

# Project Files

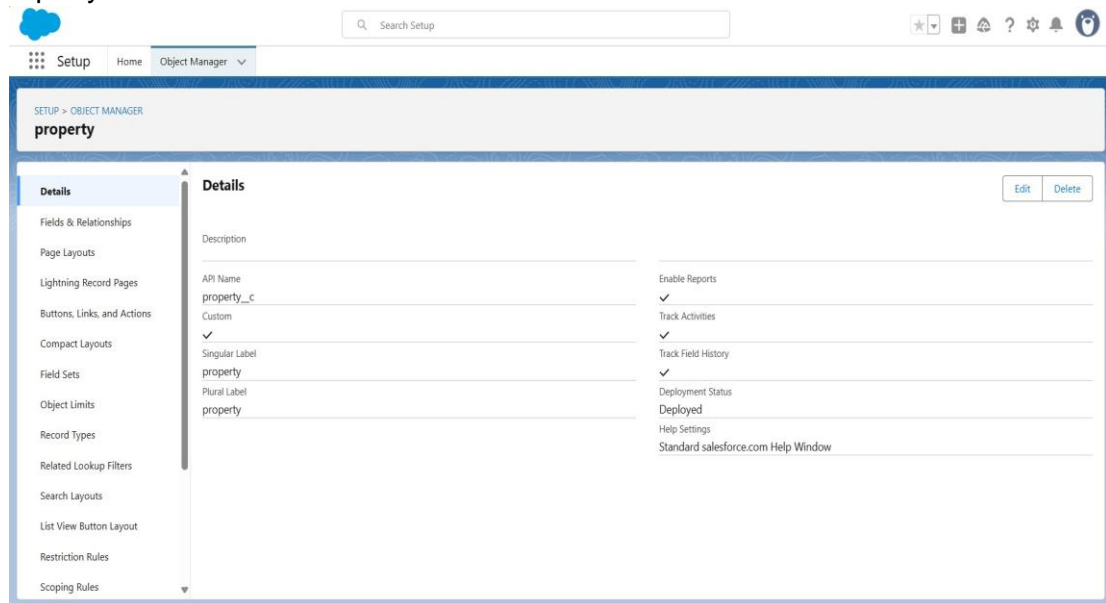
Date	25 JUNE 2025
Team ID	LTVIP2025TMID30578
Project Name	Lease Management
Maximum Marks	

This is a **Salesforce-based implementation**, so the "executables" are not traditional .exe files but configurations and custom code within the Salesforce ecosystem. The project includes:

## A. Custom Salesforce Objects

Created via Salesforce Setup:

