

# LEASE MANAGEMENT

**College Name:** NANDHA ARTS AND SCIENCE COLLEGE ERODE-52

**College Code:** bru4j

**TEAM ID:** NM2025TMID22930

**TEAM MEMBERS:** 5

**Team LeaderName:** Varshini v

**Email:** vairavelvarshini@gmail.com

**Team Member:** Karthika E

**Email:** karthikaeswaran2005@gmail.com

**Team Member:** Antony P

**Email:** antony2005anto123@gmail.com

**Team Member:** Arulkumaran T

**Email:** arulkumaran0311@gmail.com

**Team Member:** Madhumitha M

**Email:** madhumitha05236@gmail.com

# 1.INTRODUCTION

## 1.1 Project Overview

The Lease Management System is a Salesforce-based application designed to streamline 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.



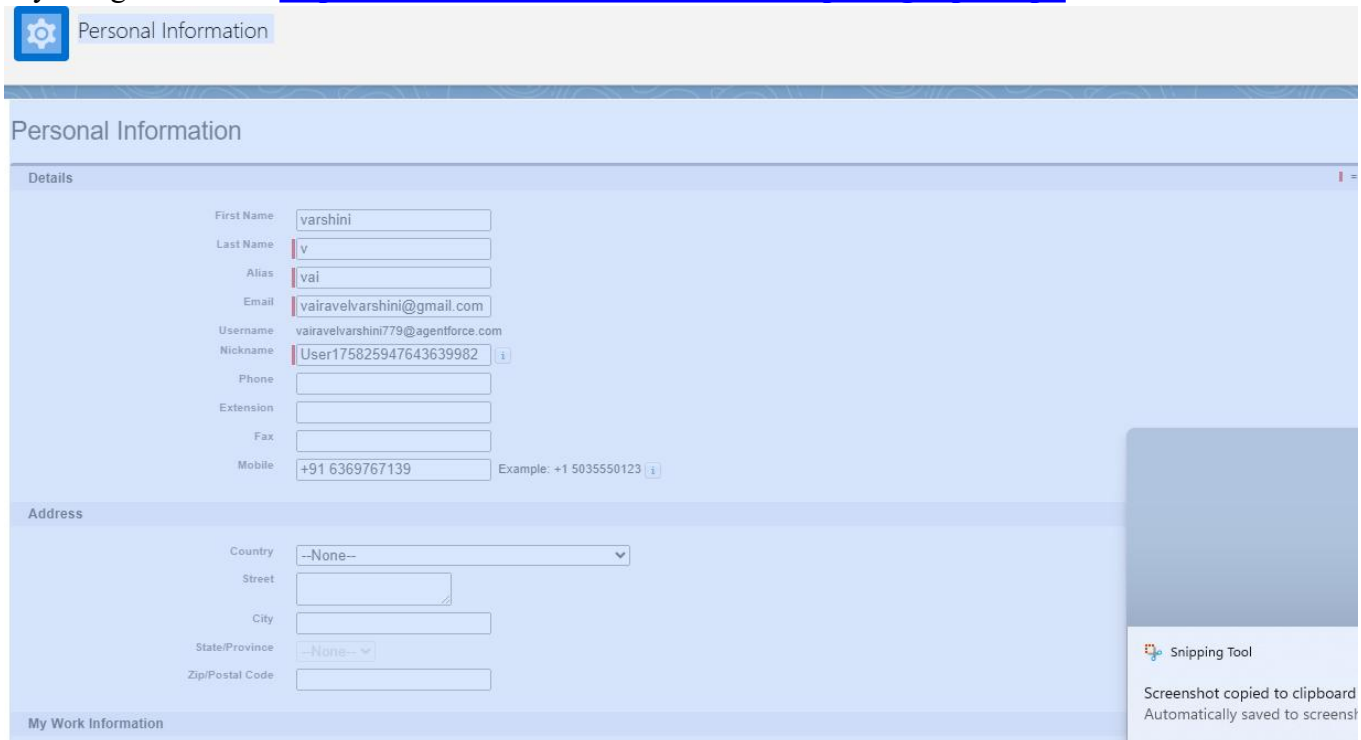
## 1.2 Purpose

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 screenshot shows the 'Personal Information' section of the Salesforce Developer Account creation form. The form is titled 'Personal Information' and has a 'Details' tab selected. The form fields are as follows:

Field	Value
First Name	varshini
Last Name	v
Alias	vai
Email	vairavelvarshini@gmail.com
Username	vairavelvarshini779@agentforce.com
Nickname	User175825947643639982
Phone	
Extension	
Fax	
Mobile	+91 6369767139

Below the 'Details' section is the 'Address' section, which includes fields for Country, Street, City, State/Province, and Zip/Postal Code. The 'Country' field is set to '--None--'. The 'State/Province' field is also set to '--None--'. The 'Zip/Postal Code' field is empty.

At the bottom of the form is the 'My Work Information' section, which is currently empty.

A 'Snipping Tool' watermark is visible in the bottom right corner of the screenshot, indicating that the screenshot was copied to the clipboard and saved to the screen.



Username

Password


Log In

☒ Remember me

Forgot Your Password?

Use Custom Domain

or

 Log In with Email


Not a customer?

Try for Free

## 40% of marketers avoid third-party data.


Discover emerging trends in data, AI, personalisation and more – collected from nearly 5,000 marketers worldwide.


