

LeaseManagement

**College Name: AAAYYAN THIRUVALLUVAR ARTS
AND SCIENCE COLLEGE**

College Code: BRU4M

TEAM ID : NM2025TMID23467

TEAM MEMBERS : 4

Team Leader : BARANEEDHARAN B

Email: bbaraneearshad@gmail.com

TeamMember1 : VISHOK S

Email: emailsvishoksvishok@gmail.com

TeamMember2 : R A M A K R I S H N A C

Email: ramakrishna4567@gmail.com

TeamMember3: KESAVARAMBABU S

Email: kessvarmragul@gmail.com

1.INTRODUCTION

Project Overview

The Lease Management System is a Sales force –based application designed to stream line the processes associated with leasing real estate properties. It handles tenant management, lease

contracts, payments, and communication with automation features such as flows, approval processes, and email alerts.



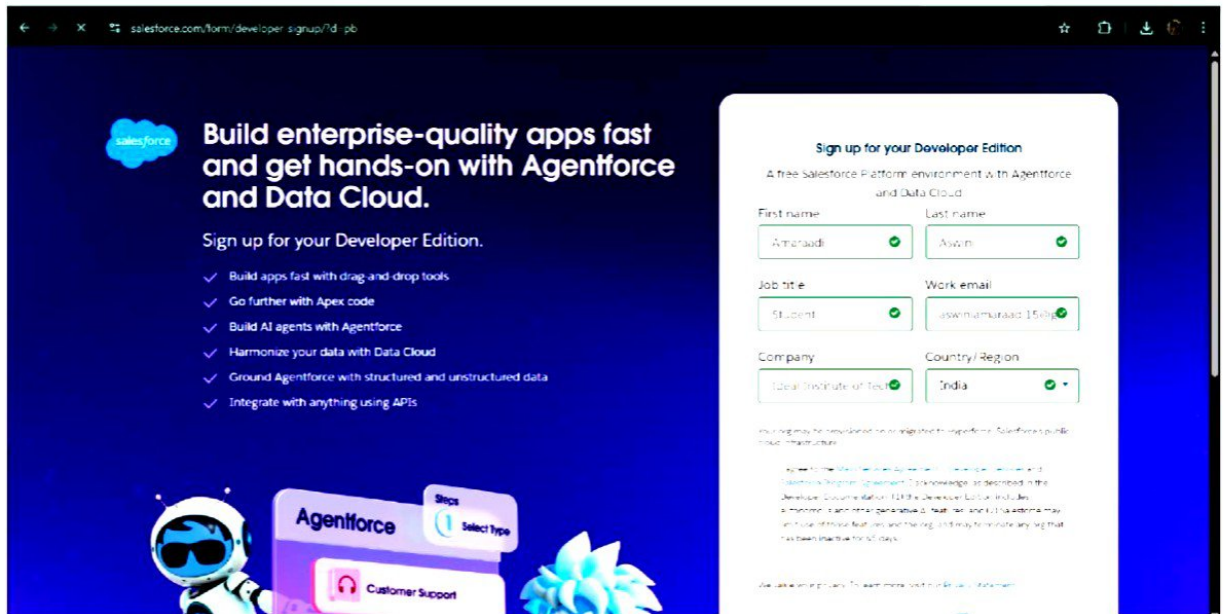
Purpose

The main objective of the project is to enable organizations to efficiently manage properties, tenants, and lease-related activities. It reduces manual intervention, improves accuracy, and ensures better compliance and communication.

DEVELOPMENT PHASE

Creating Developer Account :

By using this URL - <https://www.salesforce.com/form/developer-signup/?d=pb>



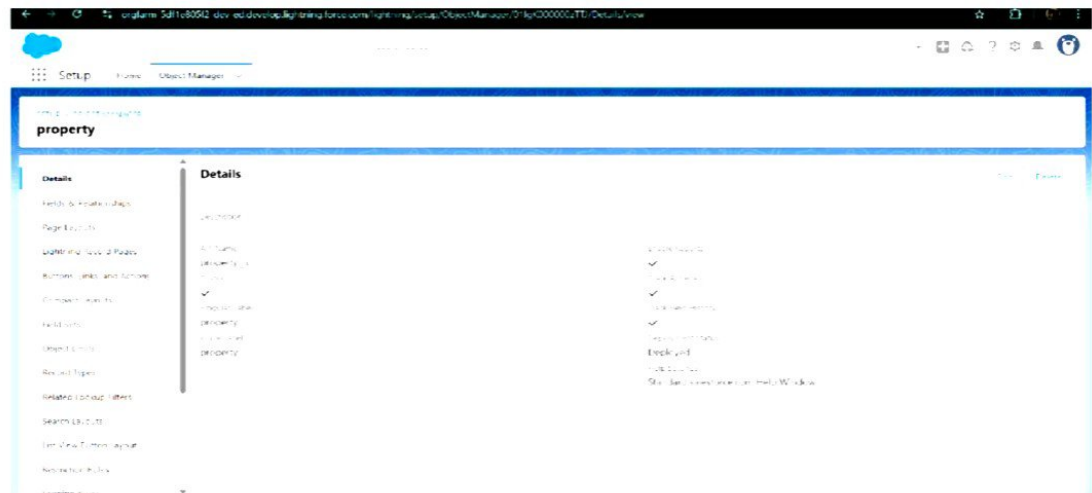
The image shows the Salesforce Developer Edition sign-up page. On the left, there's a blue background with the Salesforce logo and text: "Build enterprise-quality apps fast and get hands-on with Agentforce and Data Cloud." Below this, it says "Sign up for your Developer Edition." and lists five benefits: "Build apps fast with drag-and-drop tools", "Go further with Apex code", "Build AI agents with Agentforce", "Harmonize your data with Data Cloud", and "Ground Agentforce with structured and unstructured data". At the bottom left, there's an illustration of an astronaut and a robot labeled "Agentforce".

On the right, there's a white sign-up form titled "Sign up for your Developer Edition". It includes a sub-header: "A free Salesforce Platform environment with Agentforce and Data Cloud." The form has the following fields:

- First name:
- Last name:
- Job title:
- Work email:
- Company:
- Country/Region:

Below the form, there's a disclaimer: "Your org may be resuspended or migrated to a different Salesforce public cloud infrastructure." and a link to the "Agree to the Developer Edition Agreement". At the bottom, it says "Get started with your Developer Edition." and "Get started with your Developer Edition."

- Created objects : Property, Tenant ,Lease, Payment



The image shows the Salesforce Object Manager setup page for the 'property' object. The page is titled 'property' and has a sub-header 'Details'. On the left, there's a sidebar with navigation links: 'Setup', 'Tools', 'Object Manager', and 'Details'. The 'Details' section is expanded, showing a list of fields and their data types. The fields are: 'id' (Text), 'name' (Text), 'description' (Text), 'status' (Text), 'created_date' (Text), 'created_by' (Text), 'last_modified_date' (Text), 'last_modified_by' (Text), 'is_active' (Text), 'is_deleted' (Text), 'created_date_time' (Text), 'created_by_name' (Text), 'last_modified_date_time' (Text), 'last_modified_by_name' (Text), 'is_active_name' (Text), and 'is_deleted_name' (Text). The 'is_active' and 'is_deleted' fields are marked as 'Required' and 'Unique'.

On the right, there's a 'Details' section with a table of fields and their data types. The table has two columns: 'Field Name' and 'Data Type'. The fields are: 'id' (Text), 'name' (Text), 'description' (Text), 'status' (Text), 'created_date' (Text), 'created_by' (Text), 'last_modified_date' (Text), 'last_modified_by' (Text), 'is_active' (Text), 'is_deleted' (Text), 'created_date_time' (Text), 'created_by_name' (Text), 'last_modified_date_time' (Text), 'last_modified_by_name' (Text), 'is_active_name' (Text), and 'is_deleted_name' (Text). The 'is_active' and 'is_deleted' fields are marked as 'Required' and 'Unique'.