- Property



- Tenant

Setup Home Object Manager

SEARCH SETUP

SETUP > OBJECT MANAGER

### Tenant

Details Edit Delete

Description

API Name: Tenant\_\_c

Custom

Singular Label: Tenant

Plural Label: Tenants

Enable Reports: ☒

Track Activities: ☒

Track Field History: ☒

Deployment Status: Deployed

Help Settings: Standard salesforce.com Help Window

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

- Lease

Setup Home Object Manager

SEARCH SETUP

SETUP > OBJECT MANAGER

### lease

Details Edit Delete

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

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

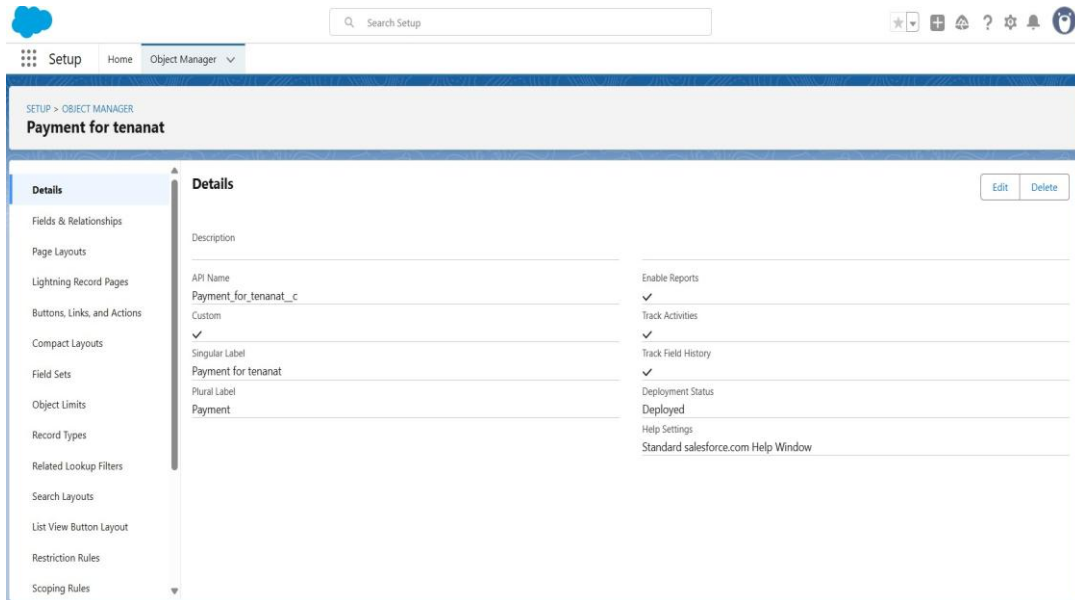
Search Layouts

List View Button Layout

Restriction Rules

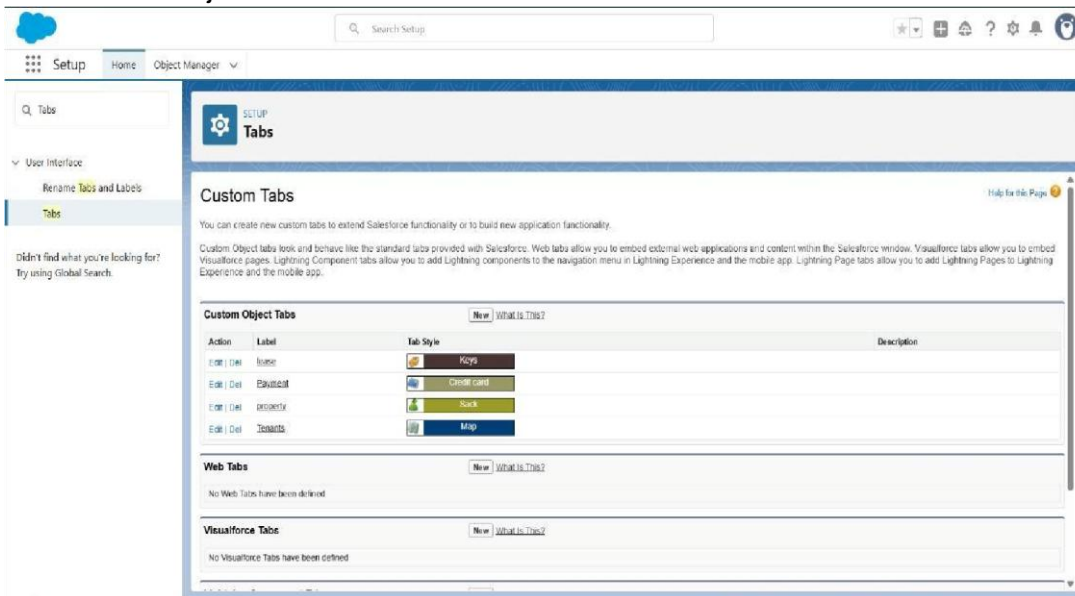
Scoping Rules

- Payment for Tenant

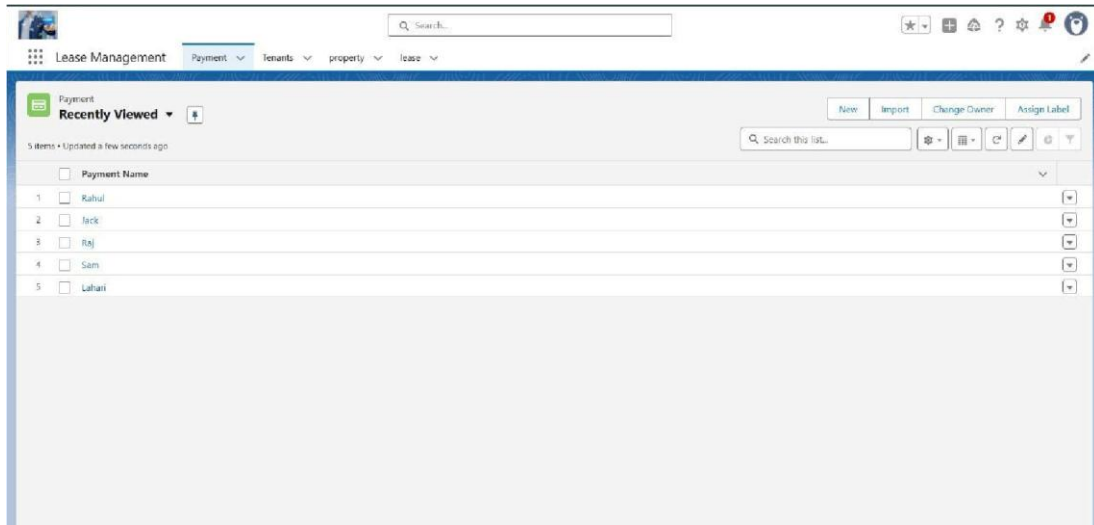


## B. Custom Tabs and App

- Tabs for each object



