GARAGE MANAGEMENT SYSTEM

BY,

TEAM: NM2024TMID01974

SARASWATHY COLLEGE OF ENGINEERING AND TECHNOLOGY

pvsvel1@gmail.com

Team Members:-

1. D.Seran - 421822104042
2. B.Sham - 421822104043
3. G.Siva - 421822104044
4. T.Sounder - 421822104045

Garage Management system

Project Description:

The Garage Management System is a valuable tool for automotive repair facilities, helping them deliver top-notch service, increase operational efficiency, and build lasting customer relationships. With its user-friendly interface and powerful features, GMS empowers garages to thrive in a competitive market while ensuring a seamless and satisfying experience for both customers and staff.

Project Flow:

In this project you can do hands on practice the configuration as well as customization with the Data modelling, App building, User Adoption & Apex Code

Milestone 1: Creation of developer account
Milestone 2: Object Creation
Milestone 3 : Tabs Creation
Milestone 4 : Create App
Milestone 5 : Fields & Relationships
Milestone 6 : Page Layout
Milestone 7 : Profile
Milestone 8 : Role and role hierarchy
Milestone 9 : Users
Milestone 10 : Permission set
Milestone 11 : User Adoption
Milestone 12 : Reports
Milestone 13 : Dashboards
Milestone 14 : Flows
Milestone 15 : Apex Classes and Triggers

What you'll learn

1. Real Time Salesforce Project
2. Object & Relationship in Salesforce
3. Formula fields and Validation rules.
4. Cross object formula fields.
5. Rollup summary fields.
6. Reports and dashboards
7. Conditional formatting.
8. Flows

9. Email alerts and email templates

Milestone 1-Salesforce :

Introduction:

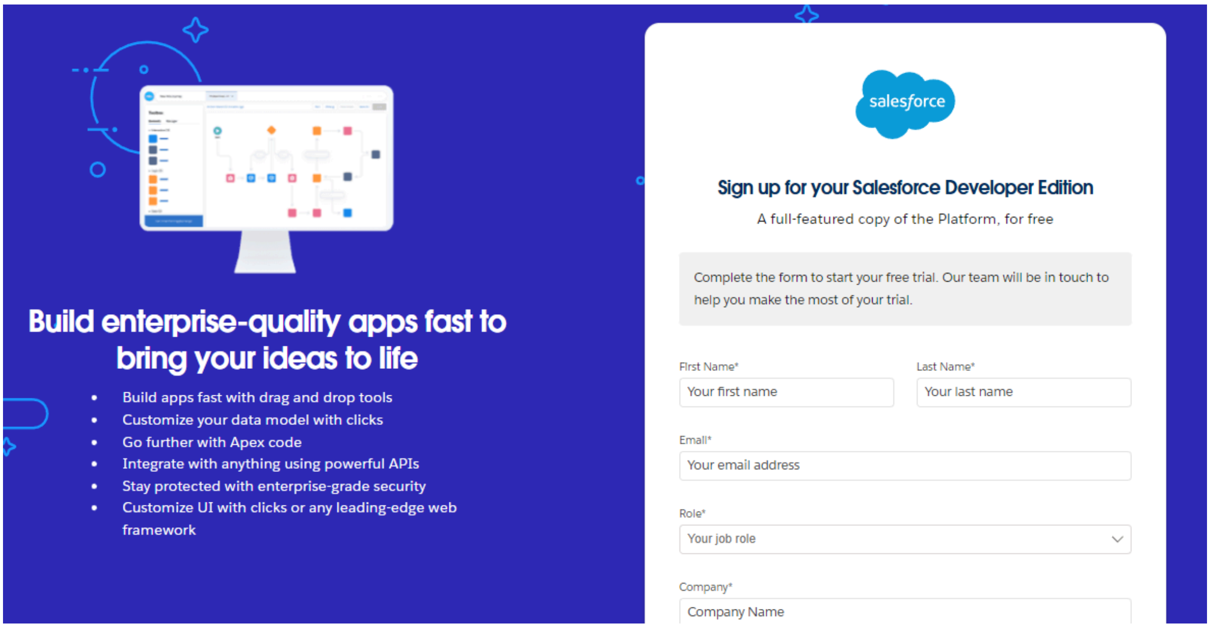
Are you new to Salesforce? Not sure exactly what it is, or how to use it? Don't know where you should start on your learning journey? If you've answered yes to any of these questions, then you're in the right place. This module is for you.
Welcome to Salesforce! Salesforce is game-changing technology, with a host of productivity-boosting features, that will help you sell smarter and faster. As you work toward your badge for this module, we'll take you through these features and answer the question, "What is Salesforce, anyway?".

What Is Salesforce?

Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers.
Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud.
So what does that really mean? Well, before Salesforce, your contacts, emails, follow-up tasks, and prospective deals might have been organized something like this:
https://youtu.be/r9EX3lGde5k

Activity 1: 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 :



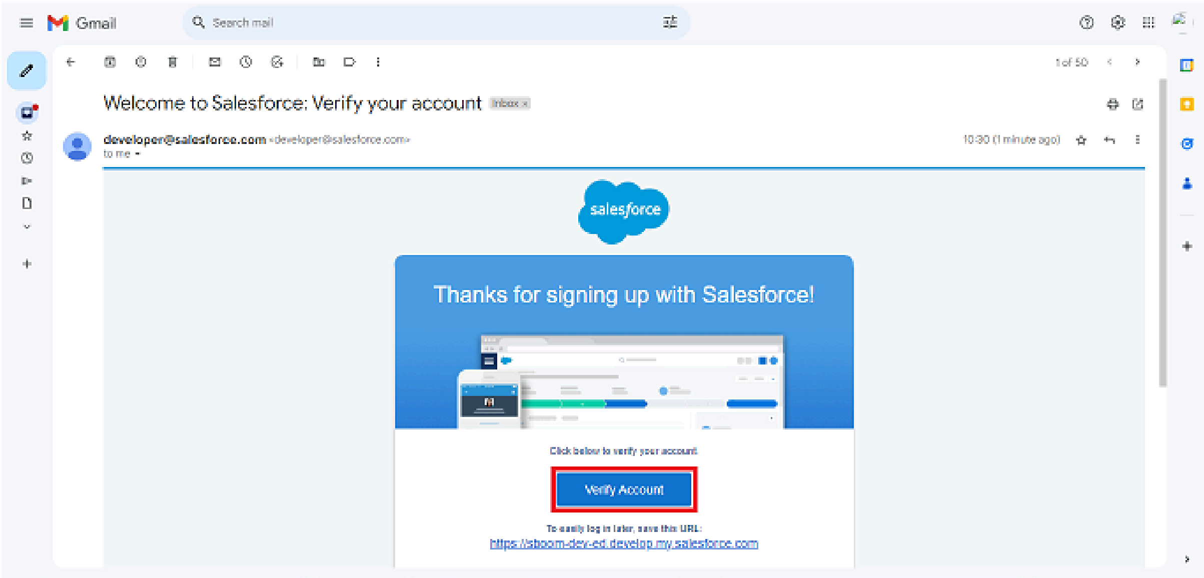
1) First name & Last name
2) Email
3) Role : Developer
4) Company : College Name
5) County : India
6) Postal Code : pin code
7) Username : should be a combination of your name and company

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

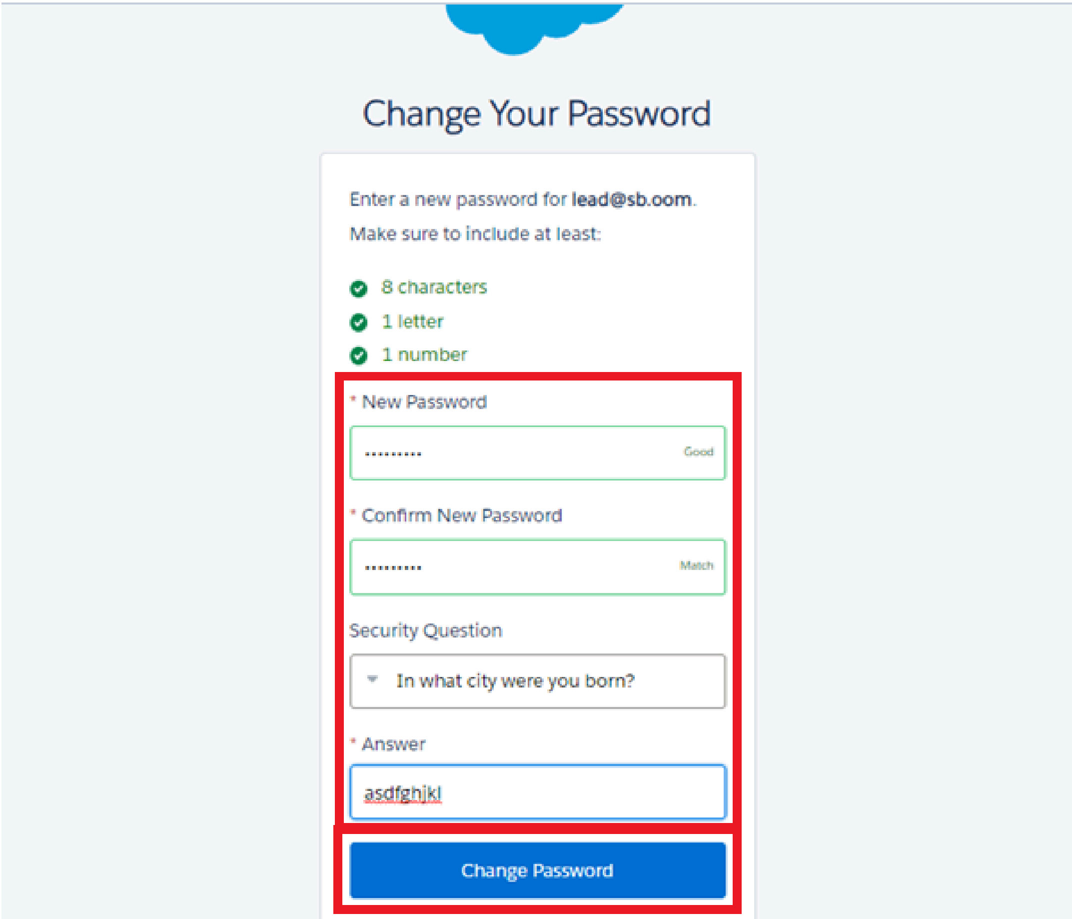
Click on sign me up after filling these.

Activity 2: Account Activation:

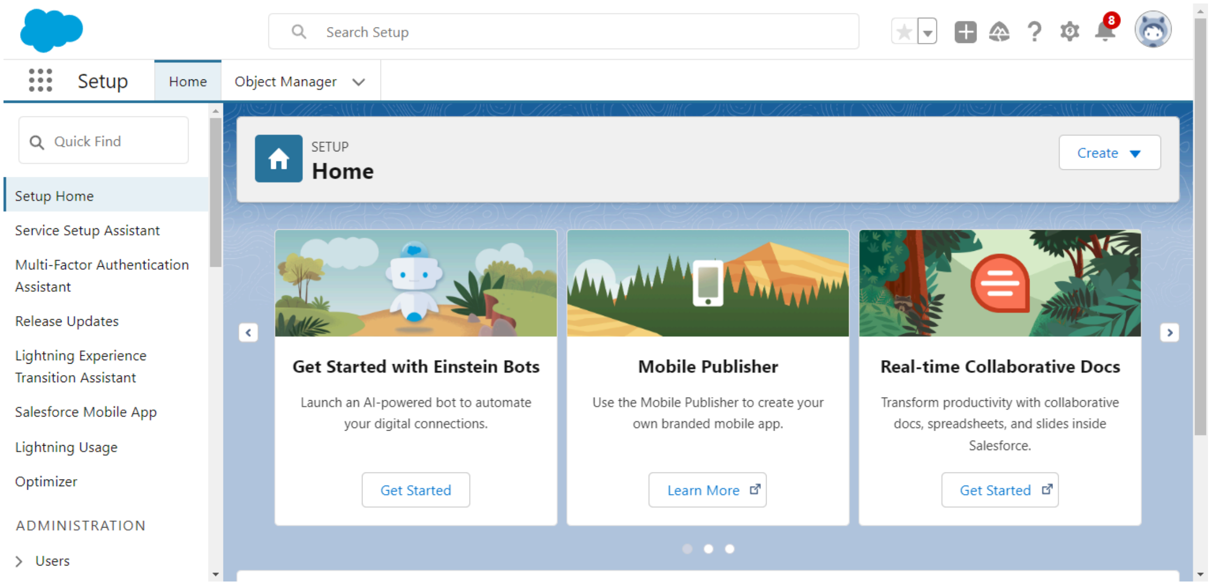
1. Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.



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



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



Milestone 2- Object

What Is an Object?

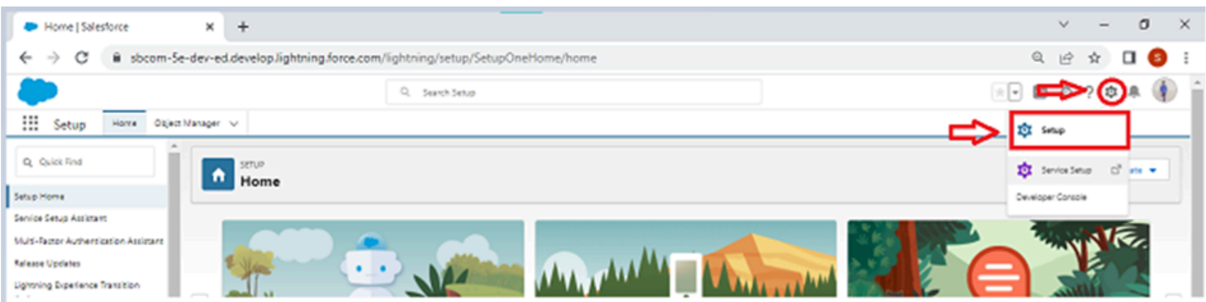
Salesforce objects are database tables that permit you to store data that is specific to an organization. What are the types of Salesforce objects

Salesforce objects are of two types:

1. **Standard Objects**: Standard objects are the kind of objects that are provided by salesforce.com such as users, contracts, reports, dashboards, etc.
2. **Custom Objects:** Custom objects are those objects that are created by users. They supply information that is unique and essential to their organization. They are the heart of any application and provide a structure for sharing data.

To Navigate to Setup page:

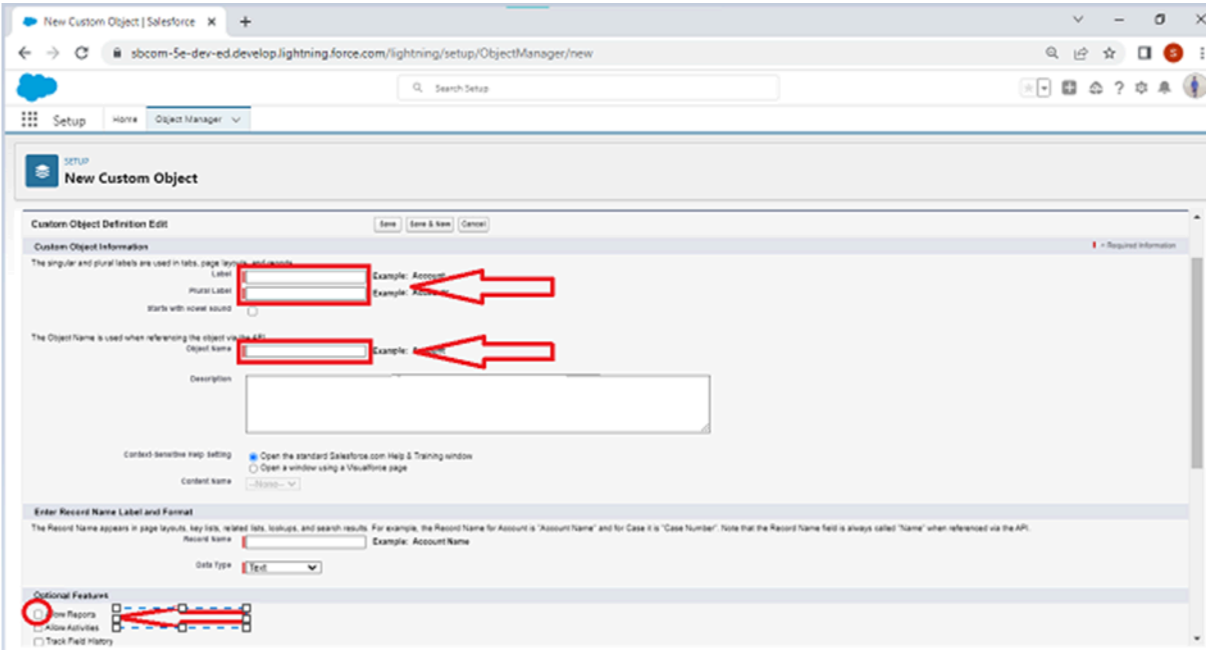
Click on gear icon → click setup.

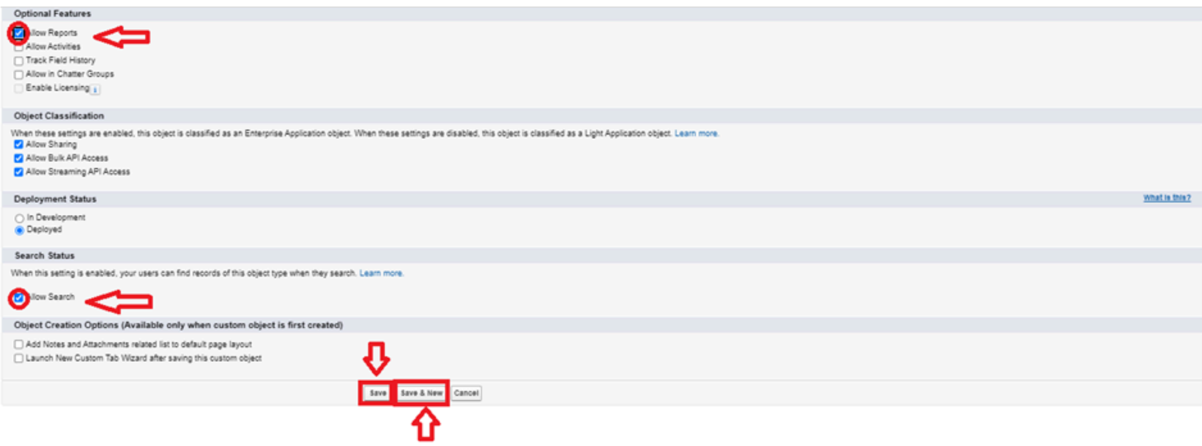


To create an object:

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.

Activity 1: Create Customer DetailsObject:

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→ Customer Details
    2) Plural label name→ Customer Details
    3) Enter Record Name Label and Format
        - Record Name → Customer Name
        - Data Type → Text
2. Click on Allow reports and Track Field History,
3. Allow search → **Save.**

Activity 2: Create Appointment 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→ Appointment
    2) Plural label name→ Appointments
    3) Enter Record Name Label and Format
        - Record Name → Appointment Name
        - Data Type → Auto Number
        - Display Format → app-{000}
        - Starting number → 1
2. Click on Allow reports and Track Field History,
3. Allow search → **Save.**

Activity 3: Create Service records Object:

To create an object:

4. From the setup page → Click on Object Manager → Click on Create → Click on Custom Object.
    4) Enter the label name→ Service records
    5) Plural label name→ Service records
    6) Enter Record Name Label and Format
        - Record Name → Service records Name
        - Data Type → Auto Number
        - Display Format → ser-{000}
        - Starting number → 1
5. Click on Allow reports and Track Field History,
6. Allow search → **Save.**

Activity 4: Create Billing details and feedback Object:

To create an object:

7. From the setup page → Click on Object Manager → Click on Create → Click on Custom Object.
    7) Enter the label name→ Billing details and feedback

8) Plural label name→ Billing details and feedback
9) Enter Record Name Label and Format
    - Record Name → Billing details and feedback Name
    - Data Type → Auto Number
    - Display Format → bill-{000}
    - Starting number → 1

8. Click on Allow reports and Track Field History,
9. Allow search → **Save.**

Milestone 3- 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.

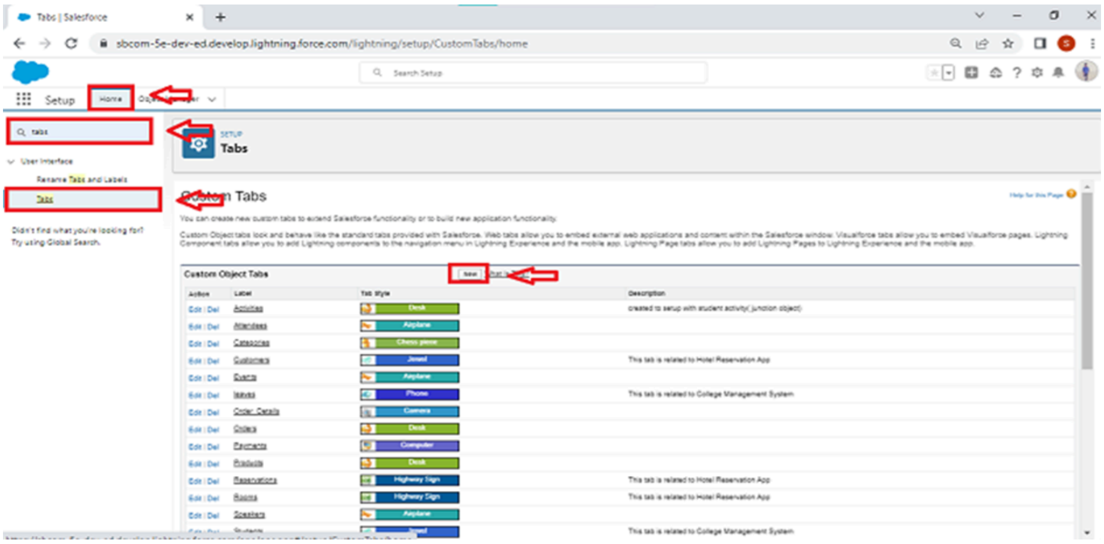
Types of Tabs:

1. **Custom Tabs**
   Custom object tabs are the user interface for custom applications that you build in salesforce.com. They look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.
2. **Web Tabs**
   Web Tabs are custom tabs that display web content or applications embedded in the salesforce.com window. Web tabs make it easier for your users to quickly access content and applications they frequently use without leaving the salesforce.com application.
3. **Visualforce Tabs**
   Visualforce Tabs are custom tabs that display a Visualforce page. Visualforce tabs look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.
4. **Lightning Component Tabs**
   Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app.
5. **Lightning Page Tabs**
   Lightning Page Tabs let you add Lightning Pages to the mobile app navigation menu.
   Lightning Page tabs don't work like other custom tabs. Once created, they don't show up on the All Tabs page when you click the Plus icon that appears to the right of your current tabs. Lightning Page tabs also don't show up in the Available Tabs list when you customise the tabs for your apps.

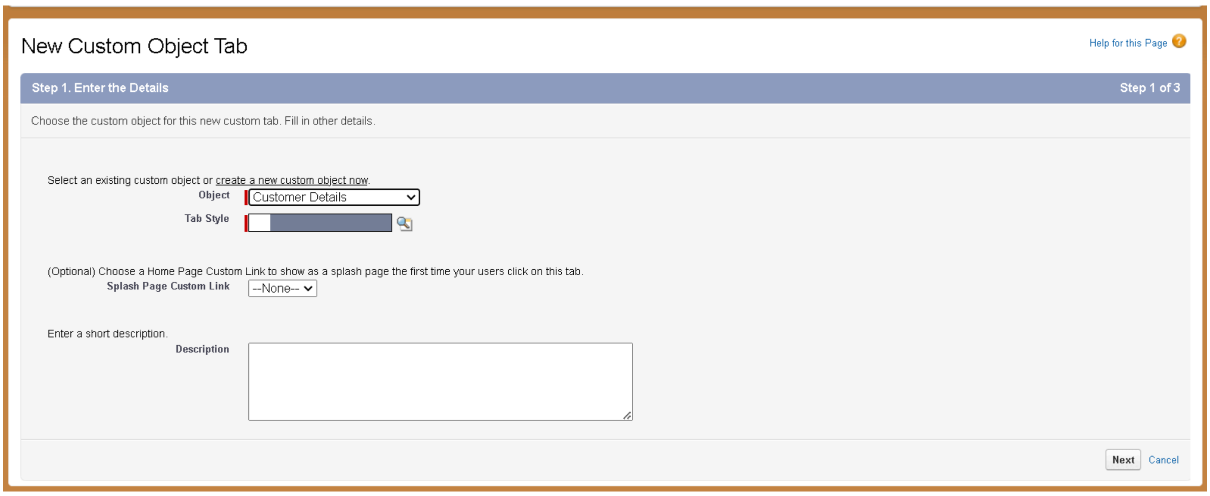
Activity 1: Creating a Custom Tab

To create a Tab:(Customer Details)

1. Go to setup page → type Tabs in Quick Find bar → click on tabs → New (under custom object tab)

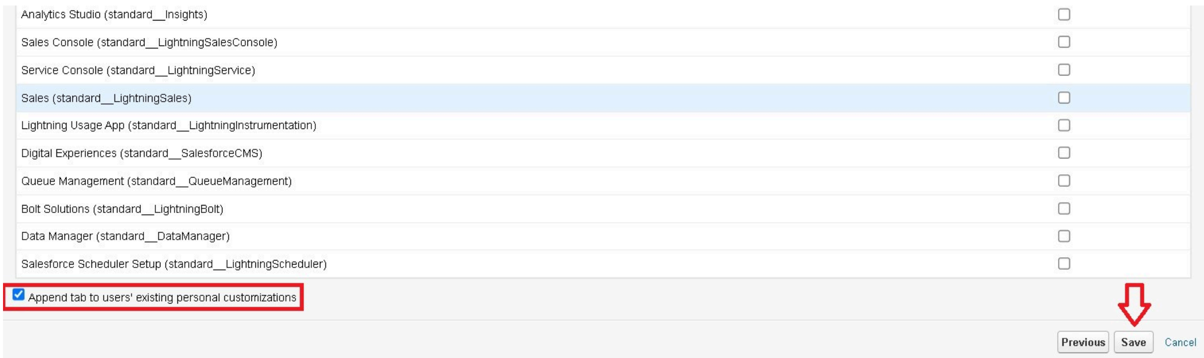


2. Select Object(Customer Details) → Select the tab style → Next (Add to profiles page) keep it as default → Next (Add to Custom App) uncheck the include tab .
3. Make sure that the Append tab to users' existing personal customizations is checked.
4. Click save.