Setup

Home

Object Manager

Payment for tenant

Details

Fields & Relationships

7 Items, Sorted by Field Label

Quick Find

New

Deleted Fields

Field Dependencies

Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Amount	Amount__c	Number(18, 0)		
check_for_payment	check_for_payment__c	Picklist		
Created By	CreatedBy	Lookup(User)		
Last Modified By	LastModifiedBy	Lookup(User)		
Owner	OwnerId	Lookup(User, Group)		✓
Payment date	Payment_date__c	Date		
Payment Name	Name	Text(80)		✓

Setup

Home

Object Manager

lease

Details

Fields & Relationships

7 Items, Sorted by Field Label

Quick Find

New

Deleted Fields

Field Dependencies

Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedBy	Lookup(User)		
End date	End_date__c	Date		
Last Modified By	LastModifiedBy	Lookup(User)		
lease Name	Name	Text(80)		✓
Owner	OwnerId	Lookup(User, Group)		✓
property	property__c	Lookup(property)		✓
start date	start_date__c	Date		

← → ↺

org:arm-5df1e0052-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01kgK000000zTbN/Details/view

☆ 📁 🏠 ? ⚙️ 🔔 👤

☁️

Setup Home Object Manager

SETUP - OBJECT MANAGER

Tenant

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Component Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Streaming Files

Details

Description

API Name

Tenant_02

Label

Tenant

Security Model

Tenant

Deployable

Tenants

Deployed

Helping to

Standard Salesforce.com Help Window

Edit

Delete

← → ↺

org:arm-5df1e0052-dev-ed.develop.lightning.force.com/lightning/setup/ObjectManager/01kgK000000zTbN/Details/view

☆ 📁 🏠 ? ⚙️ 🔔 👤

☁️

Setup Home Object Manager

SETUP - OBJECT MANAGER

lease

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Component Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Streaming Files

Details

Description

API Name

lease_02

Label

lease

Security Model

lease

Deployable

lease

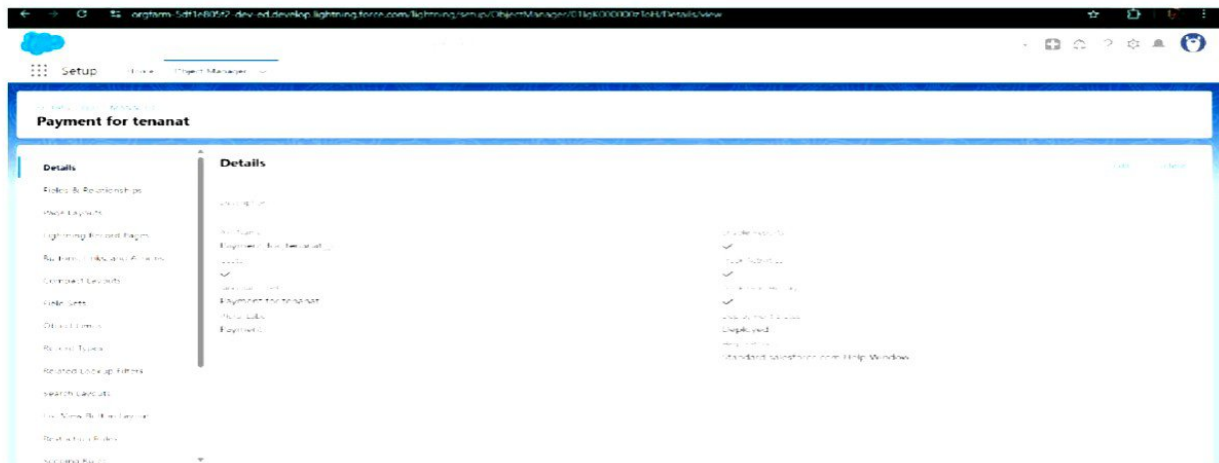
Deployed

Helping to

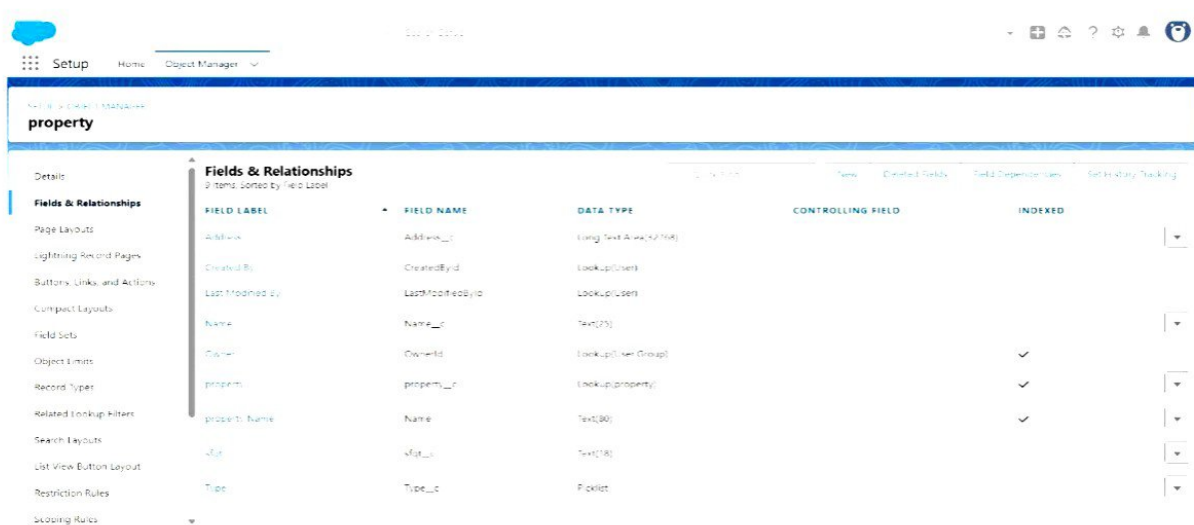
Standard Salesforce.com Help Window

Edit

Delete



● Configured fields and relationships



Send Email

The screenshot displays the Salesforce Flow Builder interface for a flow named 'monthly payment - V1'. The flow is a 'Record-Triggered Flow' that starts with a 'Start' node, followed by a 'Run Immediately' node, and then a 'Send Email' action. The 'Send Email' action is configured with the following details:

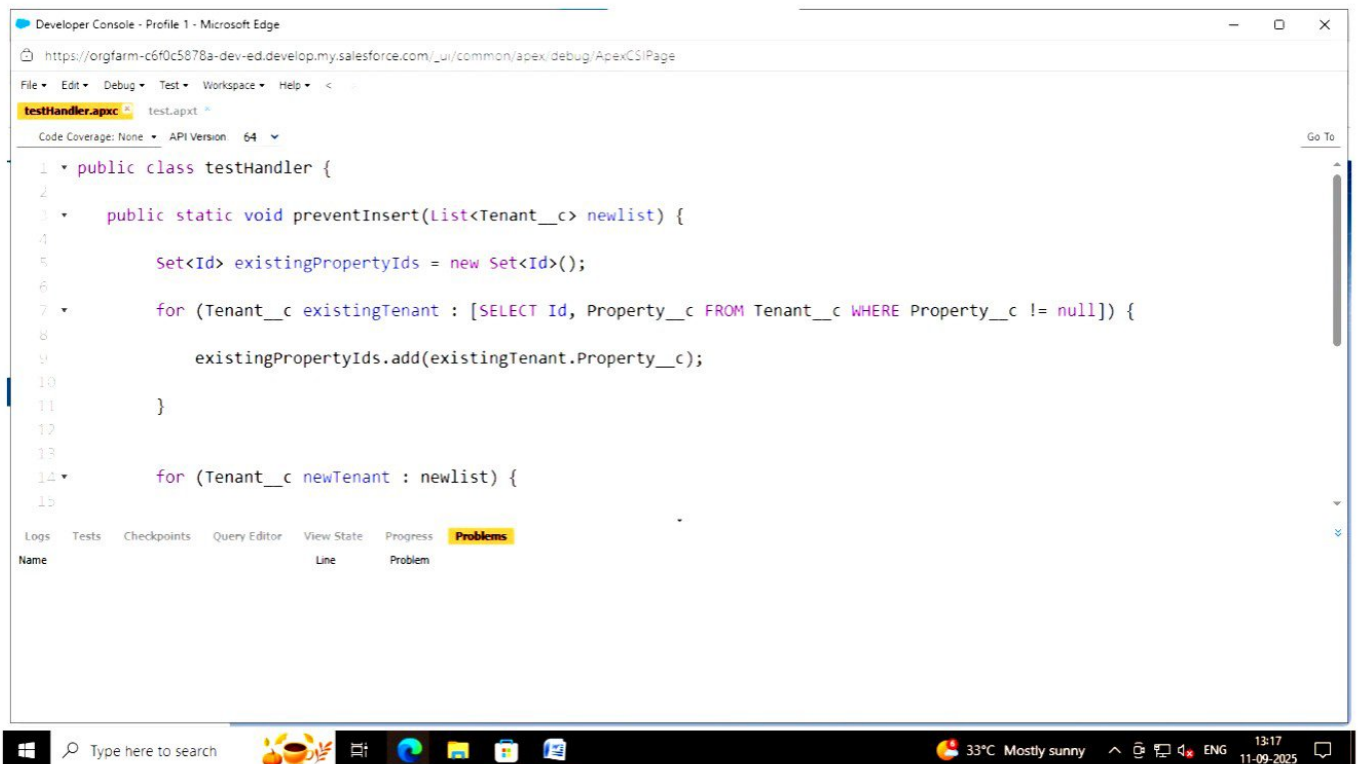
- Use:** send_email
- API Name:** send_email
- Description:**
- Set Input Values:**
- Configure Recipient Details:**
 - Recipient Address List:** A list of recipients, including 'Living Payment_for_Tenant_1' and 'Tenant_1 Email'.
 - Recipient Address Collection:** A collection of recipients, with a search bar for 'Enter value or search resources'.

