LEASE MANAGEMENT

College Name: KPR College Of Arts Science And Research

College Code: bruAZ

TEAM ID: NM2025TMID21362

TEAM MEMBERS: INDHRAJETH ARUNACHALAM , SRISIVALAKSHMI P, SARAN S M, NITHYASREE N

Team LeaderName: INDHRAJETH ARUNACHALAM

Email: 23bcs027@kprcas.ac.in

Team Member1: SARAN S M

Email: 23bcs053@kprcas.ac.in

Team Member2: SRISIVALAKSHMI P

Email: 23bcs060@kprcas.ac.in

Team Member3: NITHYASREE N

Email: 23bcs042@kprcas.ac.in

Creating Developer Account:

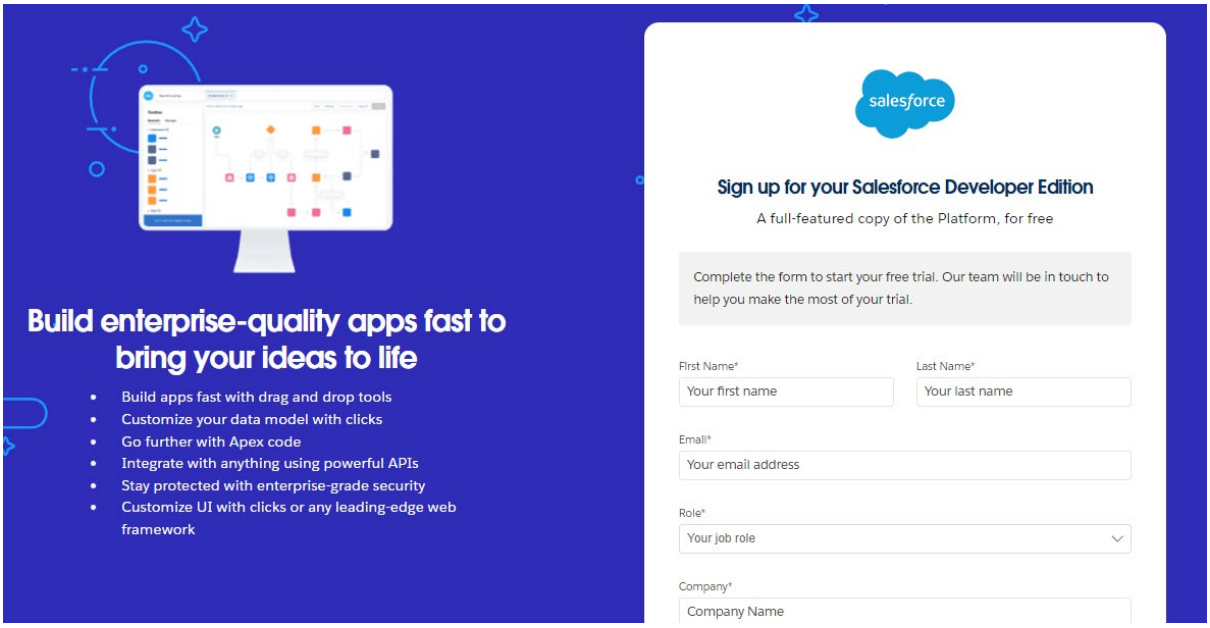
Creating a developer org in salesforce.

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

1. First name & Last name
2. Email
3. Role : Developer
4. Company : College Name
5. County : India
6. Postal Code : pin code
7. Username : should be a combination of your name and company

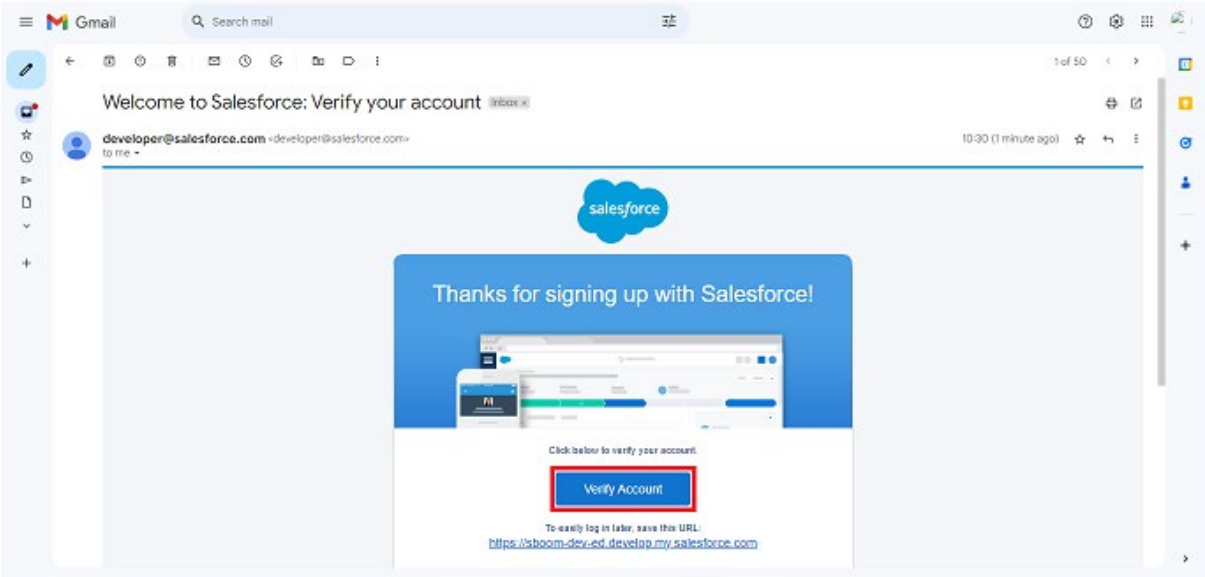
This need not be an actual email id, you can give anything in the format : username@organization.com

Click on sign me up after filling these.

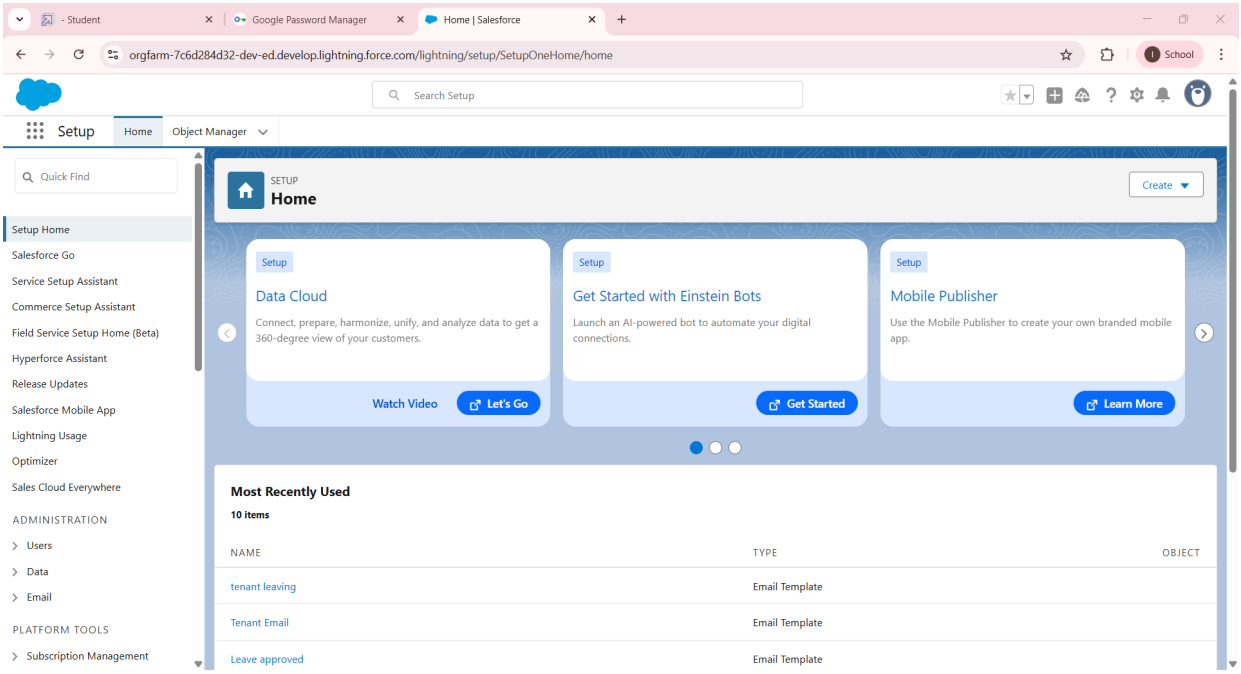


Account Activation:

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.



After the verify the account the page will show like this :



Object:

What Is an Object?

Salesforce objects are database tables that permit you to store data that is specific to an organization. What are the types of Salesforce objects.

To create an object:

1. From the setup page ? Click on Object Manager ? Click on Create ? Click on Custom Object.
2. On Custom object defining page:
3. Enter the label name, plural label name, click on Allow reports, Allow search.
4. Click on Save.