Tab Style Selector
Create your own style
Hide styles which are used on other tabs
Airplane
Alarm clock
Apple
Balls
Bank[1]
Bell
Big top
Boat[1]
Books
Bottle
Box
Bridge
Building
Building Block
Caduceus
Camera
Can
Car
Castle
CD/DVD
Cell phone
Chalkboard
Chess piece
Chip
Circle
Compass
Computer
Credit card
CRT TV
Cup
Desk[1]
Diamond
Dice
Factory
Fan
Flag
Form
Gears
Globe
Guitar
Hammer
Hands
Handsaw
Headset
Heart[1]
Helicopter
Hexagon
Highway Sign
Hot Air Balloon
Insect
IP Phone
Jewel
Keys
Laptop
Leaf
Lightning
Save
Cancel


Step 3. Add to Custom Apps
Step 3 of 3
Choose the custom apps for which the new custom tab will be available. You may also examine or alter the visibility of tabs from the detail and edit pages of each Custom App.
Custom App
Include Tab
Platform (standard__Platform)
Sales (standard__Sales)
Service (standard__Service)
Marketing (standard__Marketing)
Sample Console (standard__ServiceConsole)
High Volume Customer Portal User
Authenticated Website User
App Launcher (standard__AppLauncher)


Analytics Studio (standard__Insights)
Sales Console (standard__LightningSalesConsole)
Service Console (standard__LightningService)
Sales (standard__LightningSales)
Lightning Usage App (standard__LightningInstrumentation)
Digital Experiences (standard__SalesforceCMS)
Queue Management (standard__QueueManagement)
Bolt Solutions (standard__LightningBolt)
Data Manager (standard__DataManager)
Salesforce Scheduler Setup (standard__LightningScheduler)
Append tab to users' existing personal customizations
Previous
Save
Cancel

Activity 2: Creating Remaining Tabs

1. Now create the Tabs for the remaining Objects, they are " Appointments, Service records,Billing details and feedback".
2. Follow the same steps as mentioned in Activity -1 .

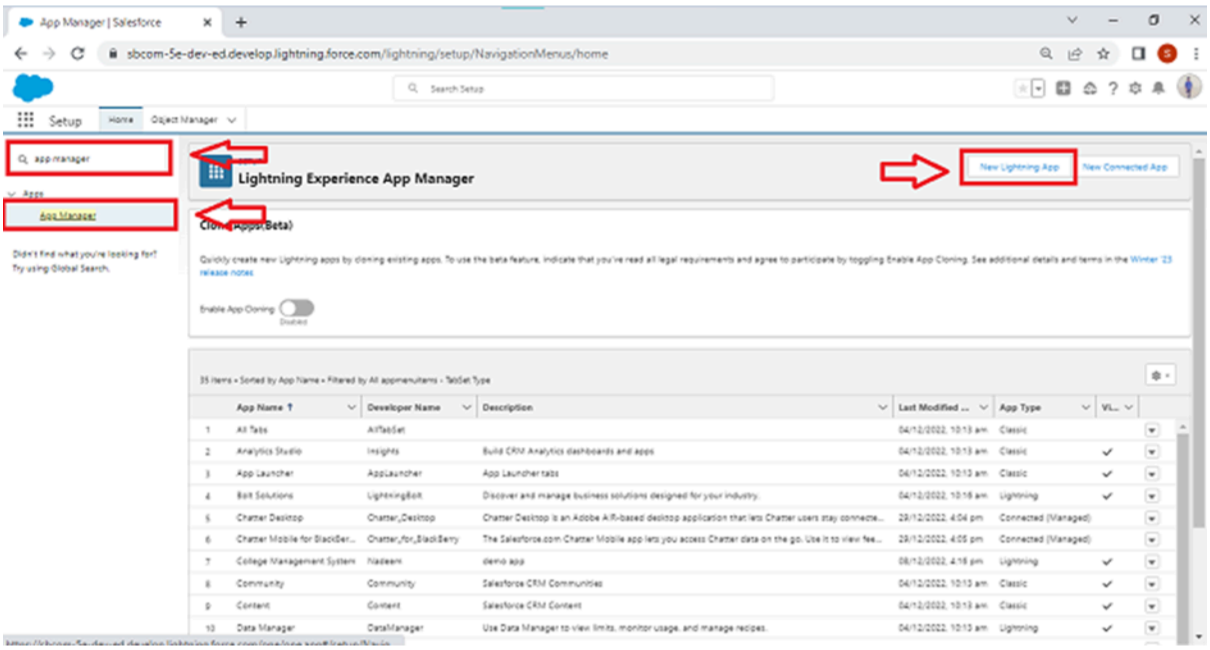
Milestone 4- The Lightning App:

An app is a collection of items that work together to serve a particular function. In Lightning Experience, Lightning apps give your users access to sets of objects, tabs, and other items all in one convenient bundle in the navigation bar.
Lightning apps let you brand your apps with a custom colour and logo. You can even include a utility bar and Lightning page tabs in your Lightning app. Members of your org can work more efficiently by easily switching between apps.

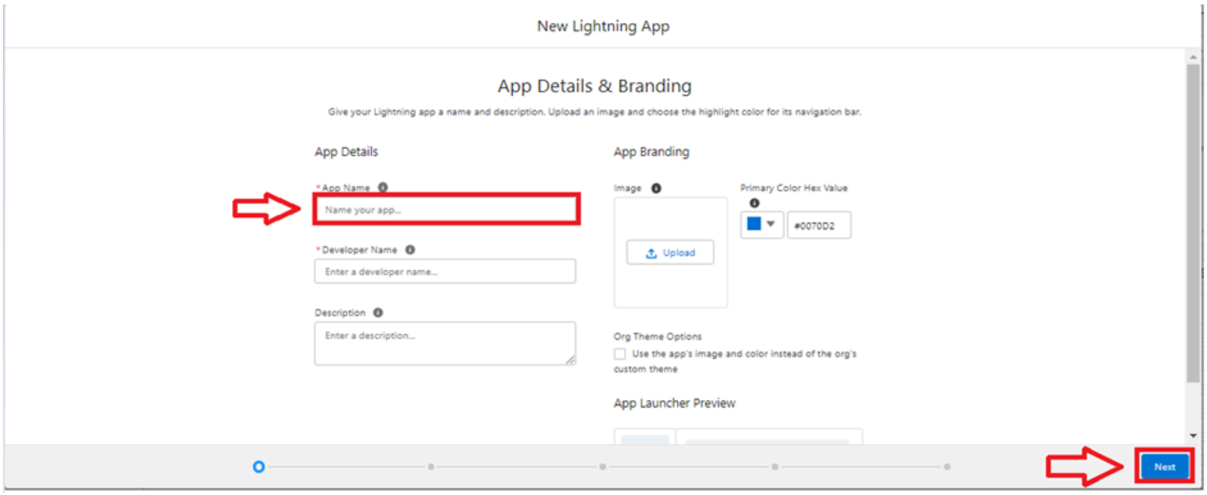
Activity 1: Create a Lightning App

To create a lightning app page:

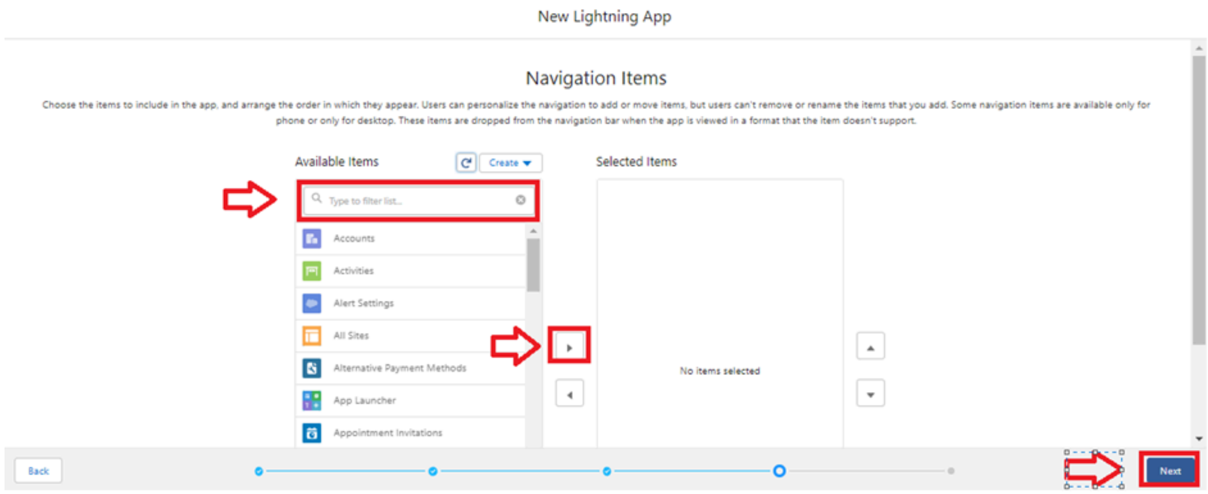
1. Go to setup page → search "app manager" in quick find → select "app manager" → click on New lightning App.



2. Fill the app name in app details as **Garage Management Application** →Next → (App option page) keep it as default → Next → (Utility Items) keep it as default → Next.

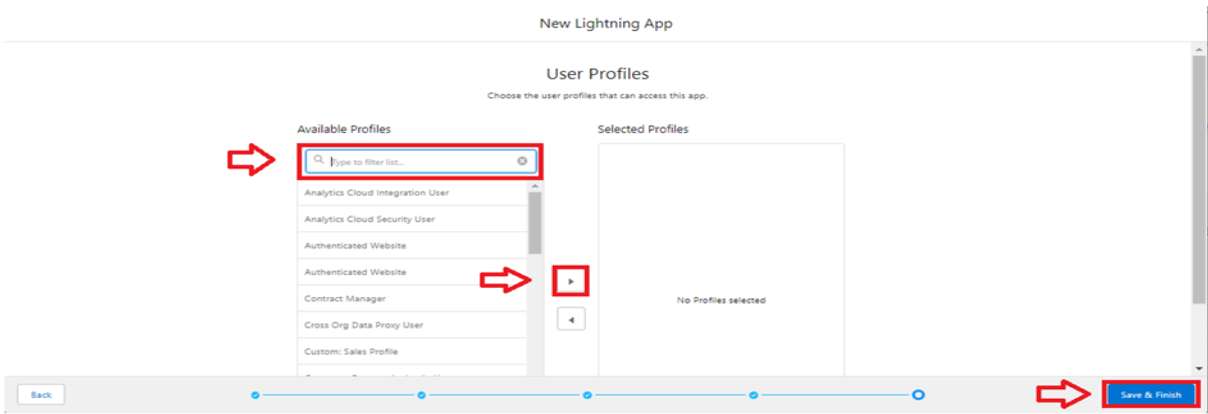


3. To Add Navigation Items:



Select the items (Customer Details,Appointments, Service records, Billing details and feedback, Reports and Dashboards) from the search bar and move it using the arrow button → Next.

4. To Add User Profiles:



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

Milestone 5 : 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.

Types of Fields

1. Standard Fields
2. Custom Fields

Standard Fields:

As the name suggests, the Standard Fields are the predefined fields in Salesforce that perform a standard task. The main point is that you can't simply delete a Standard Field until it is a non-required standard field. Otherwise, users have the option to delete them at any point from the application freely. Moreover, we have some fields that you will find common in every Salesforce application. They are,

- Created By
- Owner
- Last Modified
- Field Made During object Creation

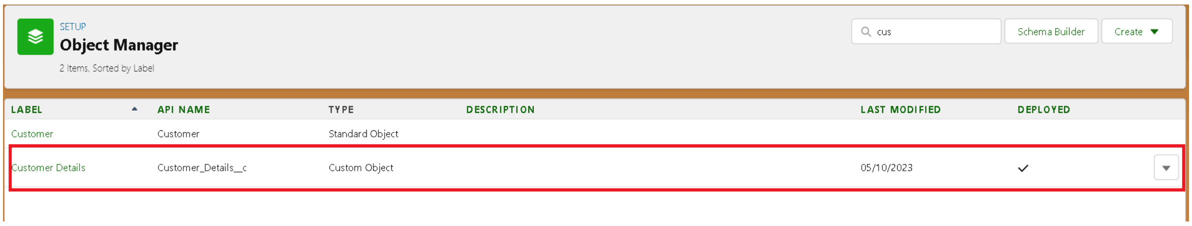
Custom Fields:

On the other side of the coin, Custom Fields are highly flexible, and users can change them according to requirements. Moreover, each organiser or company can use them if necessary. It means you need not always include them in the records, unlike Standard fields. Hence, the final decision depends on the user, and he can add/remove Custom Fields of any given form.

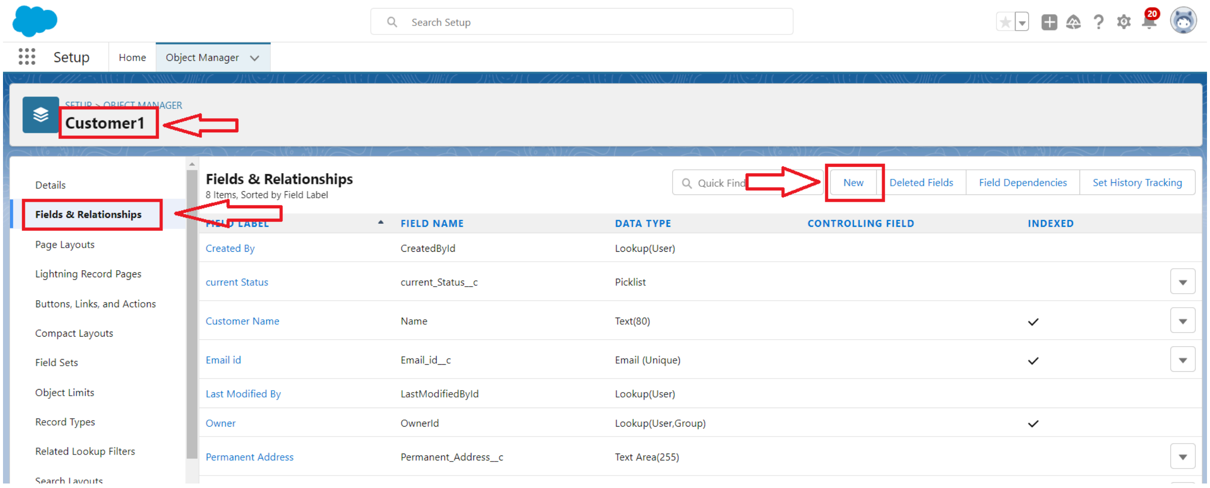
Activity 1: Creation of fields for the Customer Details object

1. To create fields in an object:

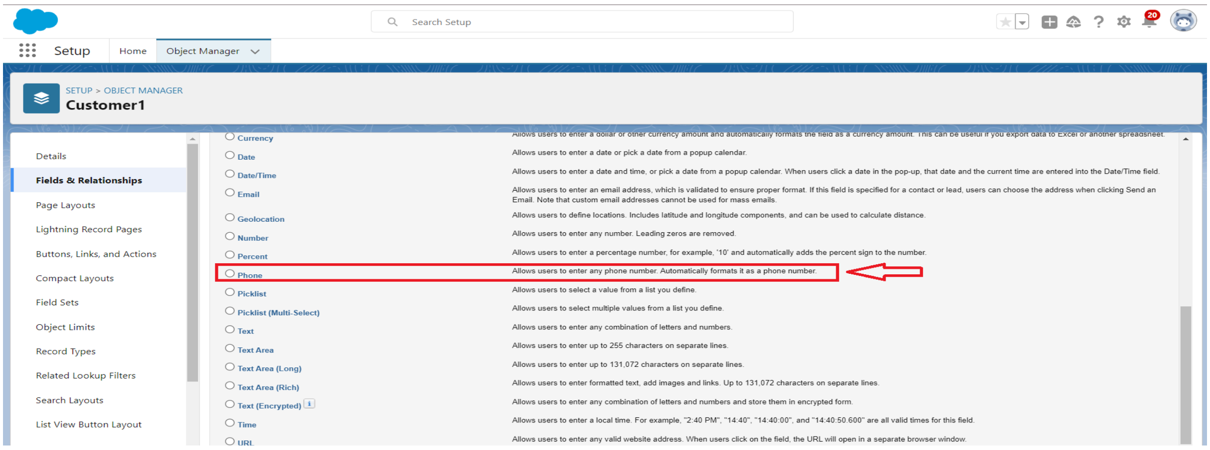
1. Go to setup → click on Object Manager → type object name(Customer Details) in search bar → click on the object.



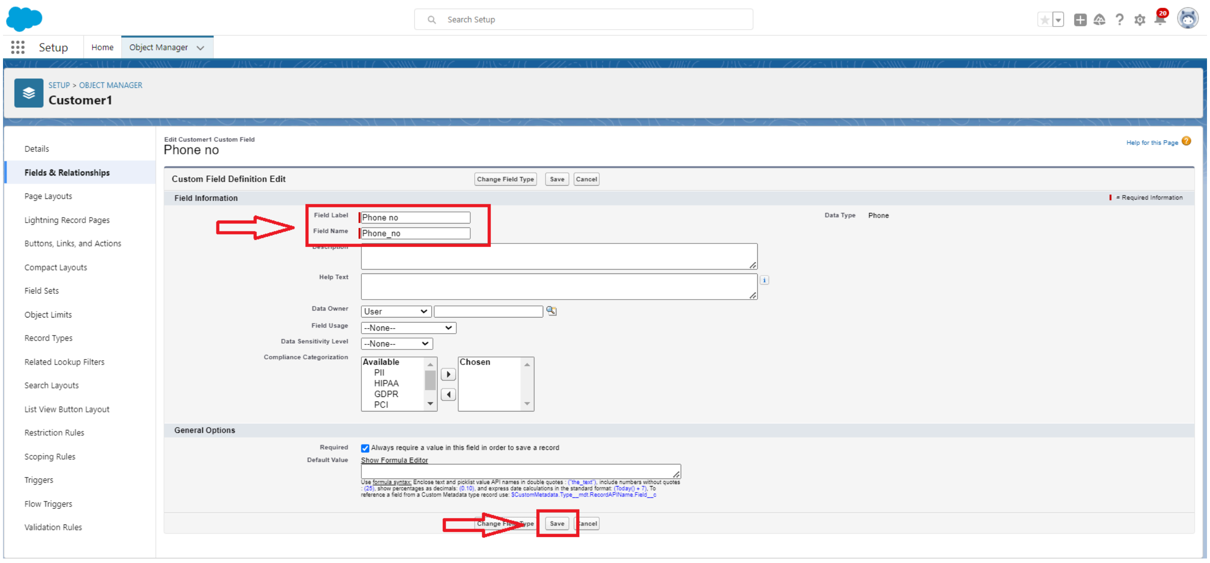
2. Now click on “Fields & Relationships” → New



3. Select Data Type as a “Phone”



4. Click on next.



5. Fill the Above as following:
    - Field Label: Phone number
    - Field Name : gets auto generated
    - Click on Next → Next → Save and new.

Note: Follow the above steps for the remaining field for the same object.

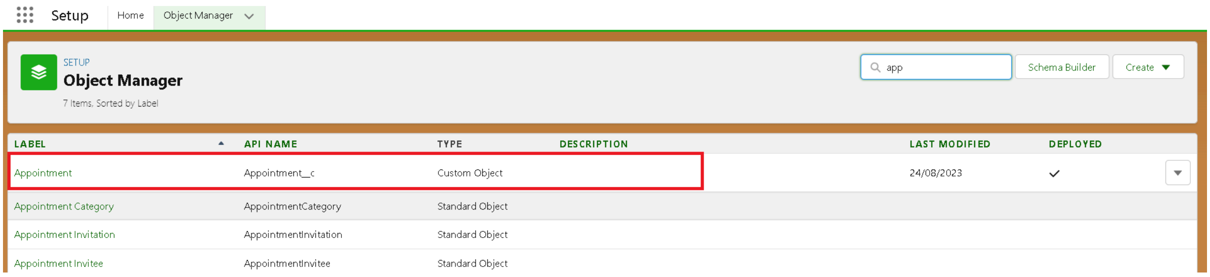
2. To create another fields in an object:

1. Go to setup → click on Object Manager → type object name(Customer Details) 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 : Gmail
    - Field Name : gets auto generated
    - Click on Next → Next → Save and new.

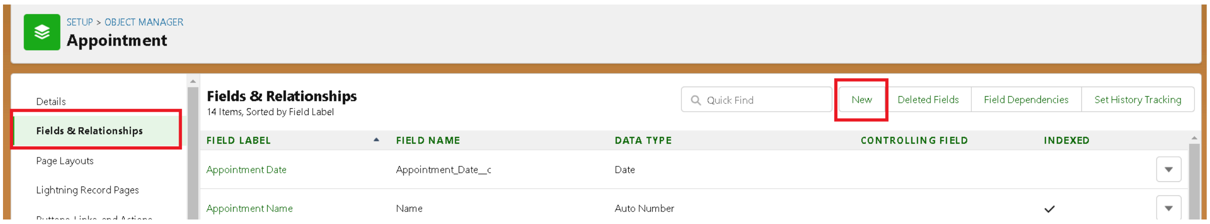
Activity 2 : Creation of Lookup Fields:

Creation of Lookup Field on Appointment Object :

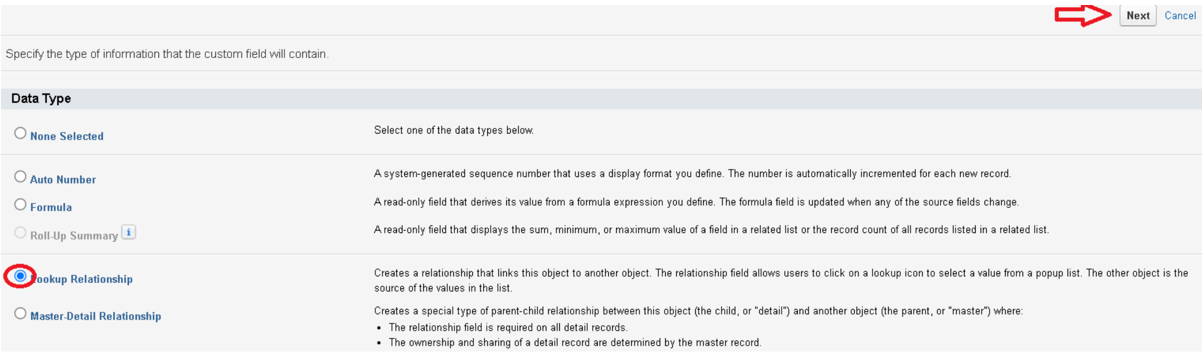
1. Go to setup → click on Object Manager → type object name(Appointment) in the search bar → click on the object.



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



3. Select “Look-up relationship” as data type and click Next.

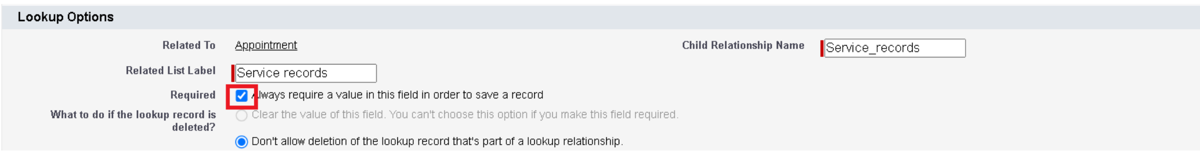


4. Select the related object “ Customer Details” and click next.
5. Next → Next → Save.

Note: Make sure you complete Activity 4 Before continuing.

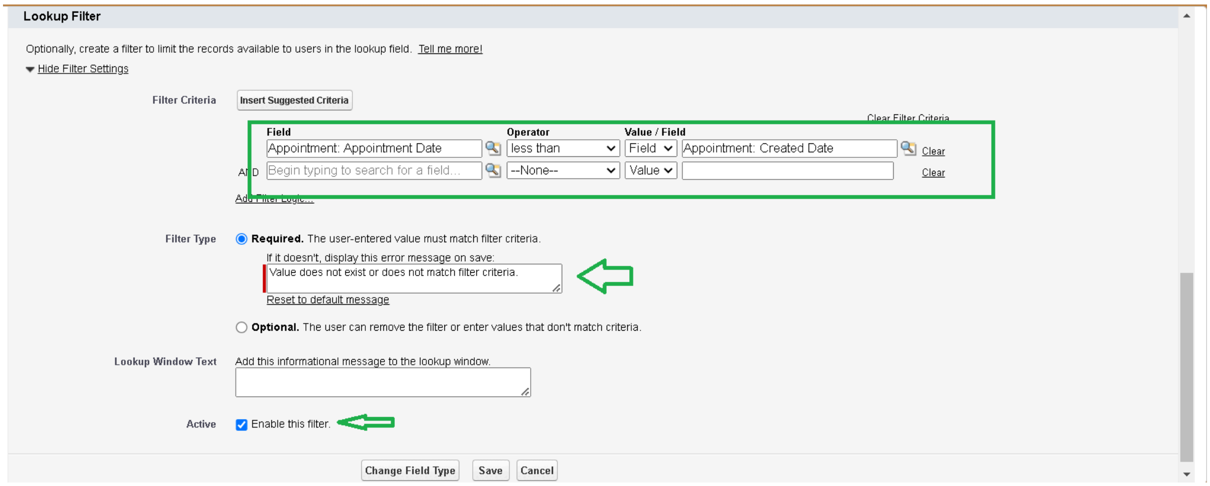
Creation of Lookup Field on Service records Object :

1. Go to setup → click on Object Manager → type object name(**Service records**) in search bar → click on the object.
2. Now click on “Fields & Relationships” → New
3. Select “Look-up relationship” as data type and click Next.
4. Select the related object “ Appointment ” and click next.
5. Make it a required field so click on Required.



6. Scroll down for Lookup Filter and click on Show filter settings.
7. Now add the filter criteria.
8. Field : Appointment: Appointment Date → Operator : less than →select field → Appointment: Created Date

9. Filter type should be Required.



10. Error Message : Value does not match the criteria.
11. Enable the filter by click on Active.
12. Next → Next → Save.

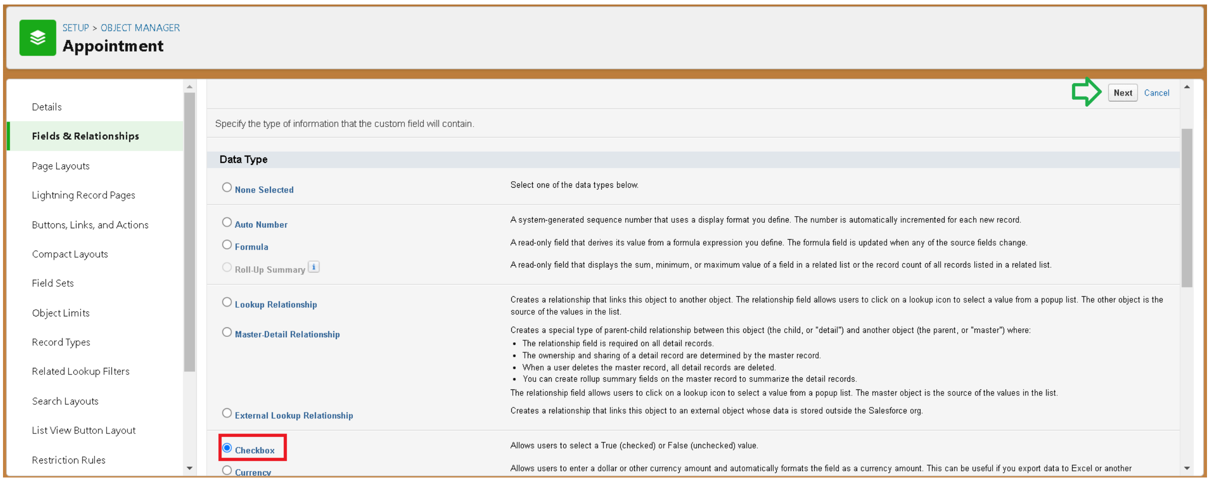
Creation of Lookup Field on **Billing details and feedback** Object :

1. Go to setup → click on Object Manager → type object name(**Billing details and feedback**) in search bar → click on the object.
2. Now click on "Fields & Relationships" → New
3. Select "Look-up relationship" as data type and click Next.
4. Select the related object " **Service records**" and click next.
5. Next → Next → Save&new.

