GARAGE MANAGEMENT SYSTEM

BY

AYYEESHA KHANNA

ayyeeshakhanna@gmail.com

ABSTRACT

Salesforce is a leading customer relationship management (CRM) software, has been transforming the way businesses interact with their customers across various industries. One such industry that has greatly benefited from Salesforce's innovative solutions is the automotive sector. The Garage Management System (GMS), a specialized tool developed by Salesforce, has been designed to cater to the unique needs of automotive repair facilities. By providing a comprehensive platform to manage customer relationships, streamline operations, and boost sales, GMS has empowered garages to deliver exceptional service, increase efficiency, and drive growth.

With GMS, garages can automate routine tasks, optimize workflows, and gain real-time visibility into their operations. The system's user-friendly interface and powerful features enable staff to work more efficiently, reducing errors and increasing productivity. Additionally, GMS provides a 360-degree view of customers, enabling garages to personalize interactions, build strong relationships, and drive loyalty. By leveraging Salesforce's advanced analytics and reporting capabilities, garages can gain valuable insights into customer behavior, sales performance, and marketing effectiveness, making data-driven decisions to drive business growth.

The GMS has been instrumental in helping automotive repair facilities stay ahead of the competition. By providing a seamless and satisfying experience for customers and staff, garages can build a positive reputation, attract new business, and increase revenue. As the automotive industry continues to evolve, Salesforce's GMS remains a vital tool for garages looking to thrive in a competitive market. With its cutting-edge technology and customer-centric approach, Salesforce is revolutionizing the way automotive businesses operate, enabling them to deliver exceptional service, drive growth, and succeed in an ever-changing landscape.

INDEX

TABLE OF CONTENTS

GARAGE MANAGEMENT SYSTEM

What is the Garage Management System (GMS)?

The GMS is a comprehensive software solution designed specifically for automotive repair facilities. Its primary goal is to help garages deliver exceptional service, streamline operations, and foster long-term customer relationships.

Key Benefits of GMS:

1. **Deliver Top-Notch Service**: GMS enables garages to provide an outstanding customer experience by streamlining service workflows, automating tasks, and ensuring that all customer interactions are prompt and professional.

2. **Increase Operational Efficiency**: By automating routine tasks, optimizing workflows, and providing real-time visibility into operations, GMS helps garages reduce waste, minimize errors, and maximize productivity.

3. **Build Lasting Customer Relationships**: GMS empowers garages to build strong, lasting relationships with their customers by providing a seamless and satisfying experience. This includes automated reminders, personalized communication, and transparent service updates.

How Does GMS Achieve These Benefits?

1. **User-Friendly Interface**: GMS features an intuitive and user-friendly interface that makes it easy for staff to navigate and use, reducing the learning curve and minimizing errors.

2. **Powerful Features**: GMS offers a range of powerful features, including:

a. Automated workflows and task management

b. Real-time reporting and analytics

c. Customer relationship management (CRM) tools

d. Inventory management and control

e. Integration with other business systems (e.g., accounting, marketing)

Competitive Advantage:

By implementing GMS, garages can gain a competitive edge in the market by:

1. **Improving Customer Satisfaction**: Delivering exceptional service and building strong relationships with customers leads to increased loyalty and retention.

2. **Increasing Efficiency**: Streamlining operations and reducing waste enables garages to reduce costs, increase productivity, and improve profitability.

3. **Enhancing Reputation**: Providing a seamless and satisfying experience for customers and staff helps to build a positive reputation and attract new business.