Create Property Object:

To create an object:

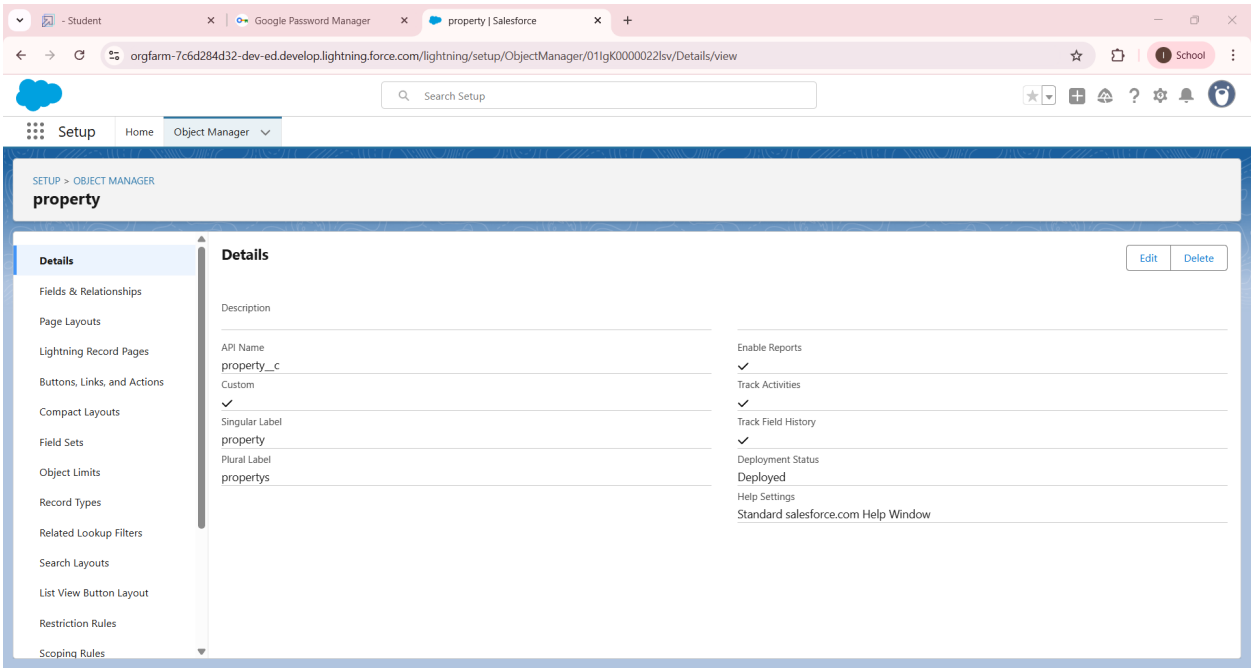
1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

1. Enter the label name>> property
2. Plural label name>> property
3. Enter Record Name Label and Format

- Record Name >>property Name
- Data Type >> Text

Click on Allow reports and Track Field History,Allow Activities

Allow search >> Save.



Create Tenant Object:

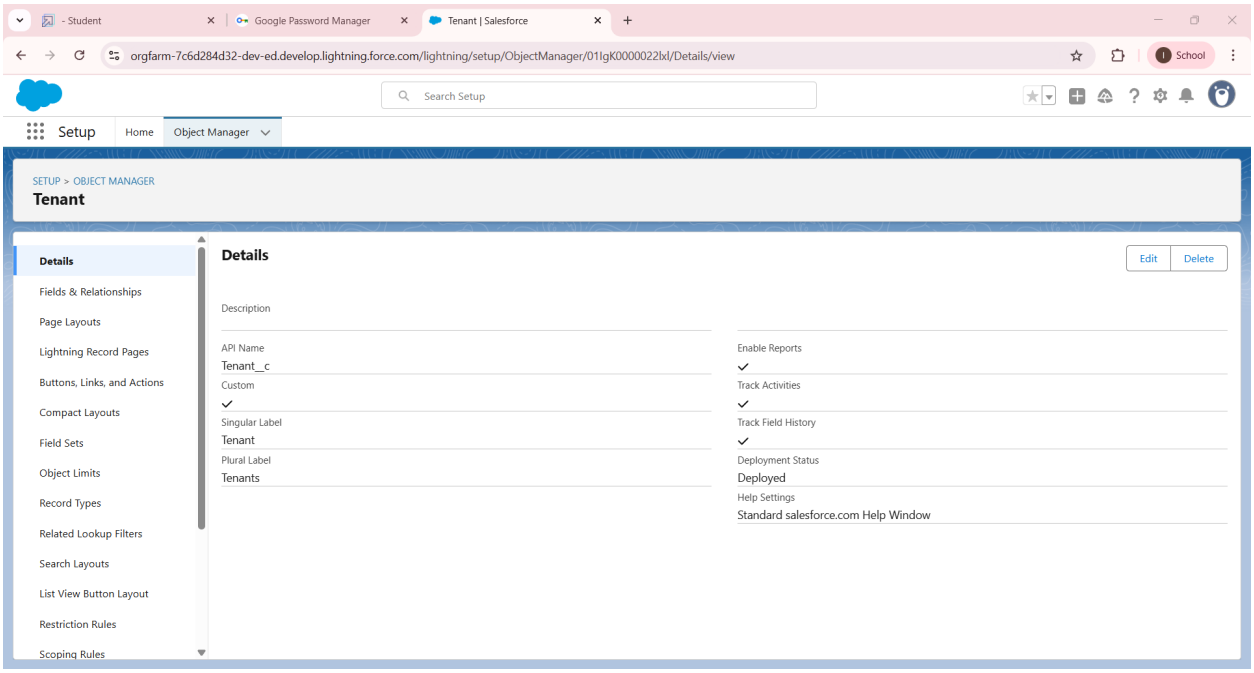
To create an object:

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

1. Enter the label name>> Tenant
2. Plural label name>> Tenants
3. Enter Record Name Label and Format

- Record Name >> Tenant Name
- Data Type >> Text

2. Click on Allow reports and Track Field History,Allow Activities
3. Allow search >> Save.



Create Payment Object

To create an object:

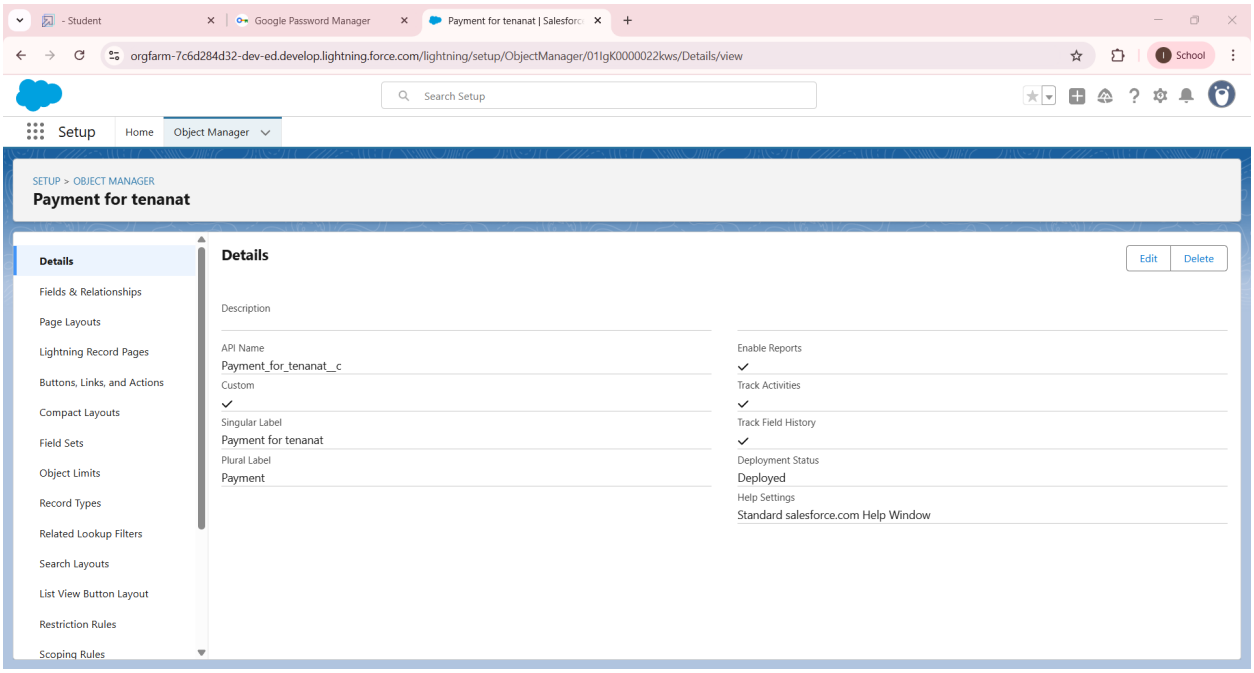
1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

1. Enter the label name>> Payment for tenanat
2. Plural label name>> Payment
3. Enter Record Name Label and Format

- Record Name >> Payment Name
- Data Type >> Text

2. Click on Allow reports and Track Field History,Allow Activities

Allow search >> Save.