Activity 3 : Creation of Checkbox Fields:

Creation of Checkbox Field on Appointment Object :

1. Go to setup → click on Object Manager → type object name(**Appointment**) in search bar → click on the object.
2. Now click on "Fields & Relationships" → New.
3. Select "Check box" as data type and click Next.



4. Give the Field Label : Maintenance service
5. Field Name : is auto populated
6. Default value : unchecked

7. Click on next → next → save.

Creation of Another Checkbox Fields on Appointment Object :

8. Repeat the steps form 1 to 3.
9. Give the Field Label : **Repairs**
10. Field Nme : is auto populated
11. Default value : unchecked
12. Click on next → next → save.

13. Follow the same and create another checkbox with given names
14. Give the Field Label : **Replacement Parts**
15. Field Nme : is auto populated
16. Default value : unchecked
17. Click on next → next → save.

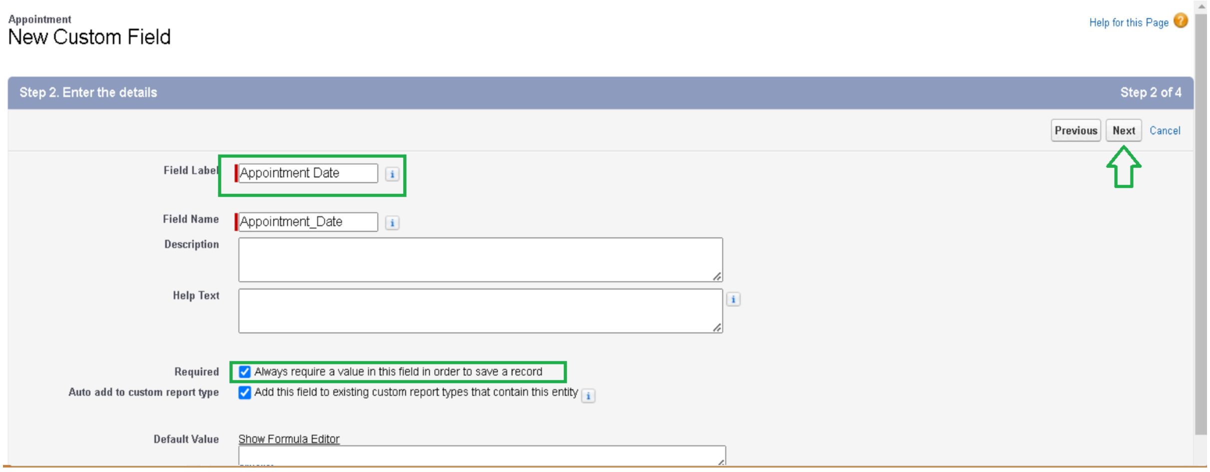
Creation of Checkbox Field on Service records Object :

1. Go to setup → click on Object Manager → type object name(**Service records**) in search bar → click on the object.
2. Now click on "Fields & Relationships" → New.
3. Select "Check box" as data type and click Next.
4. Give the Field Label : Quality Check Status
5. Field Nme : is auto populated
6. Default value : unchecked
7. Click on next → next → save

Activity 4 : Creation of date Fields:

Creation of Date Field on Appointment Object :

1. Go to setup → click on Object Manager → type object name(Appointment) in the search bar → click on the object.
2. Now click on "Fields & Relationships" → New.
3. Select "Date" as data type and click Next.
4. Give the Field Label : Appointment Date
5. Field Nme : is auto populated
6. Make it as a Required field by clicking on the Required option.
7. Click on next → next → save.

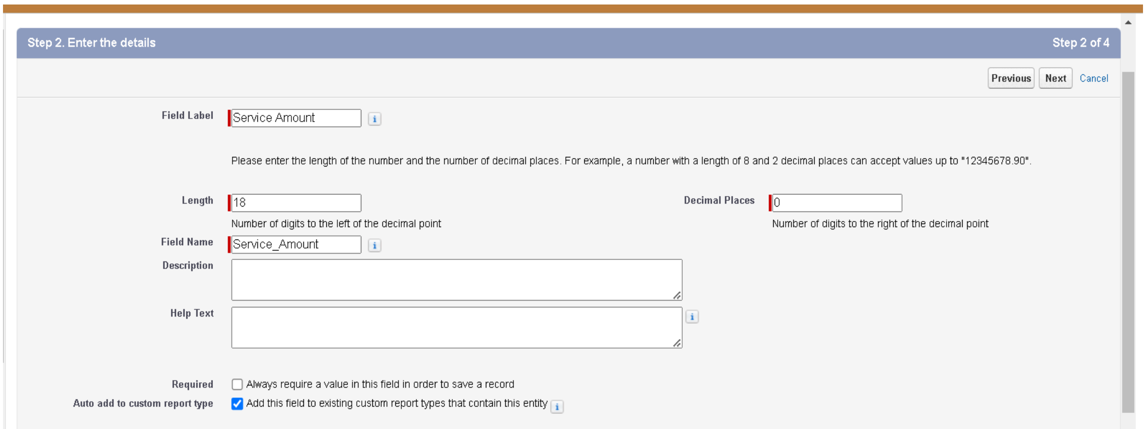


Activity 5 : Creation of Currency Fields:

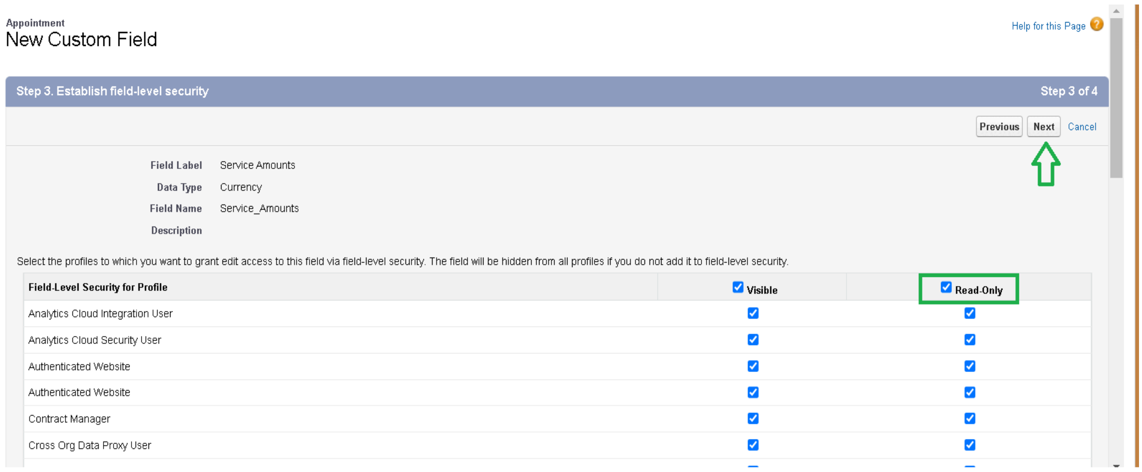
Creation of Currency Field on Appointment Object :

1. Go to setup → click on Object Manager → type object name(**Appointment**) in the search bar → click on the object.

2. Now click on “Fields & Relationships” → New.
3. Select “Currency” as data type and click Next.
4. Give the Field Label : Service Amount
5. Field Nme : is auto populated



6. Click on next
7. Give read only for all the profiles in field level security for profile.



8. Click on next → save.

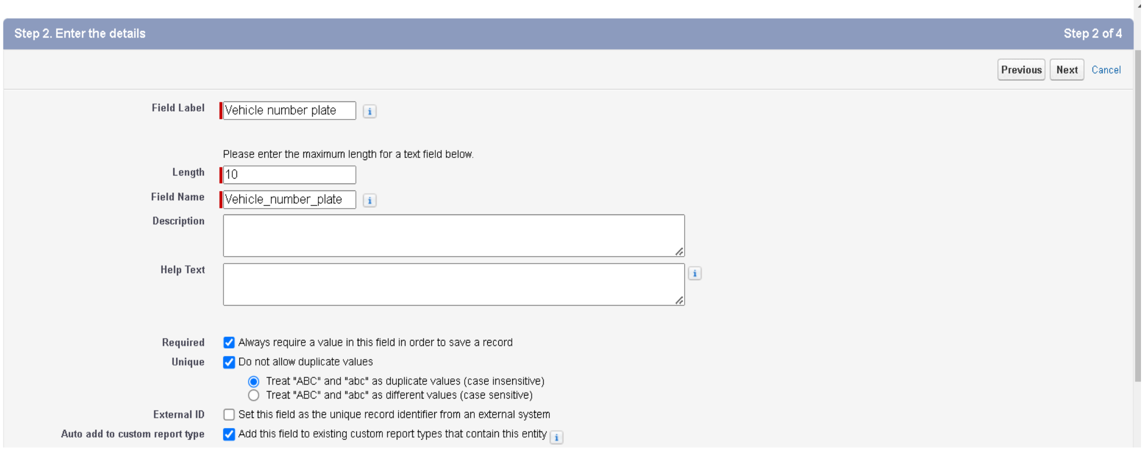
Creation of Currency Field on Billing details and feedback Object :

1. Follow the same steps as mentioned above in **Billing details and feedback Object**.
2. Change the label name as mentioned.
3. Give the Field Label : Payment Paid
4. Field Nme : is auto populated

Activity 6 : Creation of Text Fields:

1. Go to setup → click on Object Manager → type object name(**Appointment**) in the search bar → click on the object.
2. Now click on “Fields & Relationships” → New.
3. Select “Text” as data type and click Next.

4. Give the Field Label : Vehicle number plate
5. Field Name : is auto populated
6. Length : 10
7. Make the field as Required and Unique.



8. Click on next → next → save.

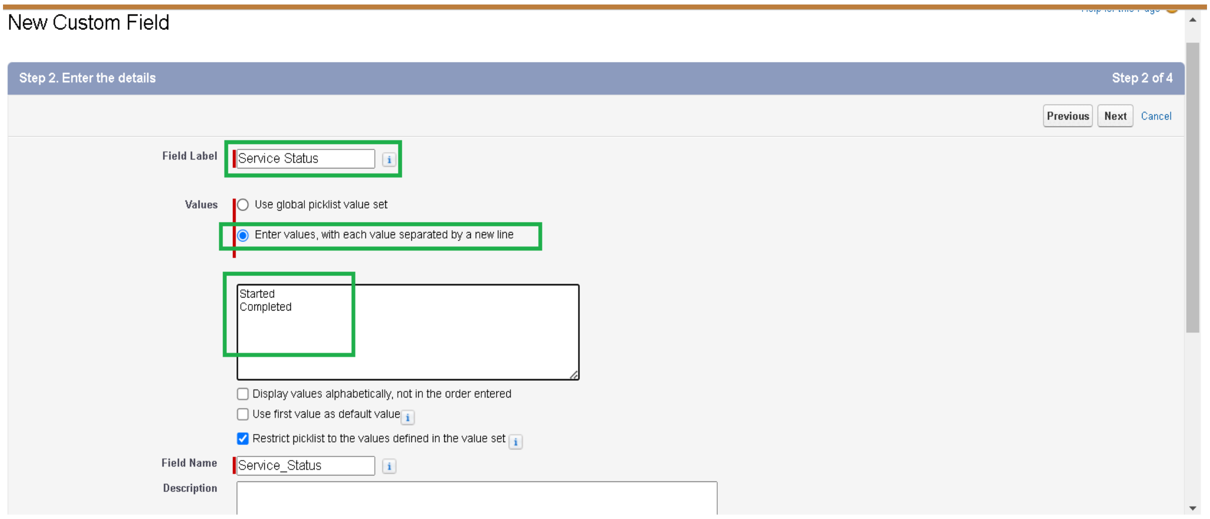
Creation of Text Fields in **Billing details and feedback object :**

1. Go to setup → click on Object Manager → type object name(**Billing details and feedback**) in search bar → click on the object.
2. Now click on "Fields & Relationships" → New.
3. Select "text" as data type and click Next.
4. Give the Field Label : Rating for service
5. Field Name : is auto populated
6. Length : 1
7. Make field as Required .
8. Click on next → next → save

Activity 7 : Creation of Picklist Fields:

Creation of Picklist Fields in **Service records object :**

1. Go to setup → click on Object Manager → type object name(**Service records**) in search bar → click on the object.
2. Click on fields & relationship → click on New.
3. Select Data type as "Picklist" and click Next.
4. Enter Field Label as "Service Status", under values select "Enter values, with each value separated by a new line" and enter values as shown below.
5. The values are: Started, Completed.



6. Click Next.
7. Next → Next → Save.

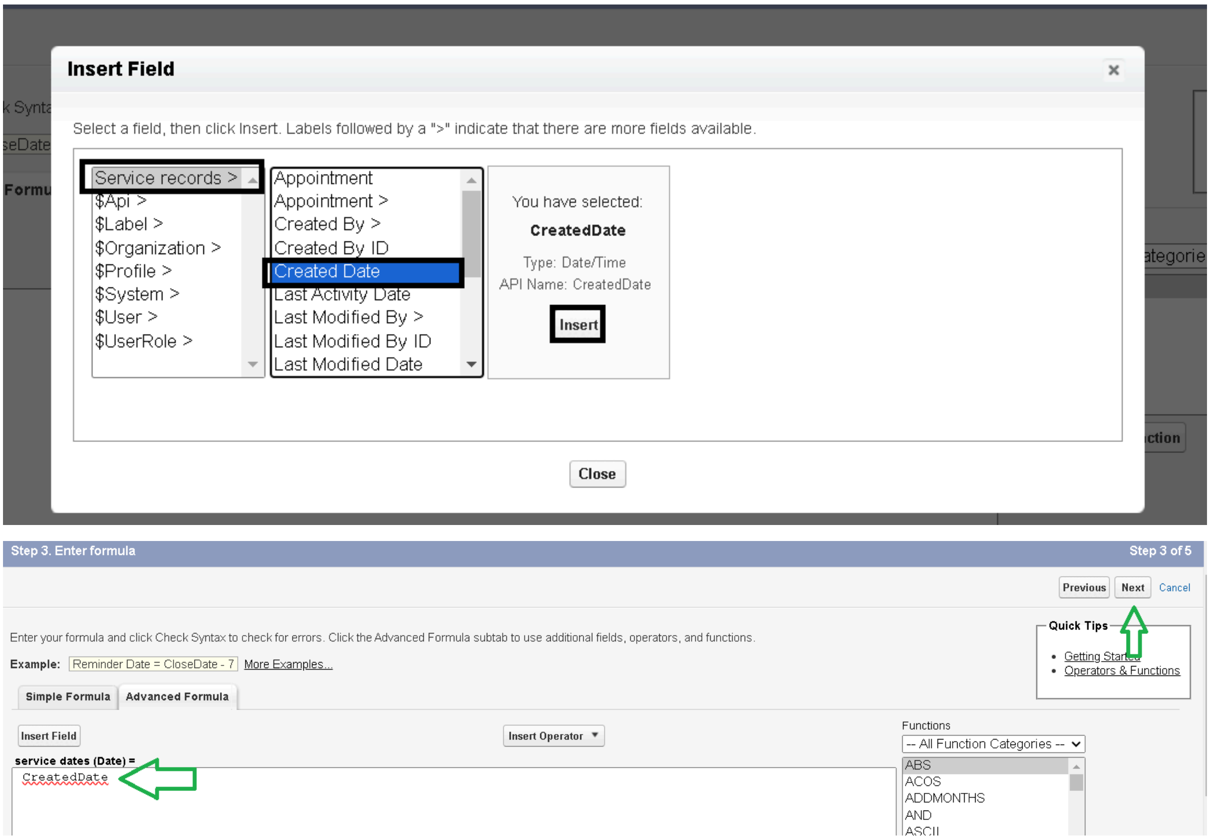
Creation of Picklist Fields in Billing details and feedback object :

1. Go to setup → click on Object Manager → type object name(**Billing details and feedback**) in search bar → click on the object.
2. Click on fields & relationship → click on New.
3. Select Data type as "Picklist" and click Next.
4. Enter Field Label as "Payment Status", under values select "Enter values, with each value separated by a new line" and enter values as shown below.
5. The values are: Pending, Completed.
6. Click Next.
7. Next → Next → Save.

Activity 8 : Creating Formula Field in Service records Object

1. Go to setup → click on Object Manager → type object name(Service records) in search bar → click on the object.
2. Click on fields & relationship → click on New.
3. Select Data type as "Formula" and click Next.
4. Give Field Label and Field Name as "service date" and select formula return type as "Date" and click next.

5. Insert field formula should be : CreatedDate



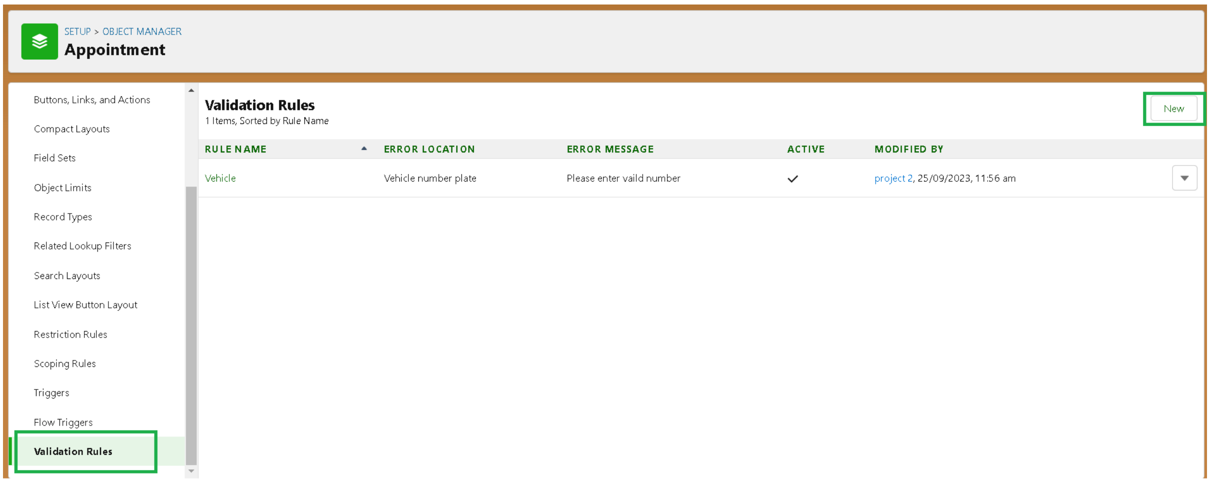
6. click “Check Syntax” .
7. Click next → next → Save.

Milestone 6: Validation rule

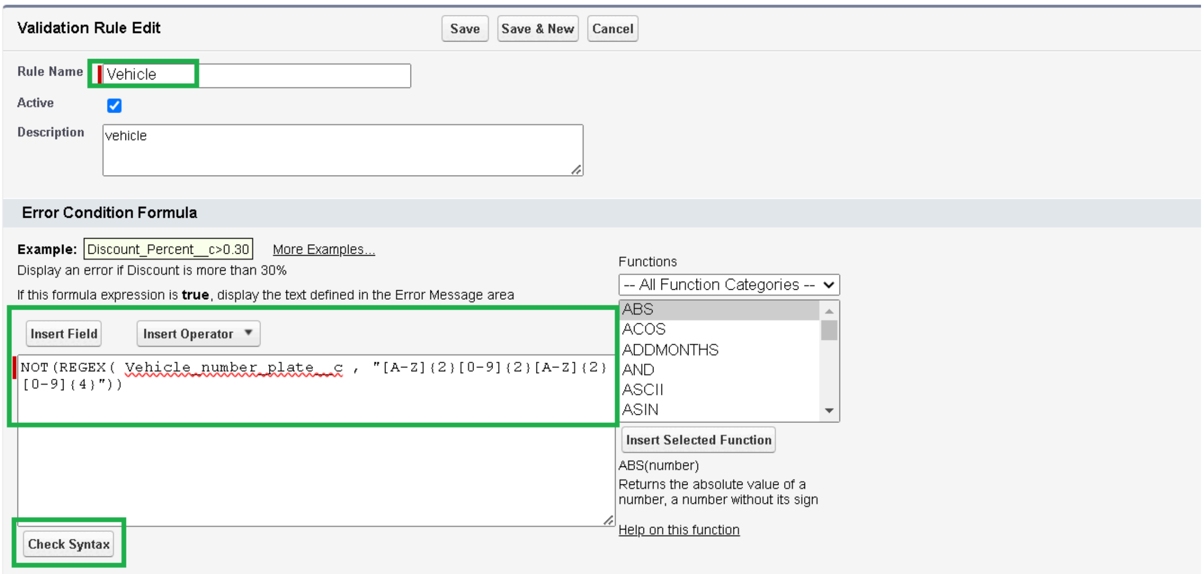
Validation rules are applied when a user tries to save a record and are used to check if the data meets specified criteria. If the criteria are not met, the validation rule triggers an error message and prevents the user from saving the record until the issues are resolved.

Activity 1: To create a validation rule to an Appointment Object

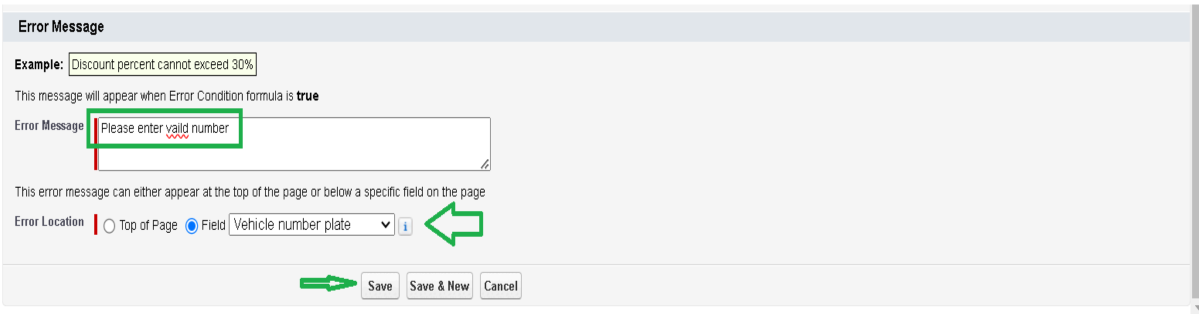
1. Go to the setup page → click on object manager → From drop down click edit for **Appointment** object.
2. Click on the validation rule → click New.



3. Enter the Rule name as " Vehicle ".
4. Insert the Error Condition Formula as : -
   NOT(REGEX(Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))

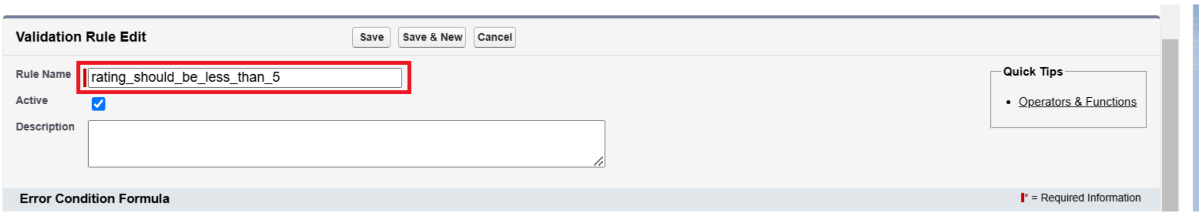


5. Enter the Error Message as "Please enter vaild number ", select the Error location as Field and select the field as "Vehicle number plate", and click Save.

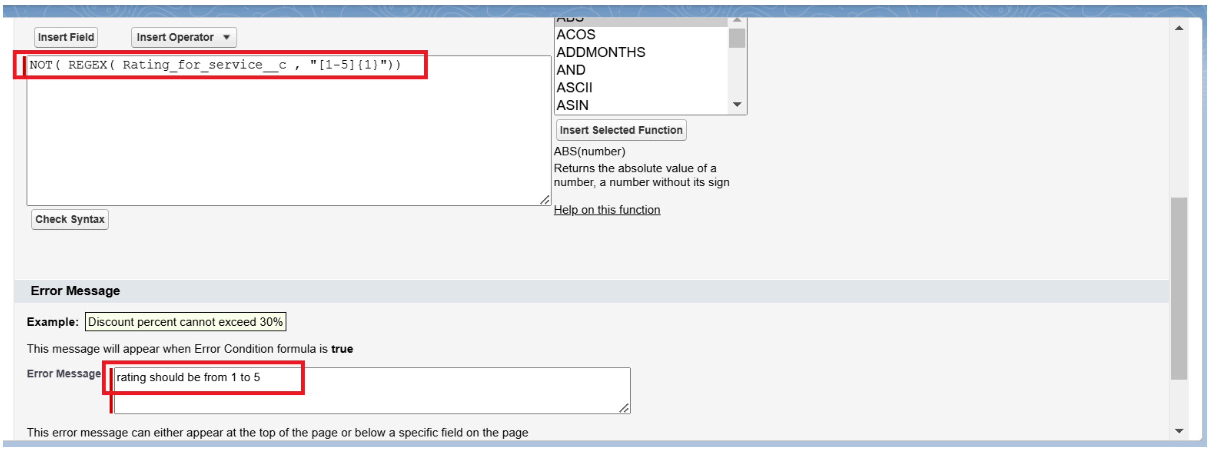


Activity 2: To create a validation rule to an Billing details and feedback Object

1. Go to the setup page → click on object manager → From drop down click edit for **Billing details and feedback** object.
2. Click on the validation rule → click New.
3. Enter the Rule name as " rating_should_be_less_than_5".
4. Insert the Error Condition Formula as : -
   NOT(REGEX(Rating_for_service__c , "[1-5]{1}"))



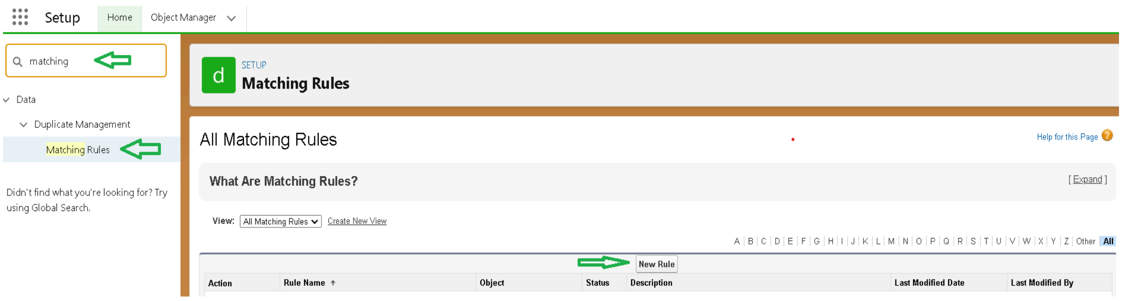
5. Enter the Error Message as "rating should be from 1 to 5", select the Error location as Field and select the field as "Rating for Service", and click Save.