Get the report



naanmudhalvan.smartinternz.com/Student/profile

Set Google Chrome as your default browser and pin it to your taskbar






VARSHINI V

Home

Projects

Support



Last Login :

### PROFILE

Name : VARSHINI V


Email: vairavelvarshini@gmail.com



NM\_ID: 88F0B565A1F4178965549C91F3613914

CATEGORY : Salesforce Developer

PROJECT TITLE : LEASE MANAGEMENT

ROCKET CHAT USERNAME : vairavelvarshini@gmail.com

ROCKET CHAT PASSWORD : 

DOWNLOAD ROCKET CHAT :  

- Created objects: Property, Tenant, Lease, Payment

The screenshot shows the Salesforce Setup interface for the 'property' object. The left sidebar contains a 'Details' menu with options: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, and Scoping Rules. The main content area is titled 'Details' and includes fields for Description, API Name (property\_\_c), Custom (checked), Singular Label (property), and Plural Label (property). On the right, there are checkboxes for Enable Reports (checked), Track Activities (checked), Track Field History (checked), and Deployment Status (Deployed). A 'Help Settings' link and a 'Standard salesforce.com Help Window' link are also present. 'Edit' and 'Delete' buttons are in the top right corner.

The screenshot shows the Salesforce Setup interface for the 'Tenant' object. The left sidebar contains a 'Details' menu with options: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, and Scoping Rules. The main content area is titled 'Details' and includes fields for Description, API Name (Tenant\_\_c), Custom (checked), Singular Label (Tenant), and Plural Label (Tenants). On the right, there are checkboxes for Enable Reports (checked), Track Activities (checked), Track Field History (checked), and Deployment Status (Deployed). A 'Help Settings' link and a 'Standard salesforce.com Help Window' link are also present. 'Edit' and 'Delete' buttons are in the top right corner.

Setup

Home

Object Manager

Q

Search Setup

SETUP > OBJECT MANAGER

lease

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Details

Description

API Name

lease\_\_c

Custom

✓

Singular Label

lease

Plural Label

lease

Enable Reports

✓

Track Activities

✓

Track Field History

✓

Deployment Status

Deployed

Help Settings

Standard salesforce.com Help Window

Edit

Delete

Setup

Home

Object Manager

Q

Search Setup

SETUP > OBJECT MANAGER

Payment for tenanat

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

Details

Description

API Name

Payment\_for\_tenanat\_\_c

Custom

✓

Singular Label

Payment for tenanat

Plural Label

Payment

Enable Reports

✓

Track Activities

✓

Track Field History

✓

Deployment Status

Deployed

Help Settings

Standard salesforce.com Help Window

Edit

Delete

- Configured fields and relationships

Setup Home Object Manager

Search Setup

SETUP > OBJECT MANAGER

**property**

Details

**Fields & Relationships**

9 Items, Sorted by Field Label

Quick Find New Deleted Fields Field Dependencies Set History Tracking

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(25)		
Owner	OwnerId	Lookup(User,Group)		✓
property	property__c	Lookup(property)		✓
property Name	Name	Text(80)		✓
sft	sft__c	Text(18)		
Type	Type__c	Picklist		

Setup Home Object Manager

Search Setup

SETUP > 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	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Payment date	Payment_date__c	Date		
Payment Name	Name	Text(80)		✓



Search Setup



Setup

Home

Object Manager

SETUP > OBJECT MANAGER

lease

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

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	CreatedById	Lookup(User)		
End date	End_date__c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
lease Name	Name	Text(80)		✓
Owner	OwnerId	Lookup(User,Group)		✓
property	property__c	Lookup(property)		✓
start date	start_date__c	Date		



Search Setup



Setup

Home

Object Manager

SETUP > OBJECT MANAGER

Tenant

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

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	CreatedById	Lookup(User)		
Email	Email__c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone	Phone__c	Phone		
status	status__c	Picklist		
Tenant Name	Name	Text(80)		✓



- Developed Lightning App with relevant tabs

← Lightning App Builder

⚙ App Settings

📄 Pages ▾

Lease Management

ⓘ Help

App Settings

App Details & Branding

App Options

Utility Items (Desktop Only)

Navigation Items

User Profiles

App Details & Branding

Give your Lightning app a name and description. Upload an image and choose the highlight color for its navigation bar.


App Details

\* App Name ⓘ  
Lease Management

\* Developer Name ⓘ  
Lease\_Management


Description ⓘ  
Application to efficiently handle the processes related to leasing real estate properties.

App Branding

Image ⓘ  
  
Clear

Primary Color Hex Value ⓘ  
#0070D2

Org Theme Options  
☐ Use the app's image and color instead of the org's custom theme

App Launcher Preview  


← Lightning App Builder

⚙ App Settings

📄 Pages ▾

Lease Management

ⓘ Help

App Settings

App Details & Branding

App Options

Utility Items (Desktop Only)

Navigation Items

User Profiles

Navigation Items

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

Available Items

🔍 Type to filter list...

Accounts

Activation Targets

Activations

All Sites

Alternative Payment Methods

Analytics

App Launcher

Appointment Categories

Appointment Invitations

Approval Requests

Selected Items

Payment

Tenants

property

lease

← Lightning App Builder

⚙ App Settings

📄 Pages

Lease Management

🔍 ? Help

App Settings

App Details & Branding

App Options

Utility Items (Desktop Only)

Navigation Items

User Profiles

User Profiles

Choose the user profiles that can access this app.

Available Profiles

Type to filter list...

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

Custom: Support Profile

Customer Community Login User

Selected Profiles

System Administrator

👤

Lease Management

Payment

Tenants

property

lease

🔍 Search...

