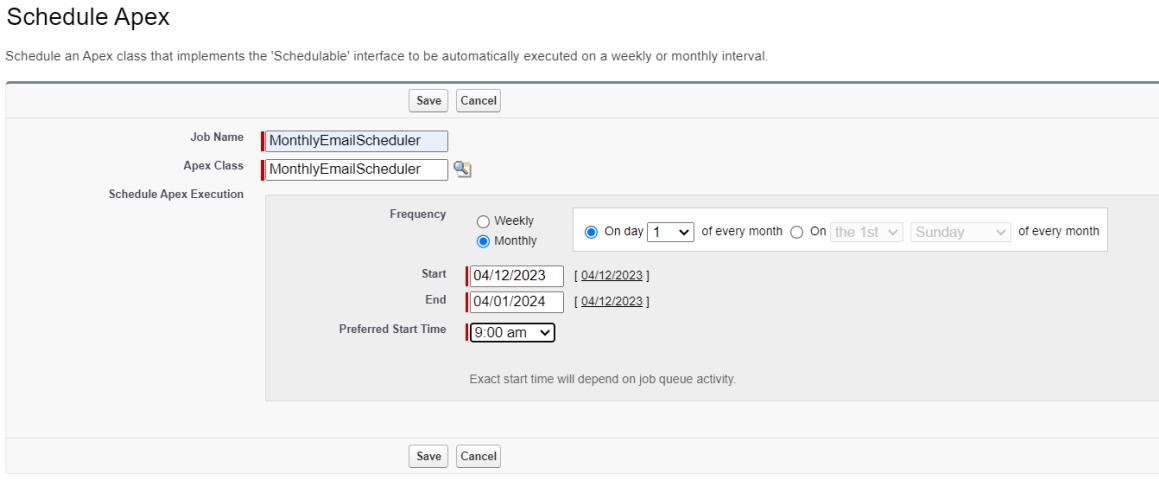
Start: [04/12/2023]

End: [04/12/2023]

Preferred Start Time:

Exact start time will depend on job queue activity.

Save Cancel



LEASE MANAGEMENT

From the set up

1.Go to ObjectManager>>Select tenant>>Click on page layouts

2.Go to the buttons tab

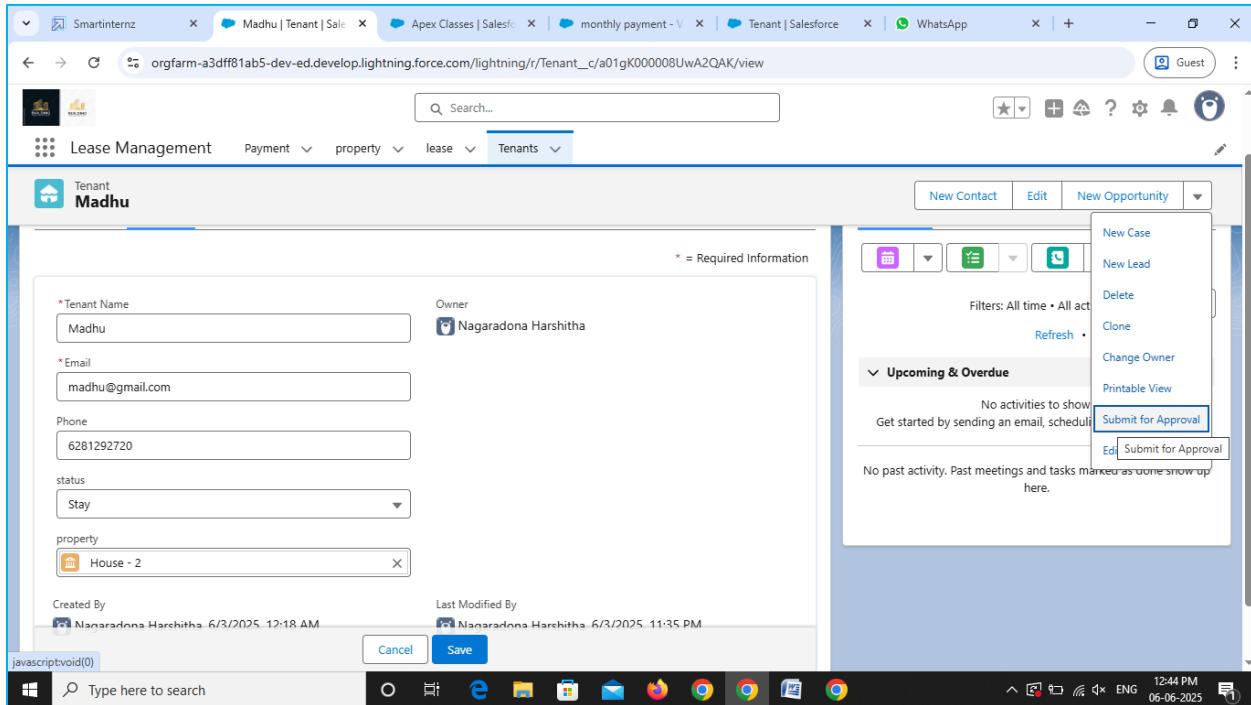
3.Drag submit for approval towards the tenant detail