The bottom of the screen shows the Windows taskbar with the search bar, task icons, and system tray information (33°C, Mostly sunny, 11-09-2025).

- Scheduled monthly reminder emails using Apex class

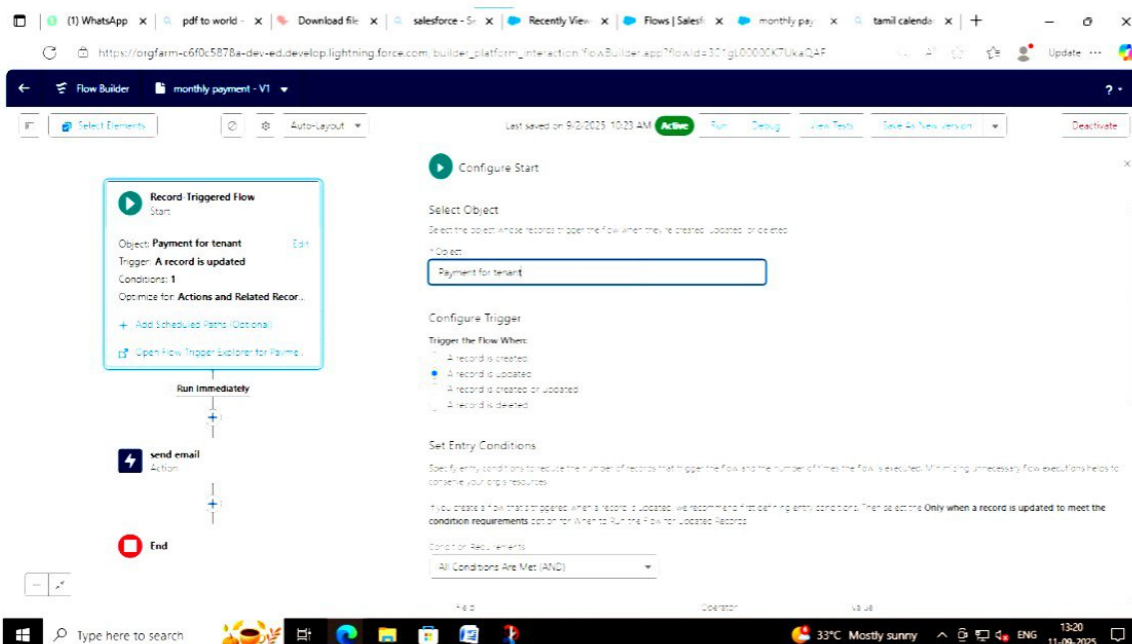
```
1 // This class is used to send scheduled reminder emails to tenants.
2 // It is a scheduled class that runs every month.
3 // It uses the EmailService to send emails.
4 // It uses the Tenant object to get tenant details.
5 // It uses the Tenant__c object to get tenant details.
6 // It uses the Tenant__c object to get tenant details.
7 // It uses the Tenant__c object to get tenant details.
8 // It uses the Tenant__c object to get tenant details.
9 // It uses the Tenant__c object to get tenant details.
10 // It uses the Tenant__c object to get tenant details.
11 // It uses the Tenant__c object to get tenant details.
12 // It uses the Tenant__c object to get tenant details.
13 // It uses the Tenant__c object to get tenant details.
14 // It uses the Tenant__c object to get tenant details.
15 // It uses the Tenant__c object to get tenant details.
16 // It uses the Tenant__c object to get tenant details.
17 // It uses the Tenant__c object to get tenant details.
18 // It uses the Tenant__c object to get tenant details.
19 // It uses the Tenant__c object to get tenant details.
20 // It uses the Tenant__c object to get tenant details.
21 // It uses the Tenant__c object to get tenant details.
22 // It uses the Tenant__c object to get tenant details.
23 // It uses the Tenant__c object to get tenant details.
24 // It uses the Tenant__c object to get tenant details.
25 // It uses the Tenant__c object to get tenant details.
26 // It uses the Tenant__c object to get tenant details.
27 // It uses the Tenant__c object to get tenant details.
28 // It uses the Tenant__c object to get tenant details.
29 // It uses the Tenant__c object to get tenant details.
30 // It uses the Tenant__c object to get tenant details.
31 // It uses the Tenant__c object to get tenant details.
32 // It uses the Tenant__c object to get tenant details.
33 // It uses the Tenant__c object to get tenant details.
34 // It uses the Tenant__c object to get tenant details.
35 // It uses the Tenant__c object to get tenant details.
36 // It uses the Tenant__c object to get tenant details.
37 // It uses the Tenant__c object to get tenant details.
38 // It uses the Tenant__c object to get tenant details.
39 // It uses the Tenant__c object to get tenant details.
40 // It uses the Tenant__c object to get tenant details.
41 // It uses the Tenant__c object to get tenant details.
42 // It uses the Tenant__c object to get tenant details.
43 // It uses the Tenant__c object to get tenant details.
44 // It uses the Tenant__c object to get tenant details.
45 // It uses the Tenant__c object to get tenant details.
46 // It uses the Tenant__c object to get tenant details.
47 // It uses the Tenant__c object to get tenant details.
48 // It uses the Tenant__c object to get tenant details.
49 // It uses the Tenant__c object to get tenant details.
50 // It uses the Tenant__c object to get tenant details.
51 // It uses the Tenant__c object to get tenant details.
52 // It uses the Tenant__c object to get tenant details.
53 // It uses the Tenant__c object to get tenant details.
54 // It uses the Tenant__c object to get tenant details.
55 // It uses the Tenant__c object to get tenant details.
56 // It uses the Tenant__c object to get tenant details.
57 // It uses the Tenant__c object to get tenant details.
58 // It uses the Tenant__c object to get tenant details.
59 // It uses the Tenant__c object to get tenant details.
60 // It uses the Tenant__c object to get tenant details.
61 // It uses the Tenant__c object to get tenant details.
62 // It uses the Tenant__c object to get tenant details.
63 // It uses the Tenant__c object to get tenant details.
64 // It uses the Tenant__c object to get tenant details.
65 // It uses the Tenant__c object to get tenant details.
66 // It uses the Tenant__c object to get tenant details.
67 // It uses the Tenant__c object to get tenant details.
68 // It uses the Tenant__c object to get tenant details.
69 // It uses the Tenant__c object to get tenant details.
70 // It uses the Tenant__c object to get tenant details.
71 // It uses the Tenant__c object to get tenant details.
72 // It uses the Tenant__c object to get tenant details.
73 // It uses the Tenant__c object to get tenant details.
74 // It uses the Tenant__c object to get tenant details.
75 // It uses the Tenant__c object to get tenant details.
76 // It uses the Tenant__c object to get tenant details.
77 // It uses the Tenant__c object to get tenant details.
78 // It uses the Tenant__c object to get tenant details.
79 // It uses the Tenant__c object to get tenant details.
80 // It uses the Tenant__c object to get tenant details.
81 // It uses the Tenant__c object to get tenant details.
82 // It uses the Tenant__c object to get tenant details.
83 // It uses the Tenant__c object to get tenant details.
84 // It uses the Tenant__c object to get tenant details.
85 // It uses the Tenant__c object to get tenant details.
86 // It uses the Tenant__c object to get tenant details.
87 // It uses the Tenant__c object to get tenant details.
88 // It uses the Tenant__c object to get tenant details.
89 // It uses the Tenant__c object to get tenant details.
90 // It uses the Tenant__c object to get tenant details.
91 // It uses the Tenant__c object to get tenant details.
92 // It uses the Tenant__c object to get tenant details.
93 // It uses the Tenant__c object to get tenant details.
94 // It uses the Tenant__c object to get tenant details.
95 // It uses the Tenant__c object to get tenant details.
96 // It uses the Tenant__c object to get tenant details.
97 // It uses the Tenant__c object to get tenant details.
98 // It uses the Tenant__c object to get tenant details.
99 // It uses the Tenant__c object to get tenant details.
100 // It uses the Tenant__c object to get tenant details.
```