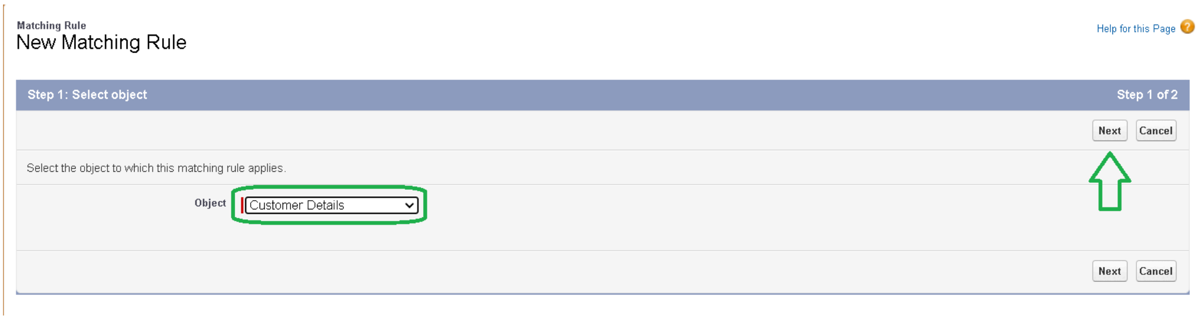
Milestone 7: Duplicate rule

Activity 1: To create a matching rule to an Customer details Object

1. Go to quick find box in setup and search for **matching Rule.**
2. Click on matching rule → click on New Rule.



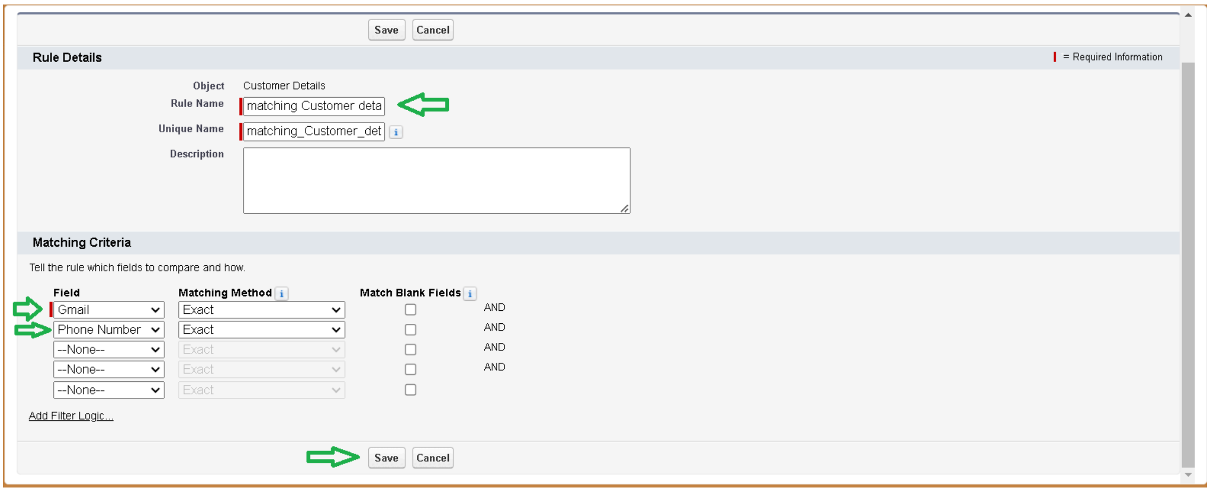
3. Select the object as Customer details and click Next.

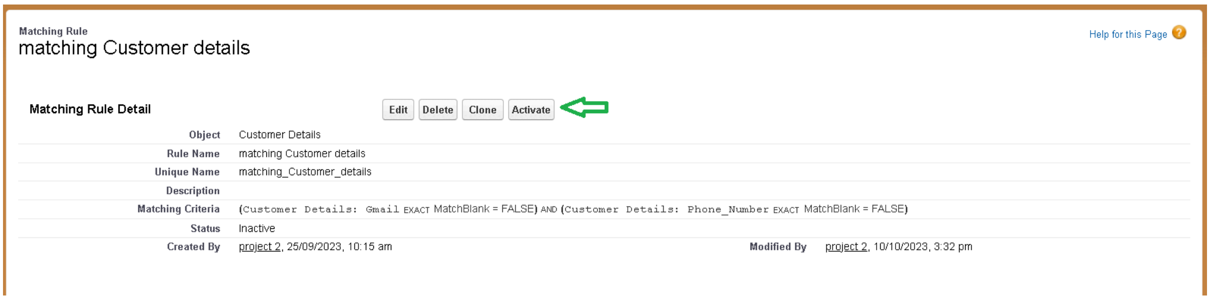


4. Give the Rule name : Matching customer details
5. Unique name : is auto populated
6. Define the matching criteria as
7.

Field	Matching Method
1. Gmail	Exact
2. Phone Number	Exact

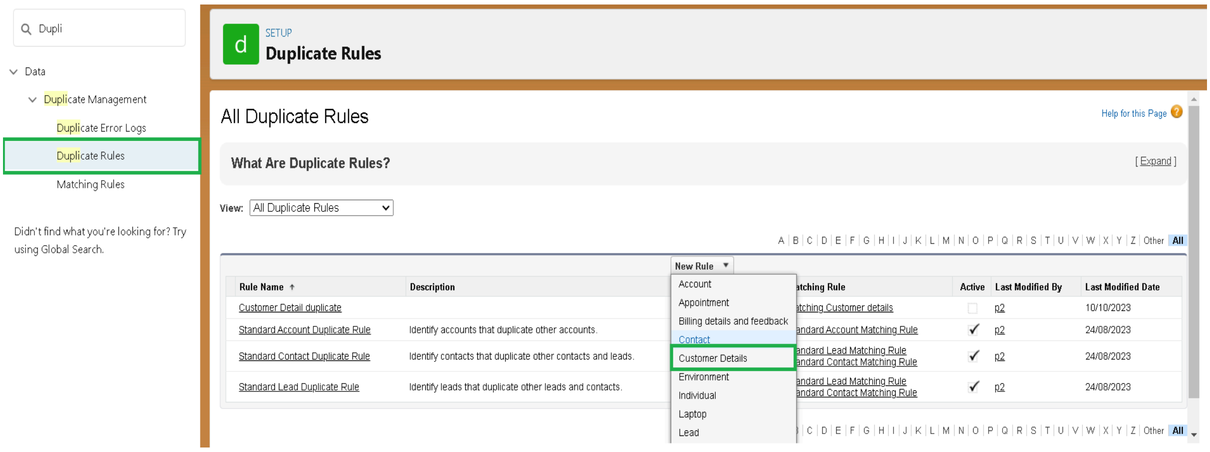
8. Click save.
9. After Saving Click on Activate.



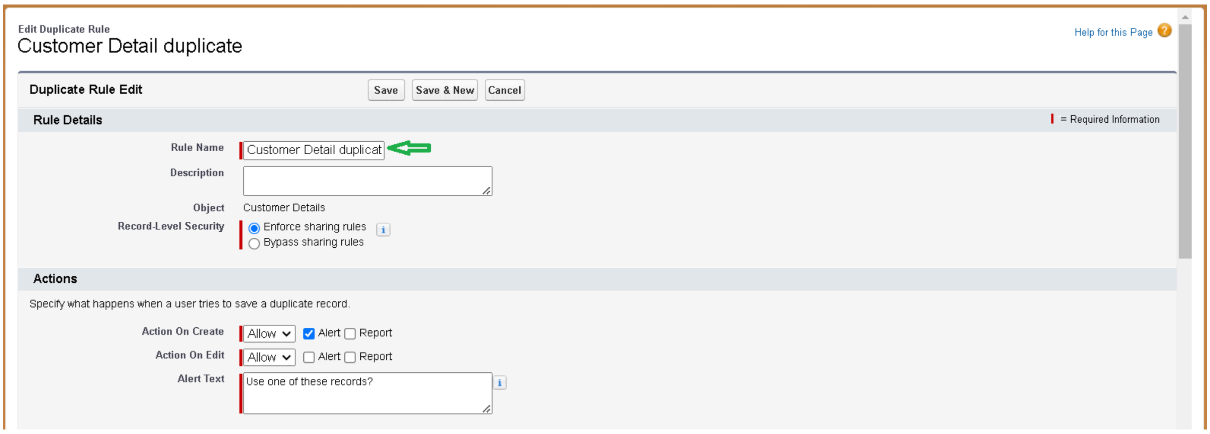


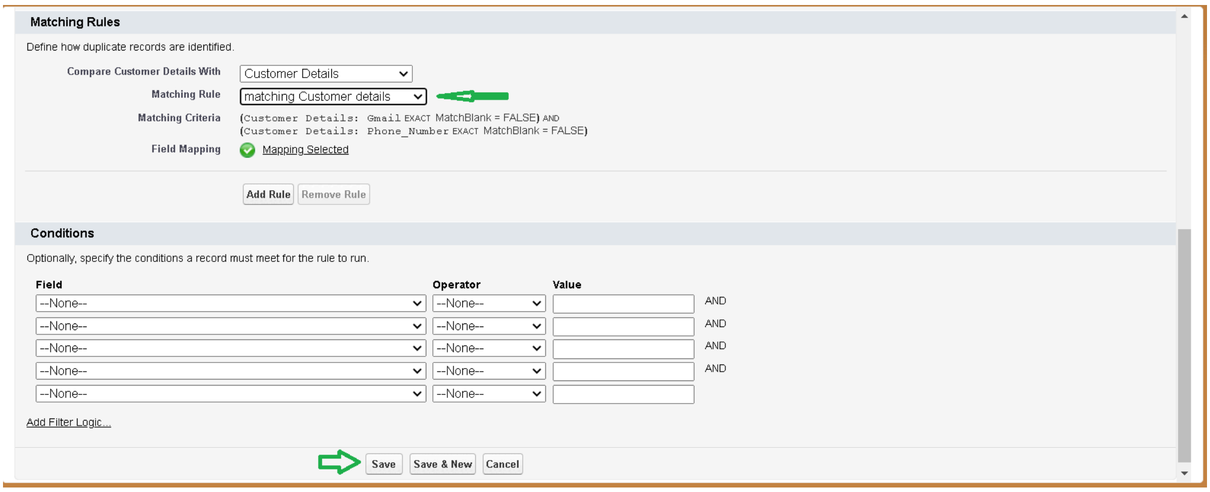
Activity 2: To create a Duplicate rule to an Customer details Object

1. Go to quick find box in setup and search for **Duplicate rules.**
2. Click on Duplicate rule → click on New Rule → select customer details object.



3. Give the Rule name as : Customer Detail duplicate
4. Scroll a little in Matching rule section
5. Select the matching rule : Matching customer details
6. And Click on save.
7. After saving the Duplicate Rule, Click on Activate.





Milestone 8 : Profiles

A profile is a group/collection of settings and permissions that define what a user can do in salesforce. Profile controls "Object permissions, Field permissions, User permissions, Tab settings, App settings, Apex class access, Visualforce page access, Page layouts, Record Types, Login hours & Login IP ranges. You can define profiles by the user's job function. For example System Administrator, Developer, Sales Representative.

Types of profiles in salesforce

1. **Standard profiles:**
   By default salesforce provides below standard profiles.
   - Contract Manager
   - Read Only
   - Marketing User
   - Solutions Manager
   - Standard User
   - System Administrator.

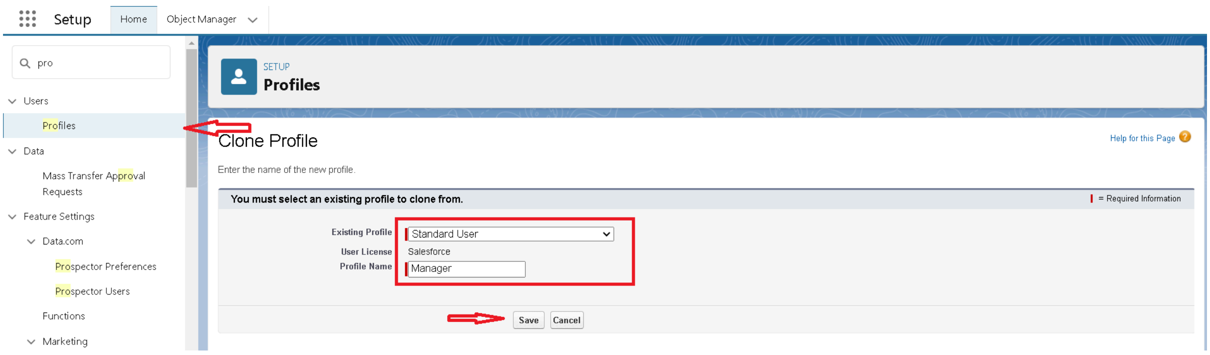
   We cannot deleted standard ones
   Each of these standard ones includes a default set of permissions for all of the standard objects available on the platform.

2. **Custom Profiles:**
   Custom ones defined by us.
   They can be deleted if there are no users assigned with that particular one.

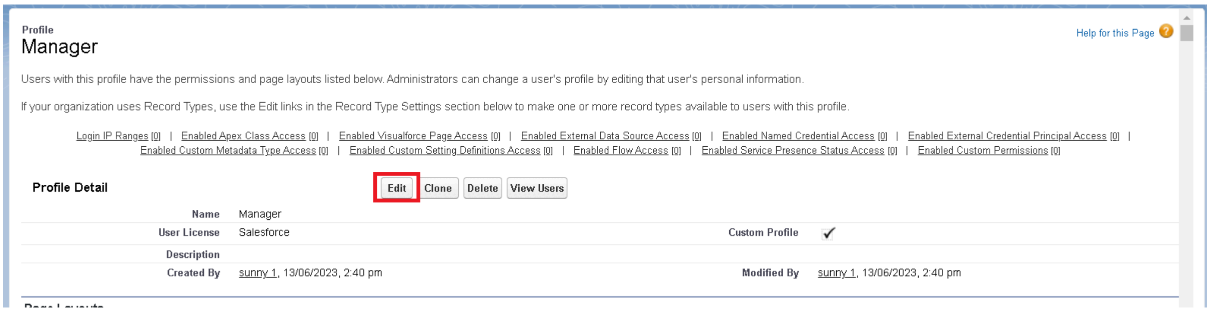
Activity 1: Manager Profile

To create a new profile:

1. Go to setup → type profiles in quick find box → click on profiles → clone the desired profile (Standard User) → enter profile name (Manager) → Save.

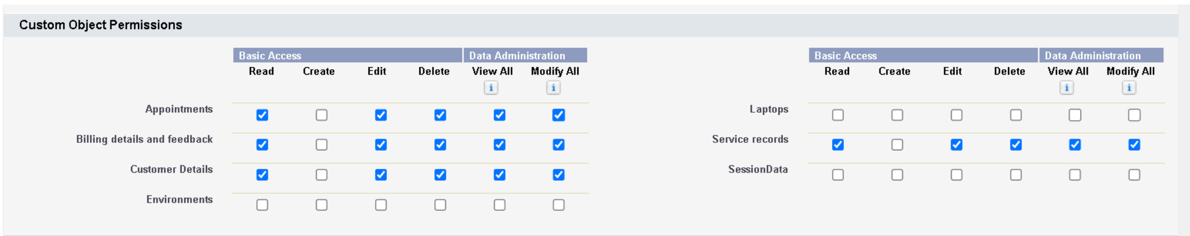


2. While still on the profile page, then click Edit.



3. Select the Custom App settings as default for the Garage management.

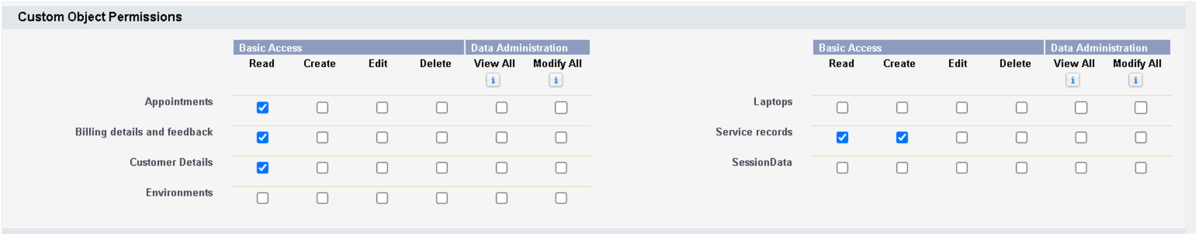
4. Scroll down to Custom Object Permissions and Give access permissions for Appointments,Billing details and feedback , service records and customer details objects as mentioned in the below diagram.



5. Changing the session times out after should be “ 8 hours of inactivity”.
6. Change the password policies as mentioned :
7. User passwords expire in should be “ never expires ”.
8. Minimum password length should be “ 8 ”, and click save.

Activity 2: sales person Profile

1. Go to setup → type profiles in quick find box → click on profiles → clone the desired profile (Salesforce Platform User) → enter profile name (sales person) → Save.
2. While still on the profile page, then click Edit.
3. Select the Custom App settings as default for the GArage management.
4. Scroll down to Custom Object Permissions and Give access permissions for Appointments,Billing details and feedback , service records and customer details objects as mentioned in the below diagram.



5. And click save.

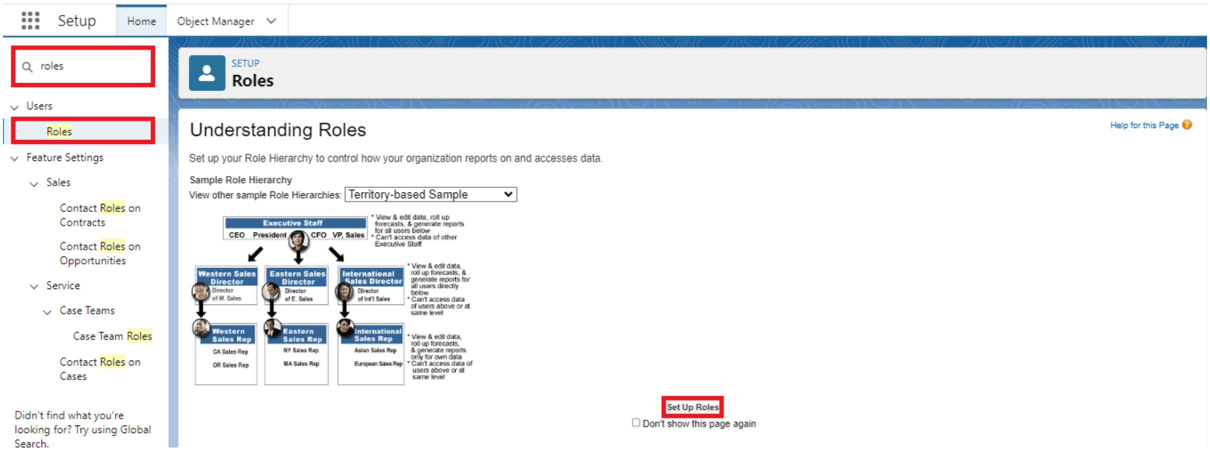
Milestone 9 : Role & Role Hierarchy

A role in Salesforce defines a user's visibility access at the record level. Roles may be used to specify the types of access that people in your Salesforce organization can have to data. Simply put, it describes what a user could see within the Salesforce organization.

Activity 1: Creating Manager Role

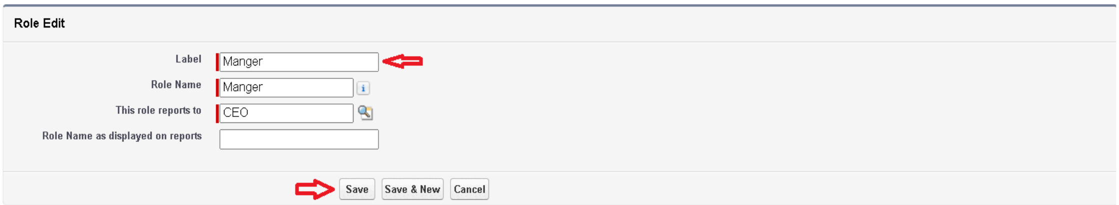
Creating Manager Role:

1. Go to quick find → Search for Roles → click on set up roles.



2. Click on Expand All and click on add role under whom this role works.

3. Give Label as “Manager” and Role name gets auto populated. Then click on Save.



Activity 2: Creating another roles

Creating another two roles under manager

1. Go to quick find → Search for Roles → click on set up roles.
2. Click plus on CEO role, and click add role under manager.

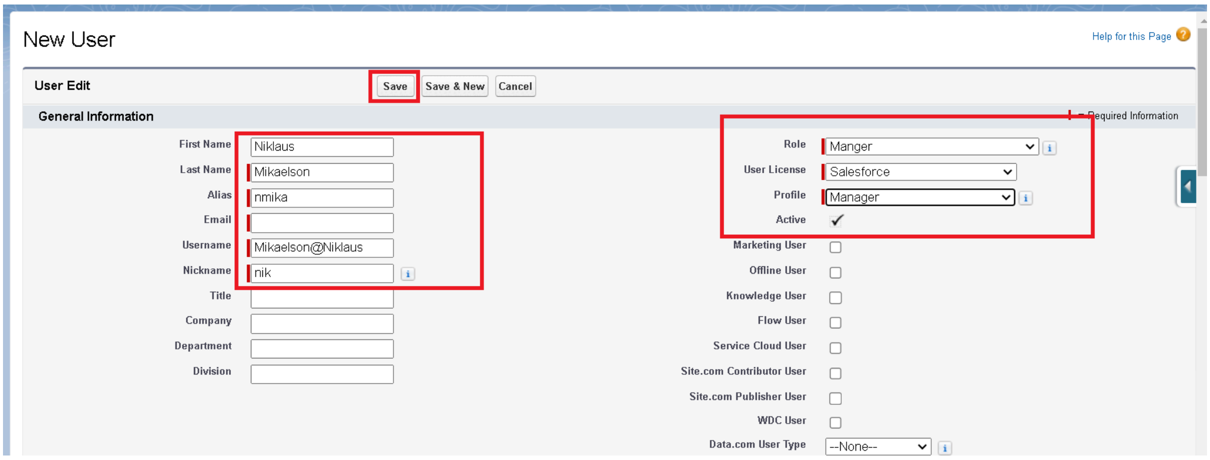
3. Give Label as “sales person” and Role name gets auto populated. Then click on Save.

Milestone 10 : Users

A user is anyone who logs in to Salesforce. Users are employees at your company, such as sales reps, managers, and IT specialists, who need access to the company's records. Every user in Salesforce has a user account. The user account identifies the user, and the user account settings determine what features and records the user can access.

Activity 1: Create User

1. Go to setup → type users in quick find box → select users → click New user.
2. Fill in the fields
    1. First Name : Niklaus
    2. Last Name : Mikaelson
    3. Alias : Give a Alias Name
    4. Email id : Give your Personal Email id
    5. Username : Username should be in this form: text@text.text
    6. Nick Name : Give a Nickname
    7. Role : Manager
    8. User licence : Salesforce
    9. Profiles : Manager



3. Save.

Activity 2: creating another users

1. Repeat the steps and create another user using
    a. Role : sales person
    b. User licence : Salesforce Platform
    c. Profile : sales person

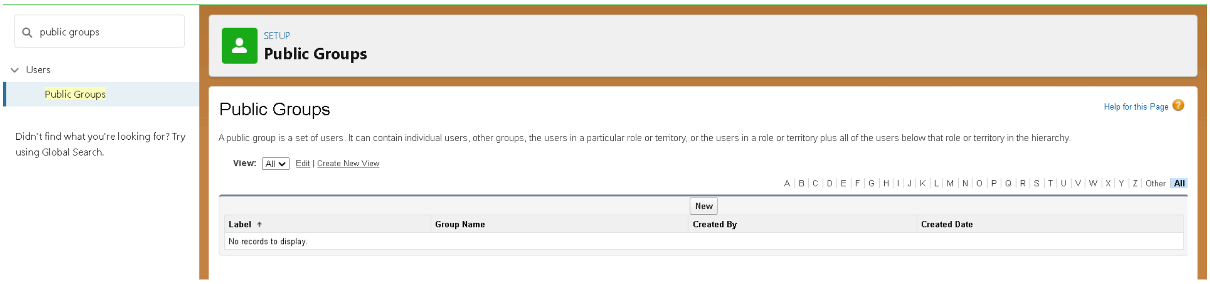
Note : create atleast 3 users with these permissions.

Milestone 11 : Public groups

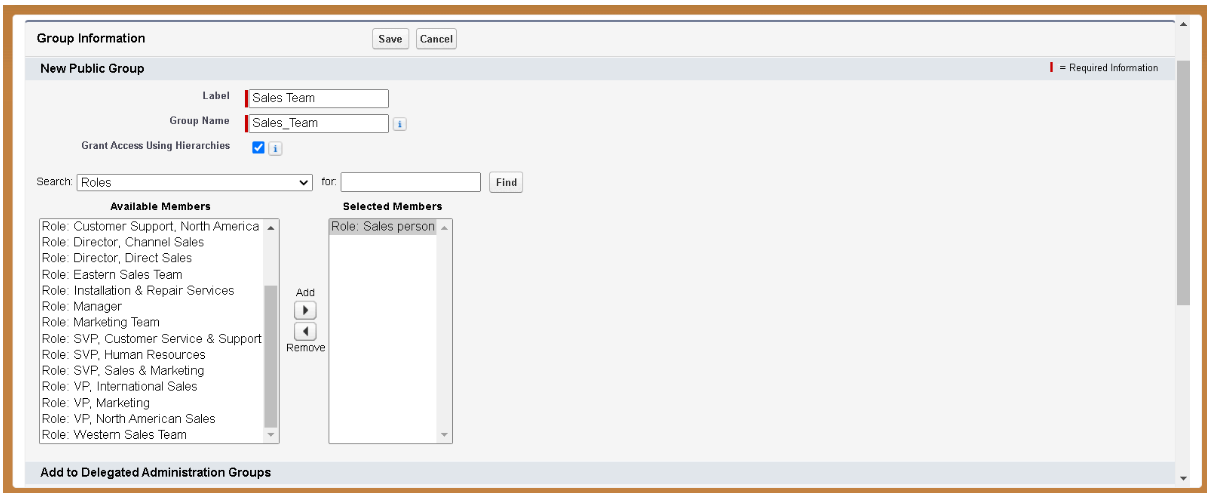
Public groups are a valuable tool for Salesforce administrators and developers to streamline user management, data access, and security settings. By creating and using public groups effectively, you can maintain a secure and organized Salesforce environment while ensuring that users have appropriate access to the resources they need.

Activity 1: Creating New Public Group

1. Go to setup → type users in quick find box → select public groups→ click New.



2. Give the Label as “sales team”.
3. Group name is autopopulated.
4. Search for Roles.
5. In Available Members select Sales person and click on add it will be moved to selected member.
6. Click on save.



Milestone 12: Sharing Setting

Salesforce allows you to configure sharing settings to control how records are accessed and shared within your organization. These settings are crucial for maintaining data security and privacy. Salesforce provides a variety of tools and mechanisms to define and enforce sharing rules, such as:

Organization-Wide Default (OWD) Settings:

These settings define the default level of access for all objects within your Salesforce org.

OWD settings include Private, Public Read-Only, Public Read/Write, and Controlled by Parent.
OWD settings can be configured for each standard and custom object.

Role Hierarchy:

Salesforce uses a role hierarchy to determine record access.
Users at higher levels in the hierarchy have greater access to records owned by or shared with users lower in the hierarchy.
The role hierarchy is often used in combination with OWD settings to grant different levels of access.

Profiles and Permission Sets:

Profiles and permission sets allow administrators to specify object-level and field-level permissions for users.
Profiles are typically used to grant general object and field access, while permission sets can be used to extend those permissions to specific users.