★ + 🔔 ? ⚙ 🔴 👤

Payment

Recently Viewed

5 items • Updated a few seconds ago

New

Import

Change Owner

Assign Label

🔍 Search this list...

⚙ 📄 🔄 ✎ ⚙ ⌵

Payment Name

1 ☐ Rahul

2 ☐ Jack

3 ☐ Raj

4 ☐ Sam

5 ☐ Lahari

⌵

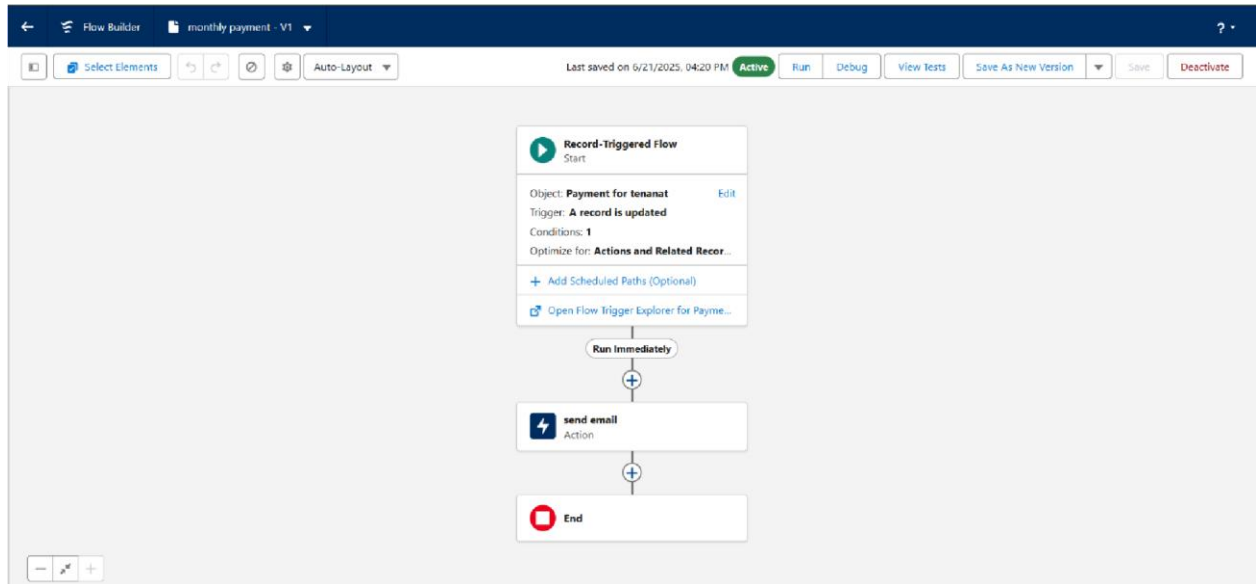
⌵

⌵

⌵

⌵

- Implemented Flows for monthly rent and payment success



- To create a validation rule to a Lease Object

The screenshot shows the Salesforce Setup interface, specifically the 'Validation Rule Edit' screen for the 'lease' object. The rule is named 'lease\_end\_date' and is currently active. The 'Error Condition Formula' section shows the formula: `ISBLANK(lease__end_date) <= ABS(lease__end_date)`. The 'Error Message' section is empty. The 'Functions' list on the right includes functions like ABS, ACOS, ADDMONTHS, AND, ASCII, and ASIN.

**Validation Rule Edit**

Rule Name: **lease\_end\_date**

Active: ☒

Description:

**Error Condition Formula**

Example: `Discount_Percent__c > 0.30` More Examples...

Display an error if Discount is more than 30%

If this formula expression is true, display the text defined in the Error Message area

Insert Field: `ISBLANK(lease__end_date) <= ABS(lease__end_date)`

Insert Operator:

Check Syntax

**Functions**

All Function Categories

ABS  
ACOS  
ADDMONTHS  
AND  
ASCII  
ASIN

Insert Selected Function

ABS(number)  
Returns the absolute value of a number, a number without its sign

Help on this function

- Added Apex trigger to restrict multiple tenants per property

[Demo Link](#)
0
[View Mentor Comments](#)
0
[View Industry Mentor Comments](#)

PROJECT DETAILS

LEASE MANAGEMENT

Salesforce

Object

Tabs

The Lightning App

Fields

Validation Rule

Email Templates

Approval Process

Apex Trigger

Create An Apex Trigger

Create An Apex Handler Class

Testing The Trigger

Flows

Schedule Class :

TASK & PROGRESS

Create An Apex Trigger

Duration: 1 Hrs

Skill Tags:

Open Resource	CTRL+SHIFT+O	Visualforce Page
Open Lightning Resources	CTRL+SHIFT+A	Visualforce Component
Open Log	CTRL+G	Static Resource
Open Raw Log	CTRL+SHIFT+G	Lightning Application
Download Log	CTRL+ALT+G	Lightning Component
Save	CTRL+S	Lightning Interface
Save All	CTRL+SHIFT+S	Lightning Event