Create Lease Object

To create an object:

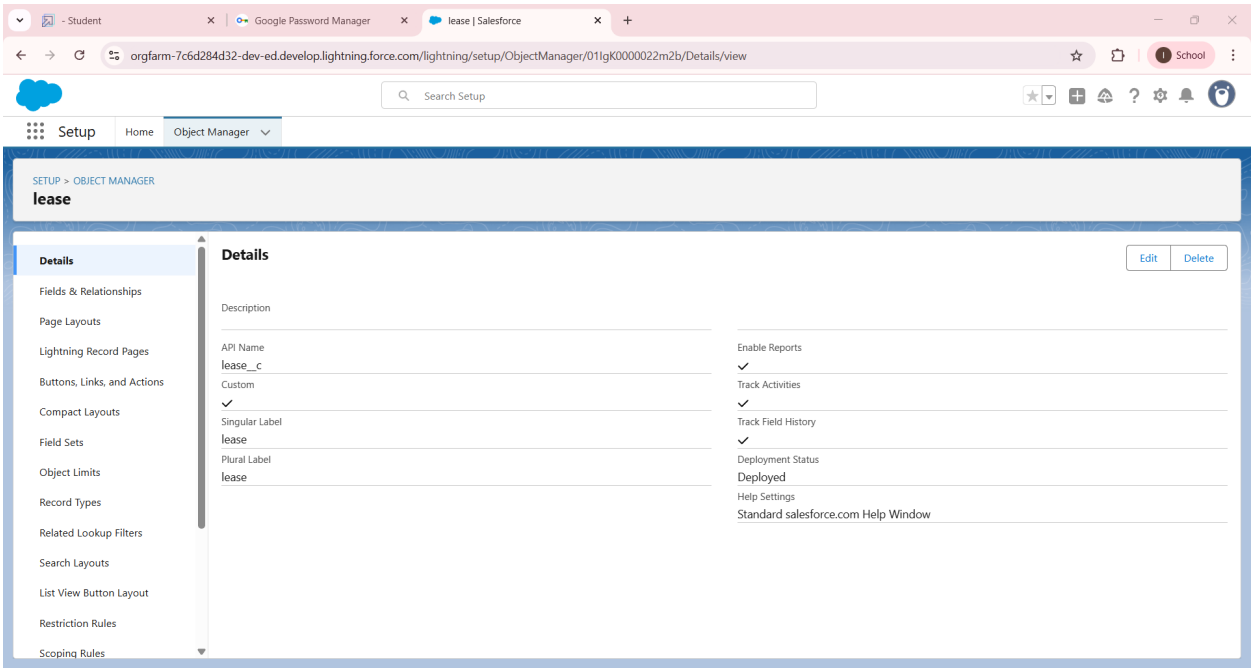
1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

1. Enter the label name>> lease
2. Plural label name>> lease
3. Enter Record Name Label and Format

- Record Name >> lease Name
- Data Type >> Text

2. Click on Allow reports and Track Field History,Allow Activities

3. Allow search >> Save.



Tabs

What is Tab :

A tab is like a user interface that is used to build records for objects and to view the records in the objects.

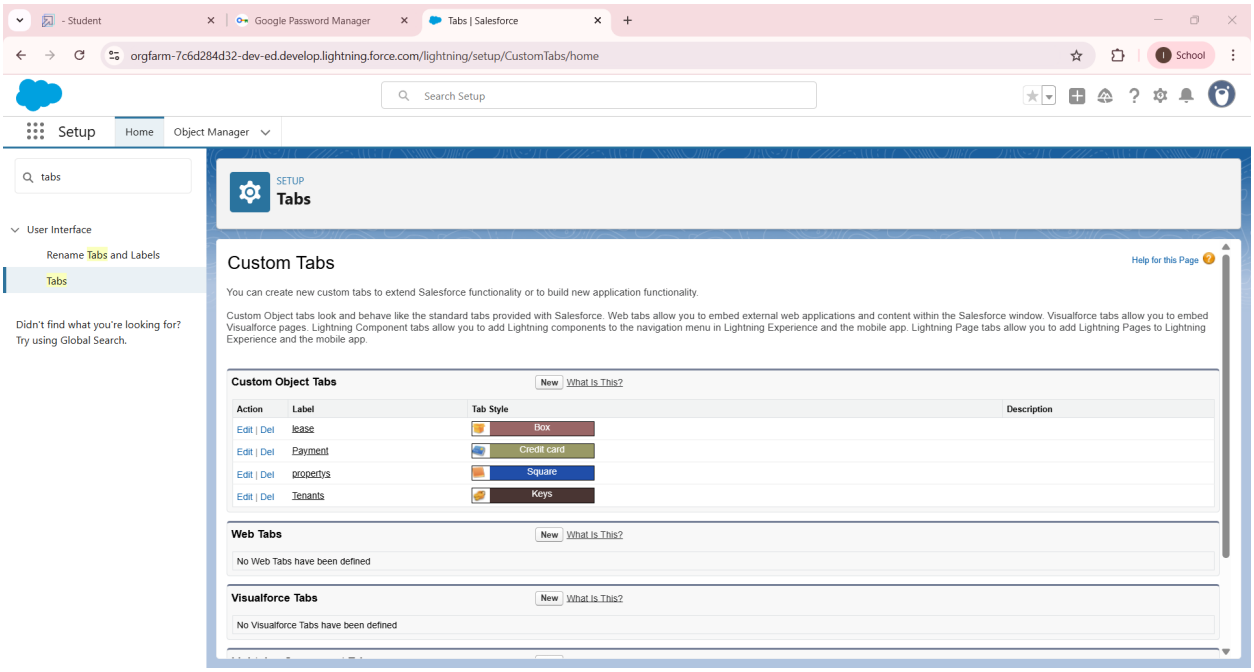
Creating a Custom Tab

To create a Tab:(Property)

1. Go to setup page >>type Tabs in Quick Find bar >> click on tabs >> New (under custom object tab)
2. Select Object(property) >> Select the tab style >> Next (Add to profiles page) keep it as default >> Next (Add to Custom App) uncheck the include tab .
3. Make sure that the Append tab to users' existing personal customizations is checked.
4. Click save

Creating Remaining Tabs

1. Now create the Tabs for the remaining Objects, they are “Payment for tenant,lease,tenant”.
2. Follow the same steps as mentioned in Activity -1 .
3. After completed the creating the tabs it show like this



Create a Lightning App

To create a lightning app page:

1. Go to setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.

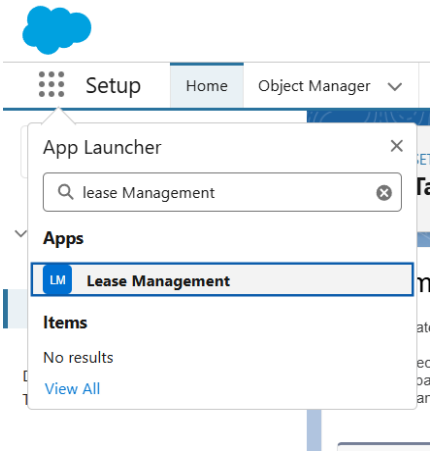
2. 2. Fill the app name in app details and branding as follow
3. App Name : Lease ManagementDeveloper Name : This will auto populatedImage : optional (if you want to give any image you can otherwise not mandatory) Primary colour hex value : keep this default.
4. 3.Then click Next >> (App option page)Set Navigation Style as Standard Navigation >> Next.
5. (Utility Items) keep it as default >> Next.
6. To Add Navigation Items:

7. 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.

8. 6. To Add User Profiles:

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

After creating the lightning app it show like this:



Fields

When we talk about Salesforce, Fields represent the data stored in the columns of a relational database. It can also hold any valuable information that you require for a specific object. Hence, the overall searching, deletion, and editing of the records become simpler and quicker.

Creation of fields for the property object

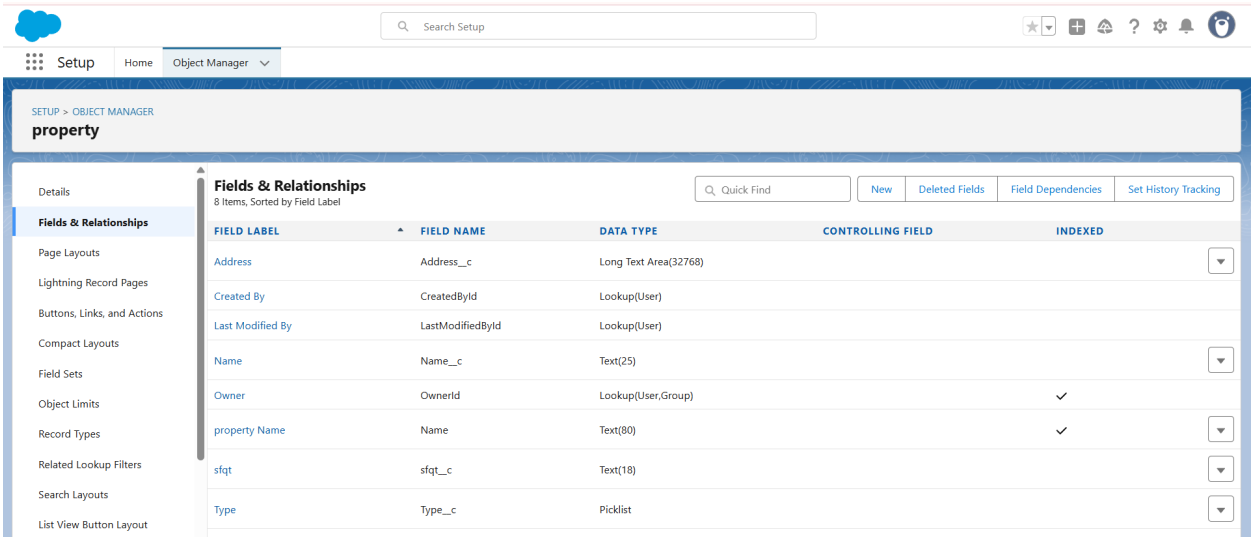
To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(property) in search bar >>click on the object.