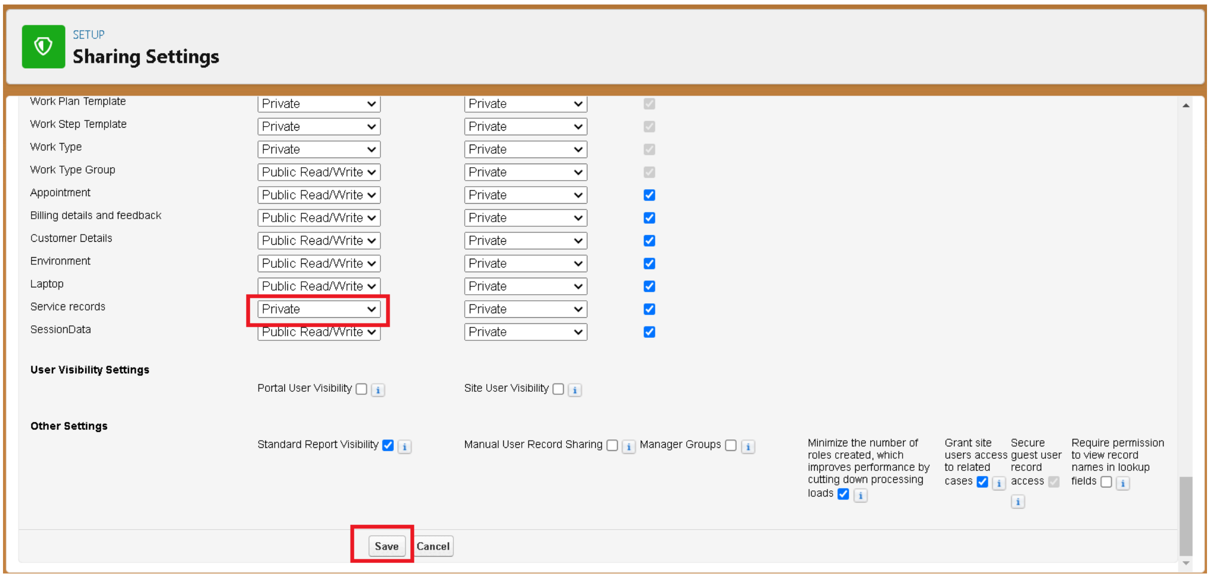
Sharing Rules:

Sharing rules are used to extend access to records for users who meet specific criteria.
They can be used to grant read-only or read-write access to records owned by other users.
Manual Sharing:

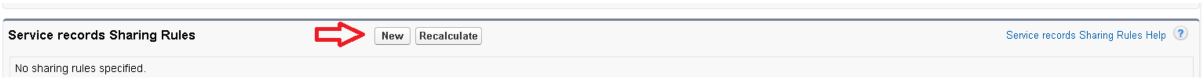
Administrators and record owners can manually share specific records with other users or groups.

Activity 1: Creating Sharing settings

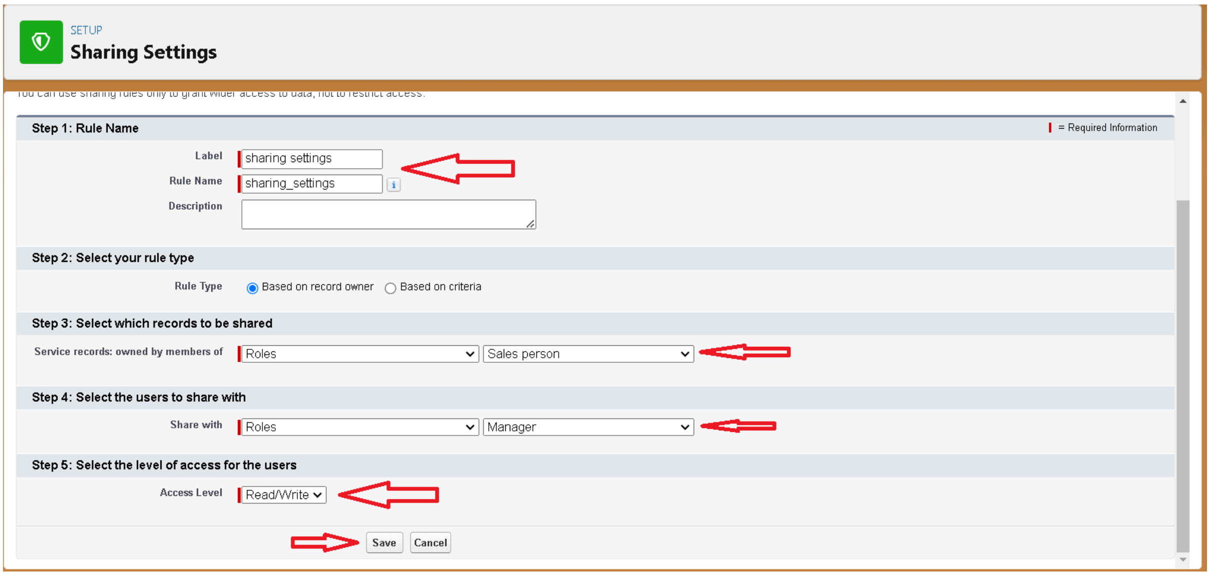
1. Go to setup → type users in quick find box → select Sharing Settings→ click Edit.
2. Change the OWD setting of the Service records Object to private as shown in fig.



3. Click on save and refresh.
4. Scroll down a bit, Click new on Service records sharing Rules.
5.



6. Give the Label name as “ Sharing setting”
7. Rule name is auto populated.
8. In step 3 : Select which records to be shared, members of “ Roles ” >> “ Sales person”
9. In step 4: share with, select “ Roles ” >> “ Manager ”
10. In step 5 : Change the access level to “ Read / write ”.
11. Click on save.

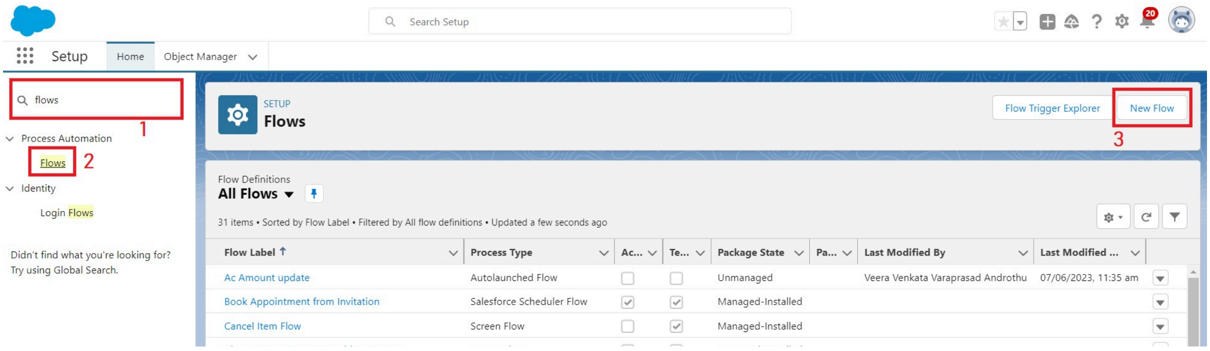


Milestone 13: Flows

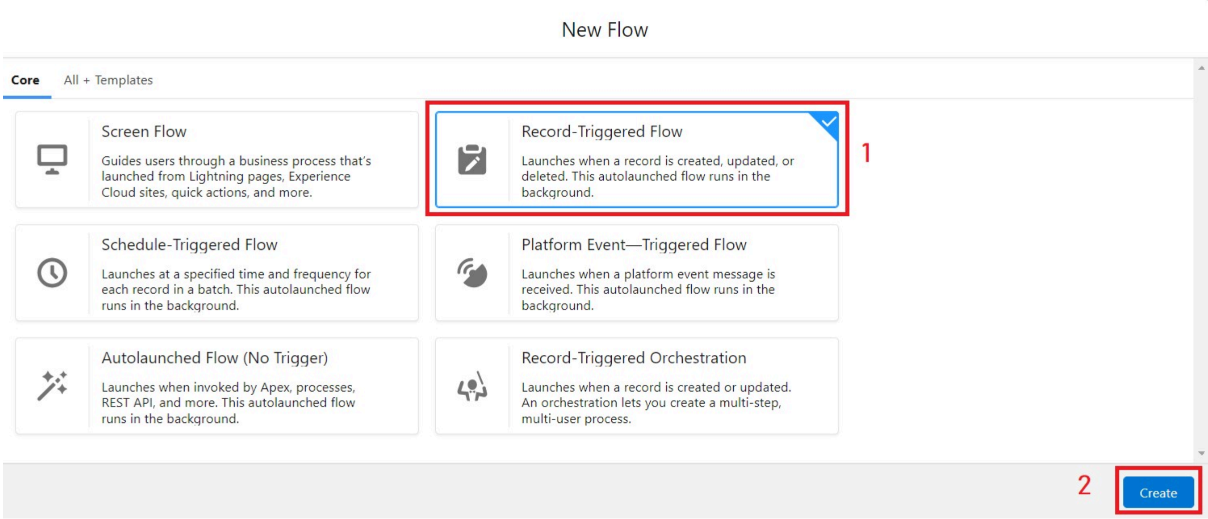
In Salesforce, a flow is a powerful tool that allows you to automate business processes, collect and update data, and guide users through a series of screens or steps. Flows are built using a visual interface and can be created without any coding knowledge.

Activity 1: Create a Flow

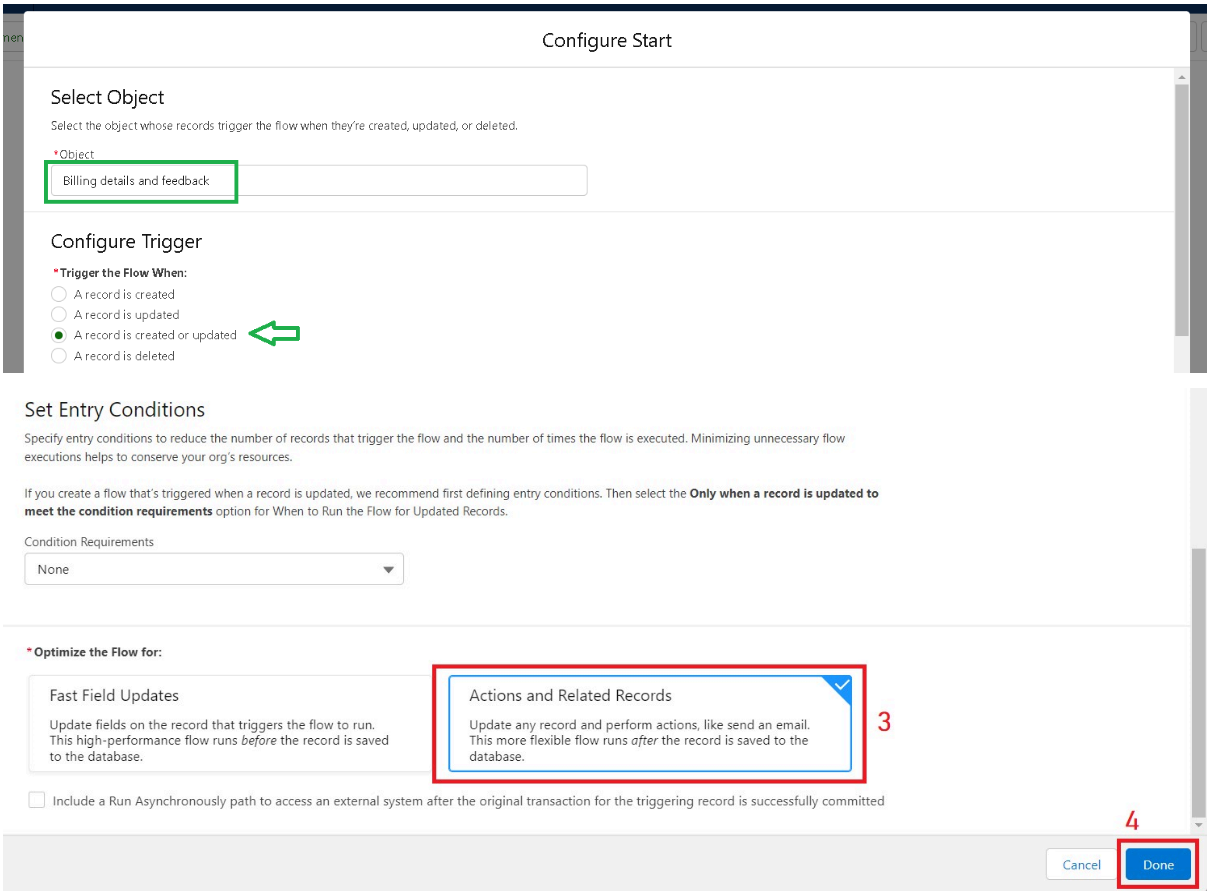
1. Go to setup → type Flow in quick find box → Click on the Flow and Select the New Flow.



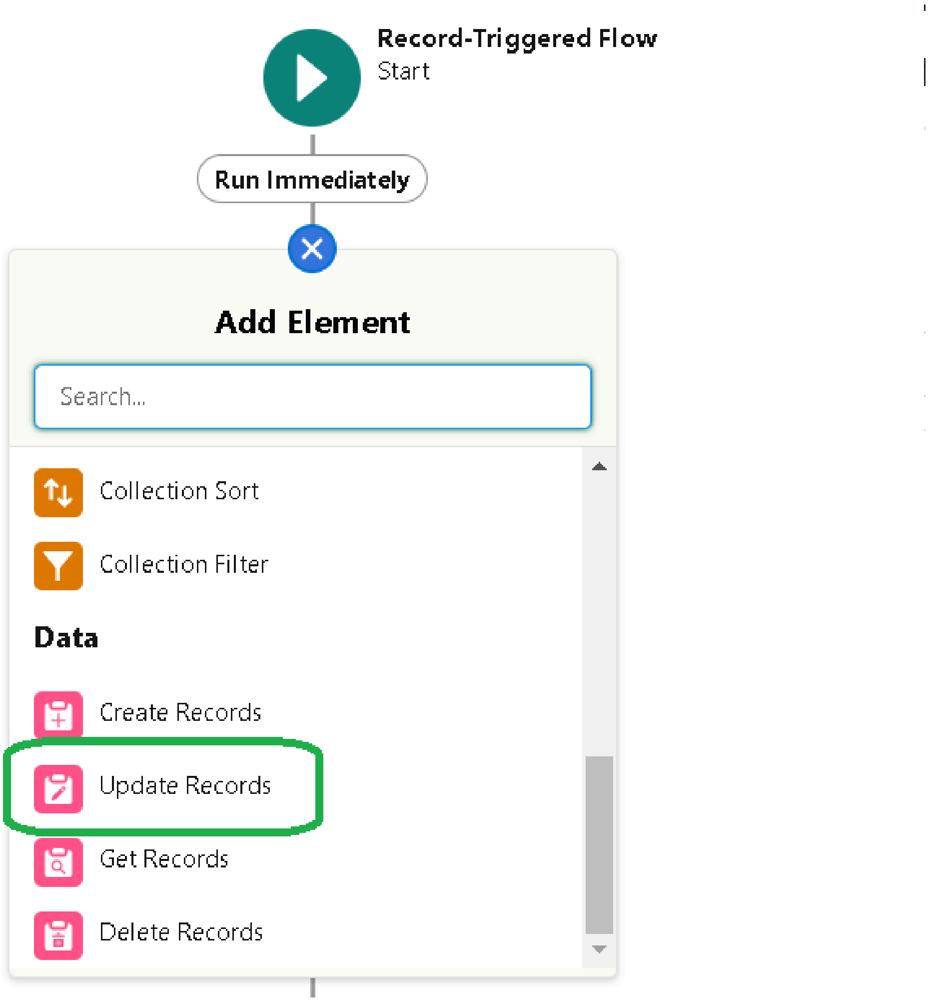
2. Select the Record-triggered flow and Click on Create.



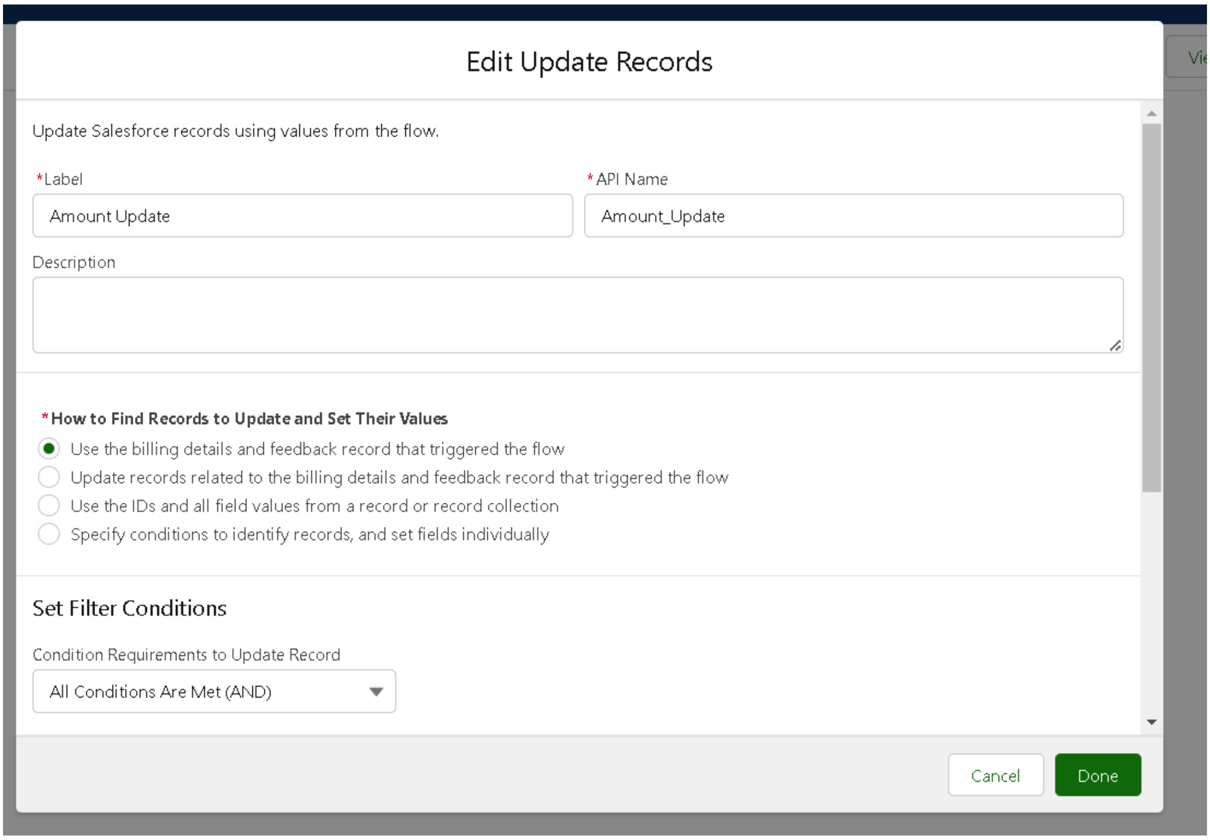
3. Select the Object as “Billing details and feedback”in the Drop down list.
4. Select the Trigger Flow when: “A record is Created or Updated”.
5. Select the Optimize the flow for: “Actions and Related Records” and Click on Done.

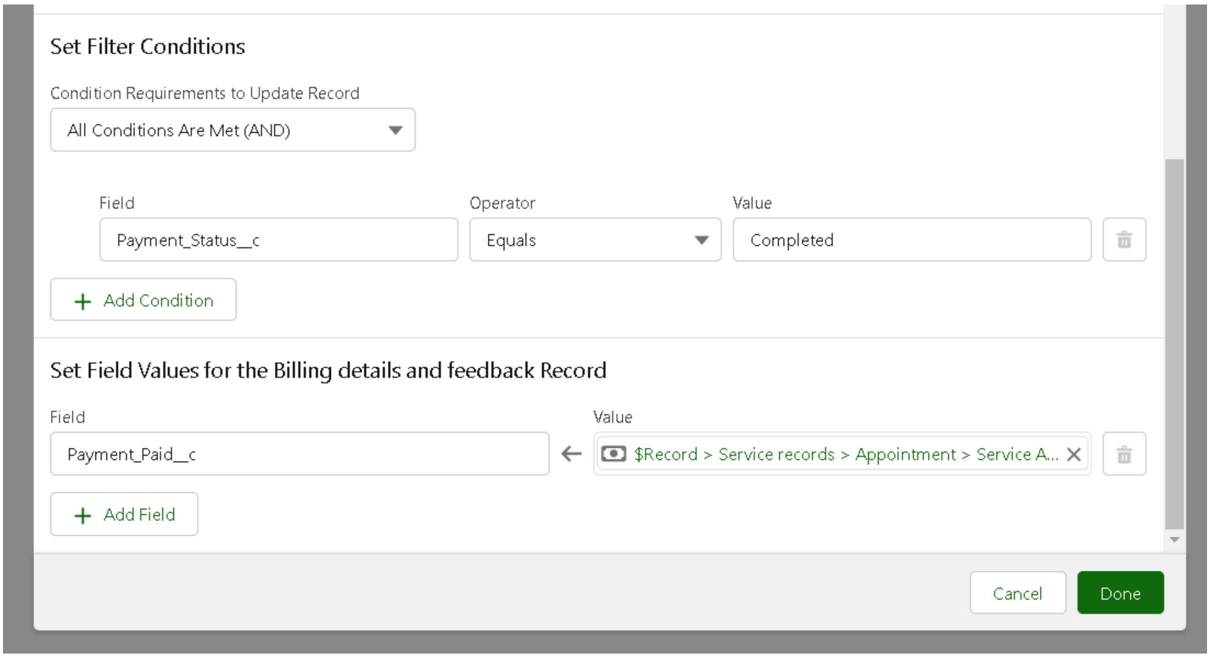


6. Under the Record-triggered Flow Click on "+" Symbol and In the Drop down List select the "Update records Element".



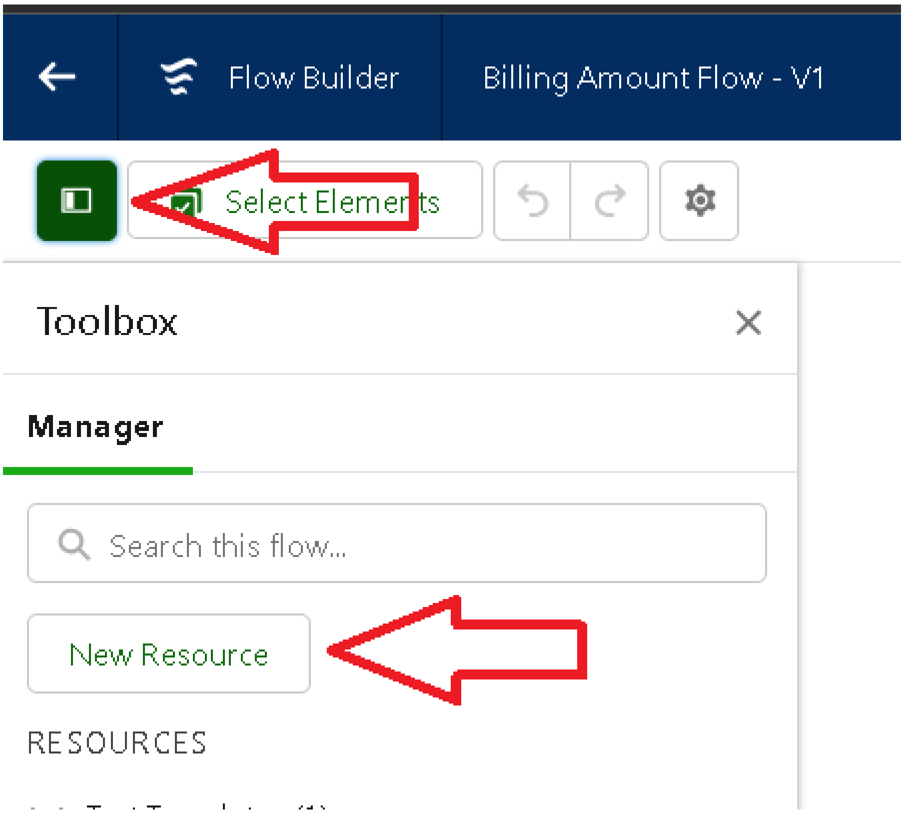
7. Give the Label Name : Amount Update
8. Api name : is auto populated





9. Set a filter condition : All Conditions are met(AND)
10. Field : Payment_Status__c
11. Operator : Equals
12. Value : Completed
13. And Set Field Values for the Billing details and feedback Record
14. Field : Payment_Paid__c
15. Value : {!$Record.Service_records__r.Appointment__r.Service_Amount__c}
16. Click On Done.

17. Before creating another Element. Create a New Resource form Toolbox form top left.



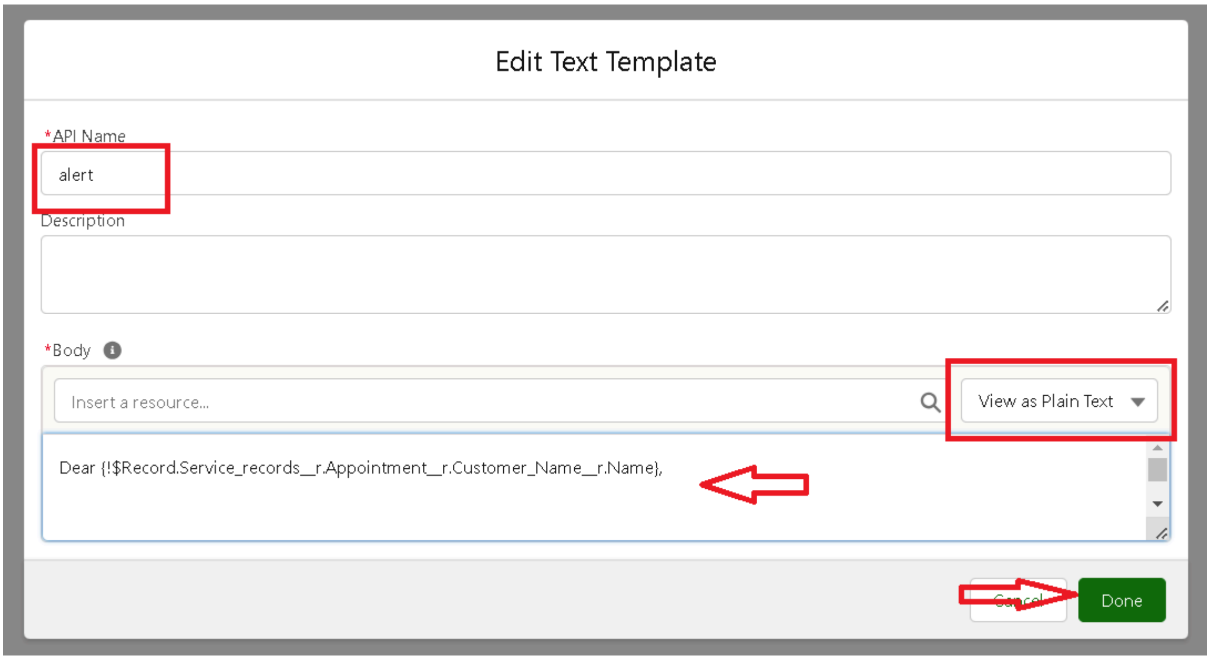
18. Click on the New Resource, And select Variable.
19. Select the resource type as text template.
20. Enter the API name as " alert".
21. Change the view as Rich Text → View to Plain Text.
22. In body field paste the syntax that given below.