MENTOR REVIEW

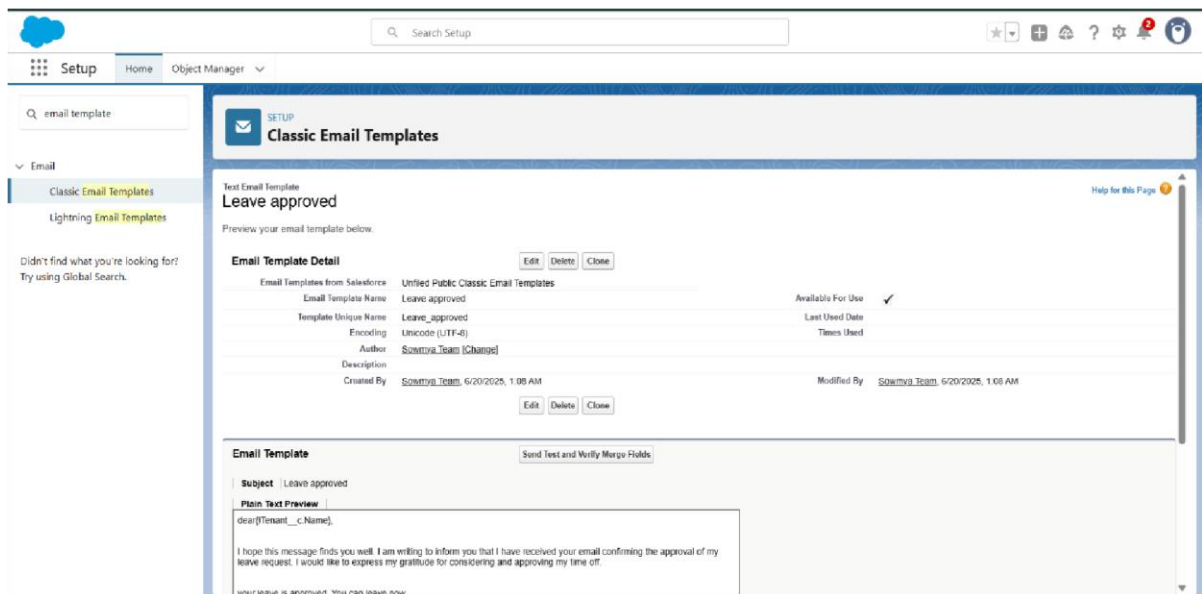
- Scheduled monthly reminder emails using Apex class

```

1 • global class MonthlyEmailScheduler implements Schedulable {
2
3 •   global void execute(SchedulableContext sc) {
4
5       Integer currentDay = Date.today().day();
6
7       if (currentDay == 1) {
8
9           sendMonthlyEmails();
10
11       }
12
13   }
14
15   public static void sendMonthlyEmails() {
16
17       List<Tenant_c> tenants = [SELECT Id, Email__c FROM Tenant__c];
18
19       for (Tenant__c tenant : tenants) {
20
21           String recipientEmail = tenant.Email__c;
22
23           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';
24
25           String emailSubject = 'Reminder: Monthly Rent Payment Due';
26
27           Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
28
29           email.setToAddresses(new String[]{recipientEmail});
30
31           email.setSubject(emailSubject);
32
33           email.setPlainTextBody(emailContent);
34
35       }
36   }
37 }

```

- Built and tested email templates for leave request, approval, rejection, payment, and reminders



SetupHomeObject Manager

Search Setup

StarGridHomeHelpUser

email template

Email

Classic Email TemplatesLightning 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

EditDeleteClone

Email Templates from Salesforce

Unified Public Classic Email Templates

Email Template Name

tenant\_leaving

Template Unique Name

tenant\_leaving

Encoding

Unicode (UTF-8)

Author

Scamya Team (Change)

Description

Created By

Scamya Team, 6/29/2025, 1:06 AM

Available For Use

✓

Last Used Date

Times Used

Modified By

Scamya Team, 6/29/2025, 1:06 AM

EditDeleteClone

Email Template

Send Text and Verify Merge Fields

Subject

request for approve the leave

Plain Text Preview

Dear {Tenant\_\_c.CreatedBy},  
  
Please approve my leave

SetupHomeObject Manager

Search Setup

StarGridHomeHelpUser

email template

Email

Classic Email TemplatesLightning Email Templates

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

Classic Email Templates

Text Email Template  
Leave rejected

Preview your email template below.

Email Template Detail

EditDeleteClone

Email Templates from Salesforce

Unified Public Classic Email Templates

Email Template Name

Leave\_rejected

Template Unique Name

Leave\_rejected

Encoding

Unicode (UTF-8)

Author

Scamya Team (Change)

Description

Created By

Scamya Team, 6/29/2025, 1:11 AM

Available For Use

✓

Last Used Date

Times Used

Modified By

Scamya Team, 6/29/2025, 1:11 AM

EditDeleteClone

Email Template

Send Text 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

Cloud Setup Home Object Manager

Search Setup

Q email template

▼ 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

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	Scoutmy Team (Change)		
Description			
Created By	Scoutmy Team, 6/20/2025, 1:13 AM	Modified By	Scoutmy Team, 6/20/2025, 1:13 AM

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

**Email Template** [Send Text and Merge Fields](#)

**Subject** Confirmation of Successful Monthly Payment

**Plain Text Preview**

Dear {{Tenant\_\_c.Email\_\_c}},

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

- Approval Process creation

For Tenant Leaving:

Cloud Setup Home Object Manager

Search Setup

Q approval

▼ 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

Approval Processes

Tenant: TenantApproval

[Back to Approval Process List](#)

**Process Definition Detail** [Edit](#) [Clone](#) [Deactivate](#)

Process Name	TenantApproval	Active	✓
Unique Name	TenantApproval	Next Automated Approver Determined By	
Description			
Entry Criteria	TENANT__STATUS__EQUALS Stay	Allow Submitters to Recall Approval Requests	<input type="checkbox"/>
Record Editability	Administrator ONLY		
Approval Assignment Email Template			
Initial Submitter	Tenant Owner		
Created By	Scoutmy Team, 6/23/2025, 3:41 AM	Modified By	Scoutmy Team, 6/26/2025, 11:57 PM