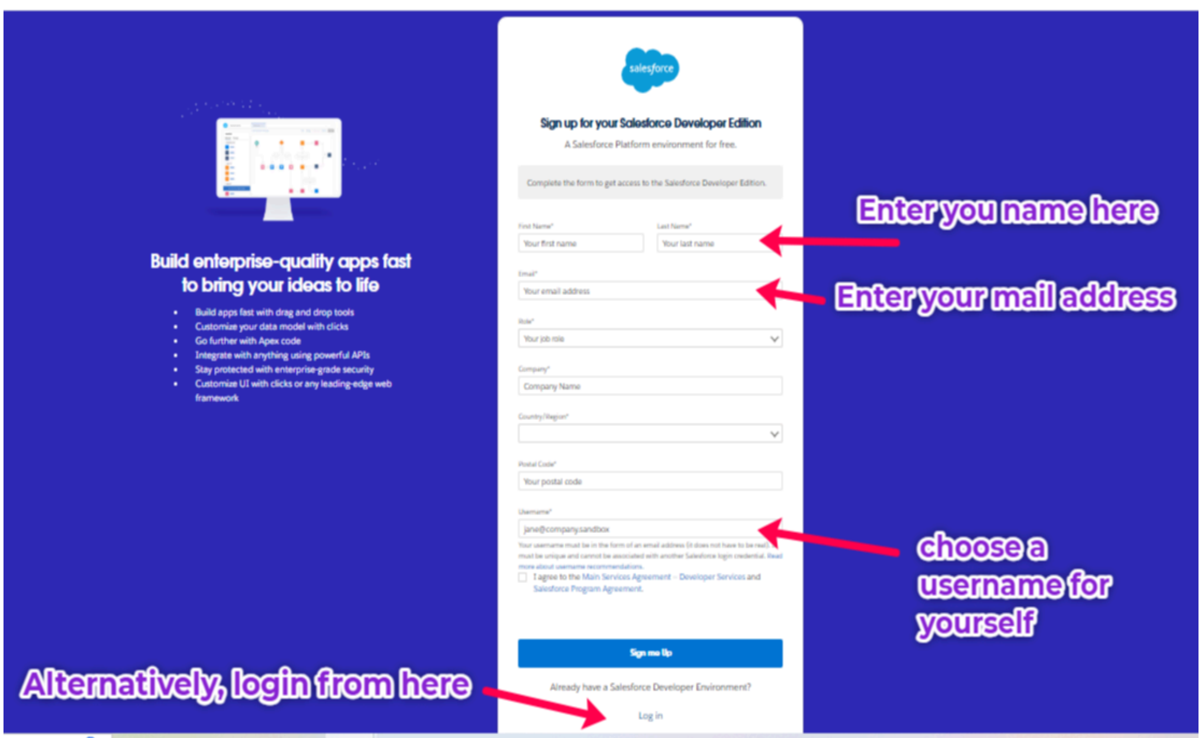
In summary, the Garage Management System is a powerful tool that helps automotive repair facilities deliver top-notch service, increase operational efficiency, and build lasting customer relationships. Its user-friendly interface and powerful features make it an essential solution for garages looking to thrive in a competitive market.

Creating Developer Account:

1. Go to https://developer.salesforce.com/signup
2. On the sign up form, enter the following details :
    a. First name & Last name
    b. Email
    c. Role : Developer
    d. Company : College Name
    e. County : India
    f. Postal Code : your area pin code
    g. 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)

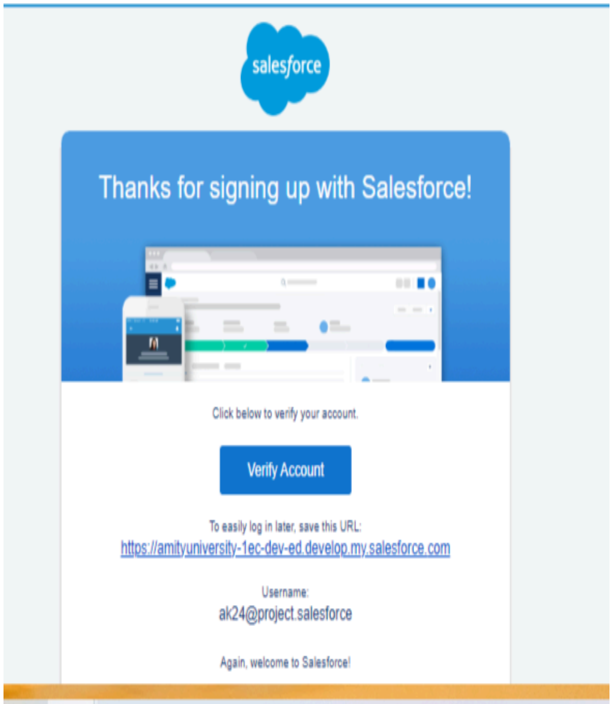
Click on sign me up after filling all the details