Dear
{!$Record.Service_records__r.Appointment__r.Customer_Name__r.Name},

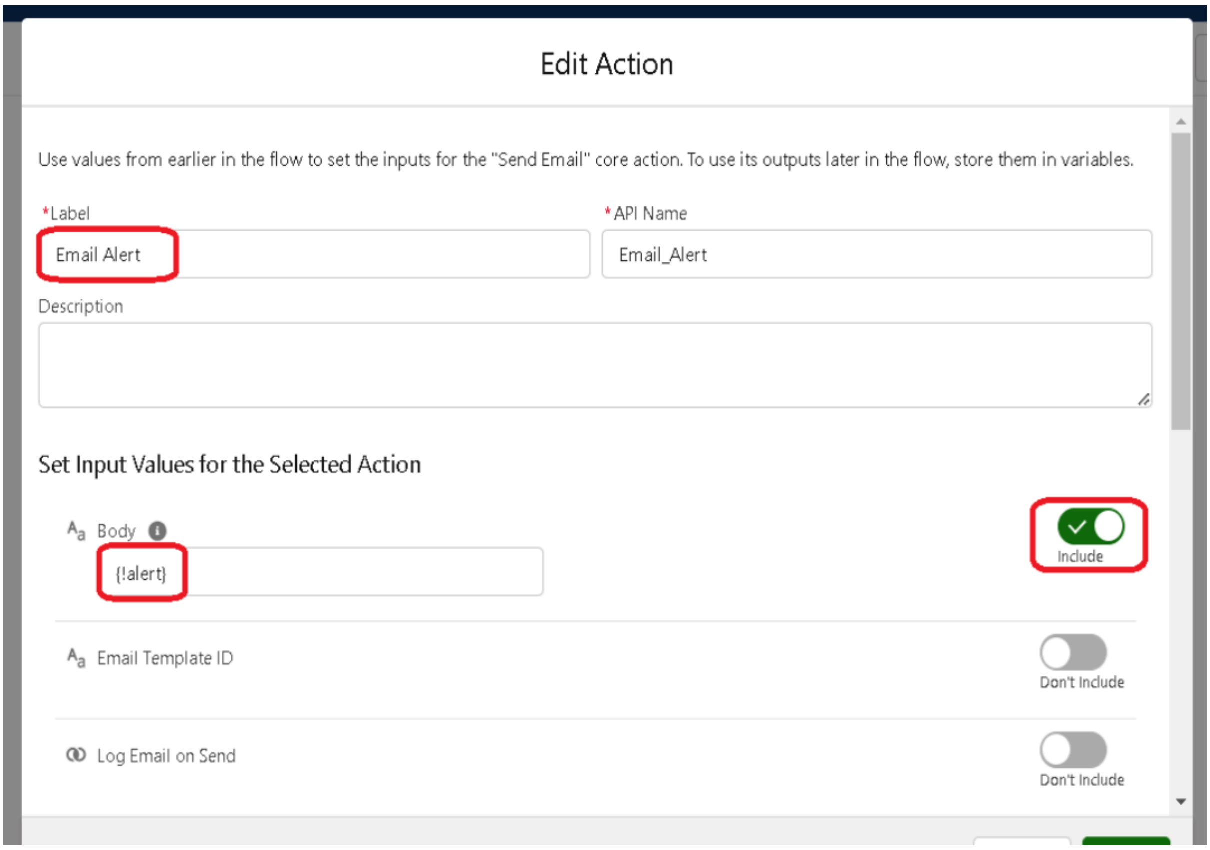
I hope this message finds you well. I wanted to take a moment to express my sincere gratitude for your recent payment for the services provided by our garage management team. Your prompt payment is greatly appreciated, and it helps us continue to provide top-notch services to you and all our valued customers.

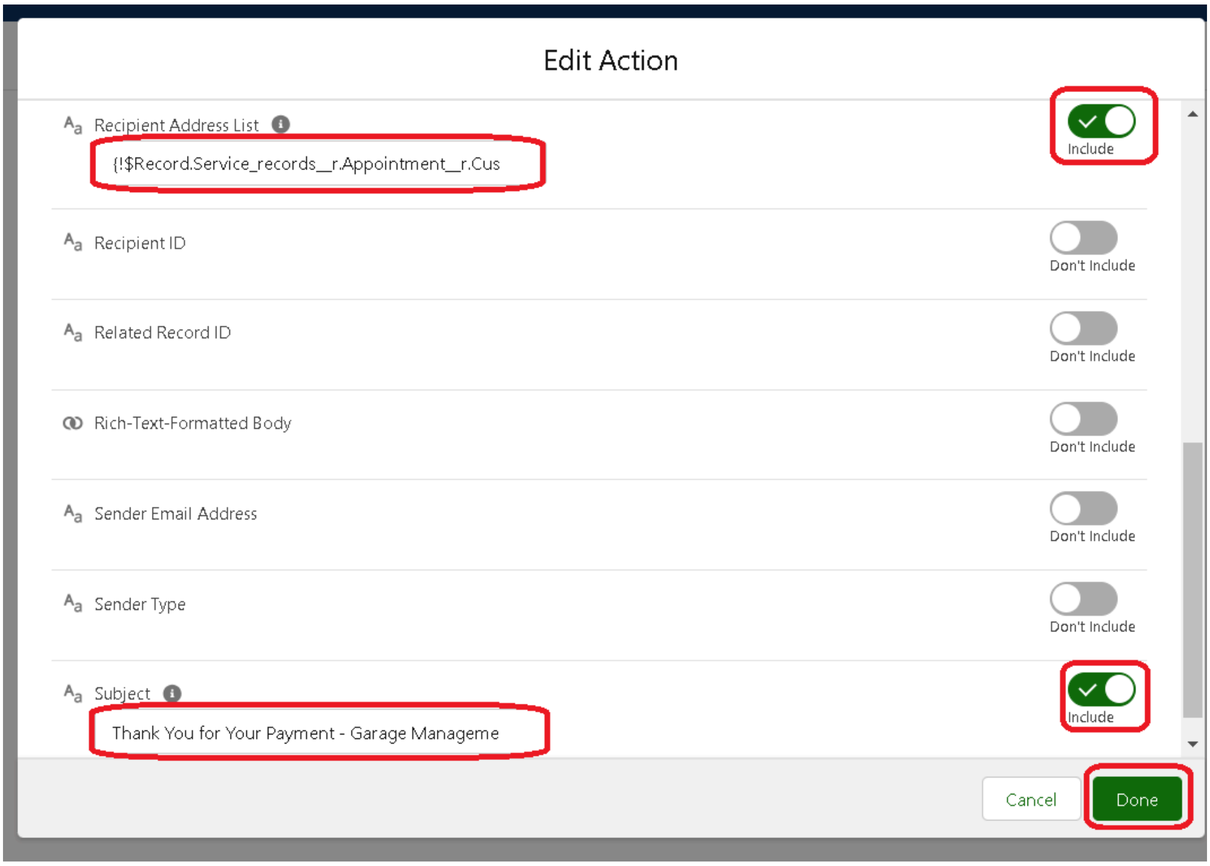
Amount paid : {!$Record.Payment_Paid__c}
Thank you for Coming .

23. Click done.

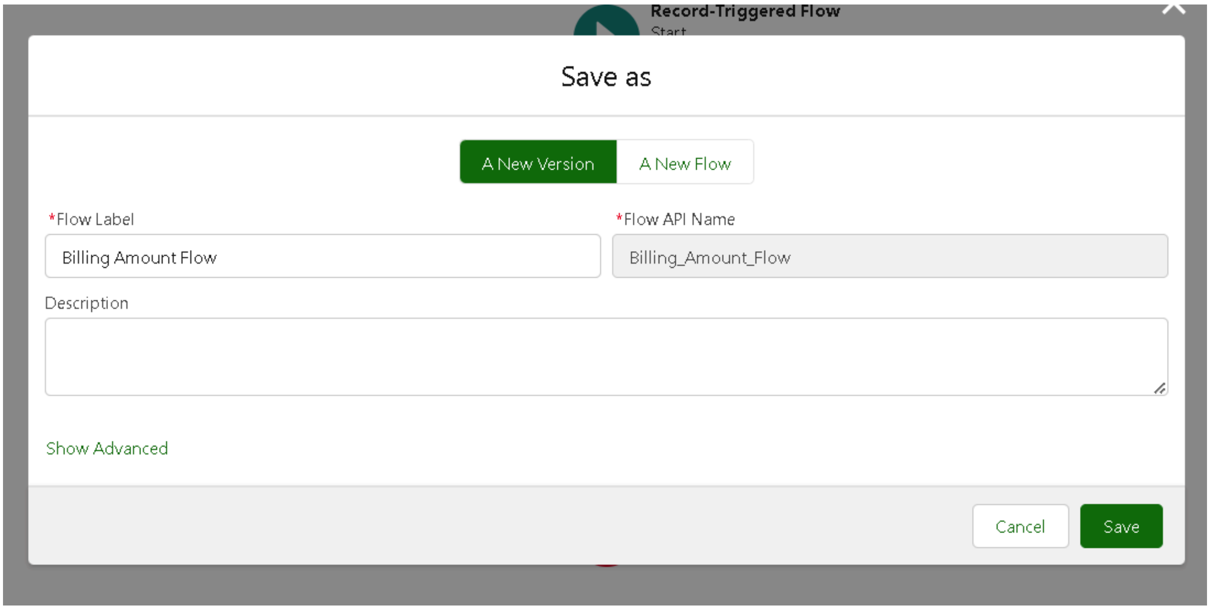


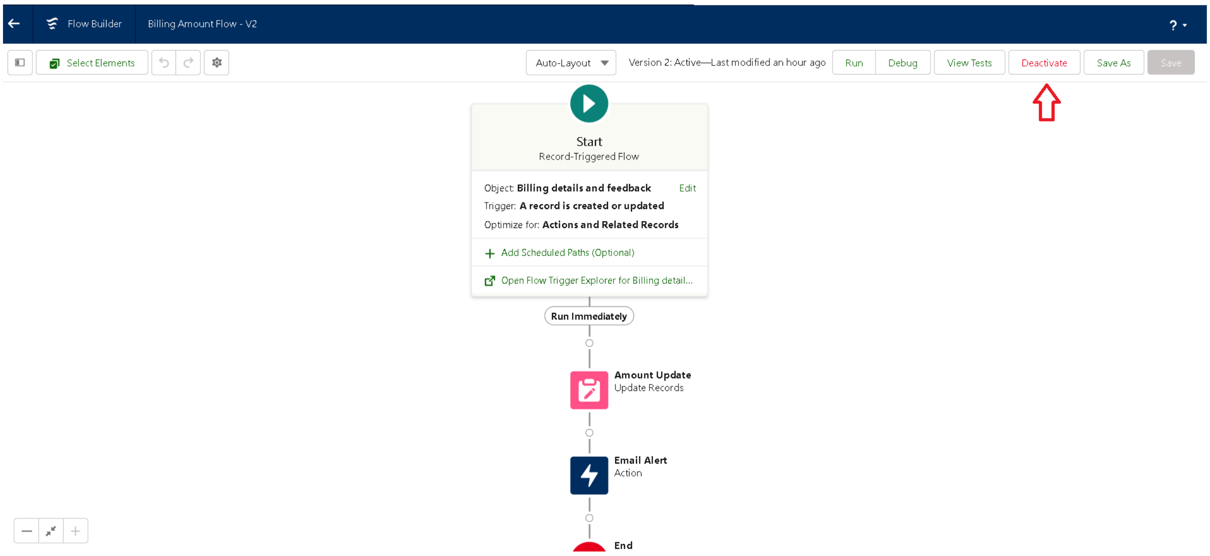
24. Now Click on Add Element , select Action.
25. Their action bar will be opened in that search for “ send email ” and click on it.
26. Give the label name as “ Email Alert”
27. API name will be auto populated.
28. Enable the body in set input values for the selected action.
29. Select the text template that created , Body : {!alert}
30. Include recipient address list select the email form the record.
31. RecipientAddressList: {!$Record.Service_records__r.Appointment__r.Customer_Name__r.Gmail__c}
32. Include subject as “ Thank You for Your Payment - Garage Management”.
33. Click done.


Edit Action
Use values from earlier in the flow to set the inputs for the "Send Email" core action. To use its outputs later in the flow, store them in variables.
*Label
Email Alert
*API Name
Email_Alert
Description
Set Input Values for the Selected Action
Body
{!alert}
Include
Email Template ID
Don't Include
Log Email on Send
Don't Include


Edit Action
Recipient Address List
{!$Record.Service_records__r.Appointment__r.Cus
Include
Recipient ID
Don't Include
Related Record ID
Don't Include
Rich-Text-Formatted Body
Don't Include
Sender Email Address
Don't Include
Sender Type
Don't Include
Subject
Thank You for Your Payment - Garage Manageme
Include
Cancel
Done

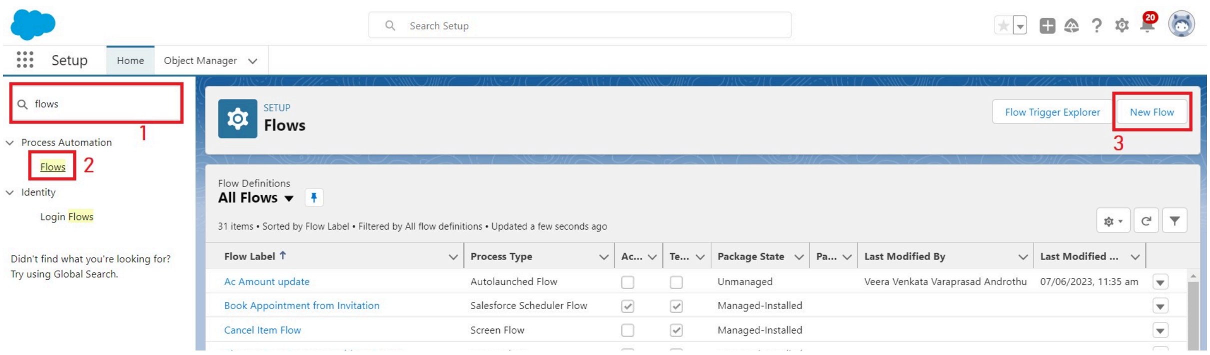
34. Click on save. Give the Flow label , Flow Api name will be autopopulated.
35. And click save, and click on activate.



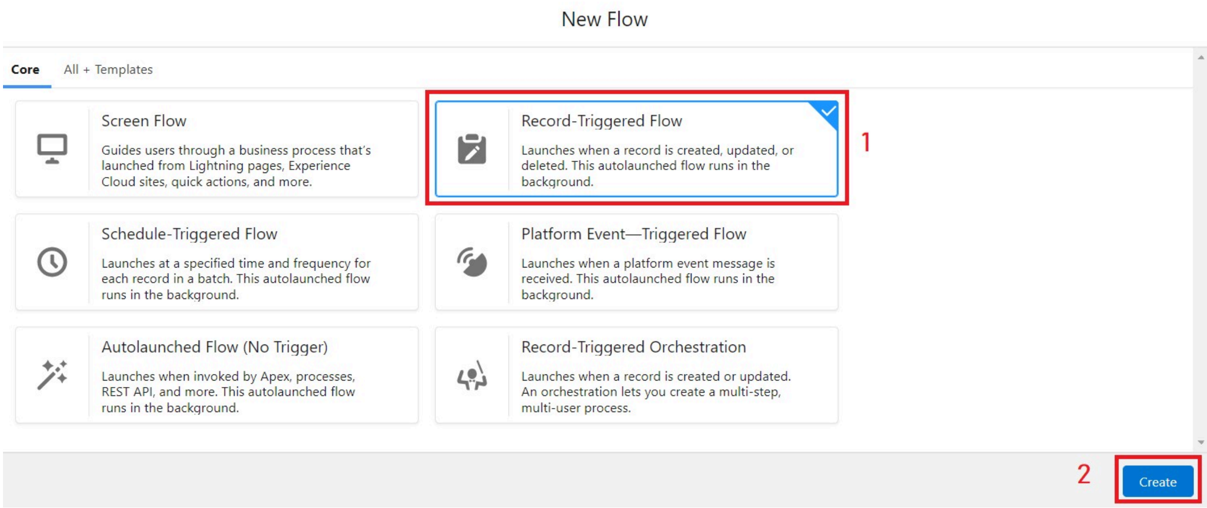


Activity 2: Create another Flow

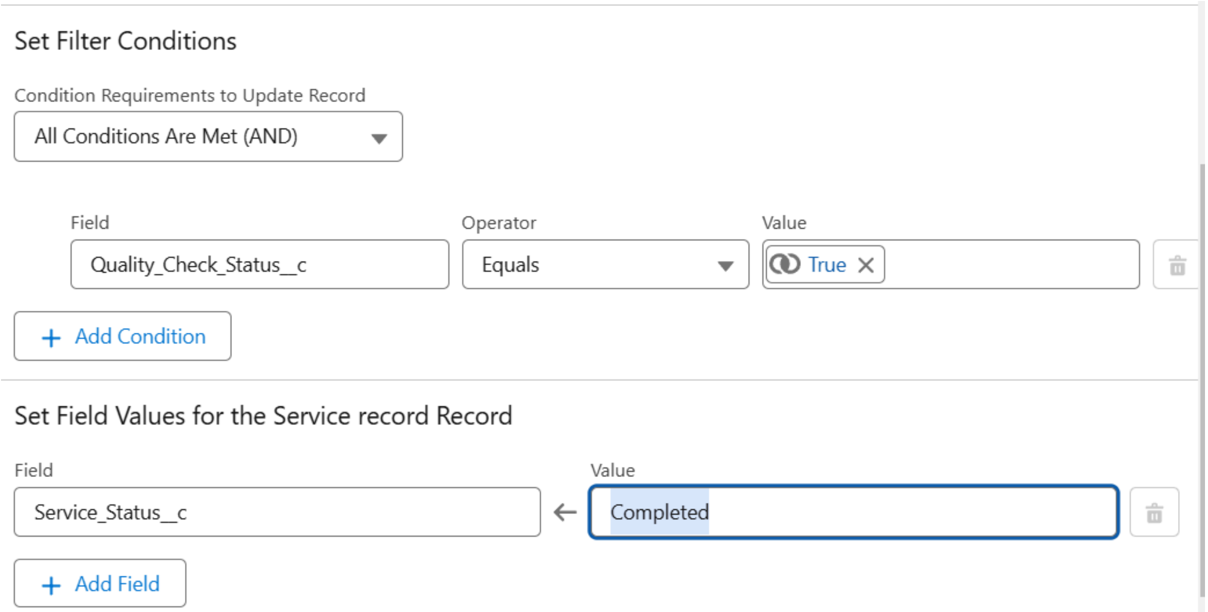
36. Go to setup → type Flow in quick find box → Click on the Flow and Select the New Flow.



37. Select the Record-triggered flow and Click on Create.



38. Select the Object as “ Service records”in the Drop down list.
39. Select the Trigger Flow when: “A record is Created or Updated”.
40. Select the Optimise the flow for: “Actions and Related Records” and Click on Done.
41. Under the Record-triggered Flow Click on “+” Symbol and In the Drop down List select the “Update records Element”.
42. Set a filter condition : All Conditions are met(AND)
43. Field : Quality_Check_Status__c
44. Operator : Equals
45. Value : True
46. And Set Field Values for the Billing details and feedback Record
47. Field : Service_Status__c
48. Value : Completed



49. Click On Done.
50. Click on save
51. Given the Flow label as **Update Service Status** , Flow Api name will be auto populated.
52. And click save, and click on activate.

Milestone 14:Apex Trigger

Apex can be invoked by using triggers. Apex triggers enable you to perform custom actions before or after changes to Salesforce records, such as insertions, updates, or deletions.
A trigger is Apex code that executes before or after the following types of operations:

- insert
- update
- delete
- merge
- upsert
- undelete

For example, you can have a trigger run before an object's records are inserted into the database, after records have been deleted, or even after a record is restored from the Recycle Bin.

You can define triggers for top-level standard objects that support triggers, such as a Contact or an Account, some standard child objects, such as a CaseComment, and custom objects. To define a trigger, from the object management settings for the object whose triggers you want to access, go to Triggers.

There are primarily two types of Apex Triggers:

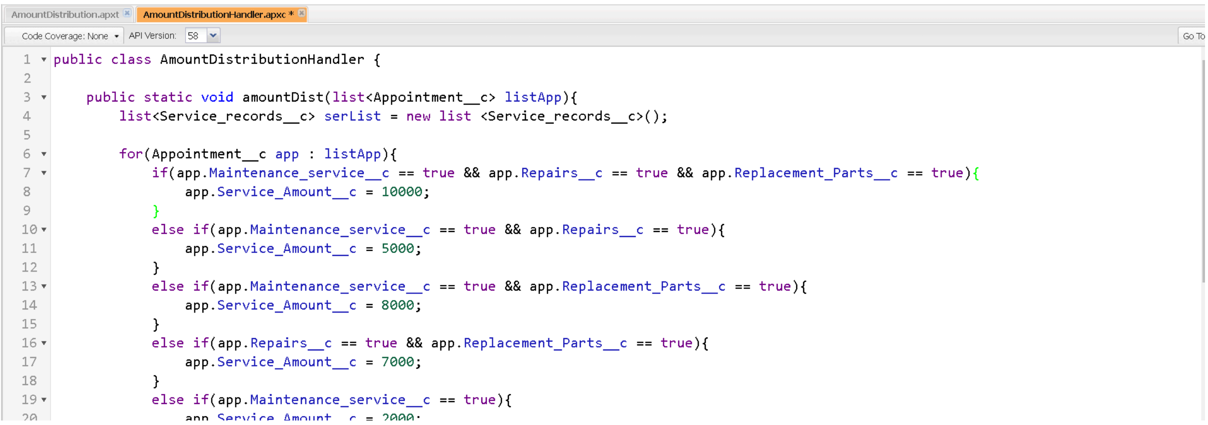
Before Trigger: This type of trigger in Salesforce is used either to update or validate the values of a record before they can be saved into the database. So, basically, the before trigger validates the record first and then saves it. Some criteria or code can be set to check data before it gets ready to be inserted into the database.

After Trigger: This type of trigger in Salesforce is used to access the field values set by the system and affect any change in the record. In other words, the after trigger makes changes to the value from the data inserted in some other record.

Activity- 1: Apex handler

UseCase : This use case works for Amount Distribution for each Service the customer selected for there Vehicle.

1) Login to the respective trailhead account and navigate to the gear icon in the top right corner.
2) Click on the Developer console. Now you will see a new console window.
3) In the toolbar, you can see FILE. Click on it and navigate to new and create New apex class.
4) Name the class as “AmountDistributionHandler ”.



AmountDistribution.apxt | AmountDistributionHandler.apxc *

Code Coverage: None | API Version: 58

```
            }
            else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
                app.Service_Amount__c = 8000;
            }
            else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
                app.Service_Amount__c = 7000;
            }
            else if(app.Maintenance_service__c == true){
                app.Service_Amount__c = 2000;
            }
            else if(app.Repairs__c == true){
                app.Service_Amount__c = 3000;
            }
            else if(app.Replacement_Parts__c == true){
                app.Service_Amount__c = 5000;
            }

        }
    }
}
```

Code:

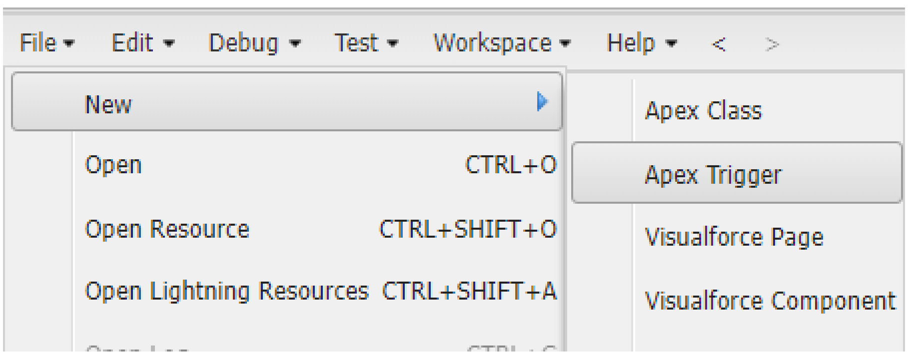
```
public class AmountDistributionHandler {

    public static void amountDist(list<Appointment__c> listApp){
        list<Service_records__c> serList = new list <Service_records__c>();

        for(Appointment__c app : listApp){
            if(app.Maintenance_service__c == true && app.Repairs__c == true &&
app.Replacement_Parts__c == true){
                app.Service_Amount__c = 10000;
            }
            else if(app.Maintenance_service__c == true && app.Repairs__c ==
true){
                app.Service_Amount__c = 5000;
            }
            else if(app.Maintenance_service__c == true &&
app.Replacement_Parts__c == true){
                app.Service_Amount__c = 8000;
            }
            else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
```

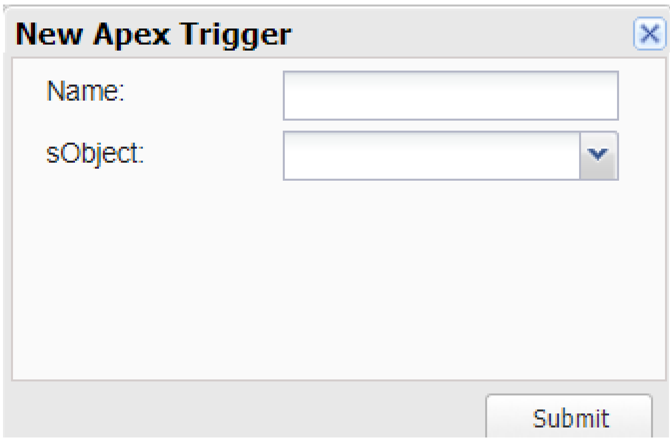
```
            app.Service_Amount__c = 7000;
        }
        else if(app.Maintenance_service__c == true){
            app.Service_Amount__c = 2000;
        }
        else if(app.Repairs__c == true){
            app.Service_Amount__c = 3000;
        }
        else if(app.Replacement_Parts__c == true){
            app.Service_Amount__c = 5000;
        }

    }
    }
}
```

Trigger Handler :

How to create a new trigger :

1) While still in the trailhead account, navigate to the gear icon in the top right corner.
2) Click on developer console and you will be navigated to a new console window.
3) Click on File menu in the tool bar, and click on new→ Trigger.
4) Enter the trigger name and the object to be triggered.
5) Name : AmountDistribution
6) sObject : Appointment__c





Syntax For creating trigger :

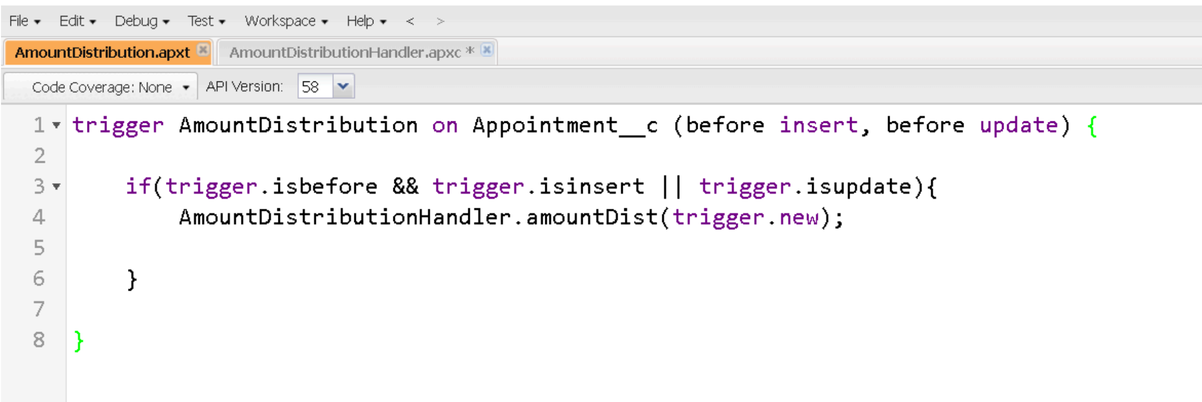
The syntax for creating trigger is :

Trigger [trigger name] on [object name](Before/After event)
{

}

In this project , trigger is called whenever the particular records sum exceed the threshold i.e minimum business requirement value. Then the code in the trigger will get executed.