**Initial Submission Actions** [Add Existing](#) [Add New](#)

Action	Type	Description
Record Lock		Lock the record from being edited

**Approval Steps** [Show Actions](#) [Edit](#)

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
	1	Step 1			User: Scoutmy Team	Final Rejection

For Check for Vacant:

The screenshot shows the Salesforce Setup interface 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 shows the configuration for a process named 'check for vacant'. The process is active and has a unique name of 'check\_for\_vacant'. The entry criteria is 'Tenants status EQUALS LEAVING'. The approval assignment email template is 'Leave Requested'. The initial submission actions include 'Record Lock' and 'Email Alert'. The approval steps table shows a single step named 'step1' with a description 'Lock the record from being edited' and a reject behavior of 'Final Rejection'.

Process Name	check for vacant	Active	✓
Unique Name	check_for_vacant	Next Automated Approver Determined By	
Description			
Entry Criteria	Tenants status EQUALS LEAVING		
Approval Assignment Email Template	Leave Requested	Allow Submitters to Recall Approval Requests	<input type="checkbox"/>
Initial Submitters	Tenant Owner		
Created By	Scoutiya Team	Modified By	Scoutiya Team

Action	Type	Description
Record Lock		Lock the record from being edited
Email Alert		PLEASE APPROVE MY LEAVE

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
1	1	step1			User:Scoutiya Team	Final Rejection

- Apex Trigger

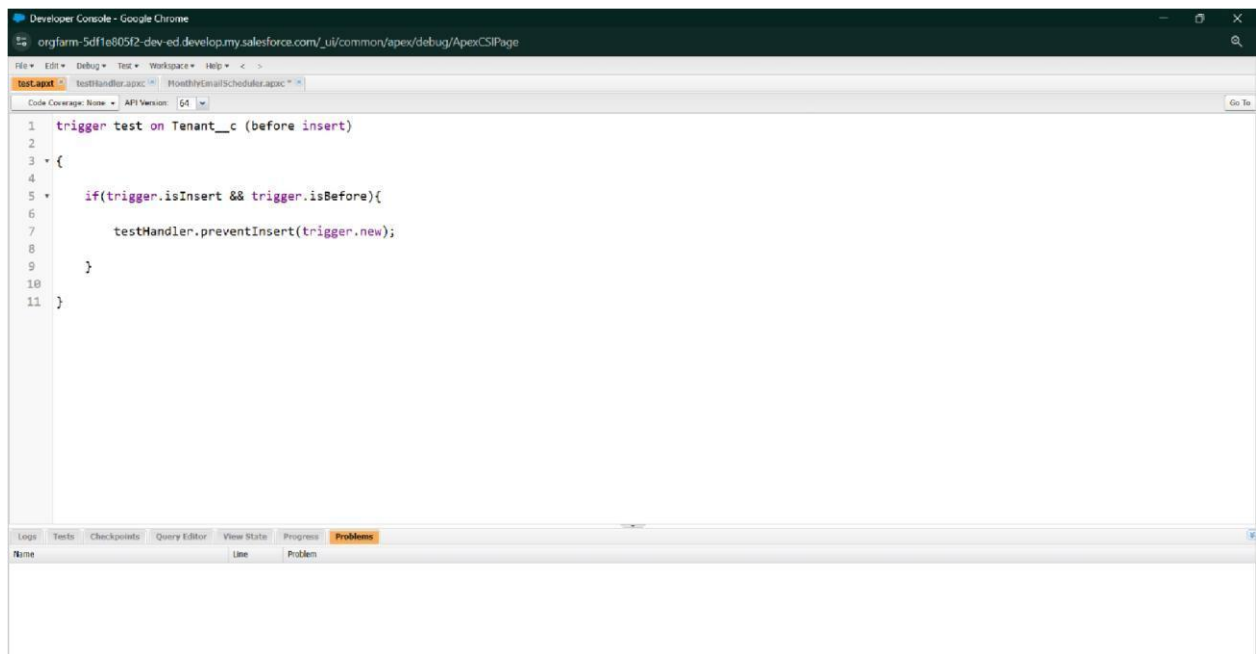
Create an Apex Trigger

The screenshot shows the Salesforce IDE with an Apex trigger named 'testHandler.apex'. The trigger is a 'before insert' trigger on the 'Tenant\_\_c' object. The code is as follows:

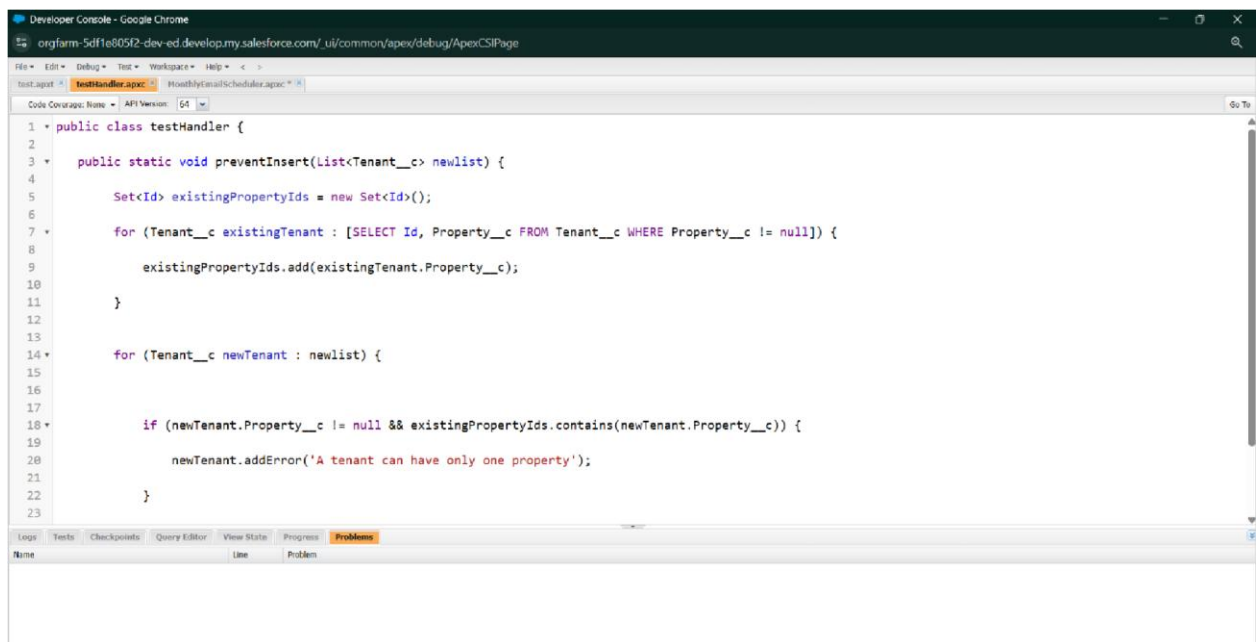
```
1 trigger test on Tenant__c (before insert)
2
3 {
4
5     if(trigger.isInsert && trigger.isBefore){
6
7         testHandler.preventInsert(trigger.new);
8
9     }
10
11 }
```

An 'Open' window is displayed, showing a search for 'test' in the 'testHandler.apex' file. The search results show a single entry for 'test' in the 'testHandler.apex' file.





Create an Apex Handler class



- FLOWS

The image displays two screenshots of the Salesforce Flow Builder interface, showing the configuration of a Record-Triggered Flow.

**Top Screenshot:**

- Flow Name:** monthly payment - V1
- Object:** Payment for tenant
- Trigger:** A record is updated
- Conditions:** 1
- Optimize for:** Actions and Related Records
- Start:** Run Immediately
- Action:** send email
- End:** End

**Configure Start Panel:**

- Field:** check for payment
- Operator:** Equals
- Value:** Ag Paid
- When to Run the Flow for Updated Records:** ☒ Every time a record is updated and meets the condition requirements
- Optimize Flow:** ☒ Actions and Related Records
- Is this flow making an external callout or connecting to an external system?** ☐ Add Asynchronous Path