3. Alternately, you can even login to your already existing developer account.

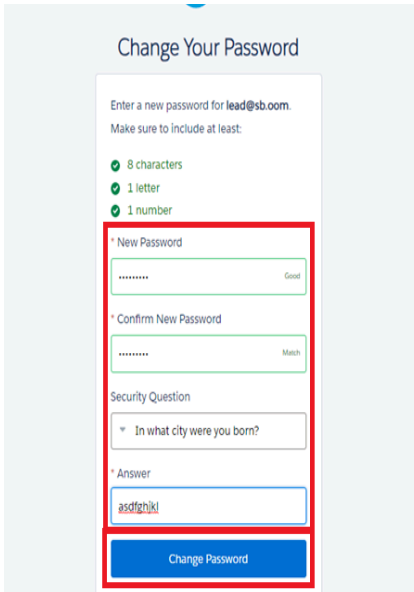


Account Activation

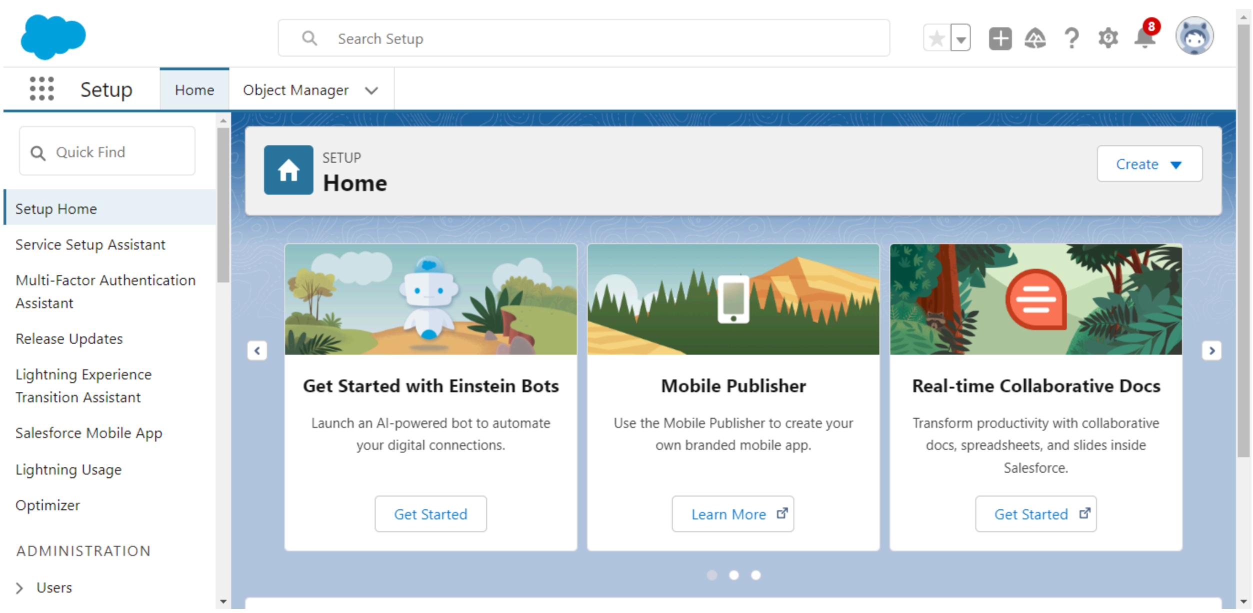
(if you have created a new developer account)



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 might take 5-10mins.
2. Click on Verify Account
3. Enter a password and answer a security question.
4. Click on change password.



5. Then you will be redirected to your salesforce setup page.



Starting with the SALESFORCE platform:

To get started with building the GMS you'll start by building the following:

1. Objects
2. Tabs
3. The Lightning App
4. Fields
5. Validation Rules
6. Duplicate Rules
7. Profiles
8. Roles & Role Hierarchy
9. Users
10. Public Groups
11. Sharing Settings
12. Flows
13. Apex Triggers
14. Reports
15. Dashboards

Let's dive into each one of them and see how the platform is built bit by bit.

CHAPTER-1

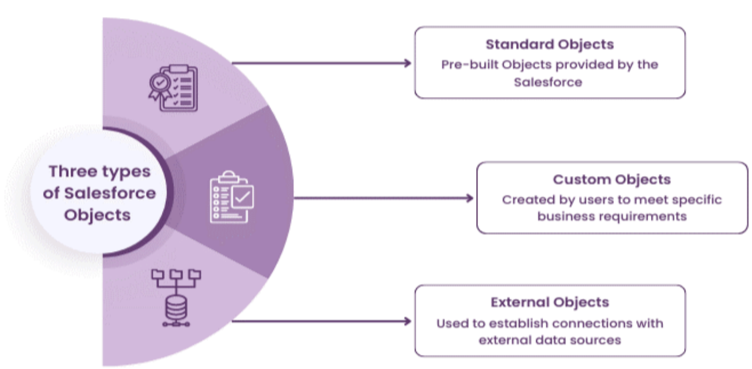
CREATION OF AN OBJECT

What Is an Object?

Salesforce objects are database tables that permit you to store data that is specific to an organization. Objects are the way you store your records in Salesforce. They are basically a container for spreadsheet data, account details, customer contacts, location and regional info, tracking status, and more, but are easier on the eyes and more customizable than a database.