1. Handler for the Appointment Object



Code:

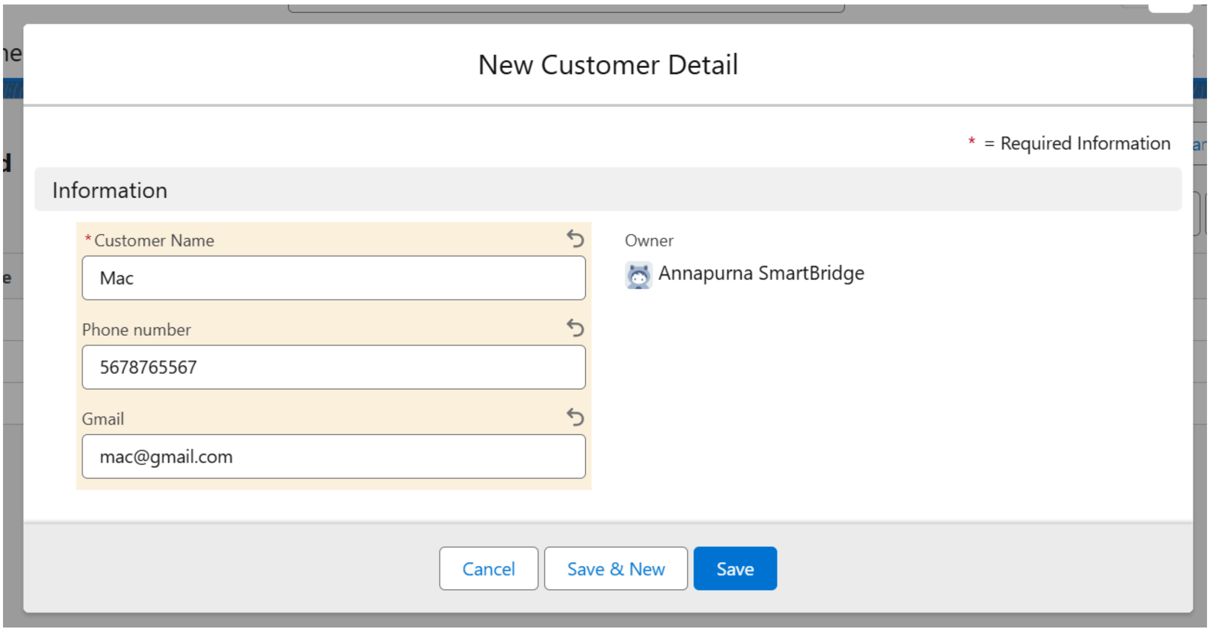
```
trigger AmountDistribution on Appointment__c (before insert, before update) {

    if(trigger.isbefore && trigger.isinsert || trigger.isupdate){
        AmountDistributionHandler.amountDist(trigger.new);

    }

}
```

Milestone-15: User Adoption

Activity 1 : creating record

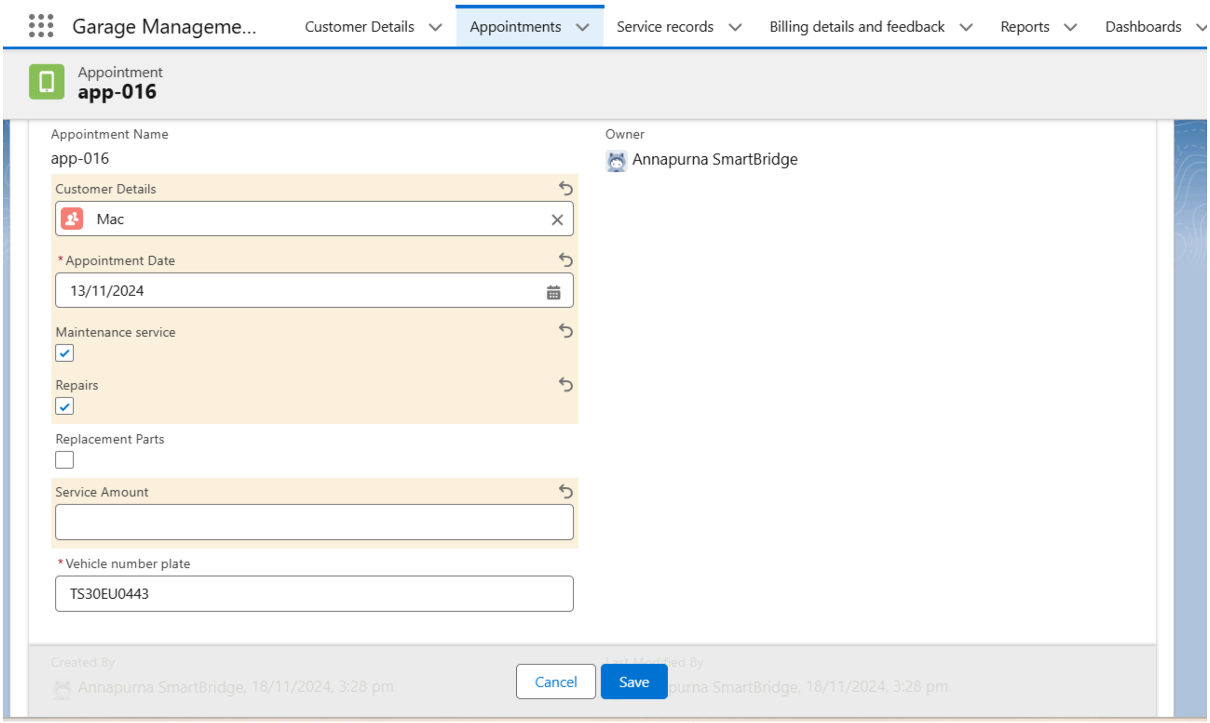
To create a record in the follow objects follow these steps

1. Click on the app launcher located at the left side of the screen.
2. Search for " Garage Management" and click on it.
3. Click on the " Consumer details tab".
4. Click on new and fill the details as shown below figs, and click save.



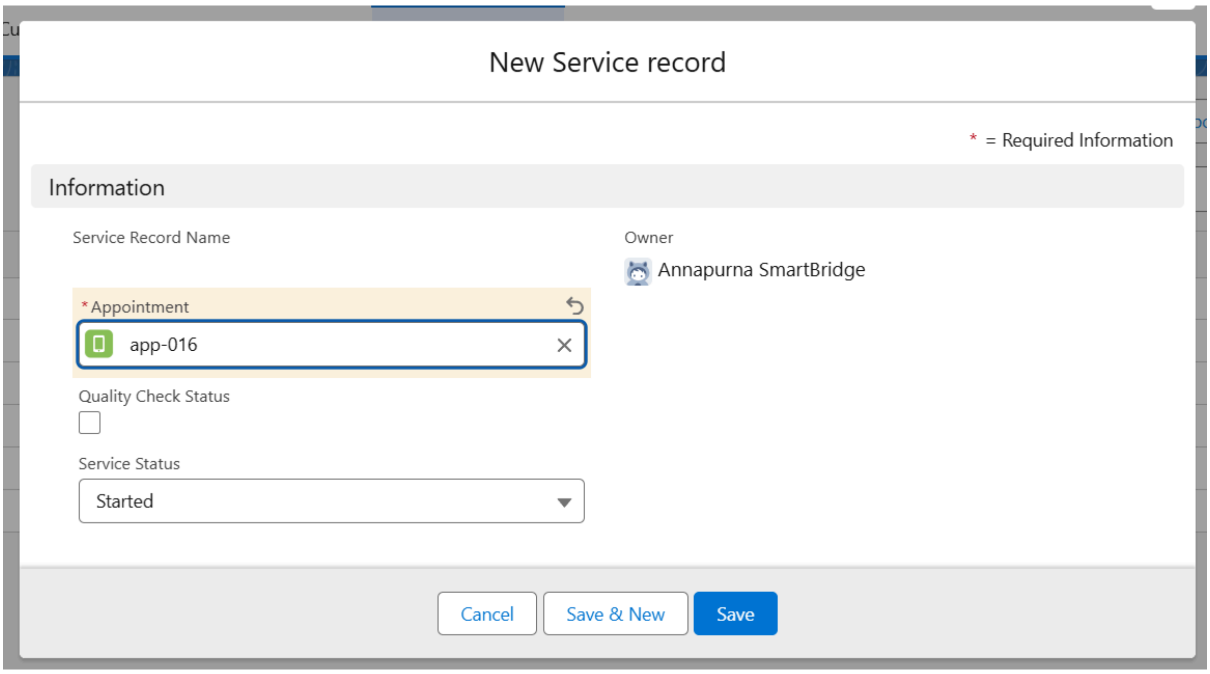
Now, Create the Appointment Record

1. Click on the "Appointment tab".
2. Enter the customer details as created, while entering **Appointment Date** enter the date less than the created date.
3. Match the validation while entering the **vehicle number plate.**
4. Select the services you need.
5. Click on save to see the **Service Amount.**

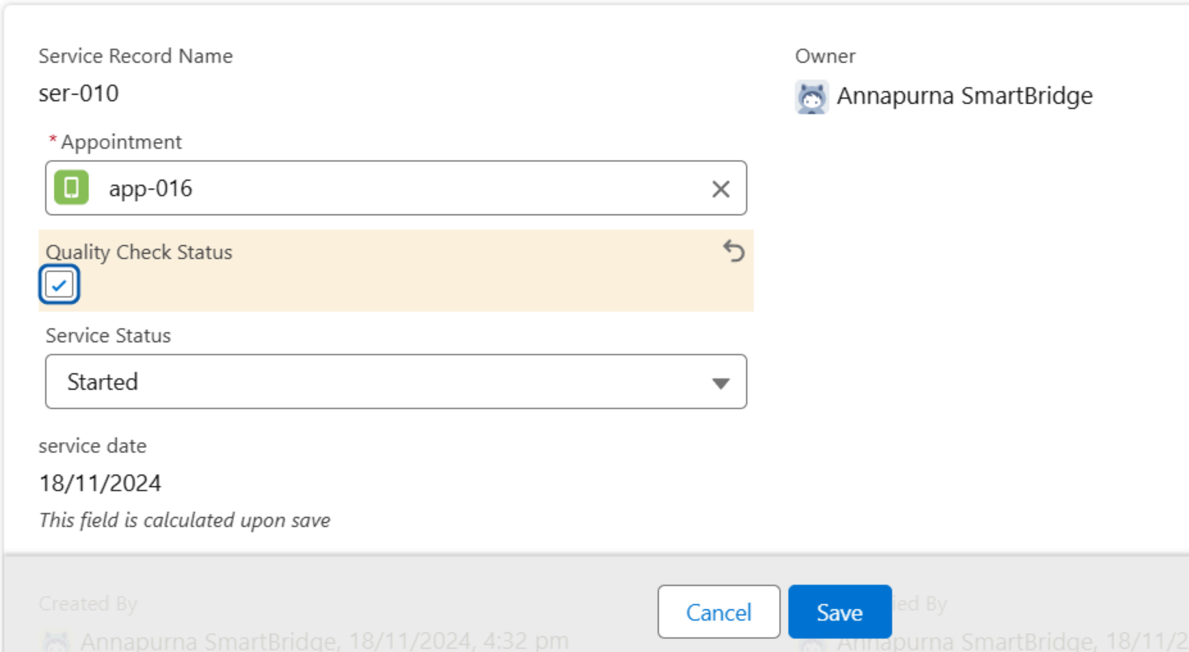


Now, Create a service Record

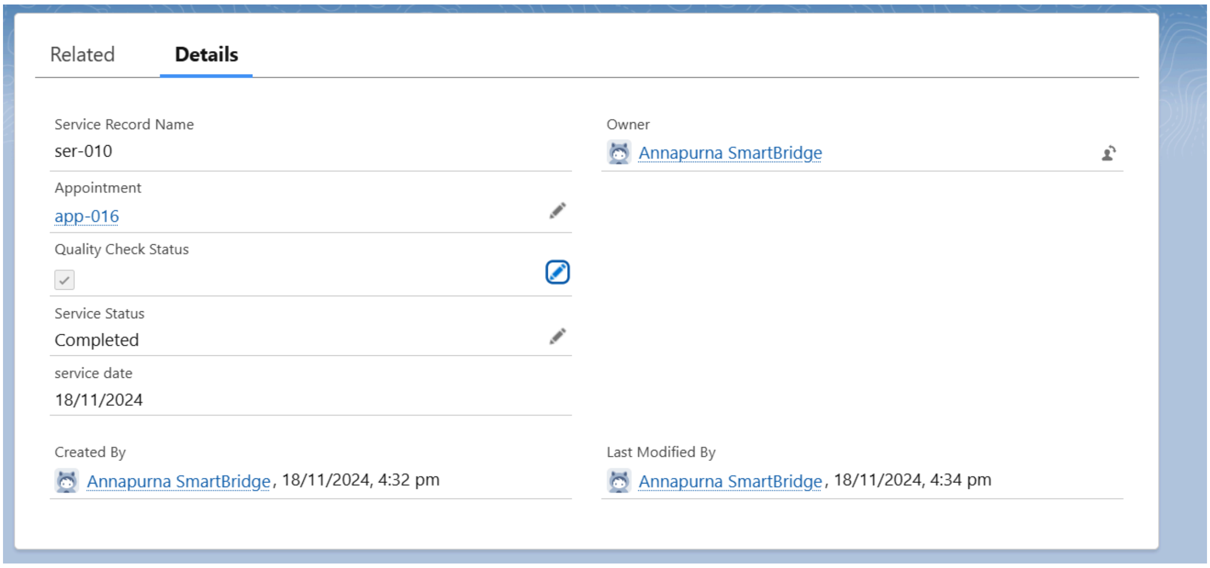
1. Click on the "Service record tab".
2. Enter the Appointment, and started is selected as default.
3. Click on save.



4. Open the record and click on Quality check status as true.
5. Click on save.



6. Now automatically Service status will be moved to completed.



Milestone 16 : Reports

Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics.

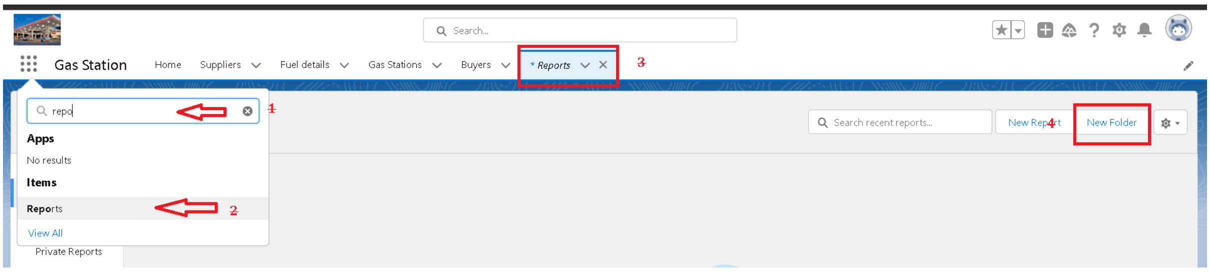
Types of Reports in Salesforce

1. Tabular
2. Summary
3. Matrix

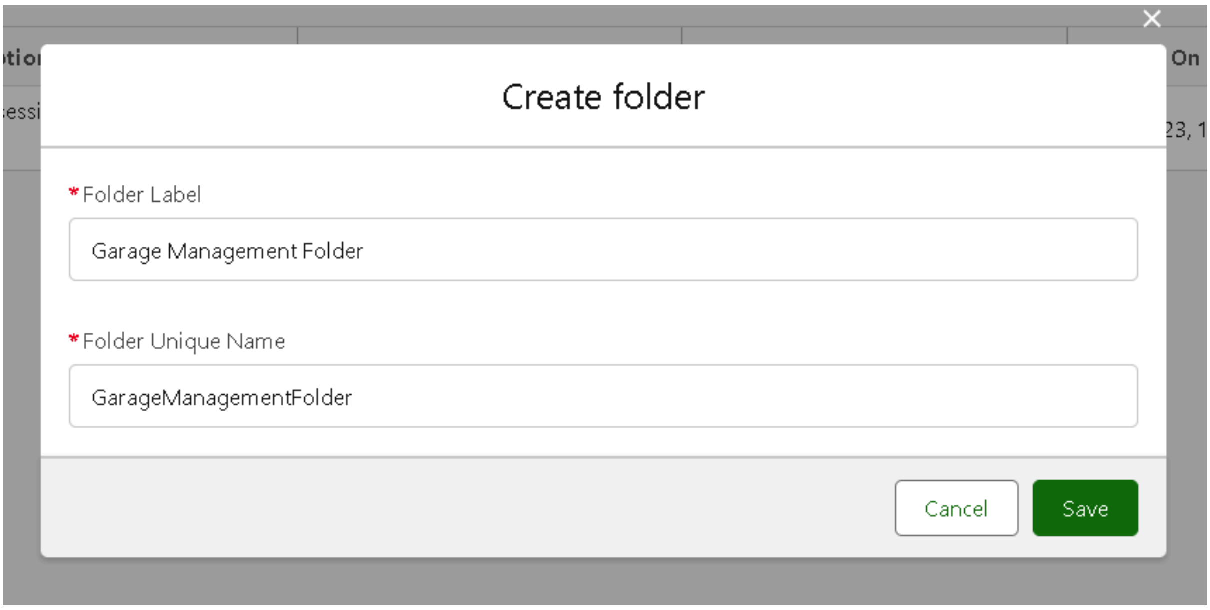
4. Joined Reports

Activity 1: create a report folder

1. Click on the app launcher and search for reports.
2. Click on the report tab, click on new folder.

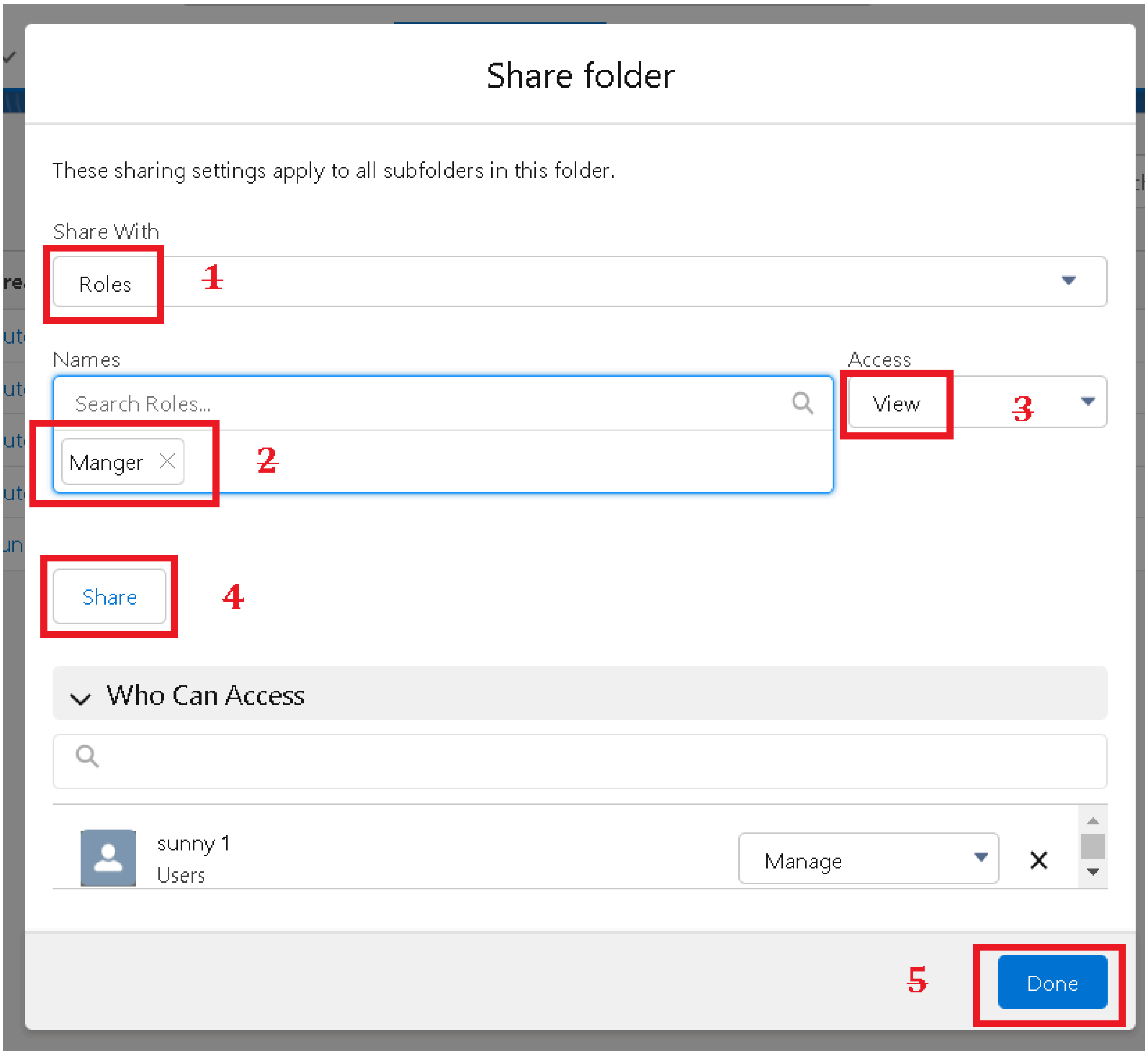


3. Give the Folder label as “Garage Management Folder”, Folder unique name will be auto populated.
4. Click save.



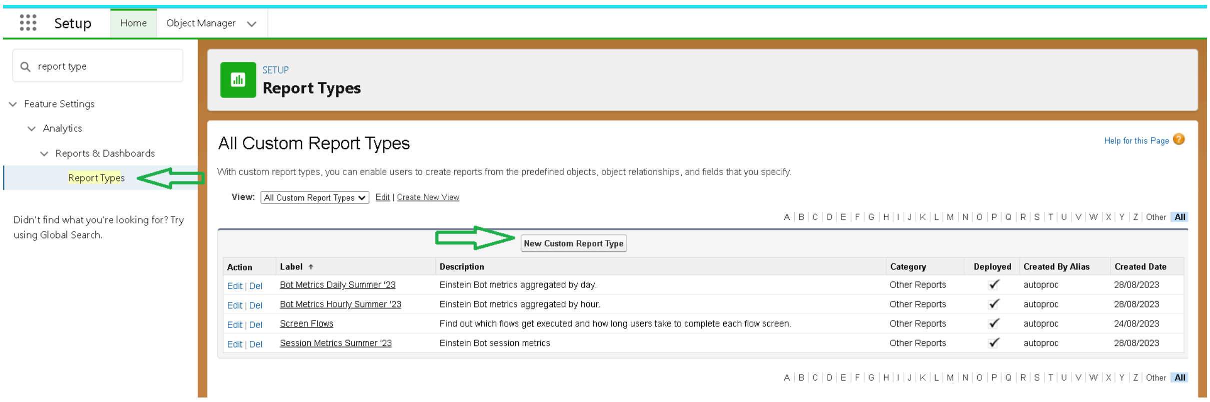
Activity 2: Sharing a report folder

1. Go to the app → click on the reports tab.
2. Click on the All folder , click on the Drop down arrow for Garage Management folder, and Click on share.
3. Select the share with as “roles”, in name field search for “manager”, give “view” as access for that role.
4. Then click share, and click on Done.

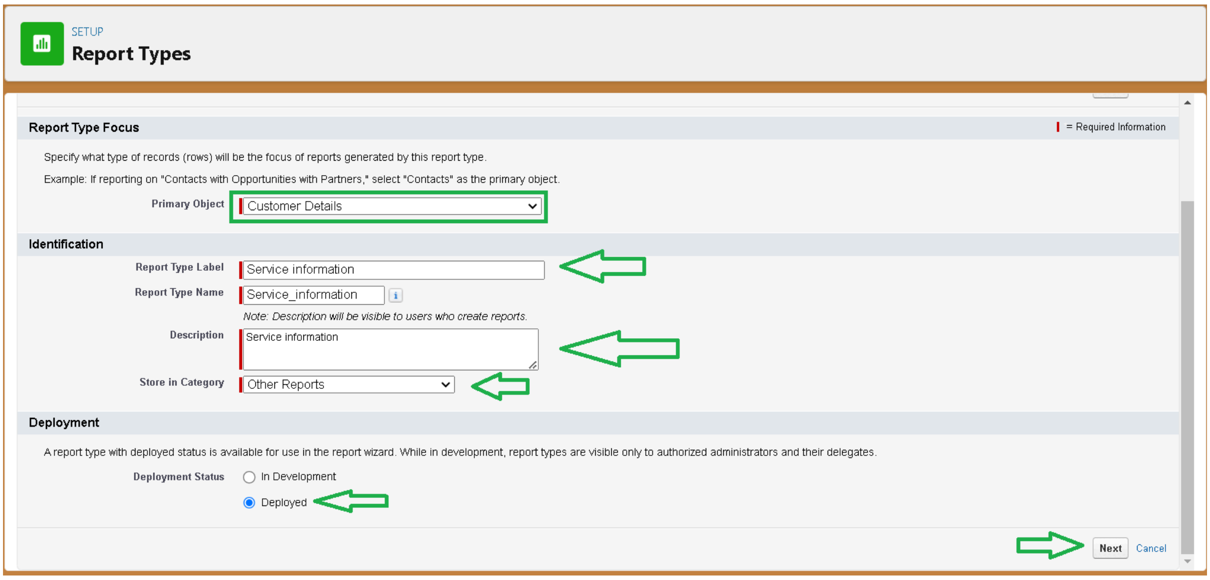


Activity 3: Create Report Type

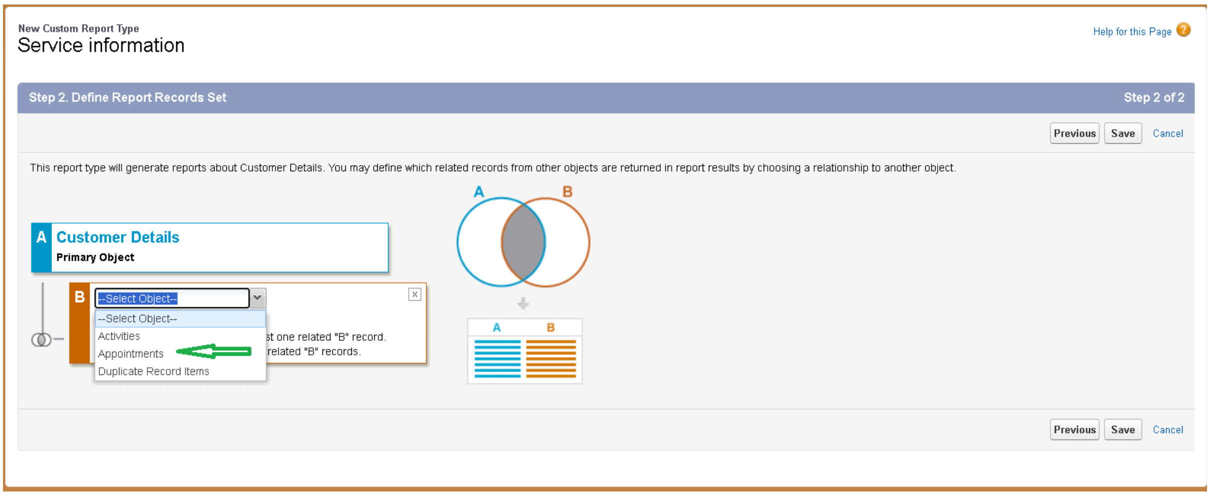
1. Go to setup → type users in quick find box → select Report Type→ click on Continue.
2. Click on new custom report type.