The screenshot shows the Salesforce Setup interface for managing page layouts. The left sidebar lists categories such as Details, Fields & Relationships, Page Layouts (which is selected), Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts. The main content area displays the 'Tenant' page layout configuration. A modal window is open over the layout editor, showing a grid of actions: Change Owner, Edit Labels, Sharing, Clone, Printable View, Delete, and Send Survey. The 'Submit for Approval' button is specifically highlighted with a blue border, indicating it is being moved or configured. Below the modal, there is a note about overriding predefined actions and a 'Tenant Detail' section with standard buttons for Edit, Delete, Clone, Change Owner, Change Record Type, Printable View, Sharing, Sharing Hierarchy, Submit for Approval, and Edit Labels.

LEASE MANAGEMENT

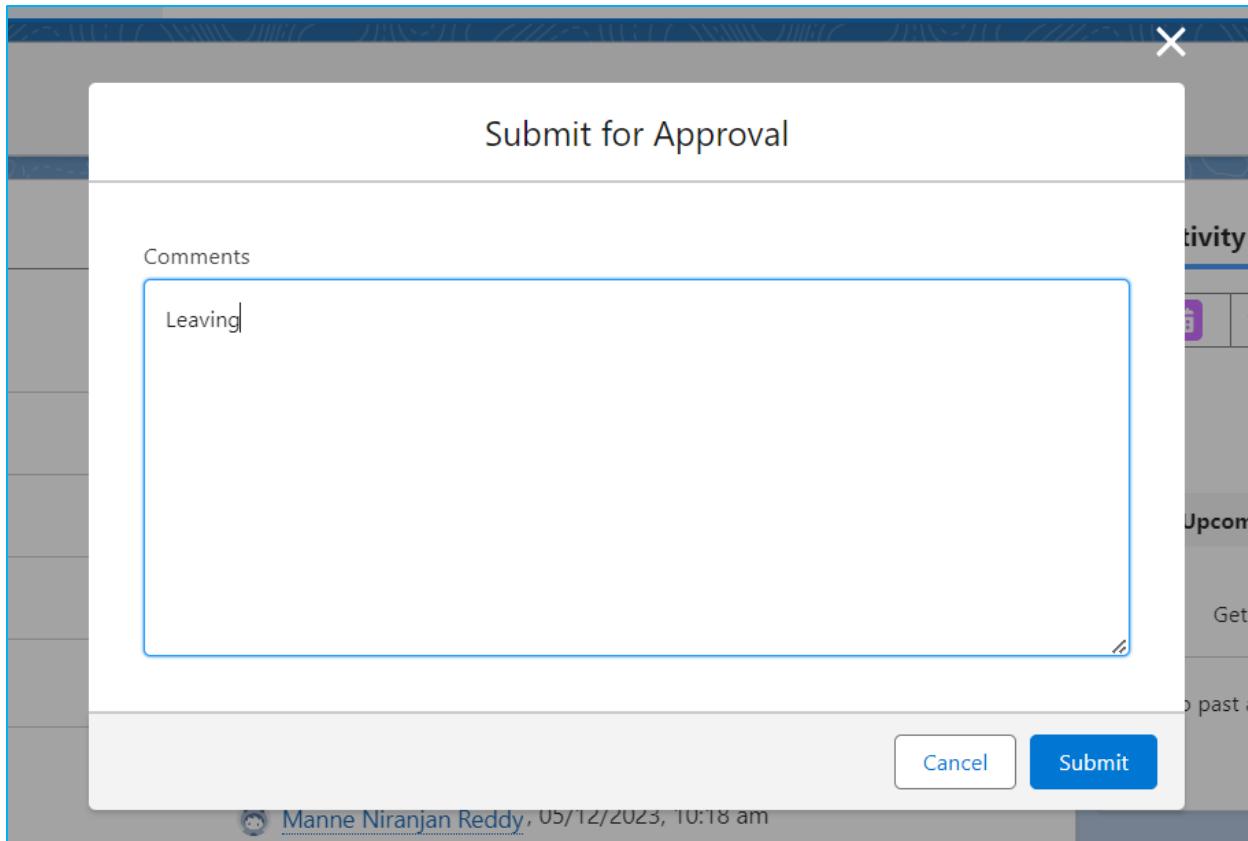
1.Go to AppLauncher>>Search Lease Management>>Select Tenant