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.
3. **External Objects**: External Objects enable integration with external data sources, accessing and interacting with data stored outside of Salesforce.



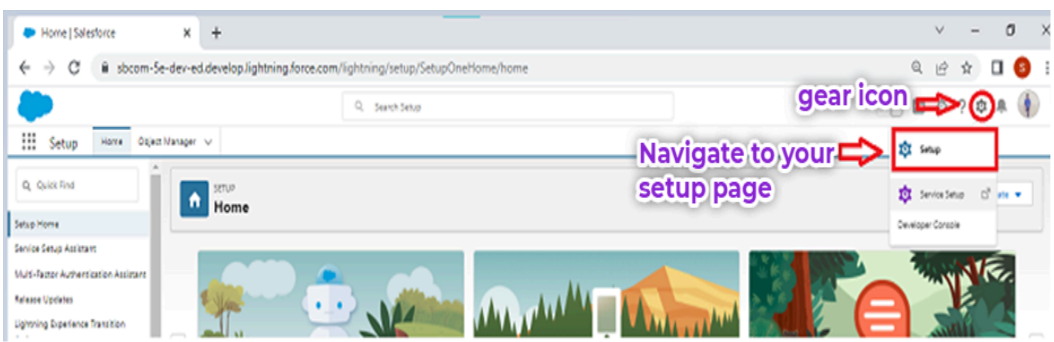
How to create an Object in salesforce ??

Object can be created in 3 ways in the Salesforce:

1. Create >> Custom Object
2. Create >> Custom Object from Spreadsheet
3. Schema Builder

Let's START !! From the home page navigate by following the instructions:

1. Navigate to Setup page:
2. Click on gear icon.
3. Then click setup.



To create an object:

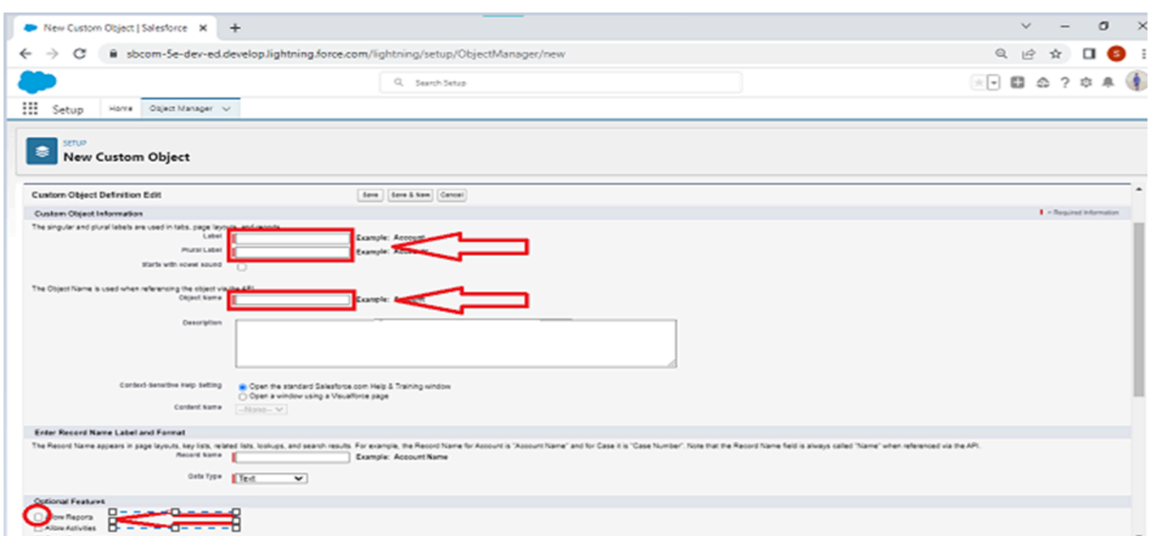
1. From the setup page
2. Click on Object Manager