2. Now click on “Fields & Relationships” >> New
3. Select Data Type as a “Text”

4. Click on next
5. 5. Fill the Above as following:
6. Field Label: Name
7. Field Name : gets auto generated
8. Length : 25
9. Required :check box
10. Click on Next >> Next >> Save and new.

11. 2. To create another fields in an object:
12. Go to setup >> click on Object Manager >>type object name(property) in search bar >>click on the object.
13. Now click on “Fields & Relationships” >>New
14. Select Data type as a “Long Text” and Click on Next
15. Fill the Above as following:
16. Field Label : Address
17. Field Name : gets auto generated
18. Click on Next >> Next >> Save and new.
19. 3. To create another fields in an object:
20. Go to setup >> click on Object Manager >>type object name(property) in search bar >> click on the object.
21. Now click on “Fields & Relationships” >> New
22. Select Data type as a “picklist” and Click on Next
23. Fill the Above as following:
24. Field Label : Type
25. Field Name : gets auto generated
26. Enter values, with each value separated by a new line
27. Enter these values
    1BHK
    2BHK
    3BHK
28. Click on Next >> Next >> Save and new.
29. To create another fields in an object:
30. Go to setup >> click on Object Manager >> type object name(property) in search bar >> click on the object.
31. Now click on “Fields & Relationships” >> New
32. Select Data type as a “ Text” and Click on Next
33. Fill the Above as following:
34. Field Label : sfqt
35. Field Name : gets auto generated
36. Length : 18
37. Click on Next >> Next >> Save.



Creation of fields for the Tenant object

1.Go to setup >> click on Object Manager >> type object name(Tenant) in search bar >> click on the object.

2. Now click on “Fields & 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
- Click on Next >> Next >> Save and new.

To create another fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Tenant) in search bar >> click on the object.
2. Now click on “Fields & 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
- Click on Next >> Next >> Save and new.

To create another fields in an object:

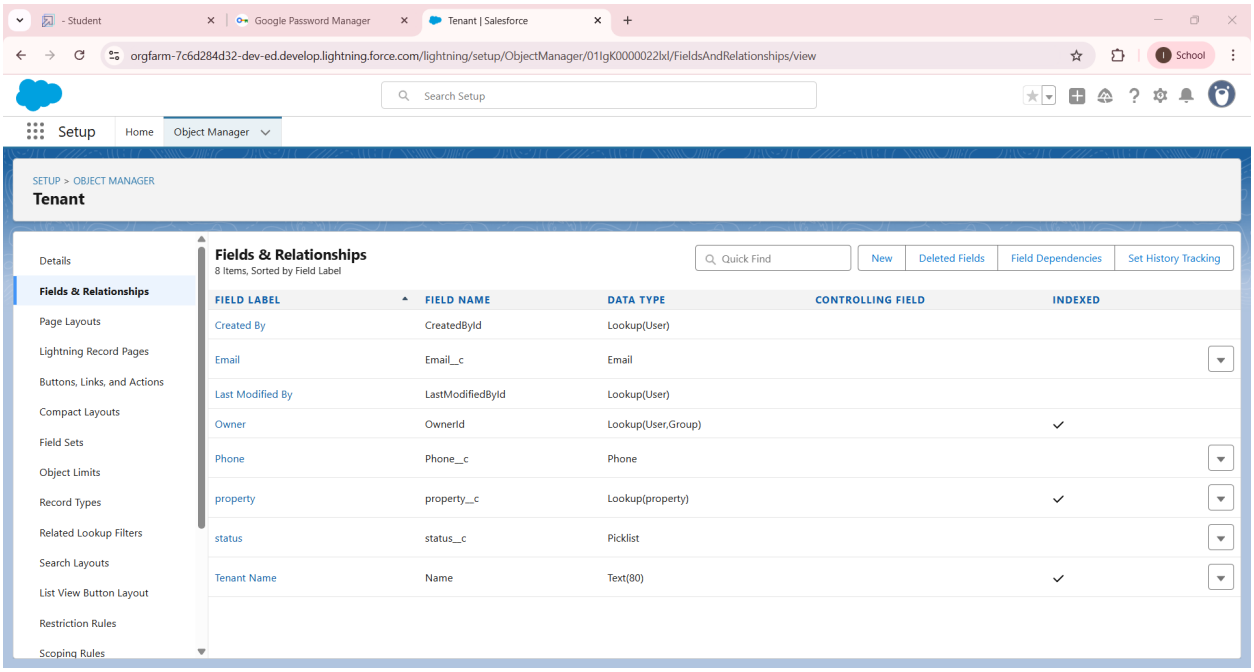
5. Go to setup >> click on Object Manager >> type object name(Tenant) in search bar >> click on the object.
6. Now click on “Fields & Relationships” >>New
7. Select Data type as a “picklist” and Click on Next
8. Fill the Above as following:

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

Stay

Leaving

- Click on Next >> Next >> Save



Creation of fields for the Lease object

1.Go to setup >> click on Object Manager >> type object name(Lease) in search bar >> click on the object.

2. Now click on “Fields & Relationships” >> New

3. Select Data type as a “Date” and Click on Next

4. Fill the Above as following:

- Field Label : start date
- Field Name : gets auto generated
- Click on Next >> Next >> Save and new.

To create another fields in an object:

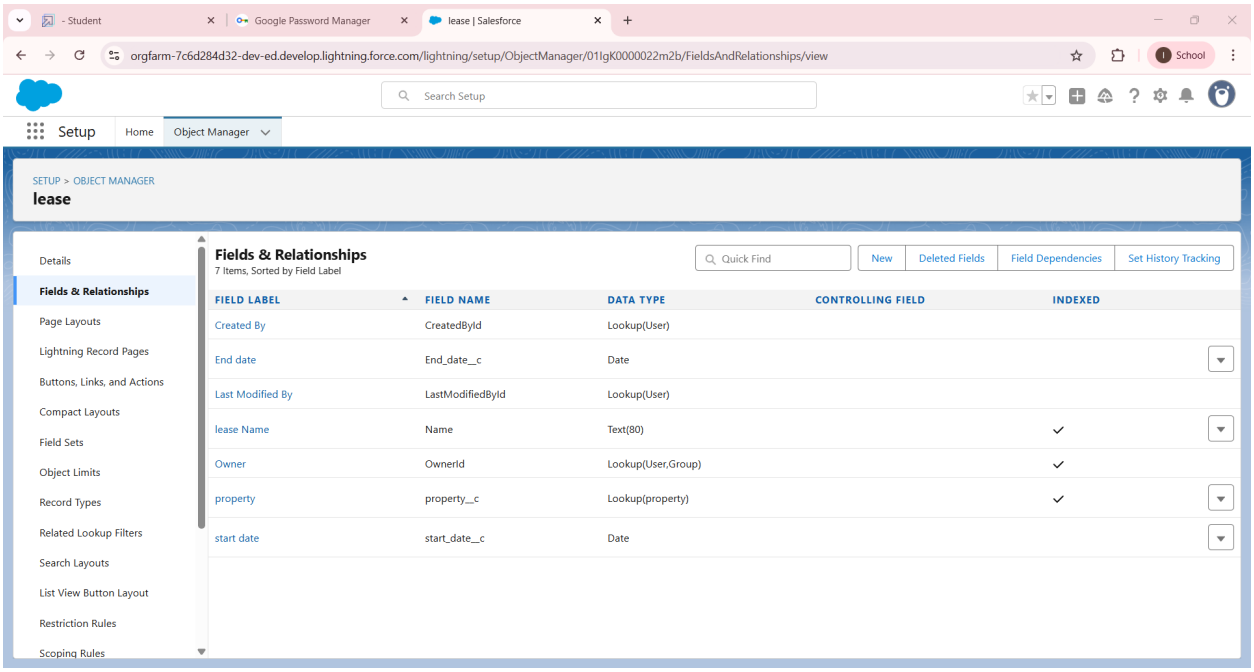
1.Go to setup >> click on Object Manager >> type object name(Lease) in search bar >> click on the object.

2. Now click on “Fields & Relationships” >> New

3. Select Data type as a “Date” and Click on Next

4. Fill the Above as following:

- Field Label : End date
- Field Name : gets auto generated
- Click on Next >> Next >> Save and new.