2.Click on the user created



3.Enter any comment am\nd click on submit

LEASE MANAGEMENT



A screenshot of a Salesforce Lease Management page. The top navigation bar includes 'Lease Management', 'Payment', 'property', 'lease', 'Tenants', and 'Madhu Approval'. A sub-header says 'Approval Request Tenant Approval Pending'. The details section shows:

Submitter	Date Submitted	Actual Approver	Assigned To
Nagaradona Harshitha	Jun 6, 2025	Nagaradona Harshitha	Nagaradona Harshitha

The 'Details' section contains an 'Approval Details' table with one row:

Tenant Name
Madhu

To the right, a 'Notifications' sidebar lists:

- Nagaradona Harshitha is requesting approval for tenant**
Tenant Name: Madhu
a few seconds ago
- Approval request for the tenant is rejected**
Madhu
Jun 5, 2025, 10:46 AM
- New Guidance Center learning resource available**
Set Up Accounts & Contacts
Start storing information about your customers with accounts and contacts.
Jun 3, 2025, 9:30 AM
- New Guidance Center learning resource available**
Define Your Sales Process
Learn how to guide reps through the sales process.
Jun 3, 2025, 9:30 AM

LEASE MANAGEMENT

Click on that notification
click on approve
Give any comment and submit

The screenshot shows a Salesforce Lightning interface for a 'Lease Management' process. A specific step titled 'Process Instance Step' is displayed, showing a 'Tenant Approval' record. The status is 'Approved'. The details section shows the submitter is 'Nagaradona Harshitha', the date submitted is 'Jun 6, 2025', the actual approver is 'Nagaradona Harshitha', and the assigned to field is also 'Nagaradona Harshitha'. To the right, a 'Notifications' sidebar lists several recent notifications:

- Approval request for the tenant is approved** Madhu (a few seconds ago)
- Approval request for the tenant is rejected** Madhu (Jun 5, 2025, 10:46 AM)
- New Guidance Center learning resource available** Set Up Accounts & Contacts (Start storing information about your customers with accounts and contacts. Jun 3, 2025, 9:30 AM)
- New Guidance Center learning resource available** Define Your Sales Process (Learn how to guide reps through the sales process. Jun 3, 2025, 9:30 AM)