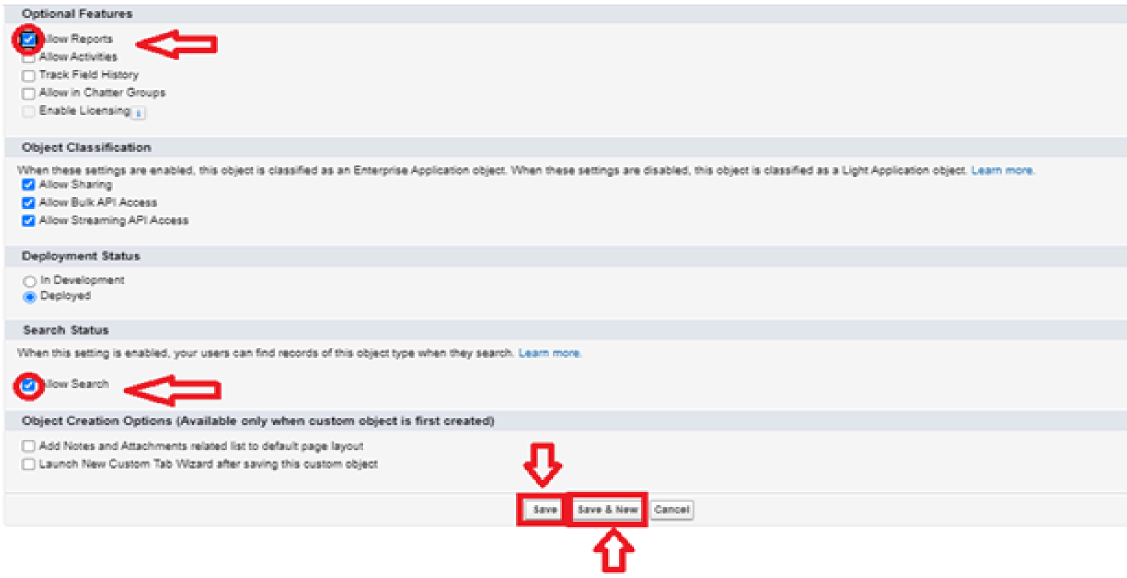
3. Click on Create
4. Click on Custom Object.

5. On Custom object defining page:
    a. Enter the label name
    b. Plural label name

Notice that the Object Name and Record Name fields auto-fill.

    c. Click on Allow reports
    d. Allow search.





6. Now, click on Save

We just learnt how to create a custom object in Salesforce. Now, we will follow the above steps to create the objects for our GMS.

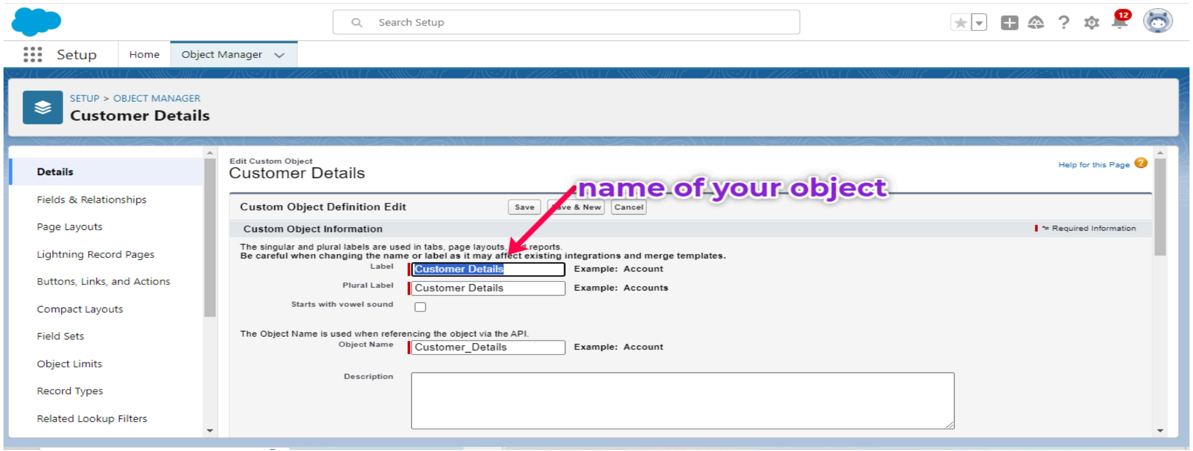
1. Create Customer Details Object
2. Create Appointment Object
3. Create Service Records Object
4. Create Billing Details and Feedback Object

Creating each of these objects and filling the appropriate details.