Creation of fields for the Payment for tenant object

1.Go to setup >> click on Object Manager >> type object name(Payment for tenant) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Date” and Click on Next
4. Fill the Above as following:

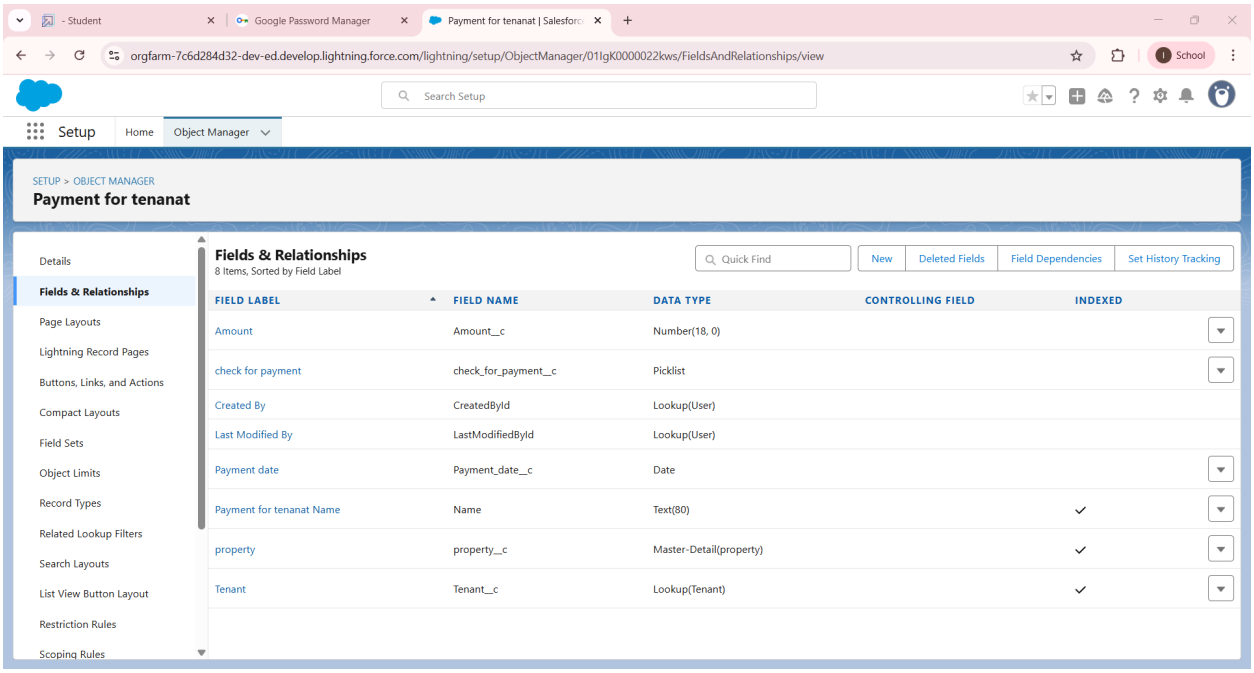
- Field Label : Payment date
- Field Name : gets auto generated
- Click on Next >> Next >> Save and new.

To create another fields in an object:
1.Go to setup >> click on Object Manager >> type object name(Payment for tenant) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Number” and Click on Next
4. Fill the Above as following:

- Field Label : Amount
- Length : 18
- Field Name : gets auto generated
- Click on Next >> Next >> Save and new.

To create another fields in an object:
1.Go to setup >> click on Object Manager >> type object name(Payment for tenant) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “picklist” and Click on Next
4. Fill the Above as following:

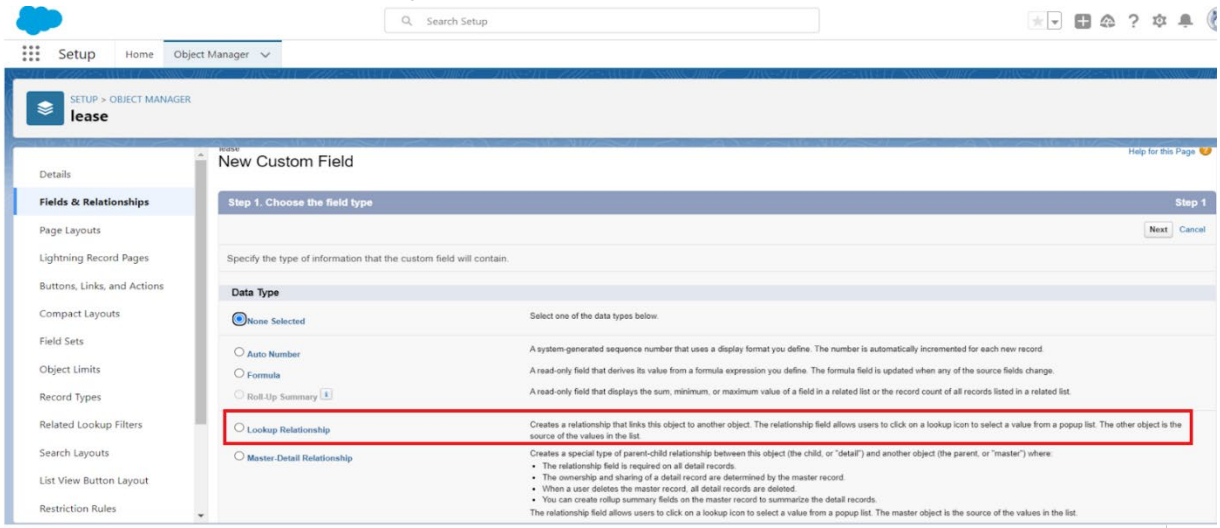
- Field Label : check for payment
- Field Name : gets auto generated
- Enter values, with each value separated by a new line
- Enter these values
  Paid
  Not paid
- Click on Next >> Next >> Save and new.



Creation of Lookup fields

Creation of Lookup Field on Lease Object :

1. Go to setup>> click on Object Manager >> type object name(Lease) in the search bar >> click on the object.



2. Now click on “Fields & Relationships” >> New
3. Select lookup relationship
4. Select the related object “ property” and click next.

5. Field Name : property
6. Field label : Auto generated
7. Next >> Next >> Save.

Creation of Lookup Field on Payment Object :

8. Go to setup >> click on Object Manager >> type object name(payment) in the search bar >> click on the object.
9. Now click on “Fields & Relationships” >> New
10. Select lookup relationship
11. Select the related object “ Tenant” and click next.
12. Field Name : Tenant
13. Field label : Auto generated
14. Next >> Next >> Save.

Creation of Lookup Field on Payment for tenant Object :

15. Go to setup>> click on Object Manager >> type object name(property) in the search bar >> click on the object.
16. Now click on “Fields & Relationships” >> New
17. Select masterdetail relationship
18. Select the related object “ property” and click next.
19. Field Name : property
20. Field label : Auto generated
21. Next >> Next >> Save.

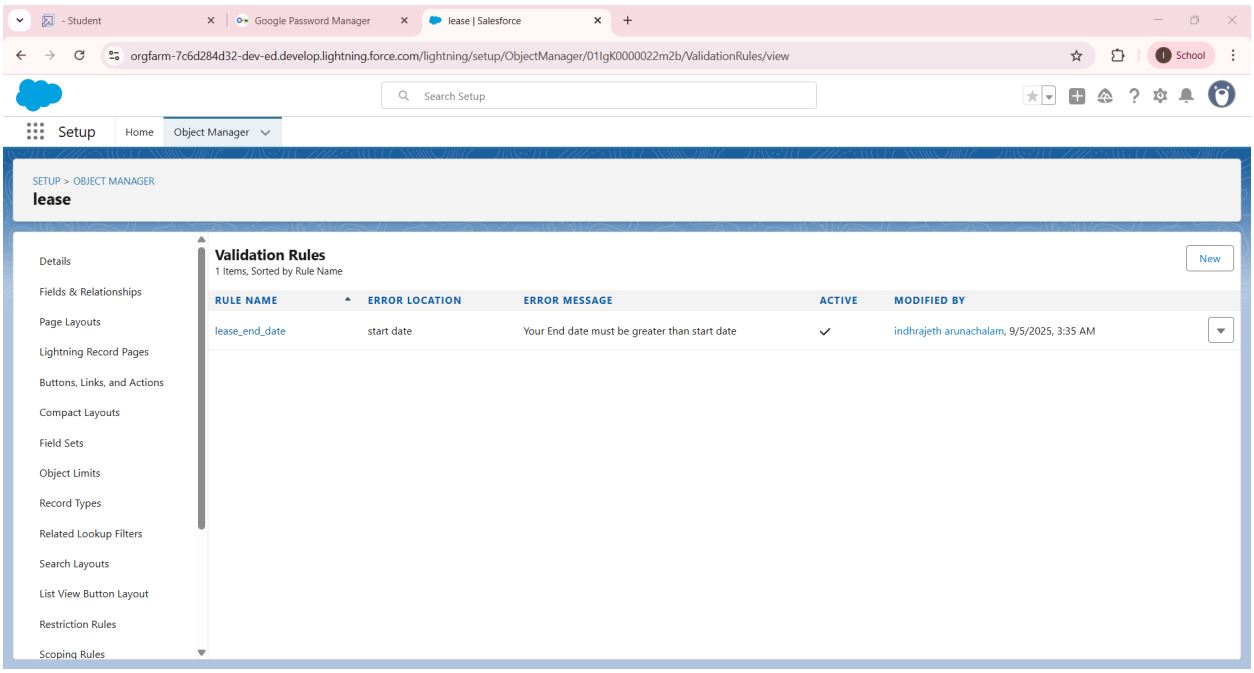
Validation rule

Validation rules are applied when a user tries to save a record and are used to check if the data meets specified criteria. If the criteria are not met, the validation rule triggers an error message and prevents the user from saving the record until the issues are resolved.

To create a validation rule to an Lease Object

1. Go to the setup page >> click on object manager >> From drop down click edit for Lease object.
2. Click on the validation rule >> click New.

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.
6. After completing this process it will show like this



Email Templates

We use email templates to increase productivity and ensure consistent messaging. Email templates with merge fields let you quickly send emails that include field data from Salesforce records like contacts, leads, or opportunities. You can use email templates when emailing groups of people—with list email or mass email—or just one person.

Salesforce email templates are the easiest way to get your emails done. They help you create and send quick emails that include merge fields from Salesforce records like Contacts, Leads, Opportunities, or Custom Objects. When you have a large number of contacts or leads in Salesforce, it can be difficult to keep track of who needs to be notified about new information.

Salesforce email templates allow you to combine all these contacts or leads into one email and then send it out simultaneously.