You will find notification like this and you will get an email check

Testing the Reject process

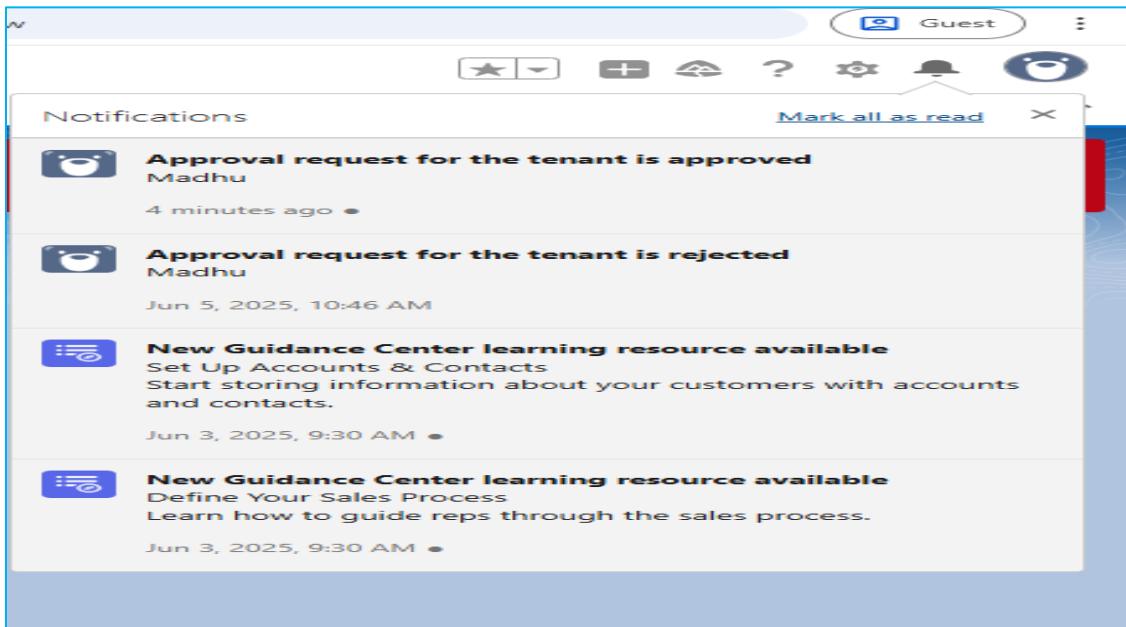
Enter any comment and click on submit

Click on that notification

Click on reject

LEASE MANAGEMENT

Give any comment and submit



You will find notification like this and you will get an email check

Outputs:

Property tab:

LEASE MANAGEMENT

The screenshot shows the 'Lease Management' application interface. On the left, there's a sidebar with a 'Recently Viewed' section for 'property'. The main area is titled 'Edit House - 2'. It contains fields for 'Name' (House - 2), 'Address' (Tirupati), 'Type' (1BHK), and 'sqft' (120). The 'Owner' field shows 'Nagaradona Harshitha'. At the bottom, it shows 'Created By' and 'Last Modified By' both as 'Nagaradona Harshitha, 6/3/2025, 12:16 AM'. There are 'Cancel', 'Save & New', and 'Save' buttons at the bottom.

Payments tab:

The screenshot shows the 'Lease Management' application interface. On the left, there's a sidebar with a 'Recently Viewed' section for 'Payment'. The main area is titled 'Edit new'. It contains fields for 'Payment Name' (new), 'Payment date' (6/3/2025), 'Amount' (empty), 'check for payment' (Paid), 'Email' (empty), and 'property' (Search property...). The 'Owner' field shows 'Nagaradona Harshitha'. At the bottom, it shows 'Created By' and 'Last Modified By' both as 'Nagaradona Harshitha, 6/3/2025, 12:00 AM'. There are 'Cancel', 'Save & New', and 'Save' buttons at the bottom.

LEASE MANAGEMENT

Lease tab:

The screenshot shows the 'Lease Management' application. In the top navigation bar, the 'Lease' tab is selected. A modal window titled 'New lease' is open, prompting for lease information. The required fields are highlighted with a yellow background: 'Lease Name' (containing 'HOUSE 1'), 'Start Date' (containing '6/18/2025'), 'End Date' (containing '7/18/2025'), and 'Property' (containing 'House - 1'). The 'Owner' field is populated with 'Nagaradona Harshitha'. At the bottom of the modal are 'Cancel', 'Save & New', and 'Save' buttons.

Tenants tab:

The screenshot shows the 'Lease Management' application. In the top navigation bar, the 'Tenants' tab is selected. A modal window titled 'Edit Madhu' is open, displaying tenant details. The required field 'Tenant Name' is highlighted with a blue border and contains the value 'Madhu'. Other visible fields include 'Email' (containing 'madhu@gmail.com'), 'Phone' (containing '6281292720'), 'Status' (set to 'Stay'), and 'Property' (containing 'House - 2'). The 'Owner' field is also populated with 'Nagaradona Harshitha'. At the bottom of the modal are 'Cancel', 'Save & New', and 'Save' buttons. Below the modal, a message states 'Nothing to see here' and 'There's nothing in your list yet. Try adding a new record.'

LEASE MANAGEMENT

GithubLink:

<https://github.com/harshithan0048/Lease-Management-LTVIP2025TMID29567>

Demo Video Link :

https://drive.google.com/file/d/1Men21Qt8LJNlyJAVvxg1oUtdMHM-sZP7/view?usp=drive_link