➤ Creating the Customer Details Object

To create an object:

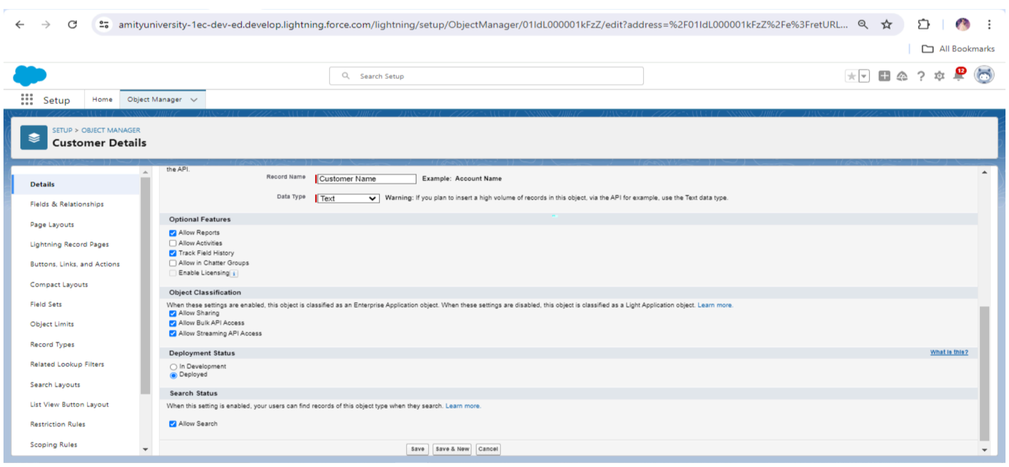
1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
    a. Enter the label name >> Customer Details
    b. Plural label name >> Customer Details



c. 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.

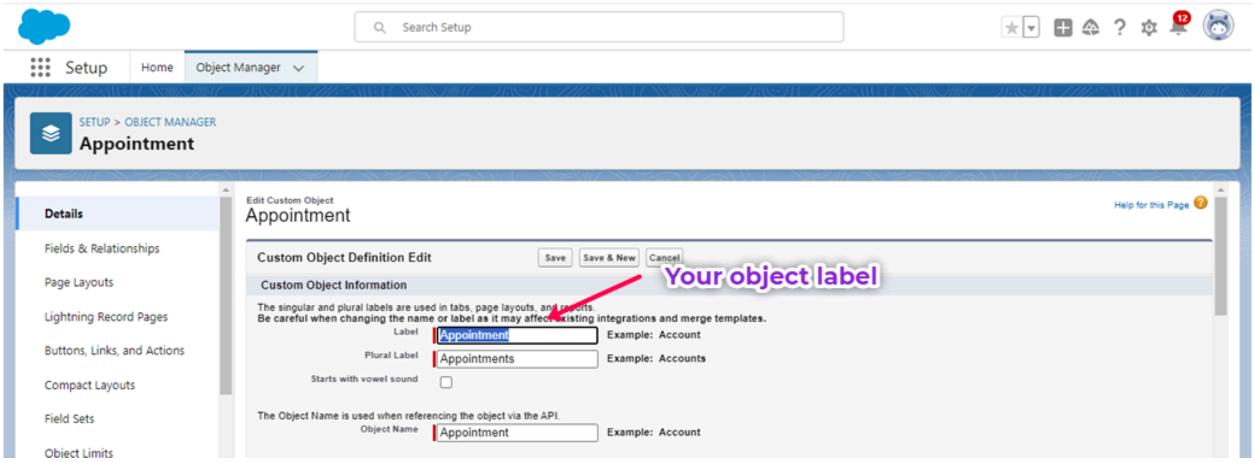


Your new Customer Details object just got created.

➤ Create the Appointment Object

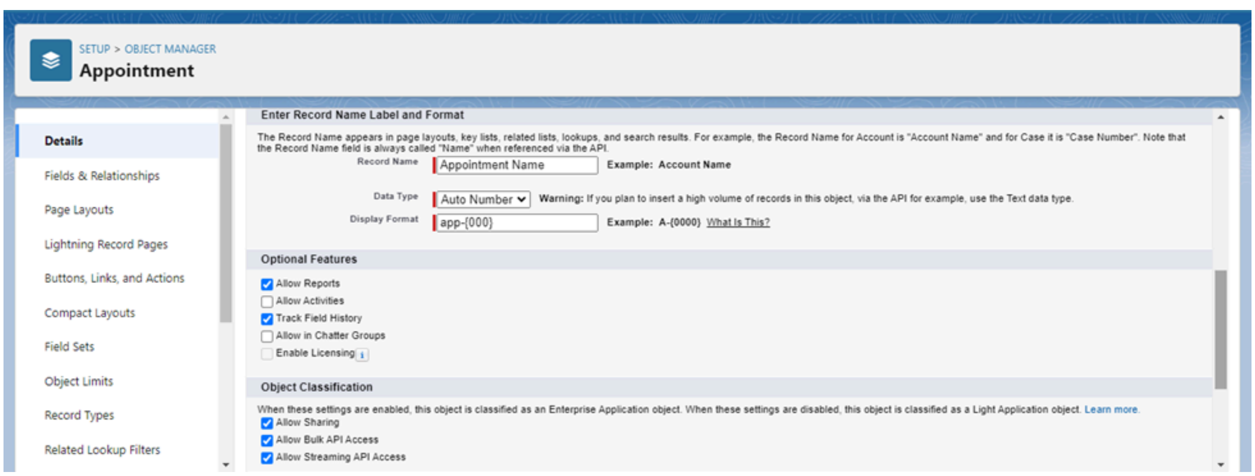
To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
   a. Enter the label name >> Appointment
   b. Plural label name >> Appointments