- A Lightning App called **Lease Management** with navigation setup



## C. Apex Code (Custom Backend Logic)

- **Apex Trigger:** test on Tenant\_\_c

```

Developer Console - Personal - Microsoft Edge
https://orgfarm-5df1e805f2-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File Edit Debug Test Workspace Help < >
test.apxt testHandler.apxc MonthlyEmailScheduler.apxc
Code Coverage: None API Version: 64

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 }

Logs Tests Checkpoints Query Editor View State Progress Problems
Name Line Problem

```

- **Apex Handler Class:** testHandler

Developer Console - Personal - Microsoft Edge  
https://orgfarm-5df1e805f2-dev-ed.develop.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage

File • Edit • Debug • Test • Workspace • Help • < • > •

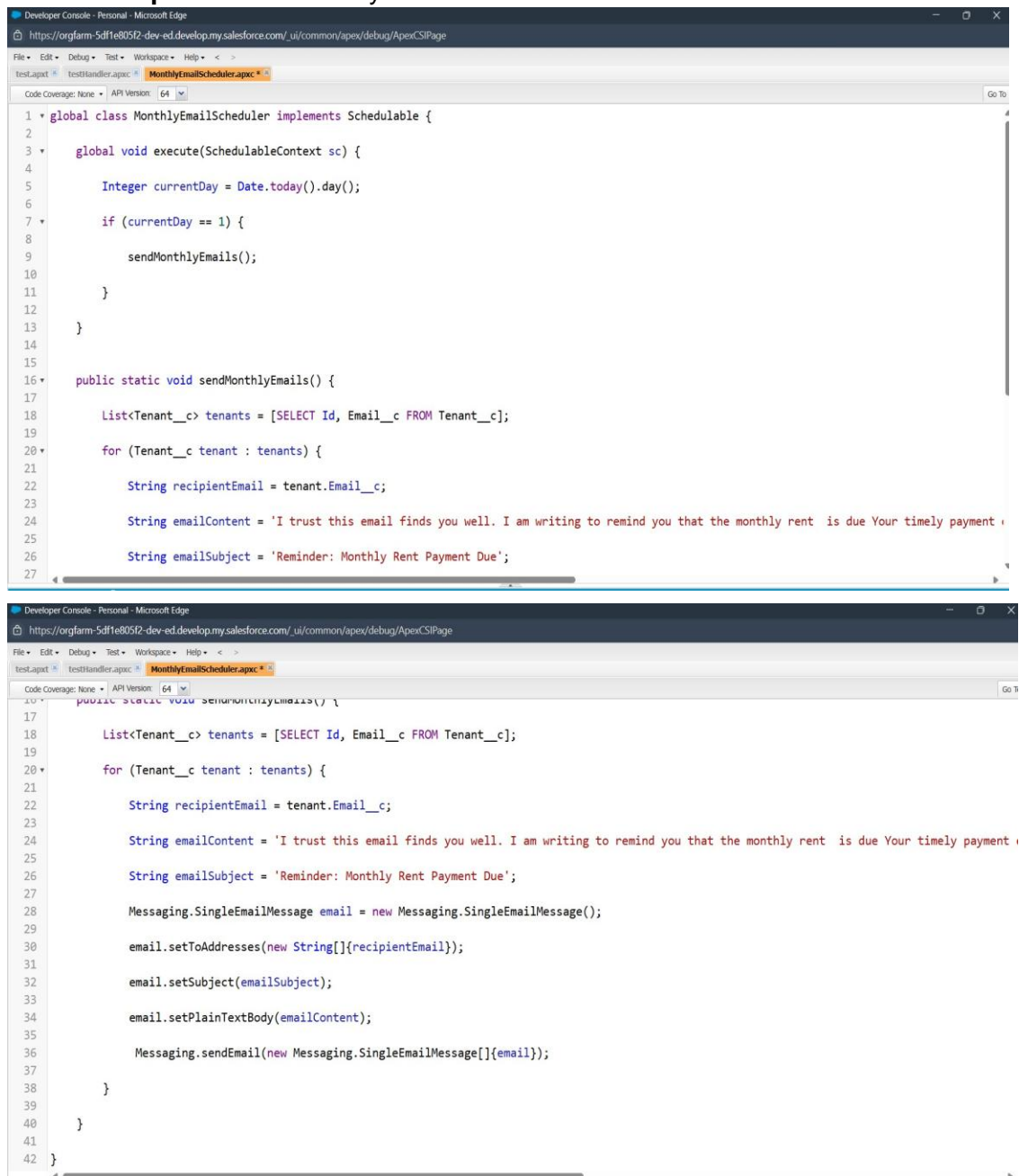
test Лагаст • testHandler.apex • MonthlyEmailScheduler.apex •