Testhandler apex



Flows

Configure Start



Browser tabs: (1) WhatsApp, pdf to world - See, Download file | il, salesforce - Search, Recently Viewed, Classic Email Tem, tamil calendar 20, +

Address bar: https://orgfarm-c6f0c5878a-dev-ed.develop.lightning.force.com/lightning/setup/CommunicationTemplatesEmail/page?address=%2F00XgL000005PGB8...

Search Setup

Setup Home Object Manager

EMAIL TE

Email

Classic Email Templates

Lightning Email Templates

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

Classic Email Templates

Text Email Template

Tenant Email

Preview your email template below

Email Template Detail [Edit](#) [Delete](#) [Clone](#)

Email Templates from Salesforce	Unified Public Classic Email Templates	Available For Use	✓
Email Template Name	Tenant Email	Last Used Date	
Template Unique Name	Tenant_Email	Times Used	
Encoding	Unicode (UTF-8)		
Author	poovarasu.p [Change]		
Description			
Created By	poovarasu.p 9/2/2025 11:26 PM	Modified By	poovarasu.p 9/2/2025 11:26 PM

[Edit](#) [Delete](#) [Clone](#)

Email Template [Send Test and Verify Merge Fields](#)

Subject Urgent: Monthly Rent Payment Reminder

Plain Text Preview

Dear {Tenant__c Name}

javascriptsrcUp(%27%2F00XgL000005PGB8%3Fsetupid%3DCommunicationTemplatesEmail%26isdt%3Op1%27);

Windows taskbar: Type here to search, Rain coming, 13:06 11-09-2025

Approval processor

Browser tabs: (1) WhatsApp, pdf to world - See, Download file | il, salesforce - Search, Recently Viewed, Approval Proces, tamil calendar 20, +

Address bar: https://orgfarm-c6f0c5878a-dev-ed.develop.lightning.force.com/lightning/setup/ApprovalProcesses/home

Search Setup

Setup Home Object Manager

appro

Data

Mass Transfer Approval Requests

Feature Settings

Approval Settings

Process Automation

Approval Processes

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

Approval Processes

[Read the help topic](#)
[View the checklist](#)
[Create a custom user hierarchical relationship field](#)
[Create email templates](#)
[Create an approval process using either the Jump Start or Standard Wizard](#)
[Add Approval History Related List to all page layouts](#)
[Activate the process to deploy to your users](#)

Manage Approval Processes For: Tenant

A listing of both active and inactive approval processes for Tenants is displayed below. To create a new approval process, click Create New Approval Process then select Use Jump Start Wizard to set up your approval process in a few short steps. Or, select Use Standard Wizard to configure all approval options.

[Create New Approval Process](#)

Active Approval Processes [Refresh](#)

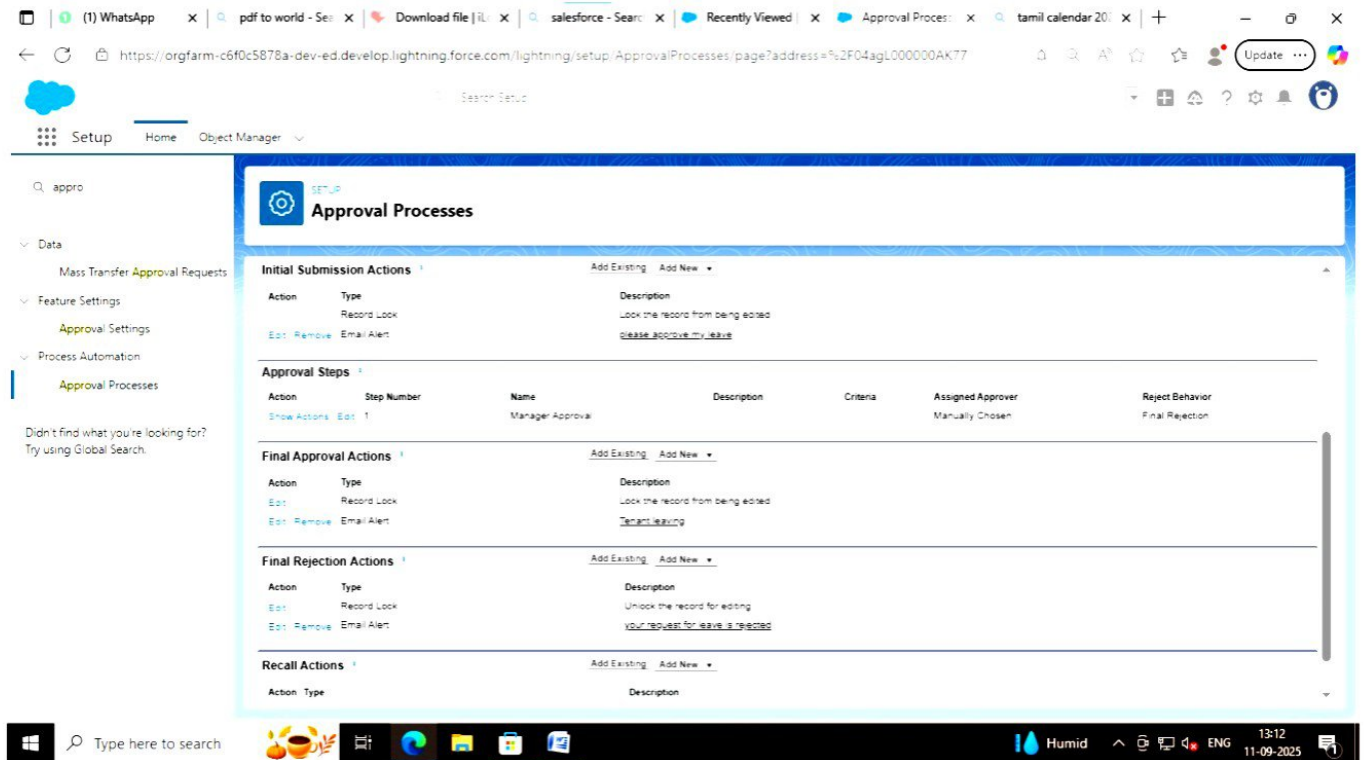
Action	Process Order	Approval Process Name	Description
Edit Deactivate	1	check for vacant	

Inactive Approval Processes

No approval processes available

Windows taskbar: Type here to search, Who is the world's ric..., 13:08 11-09-2025

Initial submission action-final approval action-final rejection action-approval set



The screenshot shows the Salesforce Setup page for Approval Processes. The left sidebar contains navigation links: Setup, Home, Object Manager, and a search bar. The main content area is titled "Approval Processes" and displays four sections: Initial Submission Actions, Approval Steps, Final Approval Actions, and Final Rejection Actions. Each section has a table of actions with columns for Action, Type, and Description. The Initial Submission Actions table has one row: "Record Lock" with description "Lock the record from being edited" and "Email Alert" with description "Please approve my leave". The Approval Steps table has one row: "Manager Approval" with description "Manager Approval" and "Manually Chosen" for the assigned approver. The Final Approval Actions table has one row: "Record Lock" with description "Lock the record from being edited" and "Email Alert" with description "Tenant leaving". The Final Rejection Actions table has one row: "Record Lock" with description "Unlock the record for editing" and "Email Alert" with description "Your request for leave is rejected". The Recall Actions section is empty. The bottom of the screen shows the Windows taskbar with the search bar, task icons, and system tray.