c. 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,

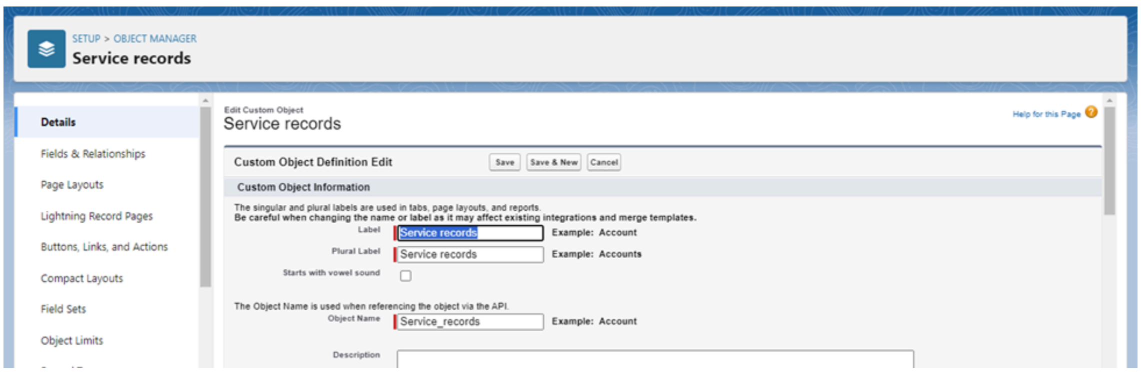
3. Allow search >> Save

Your new Appointment object just got created.

➤Create Service Records Object

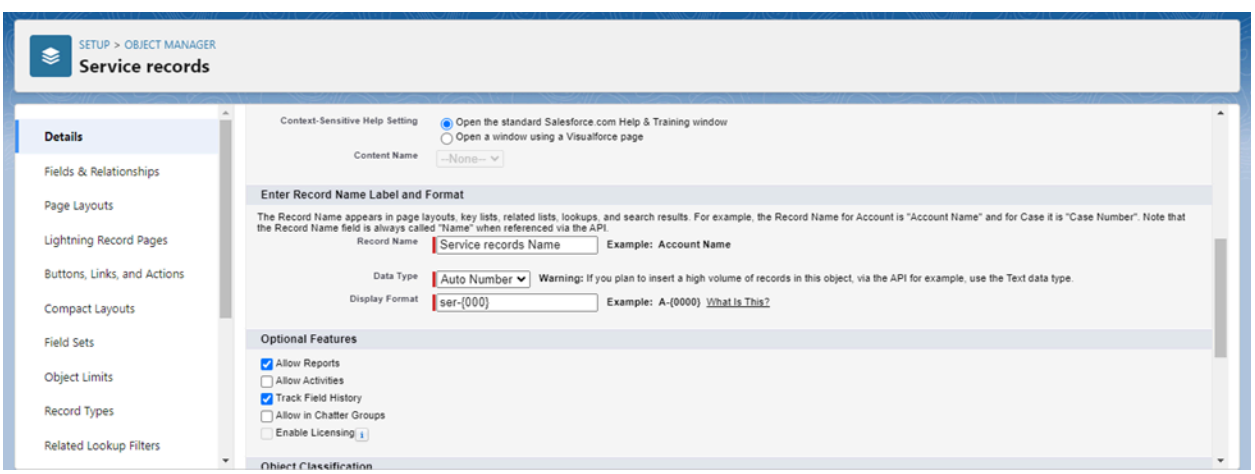
To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
    a. Enter the label name >> Service records
    b. Plural label name >> Service records



c. Enter Record Name Label and Format

- Record Name >>Service records Name
- Data Type >> Auto Number
- Display Format >> ser-{000}
- Starting number >> 1