Create Email Template For Tenant Leaving

To create Email Template:

1. Go to setup in quick find box enter email template >> click on classic Email Template.
2. Click on >> New Email Template===>Choose text
   Folder : Unfiled 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”
6. Email body :

Dear {!Tenant__c.CreatedBy},

Please approve my leave

7. Save

Create Email Template For Leave Approved

To create Email Template:

1. Go to setup in quick find box enter email template >> click on classic Email Template.
2. Click on >> New Email Template===>Choose text
   Folder : Unfiled 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

7. Save

Create Email Template For rejection for leave

To create Email Template:

1. Go to setup in quick find box enter email template >> click on classic Email Template.
2. Click on >>New Email Template===>Choose text
   Folder : Unfiled 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},

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

your leave has rejected

8. Save

Create Email Template For Monthly payment

To create Email Template:

1. Go to setup in quick find box enter email template >> click on classic Email Template.
2. Click on >> New Email Template===>Choose text
   Folder : Unfiled 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. 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

Create Email Template For successful payment

To create Email Template:

1. Go to setup in quick find box enter email template >> click on classic Email Template.
2. Click on >> New Email Template===>Choose text
   Folder : Unfiled 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

Approval Process

What Is Approval Process In Salesforce?

The Approval Process is an automated process that an org uses to approve records in Salesforce. For example, When In the organization, someone is not able to decide a particular thing then he can ask someone else for approval. So, for such frequent cases or situations, one can define the approval process. So, Users can take benefit of such an approval process whenever needed.

Records submitted for approval are approved by the user(s) in the organization. These users are called Approvers. A single Approval process is bound to a single object because when a rule is defined, this object influences the fields that will be available to set the criteria.

Create Approval Process For check for vacant

To create fields in an object:

1.Go to setup >> Approval Processes in quick find bar>>click on it.
2.Manage Approval Process For >> “Tenant” from the drop down.

3.Click on “Create New Approval Process” >> Use standard setup wizard.

4. Process Name “check for vacant” >> Click Next.

5. Field “Tenant:status” >> Operator : Not equals , Value >> Click on the lookup filter icon and select “Leaving”.

6.Click insert field,then click Next.

7.Next Automated Approver determined by “None” from the drop down.

9. Select the “Administrators ONLY can edit records during the approval process”.Then Next.
9. Click on next leave the email template click on next
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.
- Then click save.
- Click on “i’ll do this later. Take me back to the listing of all approval process for this object”
- Click go

Initial Submission Action:

1. Under in itial submission action click on add new and then select email alert.

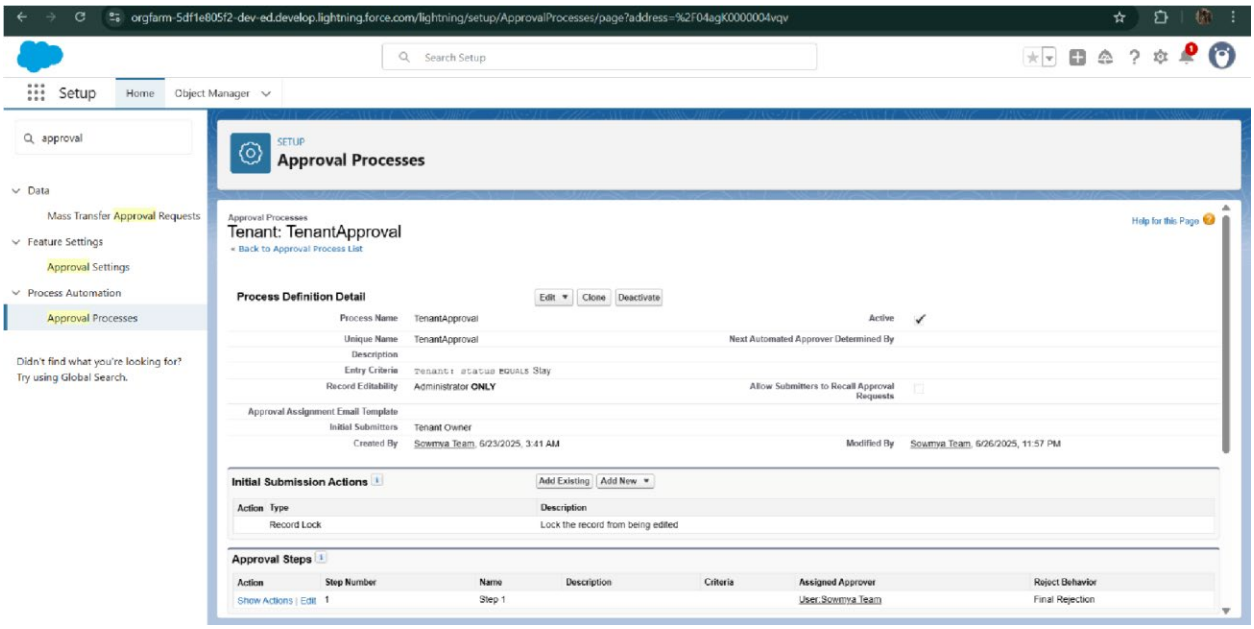
2. Description: “please approve my leave”.
3. unique name : auto populated
4. Email template : tenant leaving
5. Recipient type : Email field
6. Available Recipients : Email field : Email
7. From Email address : Current user’s email
8. Click save

Final Approval Action

1. Under Final approval action click o n 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

Final Rejection Action

1.Under final rejection action click on add 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

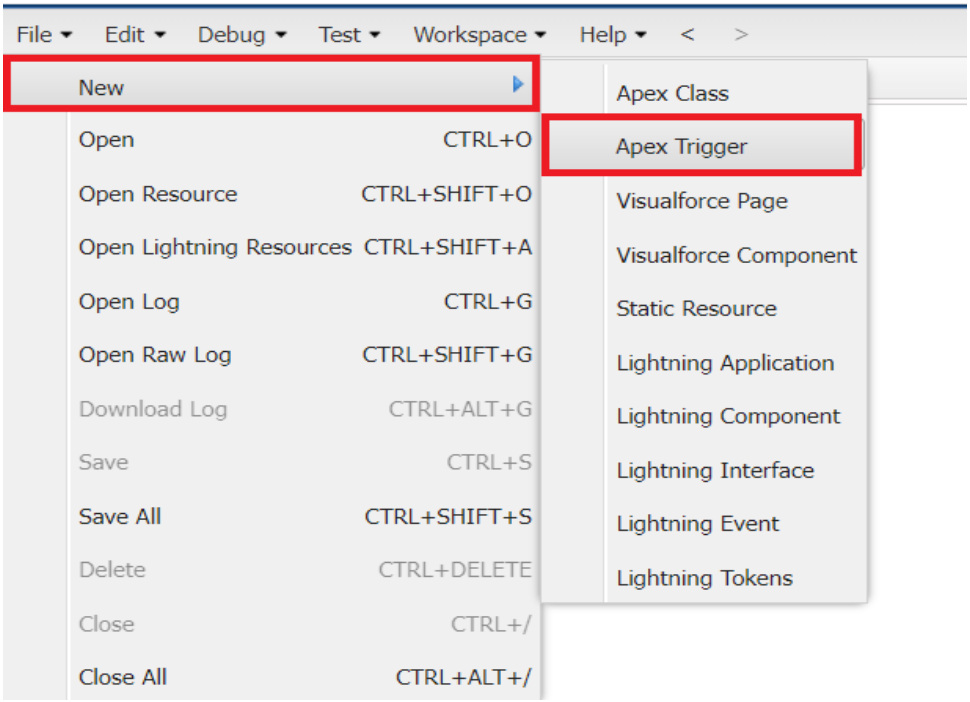


Apex Trigger

Create an Apex Trigger

1. To create a new Apex Class follow the below steps:

Click on the file >> New ? Apex Class.



1. Give the Apex Trigger name as “test”, and select “Tenant__c” from the dropdown for sObject.
2. Click Submit.
3. Now write the code logic here

Trigger Code:

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

Create an Apex Handler class

To create a new Apex Class follow the below steps:

Click on the file >> New >>Apex Class.