Action	Type	Description
Record Lock	Record Lock	Lock the record from being edited
Email Alert	Email Alert	Please approve my leave

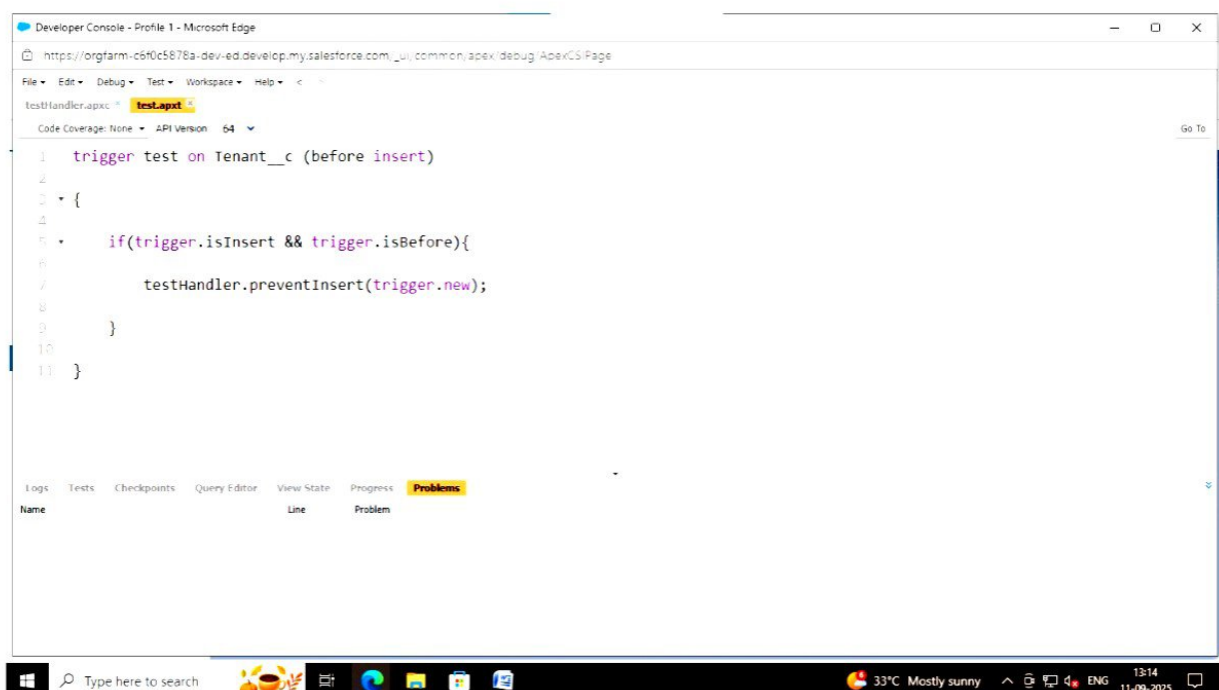
Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Manager Approval	1	Manager Approval	Manager Approval		Manually Chosen	Final Rejection

Action	Type	Description
Record Lock	Record Lock	Lock the record from being edited
Email Alert	Email Alert	Tenant leaving

Action	Type	Description
Record Lock	Record Lock	Unlock the record for editing
Email Alert	Email Alert	Your request for leave is rejected

Apex trigger

Test apex



The screenshot shows the Salesforce Developer Console with an Apex trigger named "test.apex" on the "Tenant__c" object. The trigger is a before insert trigger. The code is as follows:

```
1 trigger test on Tenant__c (before insert)
2 {
3     {
4     }
5     if(trigger.isInsert && trigger.isBefore){
6         testHandler.preventInsert(trigger.new);
7     }
8 }
9
10
11 }
```

The bottom of the screen shows the Windows taskbar with the search bar, task icons, and system tray.

Browser tabs: (1) WhatsApp, pdf to world - See, Download file | il, salesforce - Search, Recently Viewed, Classic Email Tem, tamil calendar 20, +

Address bar: https://orgfarm-c6f0c5878a-dev-ed.develop.lightning.force.com/lightning/setup/CommunicationTemplatesEmail/page?address=%2F00XgL000005PG5...

Setup | Home | Object Manager

EMAIL TE

Email

Classic Email Templates

Lightning Email Templates

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

Classic Email Templates

Email Template Detail Edit Delete Clone

Email Templates from Salesforce	Unified Public Classic Email Templates	Available For Use	✓
Email Template Name	Leave rejected	Last Used Date	
Template Unique Name	Leave_rejected	Times Used	
Encoding	Unicode (UTF-8)		
Author	poovarasu.p (Change)		
Description			
Created By	poovarasu.p 9/2/2025 11:24 PM	Modified By	poovarasu.p 9/2/2025 11:27 PM

Edit Delete Clone

Email Template Send Test and Verify Merge Fields

Subject Leave rejected

Plain Text Preview

Dear (!Tenant__c.Name),

I hope this email finds you well. Your contract has not ended. So we can't approve your leave your leave has rejected

javascriptsrcUp(%27%2F00XgL000005PG57%3Fsetupid%3DCommunicationTemplatesEmail%26isdtp%3Dp1%27);

Windows taskbar: Type here to search, Rain coming, 13:04 11-09-2025

Browser tabs: (1) WhatsApp, pdf to world - See, Download file | il, salesforce - Search, Recently Viewed, Classic Email Tem, tamil calendar 20, +

Address bar: https://orgfarm-c6f0c5878a-dev-ed.develop.lightning.force.com/lightning/setup/CommunicationTemplatesEmail/page?address=%2F00XgL000005PGD...

Setup | Home | Object Manager

EMAIL TE

Email

Classic Email Templates

Lightning Email Templates

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

Classic Email Templates

Text Email Template tenant payment Help for this Page

Preview your email template below

Email Template Detail Edit Delete Clone

Email Templates from Salesforce	Unified Public Classic Email Templates	Available For Use	✓
Email Template Name	tenant_payment	Last Used Date	
Template Unique Name	tenant_payment	Times Used	
Encoding	Unicode (UTF-8)		
Author	poovarasu.p (Change)		
Description			
Created By	poovarasu.p 9/2/2025 11:27 PM	Modified By	poovarasu.p 9/2/2025 11:28 PM

Edit Delete Clone

Email Template Send Test and Verify Merge Fields

Subject Confirmation of Successful Monthly Payment

Plain Text Preview

Dear (!Tenant__c.Email__c),

javascriptsrcUp(%27%2F00XgL000005PGD8%3Fsetupid%3DCommunicationTemplatesEmail%26isdtp%3Dp1%27);

Windows taskbar: Type here to search, Who is the world's ric..., 13:06 11-09-2025

Email Templates

Browser tabs: (1) WhatsApp, pdf to world - See, Download file | il, salesforce - Search, Recently Viewed, Classic Email Tem, tamil calendar 20, +

Address bar: https://orgfarm-c6f0c5878a-dev-ed.develop.lightning.force.com/lightning/setup/CommunicationTemplatesEmail/page?address=%2F00XgL000005PFqbl...

Search Setup

Setup Home Object Manager

EMAIL TEMPLATES

Email

Classic Email Templates

Lightning Email Templates

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

Classic Email Templates

Text Email Template

tenant leaving

Preview your email template below.

Email Template Detail		Edit	Delete	Clone
Email Templates from Salesforce	Unified Public Classic Email Templates			
Email Template Name	tenant leaving			
Template Unique Name	tenant_leaving			
Encoding	Unicode (UTF-8)			
Author	poovarasu.p [Change]			
Description				
Created By	poovarasu.p 9/2/2025 11:22 PM			
Modified By	poovarasu.p 9/2/2025 11:22 PM			

Available For Use: ☒

Last Used Date:

Times Used:

Email Template

Send Test and Verify Merge Fields

Subject: request for approve the leave

Plain Text Preview

Dear {Tenant__c.CreatedBy},

Browser tabs: (1) WhatsApp, pdf to world - See, Download file | il, salesforce - Search, Recently Viewed, Classic Email Tem, tamil calendar 20, +

Address bar: https://orgfarm-c6f0c5878a-dev-ed.develop.lightning.force.com/lightning/setup/CommunicationTemplatesEmail/page?address=%2F00XgL000005PFqbl...