Code Coverage: None • API Version: 64 • Go To

```
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
16
17
18             if (newTenant.Property__c != null && existingPropertyIds.contains(newTenant.Property__c)) {
19
20                 newTenant.addError('A tenant can have only one property');
21
22             }
23
24         }
25
26     }
27 }
```

Logs, Tests, and Problems

- **Scheduled Apex Class: MonthlyEmailScheduler**



The image displays two screenshots of the Salesforce Developer Console, showing the code for the `MonthlyEmailScheduler` Apex class. The first screenshot shows the class definition and the `execute` method, while the second screenshot shows the `sendMonthlyEmails` method.

```
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
16 public static void sendMonthlyEmails() {
17
18     List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
19
20     for (Tenant__c tenant : tenants) {
21
22         String recipientEmail = tenant.Email__c;
23
24         String emailContent = 'I trust this email finds you well. I am writing to remind you that the monthly rent is due Your timely payment';
25
26         String emailSubject = 'Reminder: Monthly Rent Payment Due';
27
28     }
29 }
```

```
17
18     List<Tenant__c> tenants = [SELECT Id, Email__c FROM Tenant__c];
19
20     for (Tenant__c tenant : tenants) {
21
22         String recipientEmail = tenant.Email__c;
23
24         String emailContent = 'I trust this email finds you well. I am writing to remind you that the monthly rent is due Your timely payment';
25
26         String emailSubject = 'Reminder: Monthly Rent Payment Due';
27
28         Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
29
30         email.setToAddresses(new String[]{recipientEmail});
31
32         email.setSubject(emailSubject);
33
34         email.setPlainTextBody(emailContent);
35
36         Messaging.sendEmail(new Messaging.SingleEmailMessage[]{email});
37
38     }
39
40 }
41
42 }
```

## D. Validation Rule

- Lease object has a rule: End\_date\_\_c > Start\_date\_\_c

The screenshot shows the Salesforce Setup interface. The left sidebar contains a navigation menu with options like 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, and Scoping Rules. The main content area is titled 'Validation Rules' for the 'lease' object. It displays a table with one rule:

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
lease_end_date	start date	Your End date must be greater than start date	✓	Sowmya Team, 6/19/2025, 5:37 AM

## E. Approval Process

- For Tenant object: “Check for vacant”

The screenshot shows the Salesforce Setup interface for an Approval Process. The left sidebar contains a navigation menu with options like Approval Processes, Process Automation, and a search bar. The main content area is titled 'Approval Processes' and shows the details for the 'Tenant: check for vacant' process. The process definition details include:

- Process Name: check for vacant
- Unique Name: check\_for\_vacant
- Description: Next Automated Approval Determined By
- Entry Criteria: Tenants status NOT EQUAL TO LEAVING
- Record Editability: Administrator ONLY
- Approval Assignment Email Template: Leave approved
- Initial Submission Actions: Record Lock, Email Alert
- Approval Steps: 1. Record Lock, 2. Email Alert

## F. Flows

- **Record-Triggered Flow** for monthly payment email when payment is marked as “Paid”

The screenshot shows the Salesforce interface for a Flow named "monthly payment". The top navigation bar includes a search bar and various tabs like Sales, Home, Opportunities, Leads, Tasks, Files, Accounts, Contacts, Campaigns, Dashboards, Reports, and Chatter. The Flow details page shows the following information:

- Type:** Record—Run After Save
- Associated Record:** (empty)
- Progress Status:** Activated
- Last Modified Date:** 6/21/2025, 3:50 AM
- Flow Owner:** Sowmya Team

The "Details" tab is selected, showing the following information:

Information	
Flow Label	monthly payment
Description	
Associated Record	
Created By	Sowmya Team, 6/21/2025, 3:50 AM
Last Modified	Sowmya Team, 6/21/2025, 3:50 AM
Category	
API Name	monthly_payment
Flow Type	Record-Triggered After Save Flow
Segment	
Created Date	6/21/2025, 3:50 AM
Last Modified Date	6/21/2025, 3:50 AM
Subcategory	

At the bottom left, there is a "To Do List" icon.