2. Enter class name as testHandler**.**

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)) {

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

            }

        }

    }

}
```

FLOWS

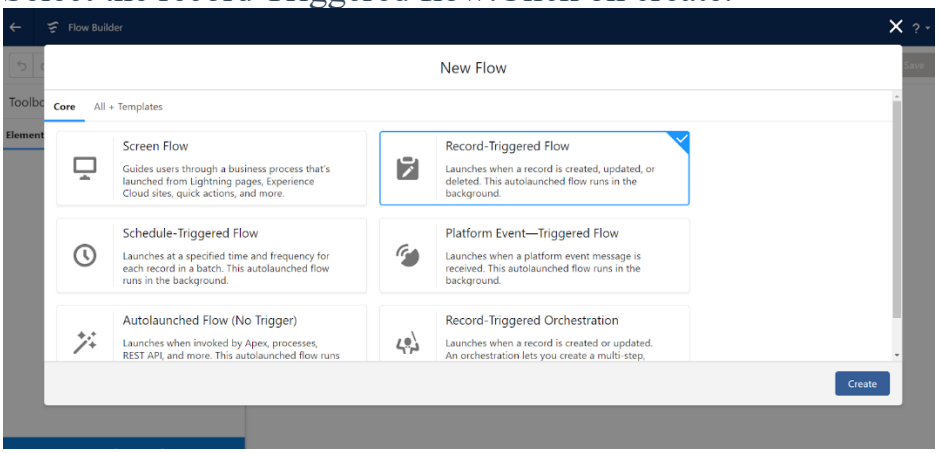
What is a flow ?

In Salesforce, a flow is a tool that automates complex business processes. Simply put, it collects data and then does something with that data.Flow Builder is the declarative interface used to build individual flows.

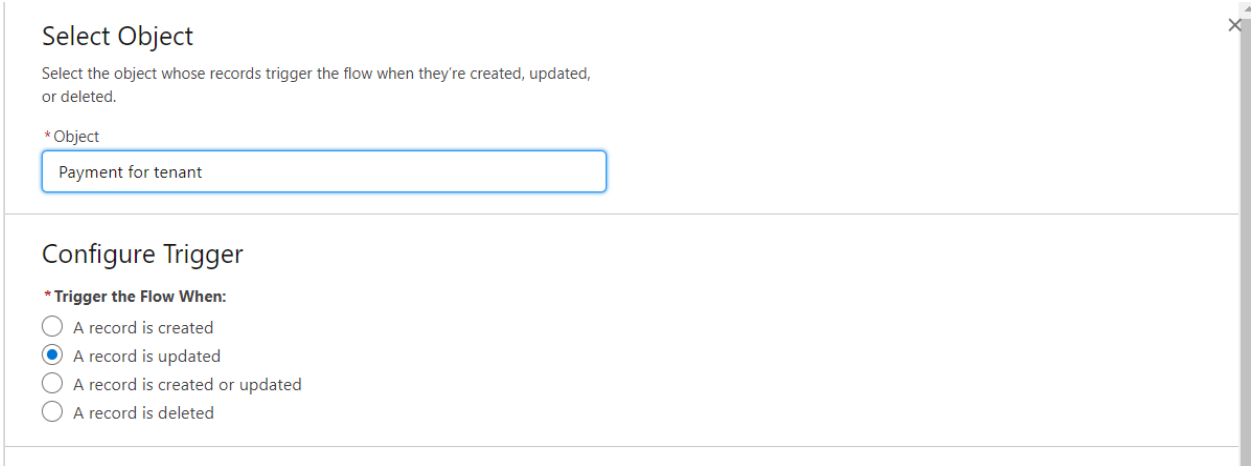
Create Flow for monthly payment

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.



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



4. Set Entry Conditions

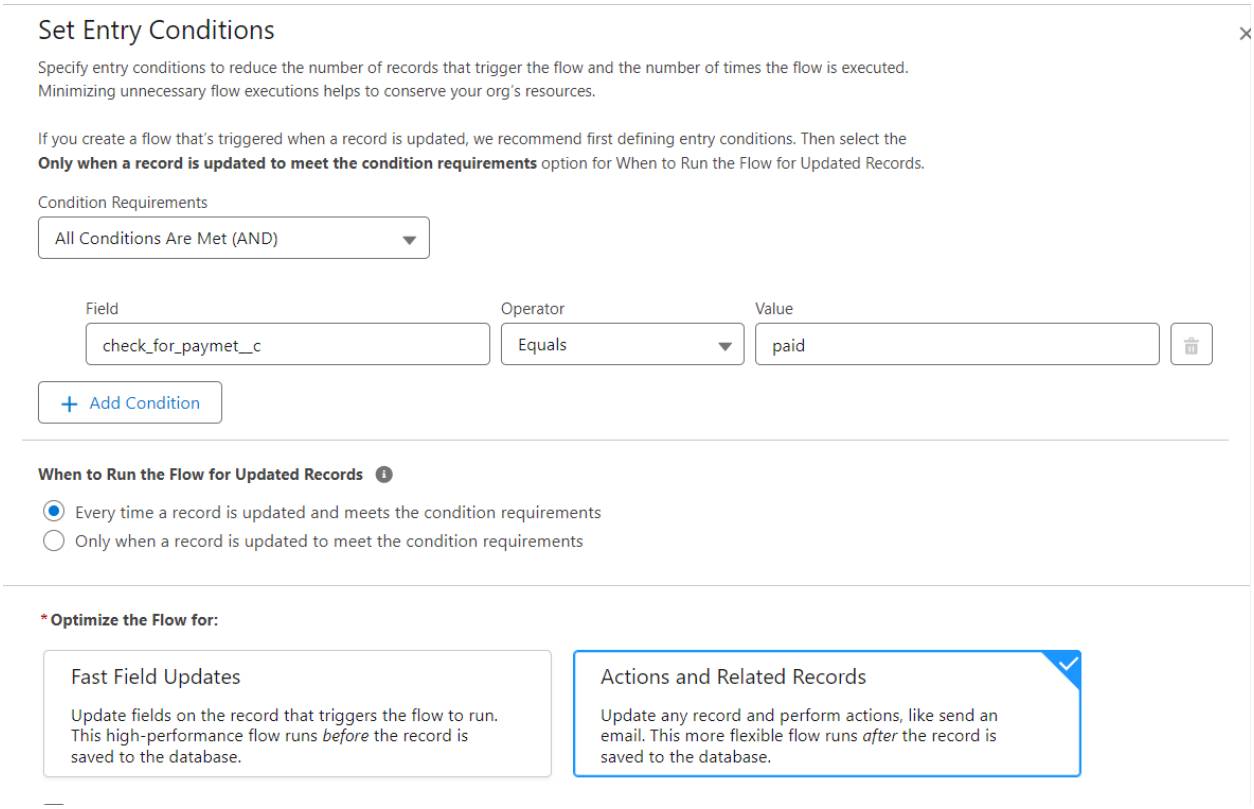
Under Condition Requirements

All Conditions are met

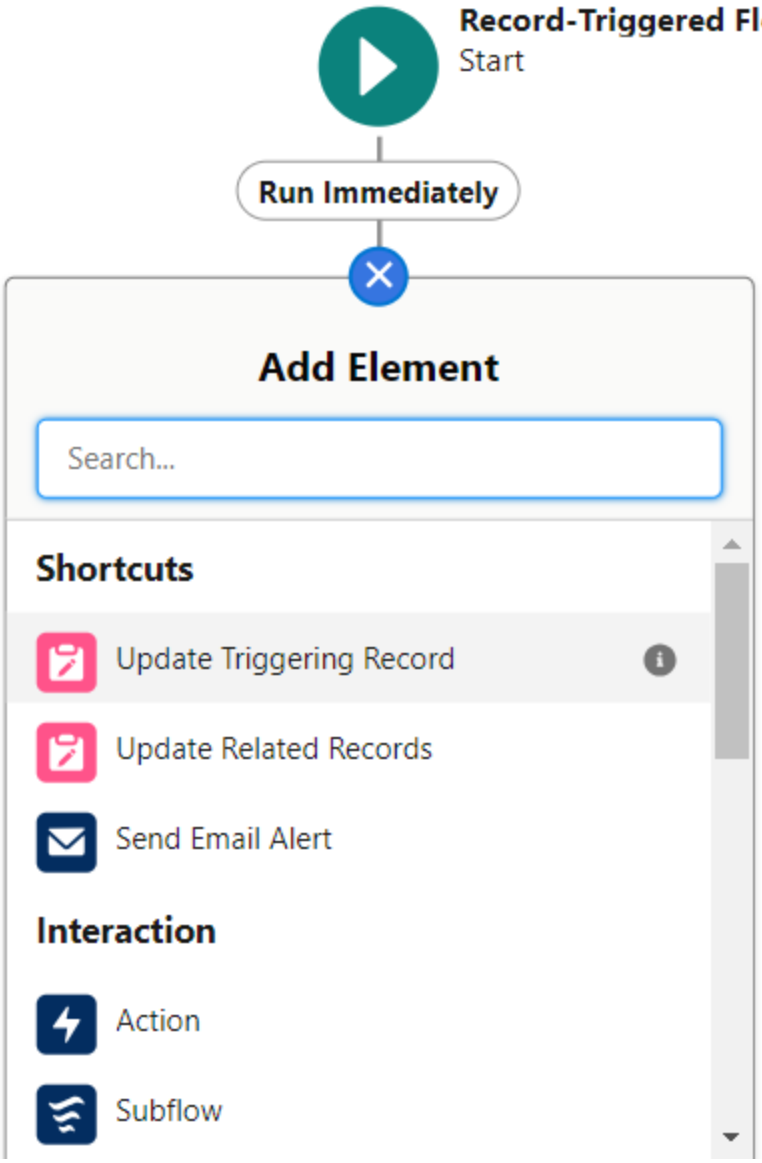
Field: check_for_payment__c	Operator: Equals	Value : paid

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

6. Click on : Actions and related records,done



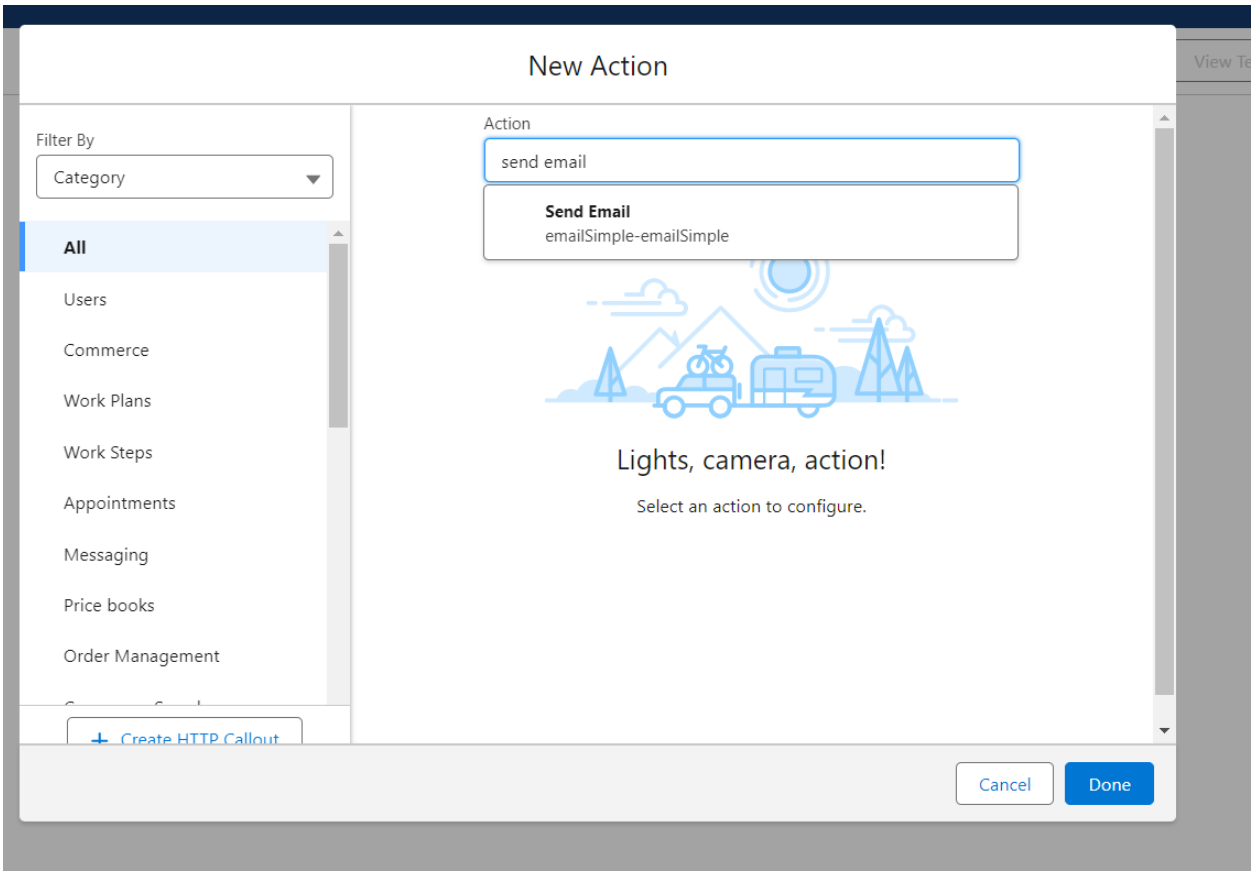
7. Under record trigger flow click on “+” icon and select action



In action search for send email then click on send email (check below image)

8. Label : send email

API Name : send_email

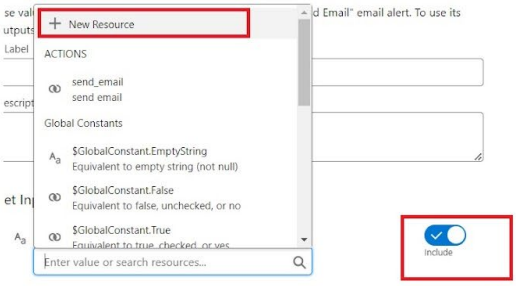


9. Label : send email

10. API Name : send_email

11. Enable Body

12. Click on new resource



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.

14. Click Done.

15. Enable recipient Address List

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

16. Click Done

17. Enable subject

Pate this >> Confirmation of Successful Monthly Payment

18. Click on save

Flow label : monthly payment

Flow API Name : monthly_payment

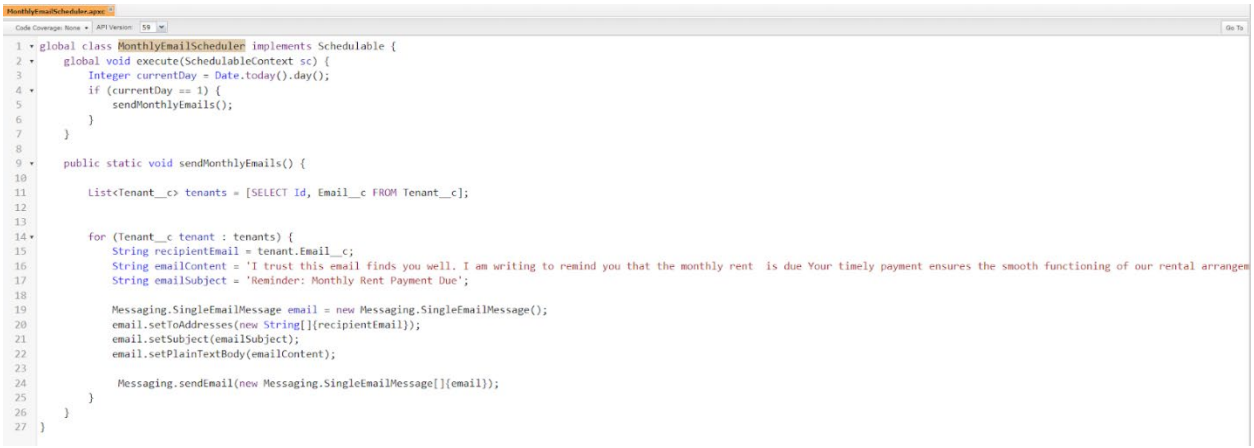
Click on activate

Create an Apex Class

1. To create a new Apex Class follow the below steps:

Click on the file >> New >> Apex Class.

2. Enter class name as MonthlyEmailScheduler.