Search Setup

Setup Home Object Manager

EMAIL TEMPLATES

Email

Classic Email Templates

Lightning Email Templates

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

Classic Email Templates

Text Email Template

Leave approved

Preview your email template below.

Email Template Detail		Edit	Delete	Clone
Email Templates from Salesforce	Unified Public Classic Email Templates			
Email Template Name	Leave approved			
Template Unique Name	Leave_approved			
Encoding	Unicode (UTF-8)			
Author	poovarasu.p [Change]			
Description				
Created By	poovarasu.p 9/2/2025 11:23 PM			
Modified By	poovarasu.p 9/2/2025 11:23 PM			

Available For Use: ☒

Last Used Date:

Times Used:

Email Template

Send Test and Verify Merge Fields

Subject: Leave approved

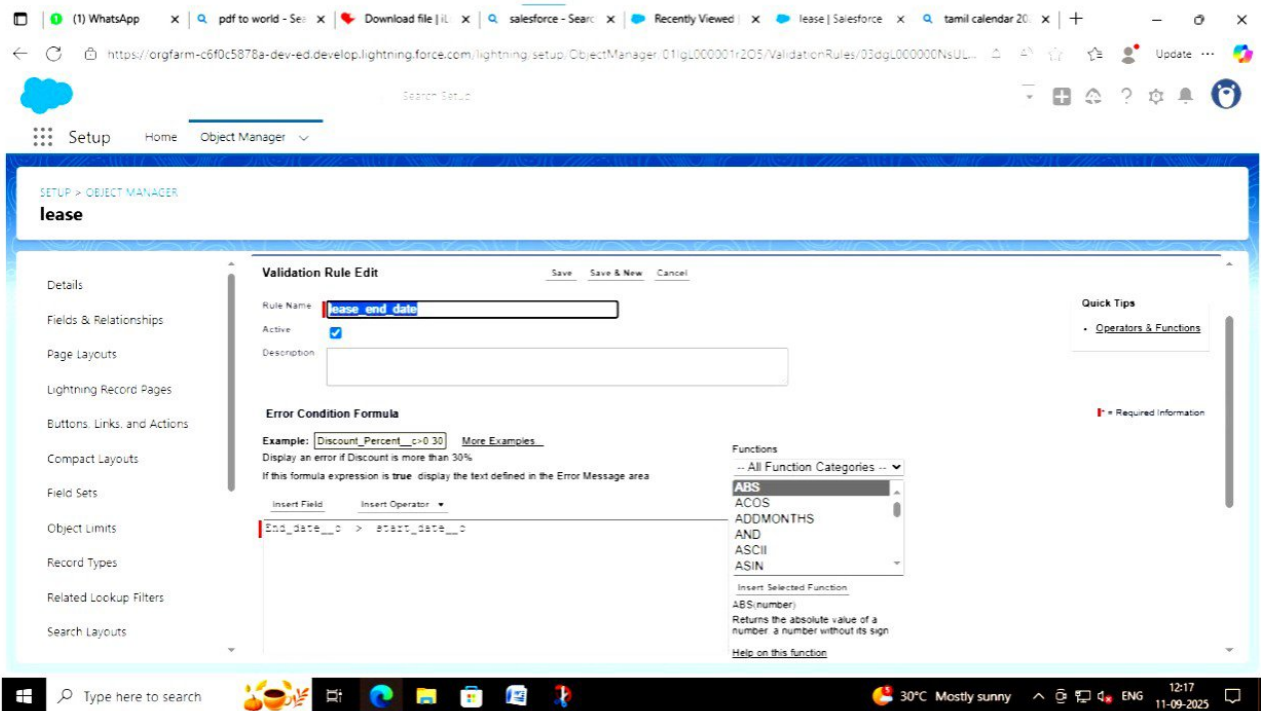
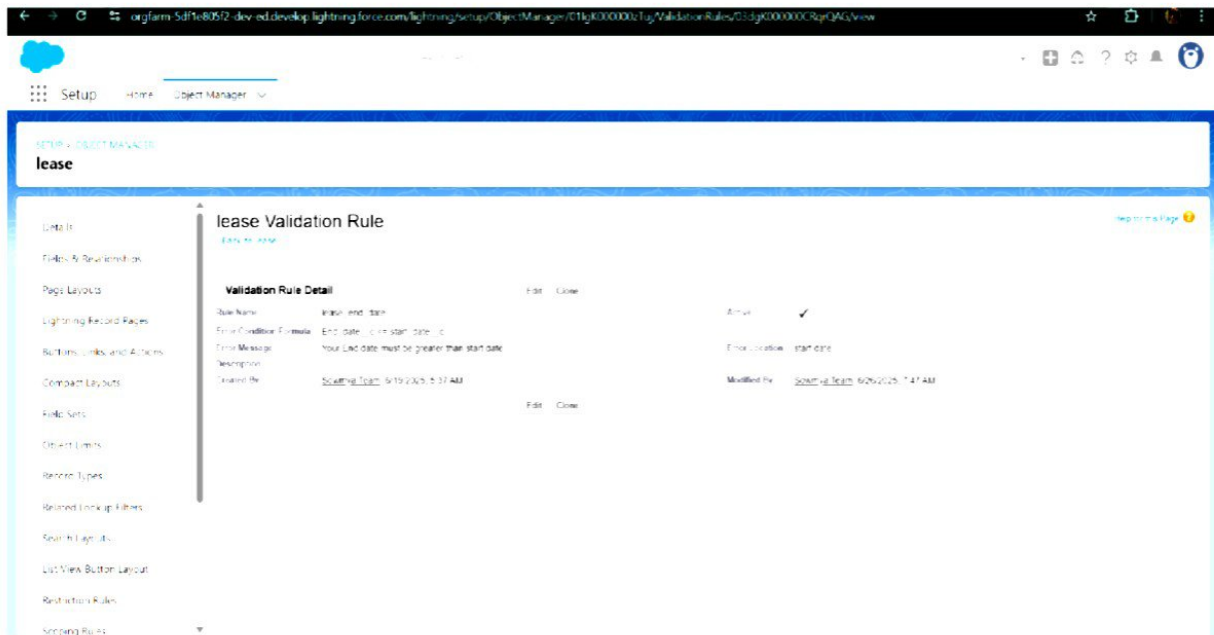
Plain Text Preview

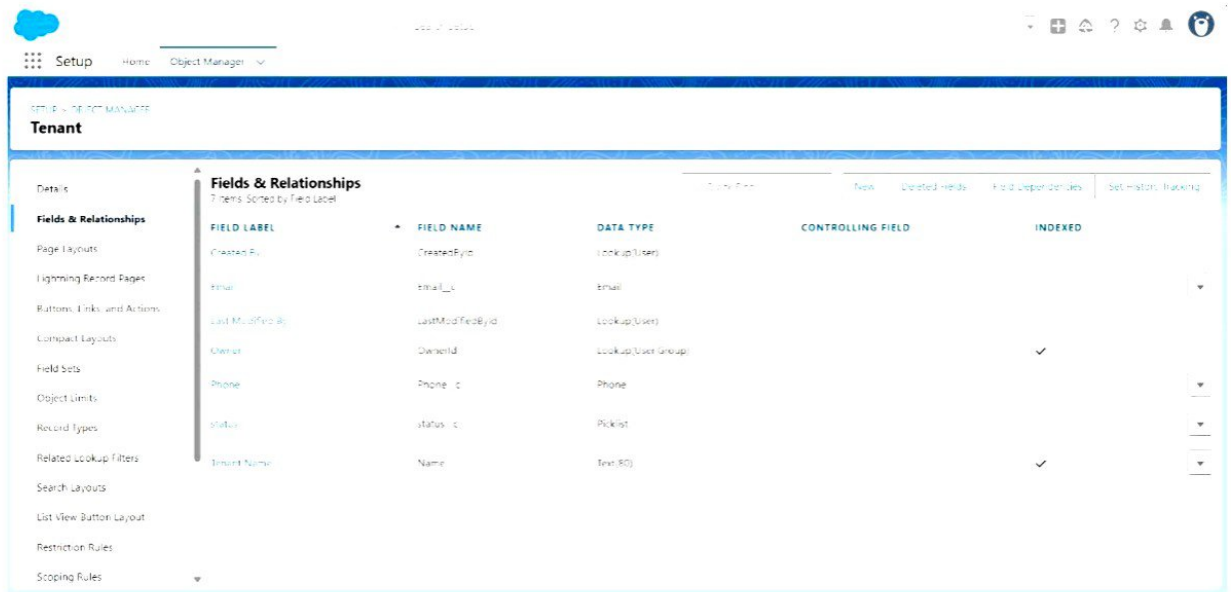
Dear {Tenant__c.Name},

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.