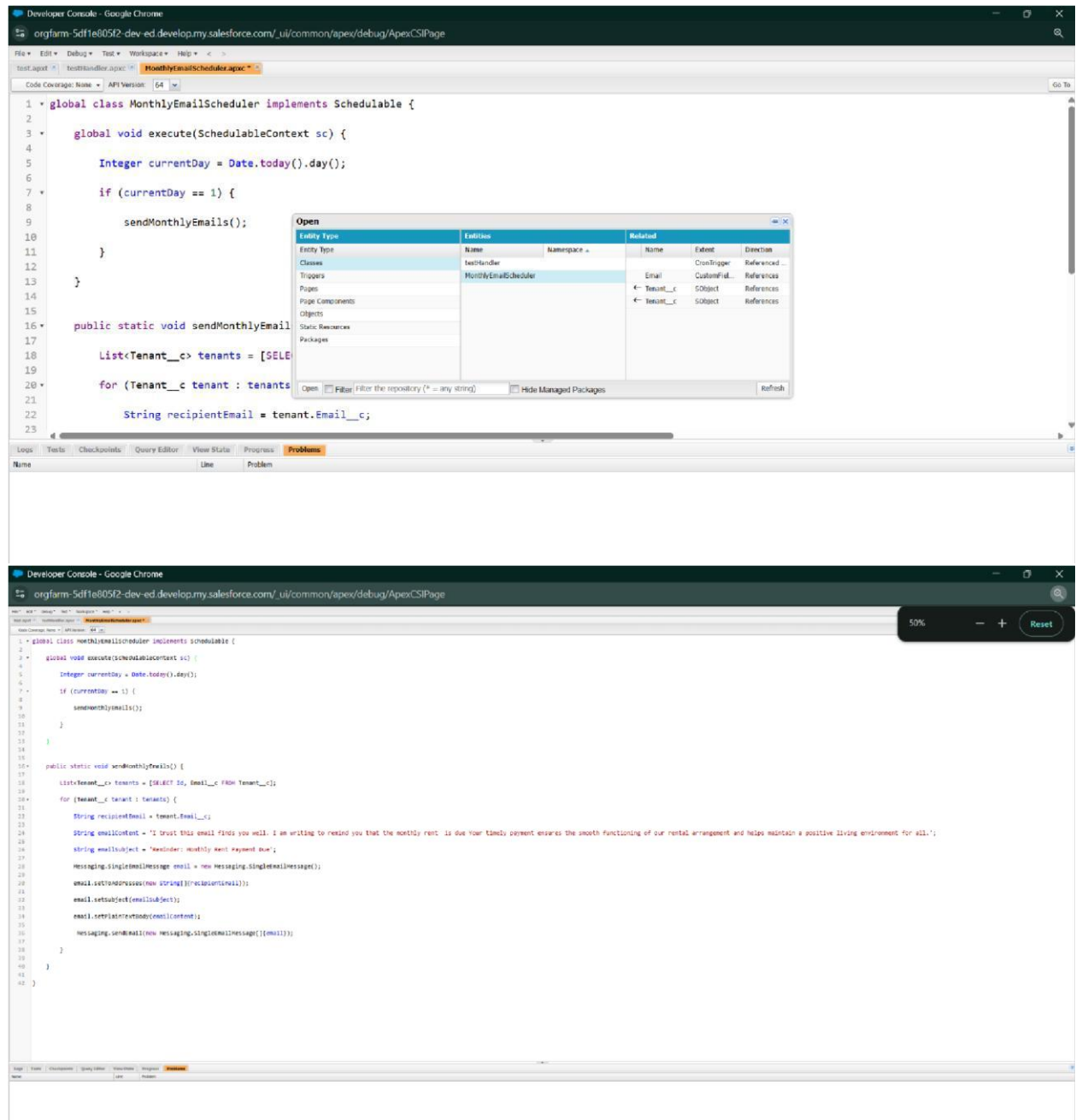
**Bottom Screenshot:**

- Flow Name:** monthly payment - V1
- Object:** Payment for tenant
- Trigger:** A record is updated
- Conditions:** 1
- Optimize for:** Actions and Related Records
- Start:** Run Immediately
- Action:** send email
- End:** End

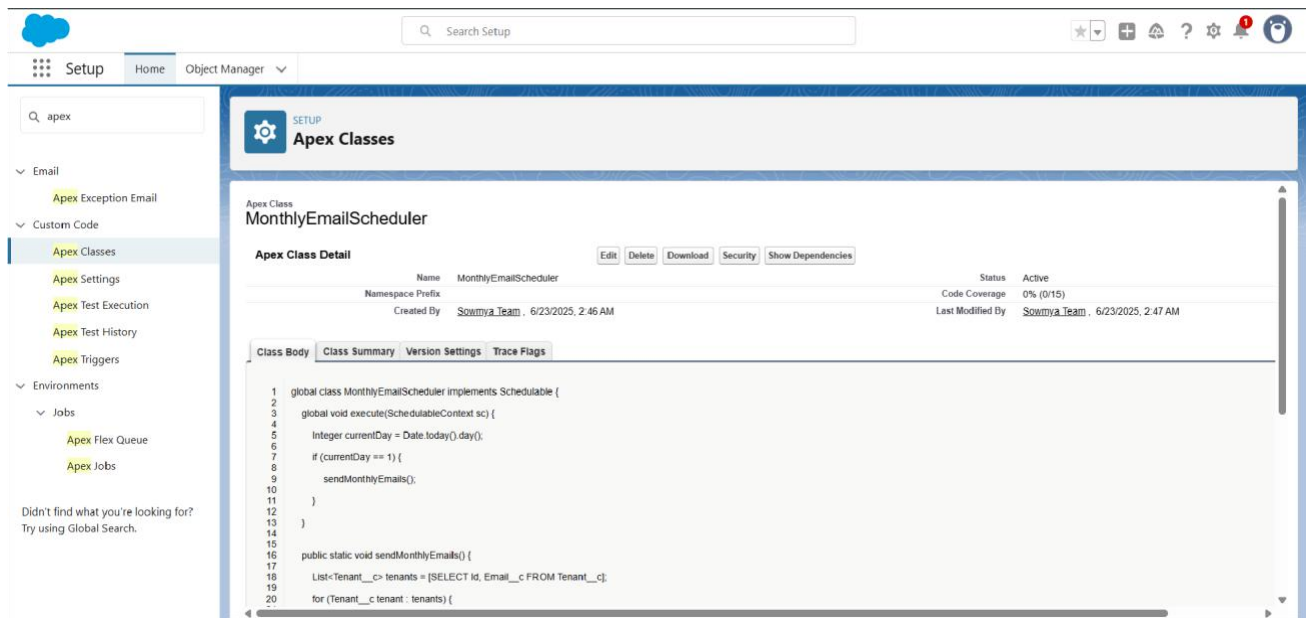
**Configure Start Panel:**

- Select Object:** Payment for tenant
- Configure Trigger:** ☒ A record is updated
- Set Entry Conditions:** All Conditions Are Met (AND)