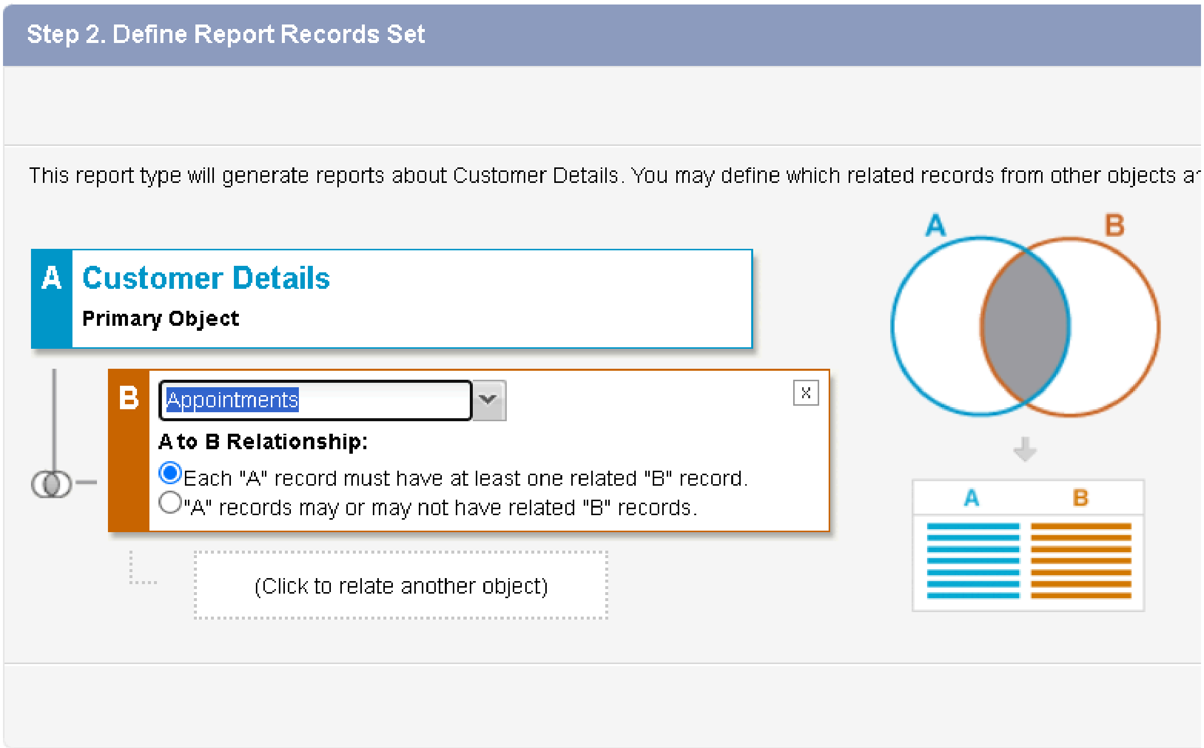


3. Select the Primary object as " Customer details" .
4. Give the Report type Label as " Service information "
5. Report type Name is autopopulated.
6. Keep the Description as same.
7. Select Store in Category as " other Reports "
8. Select the deployment status as " Depolyed ", click on Next.

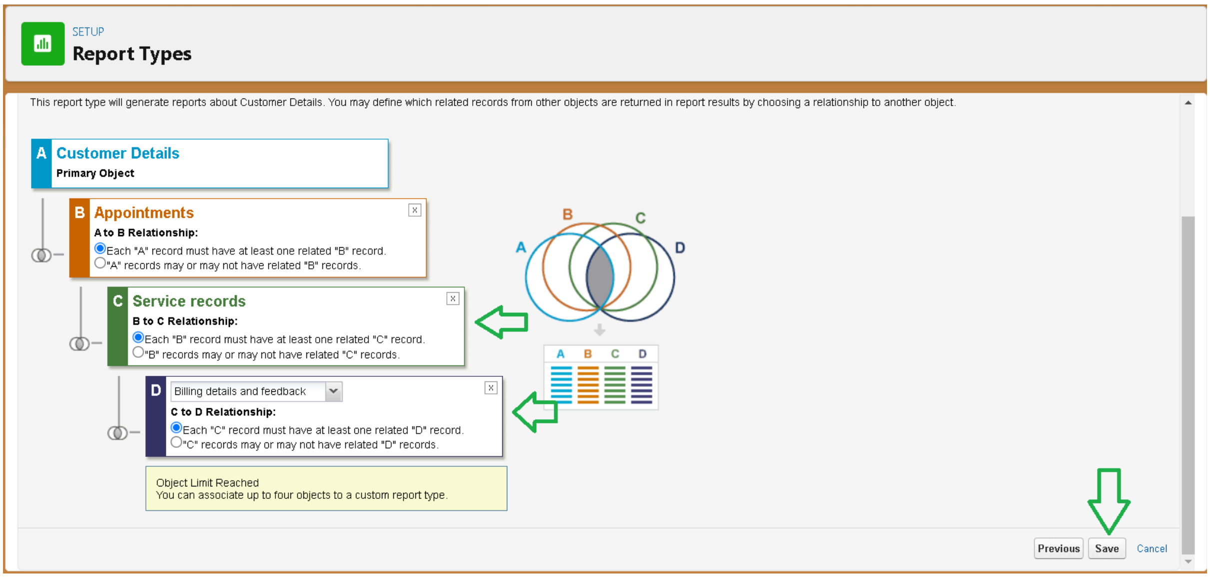


9. now , Click on Related object box.
10. Click on Select Object, choose Appointment Object as shown in fig.





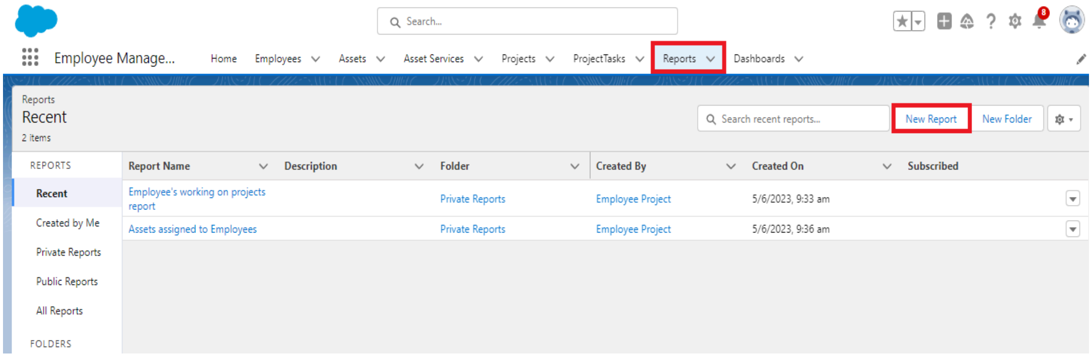
11. Again Click to relate another object.
12. And select the related object as " service records".
13. Repeat the process and select the related object as " Billing details and feedback".
14. And click on save.



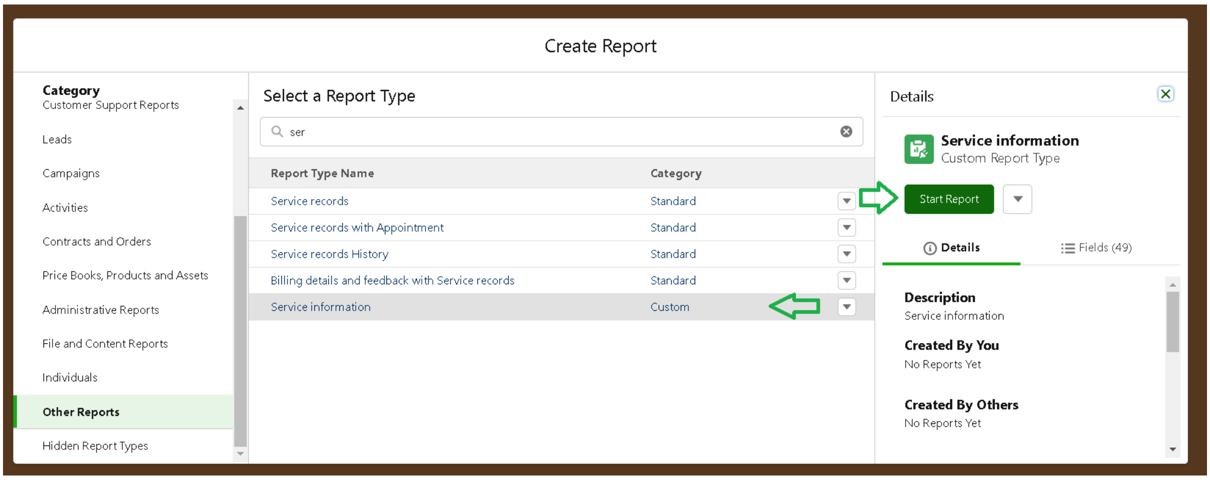
Activity 4: Create Report

Note : Before creating report, create latest “10” records in every object. Try to fill every field in each record for better experience.

1. Go to the app → click on the reports tab
2. Click New Report.

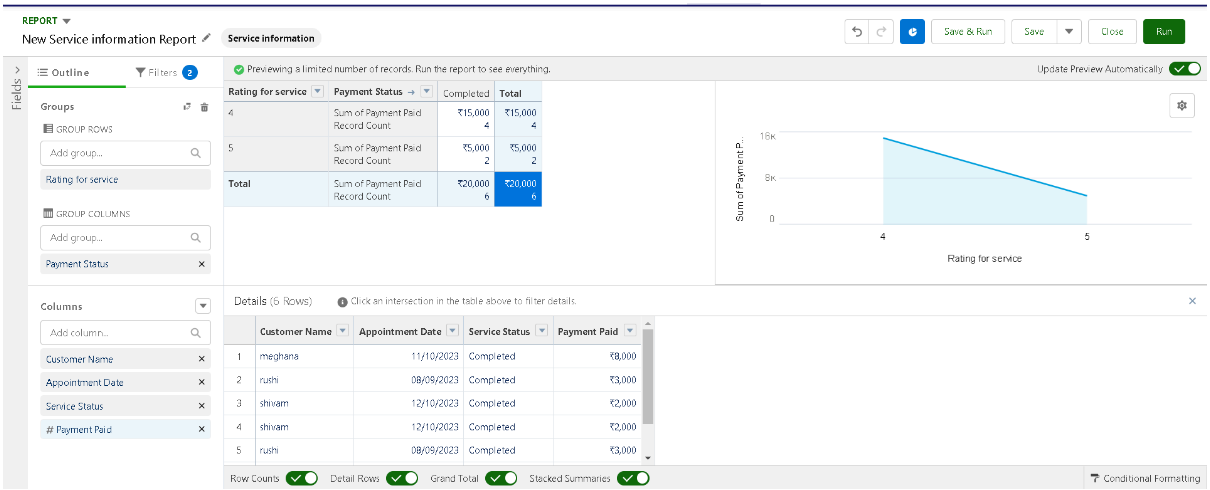


3. Select the Category as other reports, search for Service Information, select that report, click on it. And click on start report.



4. Their outline pane is opened already, select the fields that are mentioned below in the column section.
   a. Customer name
   b. Appointment Date
   c. Service Status
   d. Payment paid
5. Remove the unnecessary fields.
6. Select the fields that mentioned below in GROUP ROWS section.
   a. Rating for Service
7. Select the fields that mentioned below in GROUP ROWS section.
   a. Payment Status
8. Click on Add Chart , Select the Line Chart.

9. Click on save, Give the report Name : New Service information Report
10. Report unique Name is auto populated.
11. Select the folder the created and Click on save.



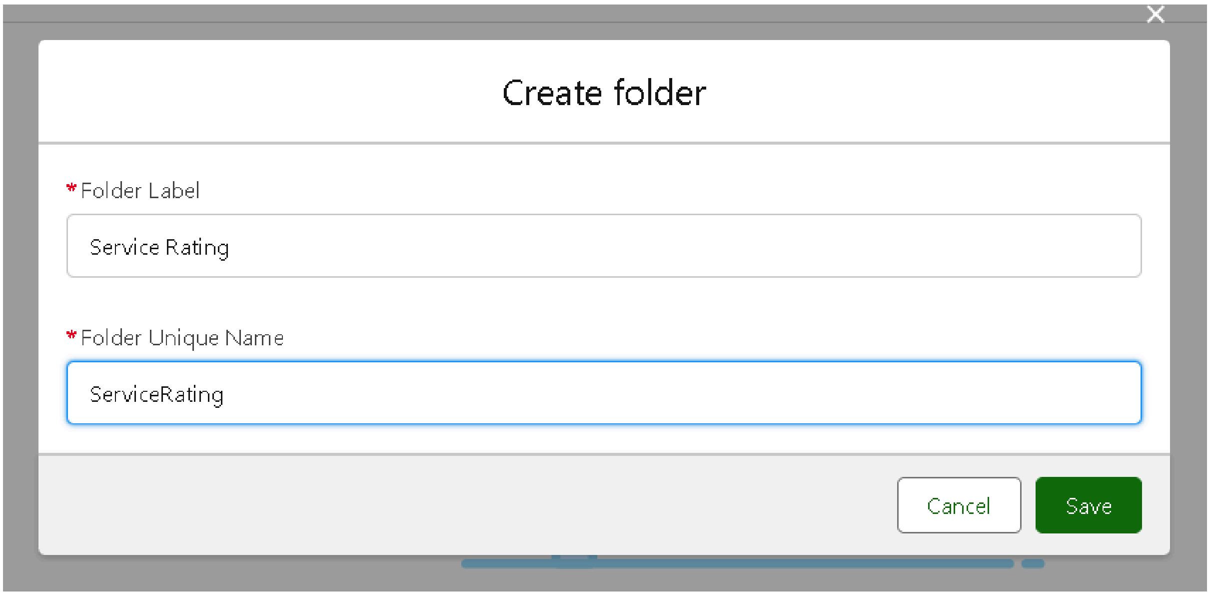
Milestone 17 : Dashboards

Dashboards help you visually understand changing business conditions so you can make decisions based on the real-time data you've gathered with reports. Use dashboards to help users identify trends, sort out quantities, and measure the impact of their activities. Before building, reading, and sharing dashboards, review these dashboard basics.

Activity 1: Create Dashboard Folder

1. Click on the app launcher and search for dashboard.

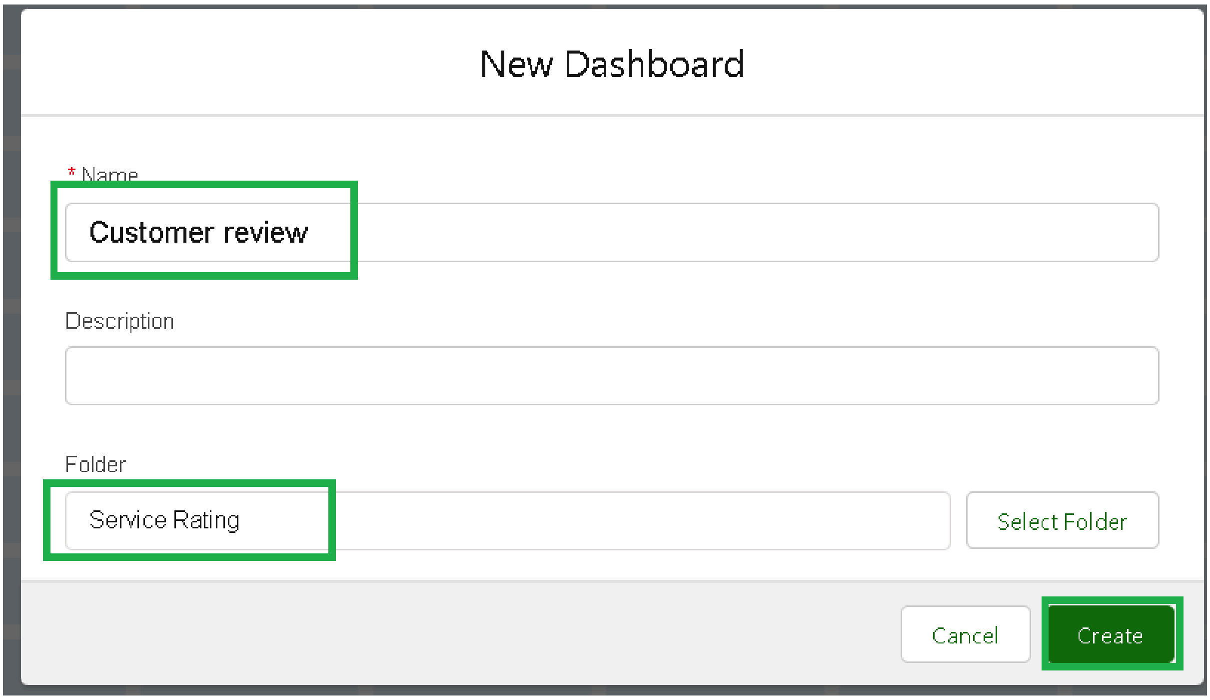
2. Click on dashboard tab.
3. Click new folder, give the folder label as “ Service Rating dashboard”.
4. Folder unique name will be auto populated.
5. Click save.



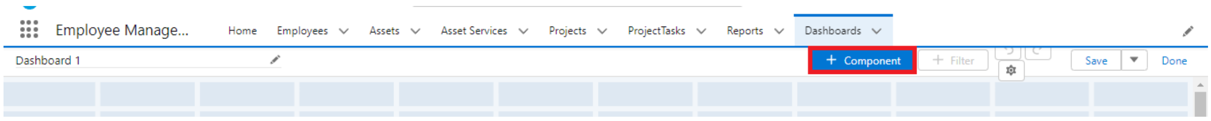
6. Follow the same steps, form milestone 15, and activity 2, and provide the sharing settings for the folder that just created.

Activity 2: Create Dashboard

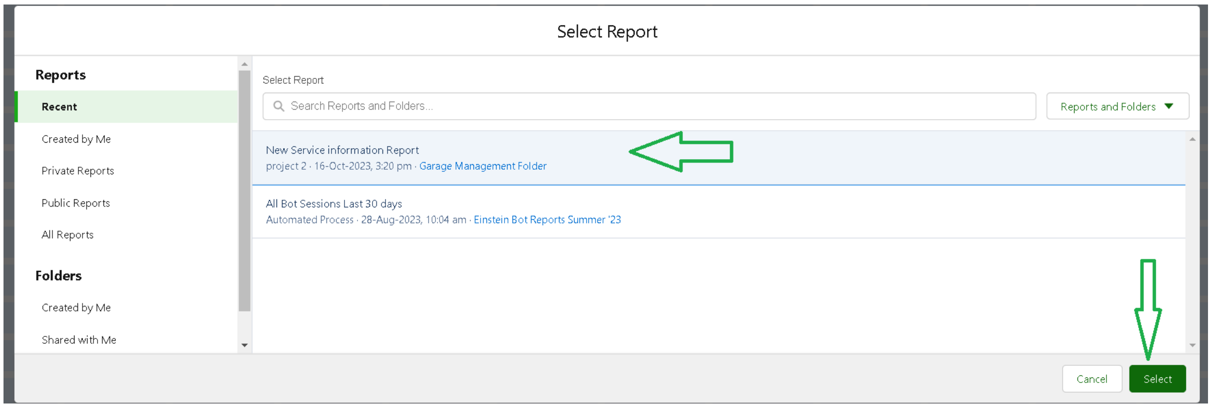
1. Go to the app → click on the Dashboards tabs.
2. Give a Name and select the folder that created, and click on create.



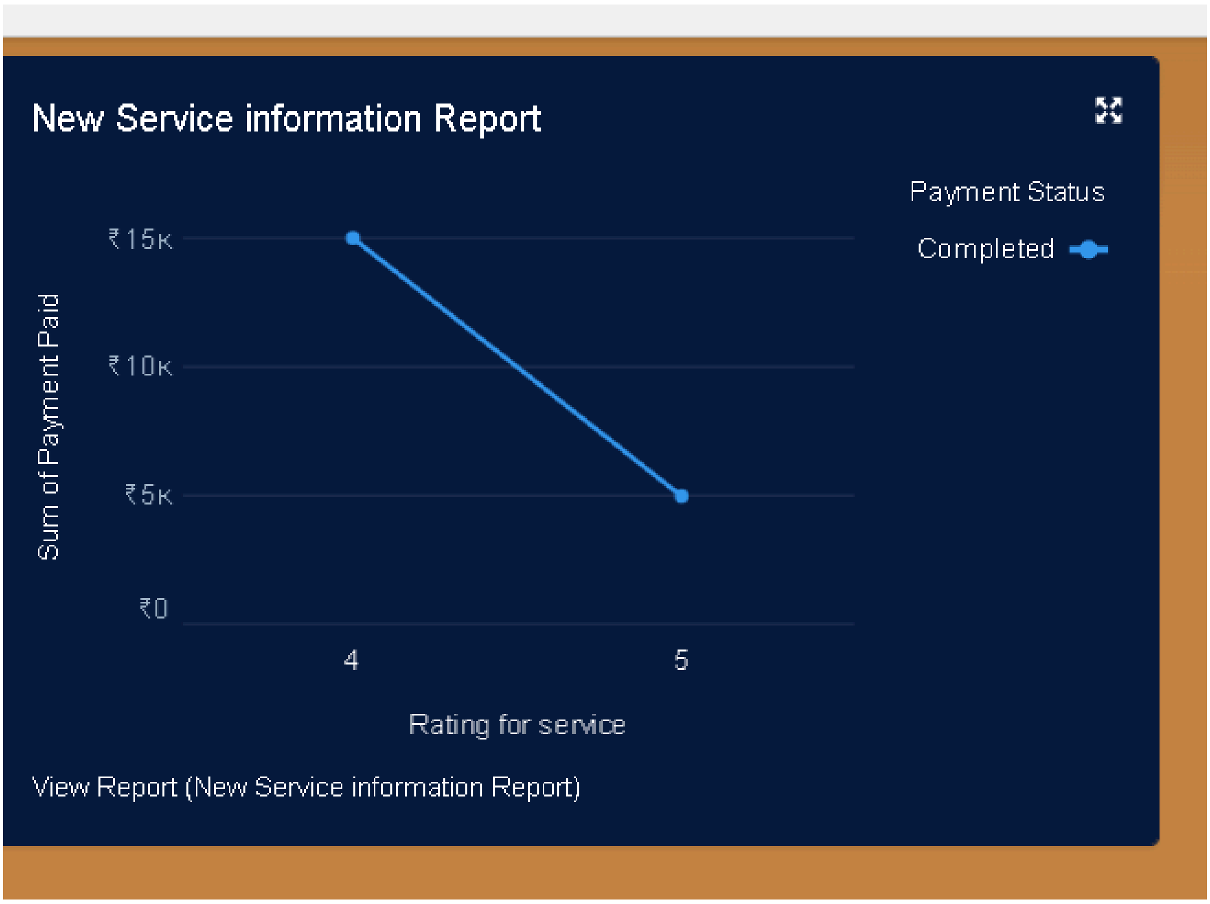
3. Select add component.



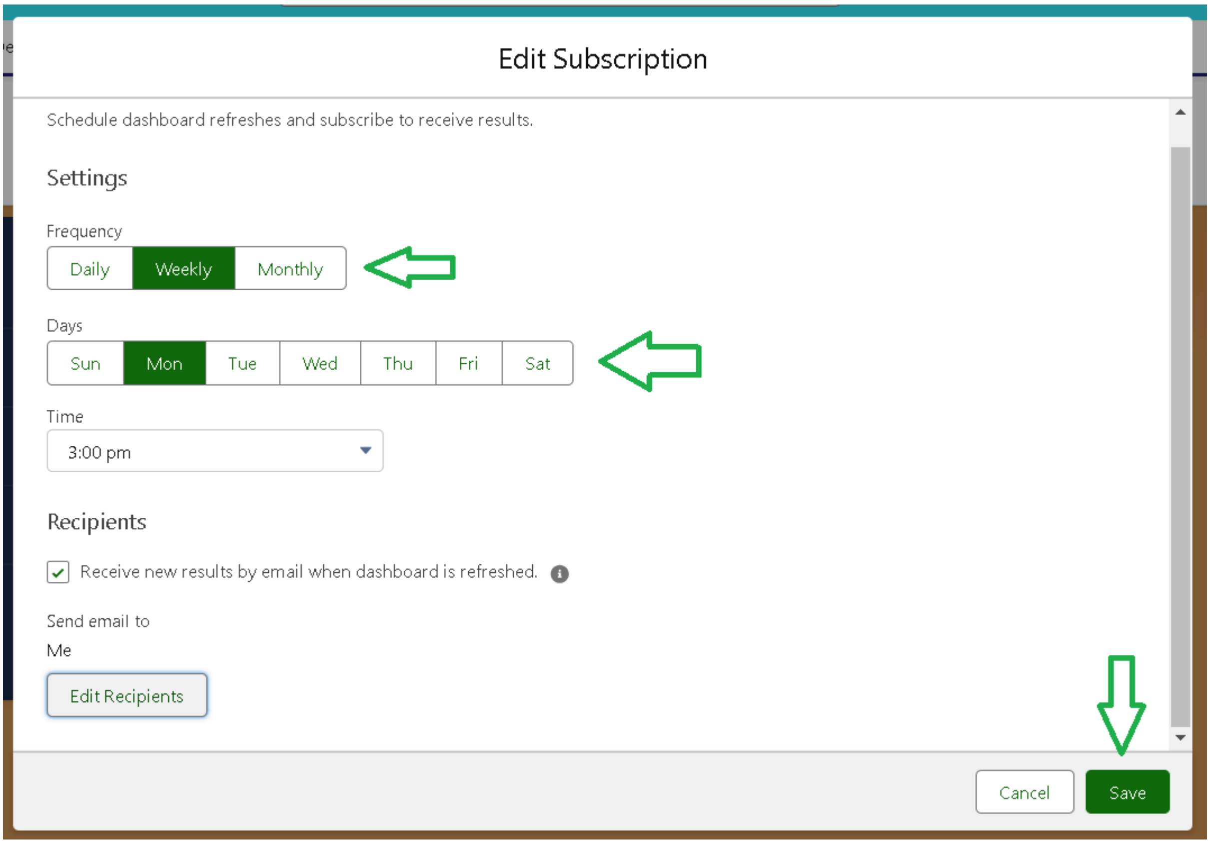
4. Select a Report and click on select.



5. Select the Line Chart. Change the theme.
6. Click Add then click on Save and then click on Done.
7. Preview is shown below.



8. After that Click on Subcribe on top right.
9. Set the Frequency as “ weekly ”.
10. Set a day as monday.
11. And Click on save.


Edit Subscription
Schedule dashboard refreshes and subscribe to receive results.
Settings
Frequency
Daily
Weekly
Monthly
Days
Sun
Mon
Tue
Wed
Thu
Fri
Sat
Time
3:00 pm
Recipients
Receive new results by email when dashboard is refreshed.
Send email to
Me
Edit Recipients
Cancel
Save