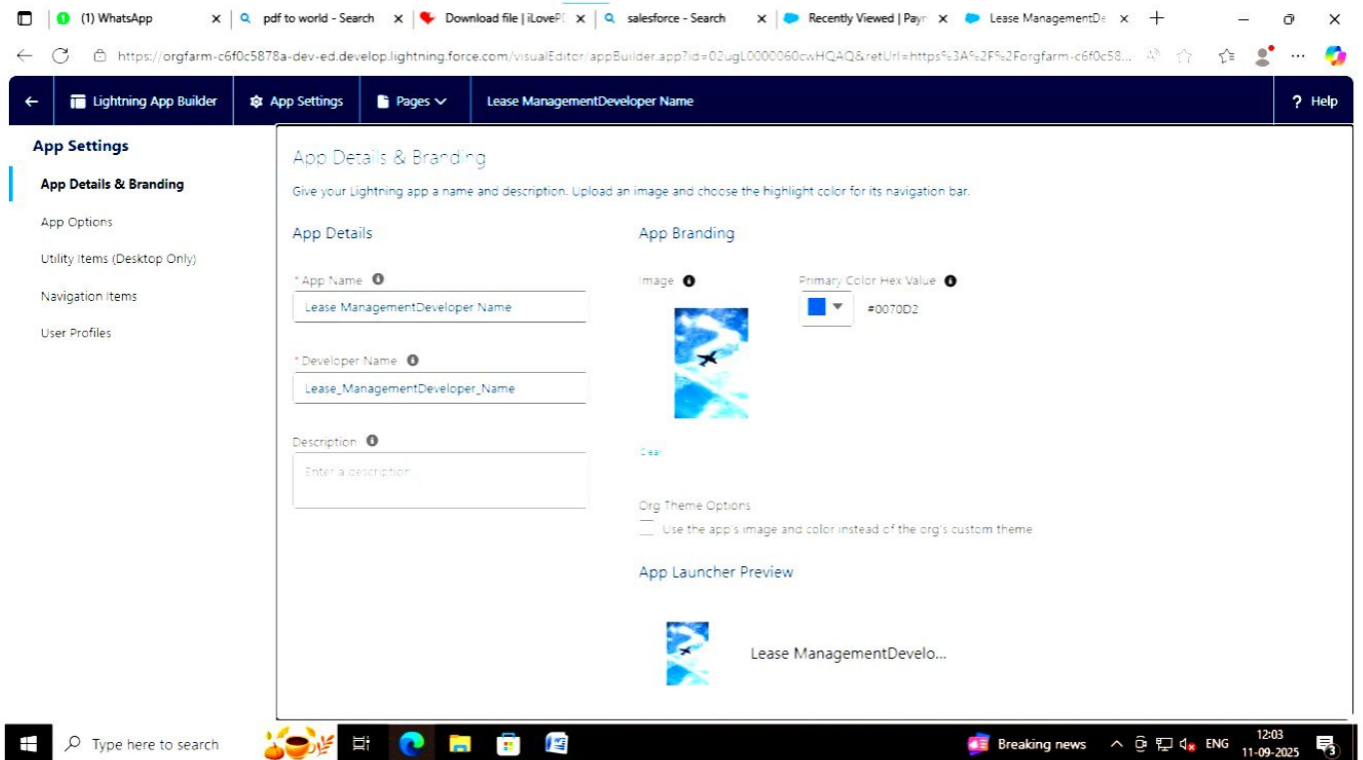
Your leave is approved. You can leave now.

To create a validation rule to a Lease Object





● Developed Lightning App with relevant tabs



```
Messaging.SingleEmailMessage email = new
Messaging.SingleEmailMessage(); email.setToAddresses(new
String[]{recipientEmail}); email.setSubject(emailSubject);
email.setPlainTextBody(emailContent);

    Messaging.sendEmail(new Messaging.SingleEmailMessage[] {email});
}
}
}
```

```

    }for(TenantcnewTenant:
newlist) {
        if (newTenant.Propertyc != null &&
existingPropertyIds.contains(newTenant.Propertyc)){newTenant.addError('A
        tenantcanhaveonlyoneproperty');
        }
    }
}
}
}

```

MonthlyEmailScheduler.apxc:

```

globalclassMonthlyEmailSchedulerimplementsSchedulable{ global
    void execute(SchedulableContext sc) { Integer currentDay =
    Date.today().day(); if (currentDay == 1) {
    sendMonthlyEmails();
    }
    }public static void
sendMonthlyEmails(){List<Tenantc>
tenants=[SELECTId,EmailcFROM
Tenantc]; for (Tenantc tenant :
tenants){
        StringrecipientEmail=tenant.Emailc;
        StringemailContent='Itrustthisemailfindsyouwell.Iamwritingtoremindyou that the
monthly rentis dueYour timely payment ensures the smooth functioning of our rental
arrangement and helps maintain a positive living environment for all.';
        StringemailSubject='Reminder:MonthlyRentPaymentDue';

```

Disadvantages of Lease Management

CONCLUSION

The Lease Management System successfully streamlines the operations of leasing through a structured, automated Salesforce application. It improves efficiency, communication, and data accuracy for both admins and tenants.

APPENDIX

● Source Code: Provided in Apex Classes and Triggers

Test.apxt:

```
triggertestonTenantc(beforeinsert){if (trigger.isInsert &&
trigger.isBefore){ testHandler.preventInsert(trigger.new);
    }}
```

testHandler.apxc:

```
public class
testHandler {
    public static void
preventInsert(List<
Tenantc>newlist)
{
    Set<Id>
existingPropertyIds
=newSet<Id>()

    for(TenantcexistingTenant:[SELECTId,PropertycFROMTenantc WHERE
Propertyc != null]) {
        existingPropertyIds.add(existingTenant.Propertyc;
```

- Approval process notifications

The screenshot shows a Salesforce Lightning interface. The main content area displays the details of a tenant named 'pooovarasu'. The details include:

- Tenant Name: pooovarasu
- Owner: pooovarasu p
- Email: pooovarasu614@gmail.com
- Phone: (556) 687-9849
- Status: Stay
- Property: pk
- Check for payment: Paid
- Created By: pooovarasu p, 9/1/2025, 10:28 PM
- Last Modified By: pooovarasu p, 9/8/2025, 5:57 AM

The right sidebar contains a 'Notifications' panel with the following messages:

- pooovarasu p is requesting approval for tenant pooovarasu. Tenant Name: pooovarasu • Owner: pooovarasu p. a few seconds ago.
- Approval request for the tenant is approved pooovarasu. a day ago.
- Approval request for the tenant is approved pooovarasu. Sep 10, 2025, 11:34 AM.
- Approval request for the tenant is approved pooovarasu. Sep 8, 2025, 5:53 PM.
- New Guidance Center learning resource available. Define Your Sales Process. Learn how to guide reps through the sales process. Sep 1, 2025, 11:55 PM.

The bottom of the screen shows a Windows taskbar with the search bar, task icons, and system tray showing the date as 11-09-2025 and time as 13:35.