- Schedule class:  
Create an Apex Class



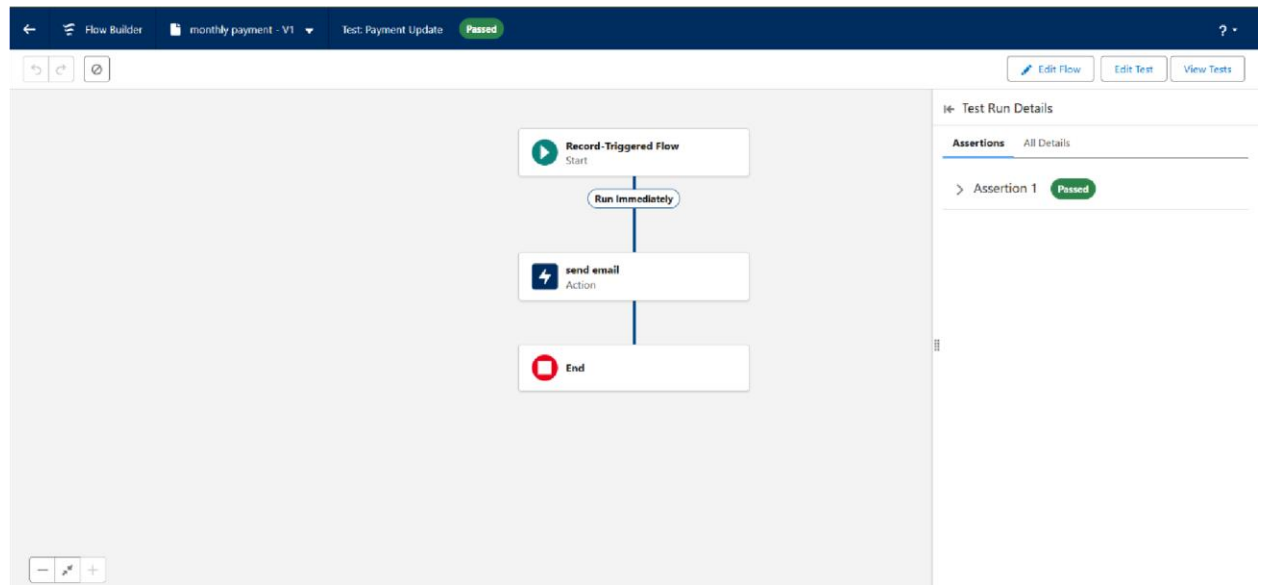
Schedule Apex class



# FUNCTIONAL AND PERFORMANCE TESTING

## Performance Testing

- Trigger validation by entering duplicate tenant-property records



- Approval process validated through email alerts and status updates

---

## RESULTS

### Output Screenshots

- Tabs for Property, Tenant, Lease, Payment

Setup Home Object Manager

Search Setup

tabs

User Interface

Rename Tabs and Labels

Tabs

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

## Custom Tabs

You can create new custom tabs to extend Salesforce functionality or to build new application functionality.

Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external web applications and content within the Salesforce window. Visualforce tabs allow you to embed Visualforce pages. Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app. Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.

### Custom Object Tabs

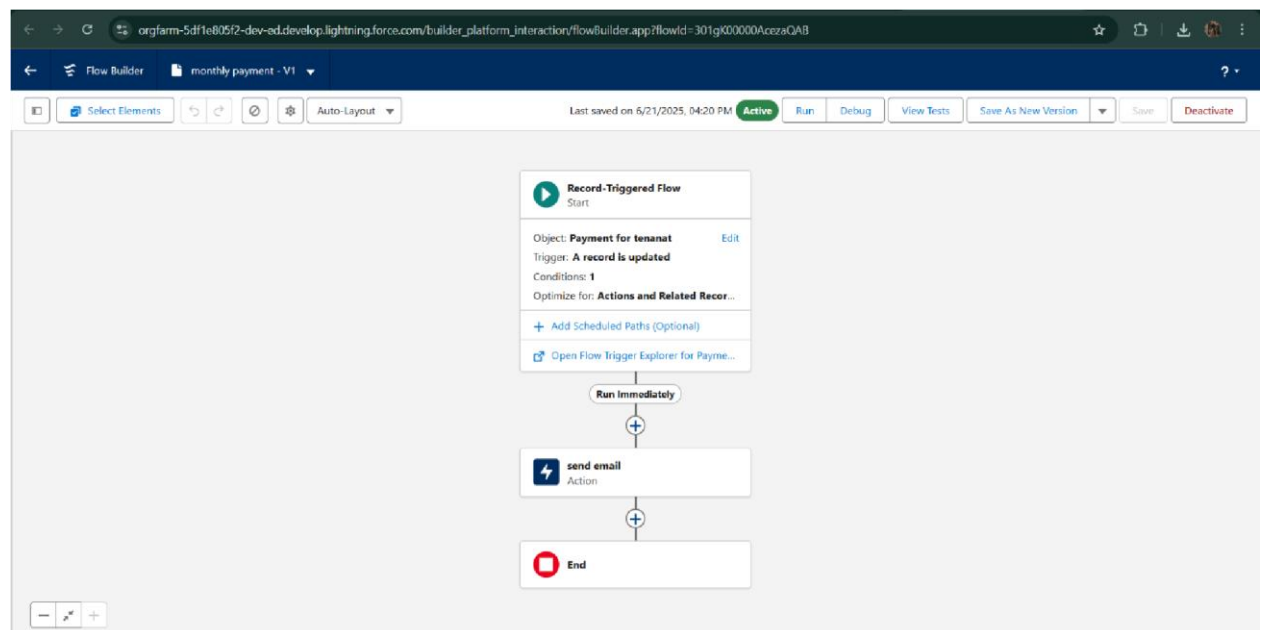
New What Is This?

Action	Label	Tab Style	Description
Edit   Del	Lease	Credit card	
Edit   Del	Payment	Box	
Edit   Del	property	Cell phone	
Edit   Del	Tenants	Globe	

### Web Tabs

New What Is This?

No Web Tabs have been defined



- Flow runs
- Trigger error messages

---

# 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:

```
trigger test on Tenant__c (before insert)

{ if (trigger.isInsert && trigger.isBefore)

{ testHandler.preventInsert(trigger.new);

    }

}
```

### testHandler.apxc:

```
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)) { newTenant.addError('A
tenant can have only one property');

            }

        }

    }

}
```

### **MonthlyEmailScheduler.apxc:**

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});
    }
}
}
```