```
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 arrangem
            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});
        }
    }
}
```

Apex logic:

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

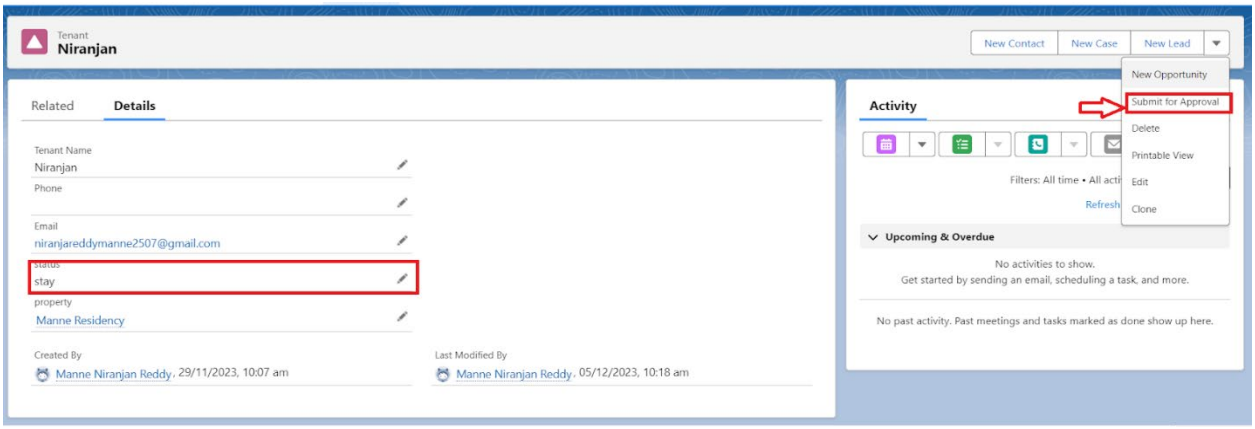
```
    }


    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});

        }

    }

}
```

Save the code.

Schedule Apex class

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

After creating the page show like this:



ADVANTAGES & DISADVANTAGES

- **Centralized Data** – All lease agreements, customer details, payment schedules, and related documents are stored in Salesforce, avoiding scattered records.
- **Automation** – Automates lease renewals, payment reminders, and approval workflows, reducing manual effort.
- **Integration** – Easily integrates with finance, accounting, and ERP systems for seamless invoicing and revenue tracking.
- **Analytics & Reporting** – Provides dashboards and reports for tracking lease revenue, expirations, utilization, and profitability.

- **Customer Experience** – Improves client interaction with quick access to lease details, digital signatures, and self-service portals.
- **Compliance & Audit Readiness** – Keeps an audit trail of all lease transactions, ensuring compliance with accounting standards (like IFRS 16 / ASC 842).
- **Scalability** – Handles multiple leases, assets, and geographies as the business grows without needing new software.
- **High Implementation Cost** – Customizing Salesforce for lease management may require additional licenses, consultants, or third-party apps.
- **Complex Setup** – Lease processes can be complex, and Salesforce often needs heavy customization to fully support them.
- **User Training Required** – Teams must be trained to use Salesforce lease modules effectively; otherwise adoption is slow.

CONCLUSION

The Lease Management system in Salesforce provides a **powerful, centralized, and automated platform** for handling lease agreements, payments, renewals, and compliance. By integrating with financial and operational systems, it enhances **efficiency, transparency, and customer experience**, while also offering **real-time insights** through reports and dashboards.

However, the project also highlights challenges such as **implementation cost, customization needs, and user training**, which must be addressed for successful adoption. Overall, Lease Management in Salesforce proves to be a **strategic solution** for organizations aiming to streamline operations, ensure compliance, and support business growth in a competitive environment.