Process Instance Step Tenant Approval

The screenshot shows a Salesforce Lightning page titled 'Process Instance Step: Tenant Approval'. The page is marked as 'Approved'. The details section shows:

- Submitter: pooovarasu p
- Date Submitted: Sep 11, 2025
- Actual Approver: pooovarasu p
- Assigned To: pooovarasu p

The 'Approval Details' section shows:

- Tenant Name: pooovarasu
- Owner: pooovarasu p

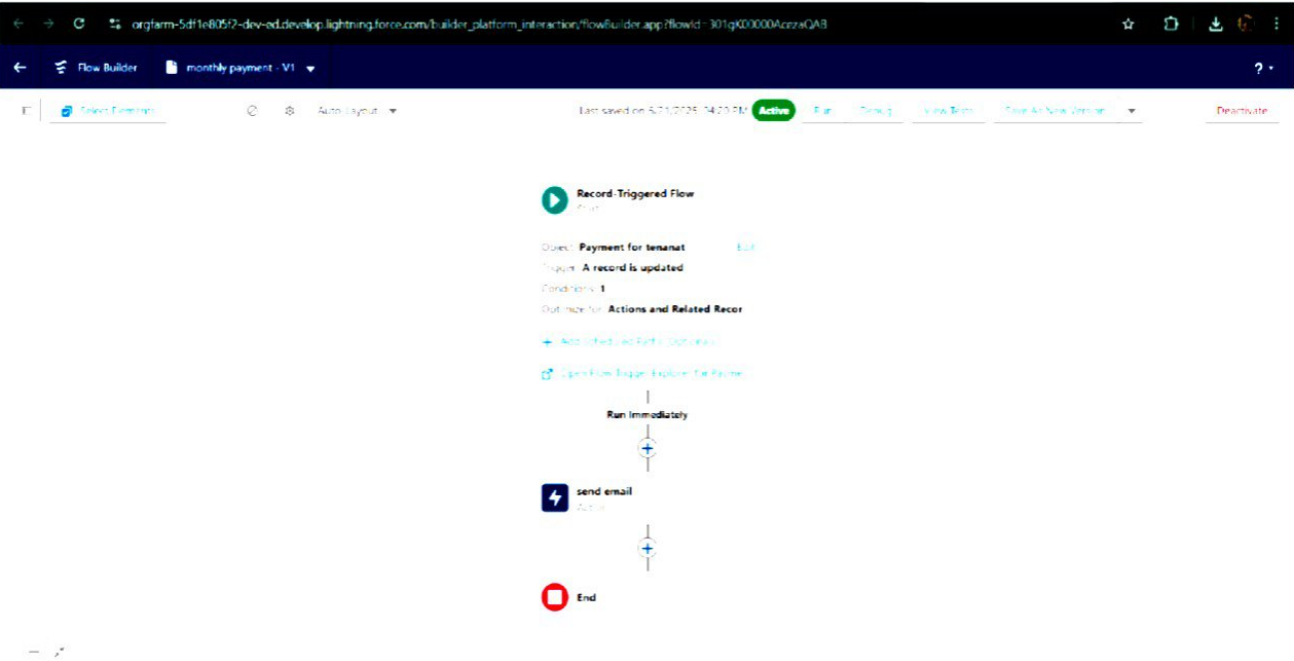
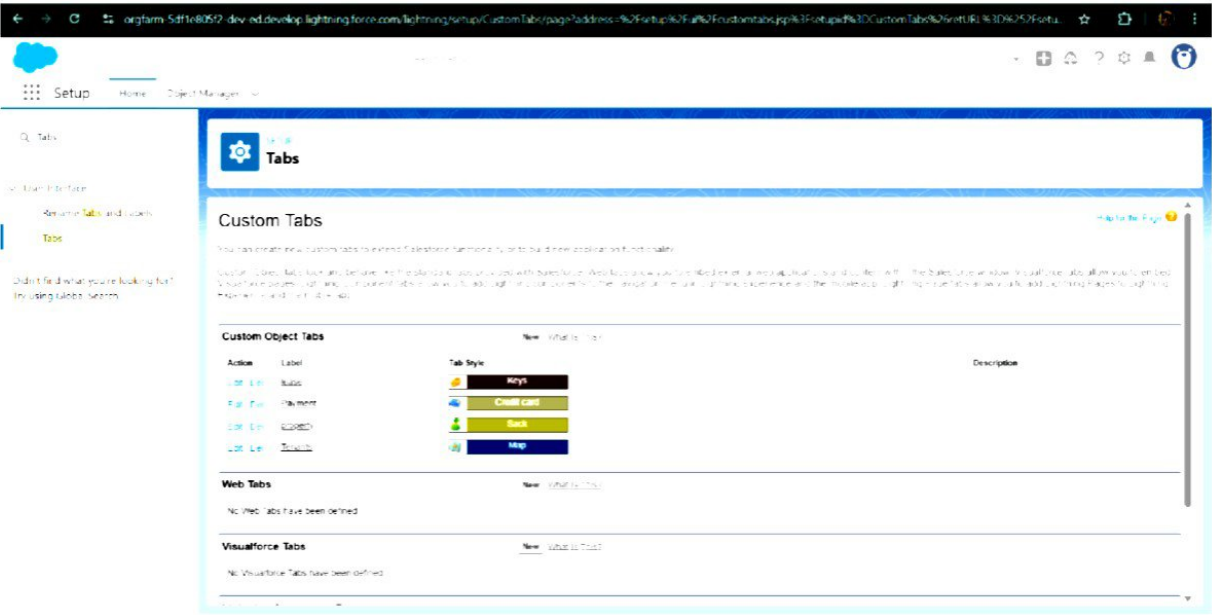
The 'Approver Comments' section shows a comment from pooovarasu p: 'hv', dated Sep 11, 2025, 11:36:59 AM.

The bottom of the screen shows a Windows taskbar with the search bar, task icons, and system tray showing the date as 11-09-2025 and time as 13:37.

RESULTS

Output Screenshots

- TabsforProperty,Tenant,Lease,Payment



ADVANTAGES&DISADVANTAGES

Advantages of Lease Management

Advantage	Description
1. Cost Predictability	Regular lease payments make budgeting easier and reduce unexpected costs.
2. Reduced Upfront Costs	Leasing avoids the need for large capital outlay, especially useful for startups or growing businesses.
3. Flexibility	Short- or long-term lease options allow businesses to scale up or down based on need.
4. Risk Mitigation	Maintenance and repair responsibilities may fall to the lessor, reducing liability for the lessee.
5. Asset Upgrades	Leases can be structured to allow regular upgrades to newer equipment or properties.

Disadvantages of Lease Management

Disadvantage	Description
1. Long-Term Cost	Over time, leasing can be more expensive than owning the asset outright.
2. Limited Control	The lessee may have restrictions on property use or modification.
3. Renewal Risks	Lease terms may change unfavorably at renewal or the lease may not be renewed at all.
4. Contract Complexity	Lease agreements can be legally complex, requiring specialized management and review.
5. Liability for Breach	Failure to comply with lease terms can result in penalties or legal consequences.

Tenant approval access

The screenshot shows a Salesforce interface with a browser window displaying a tenant record for 'poovarasu'. The browser's address bar shows the URL: `https://orgfarm-c6f0c5878a-dev-ed.develop.lightning.force.com/lightning/r/Tenant__c/a01gl00000K7Ud4QAB/view`. The Salesforce navigation bar includes tabs for 'Lease Management...', 'Payment', 'Tenants', 'property', 'lease', and 'Approval Processes'. The tenant details page has a 'Details' tab selected, showing fields for Tenant Name, Email, Phone, Status, Stay, property, check for payment, and Paid. The 'Owner' field is set to 'poovarasu p'. The 'Activity' tab on the right shows 'Upcoming & Overdue' activities, with a message: 'No activities to show. Get started by sending an email, scheduling a task, and more.' The Windows taskbar at the bottom shows the time as 13:31 on 11-09-2025.

Submit for approval

This screenshot shows the same Salesforce tenant details page for 'poovarasu', but with a 'Submit for Approval' modal dialog open in the center. The modal has a title bar and a 'Comments' section with a text area containing the text 'hey hi'. At the bottom of the modal are 'Cancel' and 'Submit' buttons. The background page is dimmed, showing the same tenant details and activity sections as in the previous screenshot. The Windows taskbar at the bottom shows the time as 13:34 on 11-09-2025.

```

globalclass Monthly Email Scheduler implements Schedulable {
    globalvoid execute(Schedulable Context sc) {
        Integer current Day = Date. today().day();
        if(current Day == 1) {
            send Monthly Emails();
        }
    }
    publicstatic void send Monthly Emails() {
        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
youthatthe monthly rent is due Your timely payment ensures the smooth functioning
ofourrental 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});
        }
    }
}

```

Testing the approval process