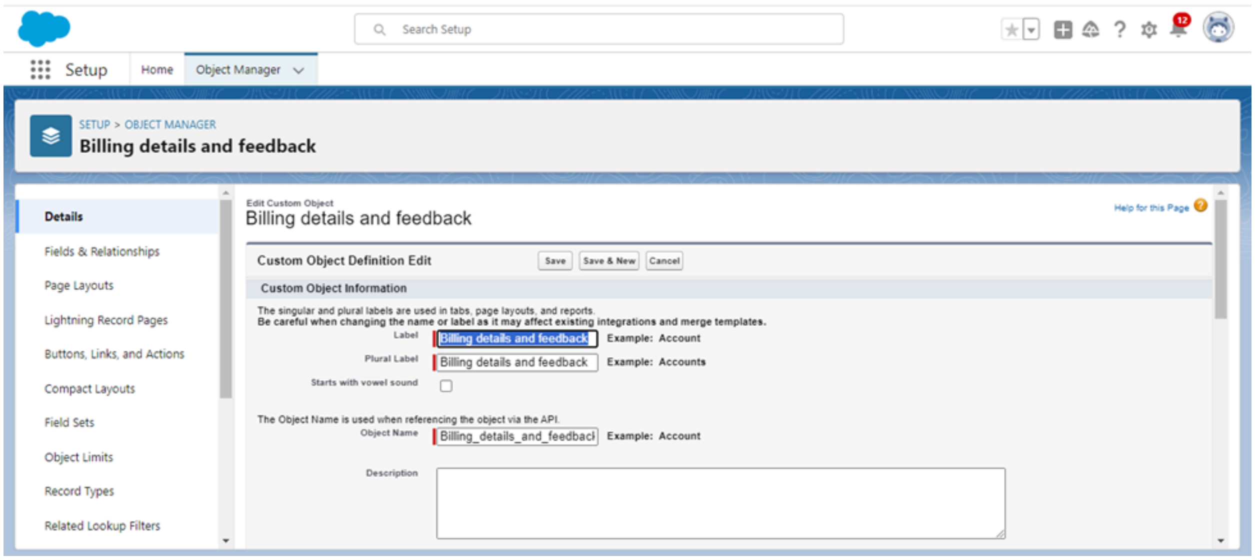
2. Click on Allow reports and Track Field History,
3. Allow search >> Save.

Your new Service Records object just got created.

➤ Create Billing Details and Feedback Object

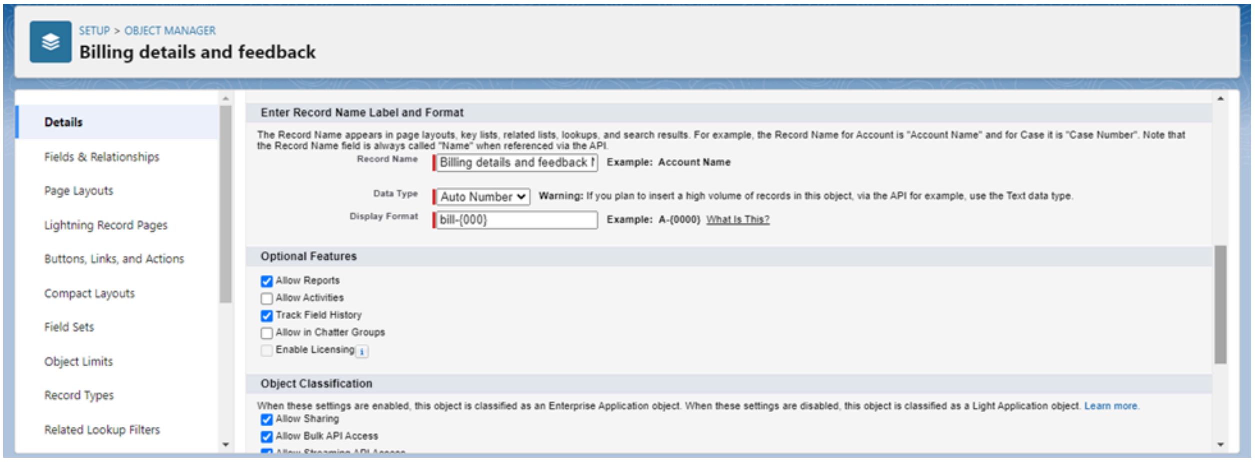
To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
    a. Enter the label name >> Billing details and feedback
    b. Plural label name >> Billing details and feedback



c. Enter Record Name Label and Format

- Record Name >> Billing details and feedback Name
- Data Type >> Auto Number
- Display Format >> bill-{000}
- Starting number >> 1



2. Click on Allow reports and Track Field History,

3. Allow search >> Save.

Your new Billing Details & Feedback object just got created.

CHAPTER-2

CREATING 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**

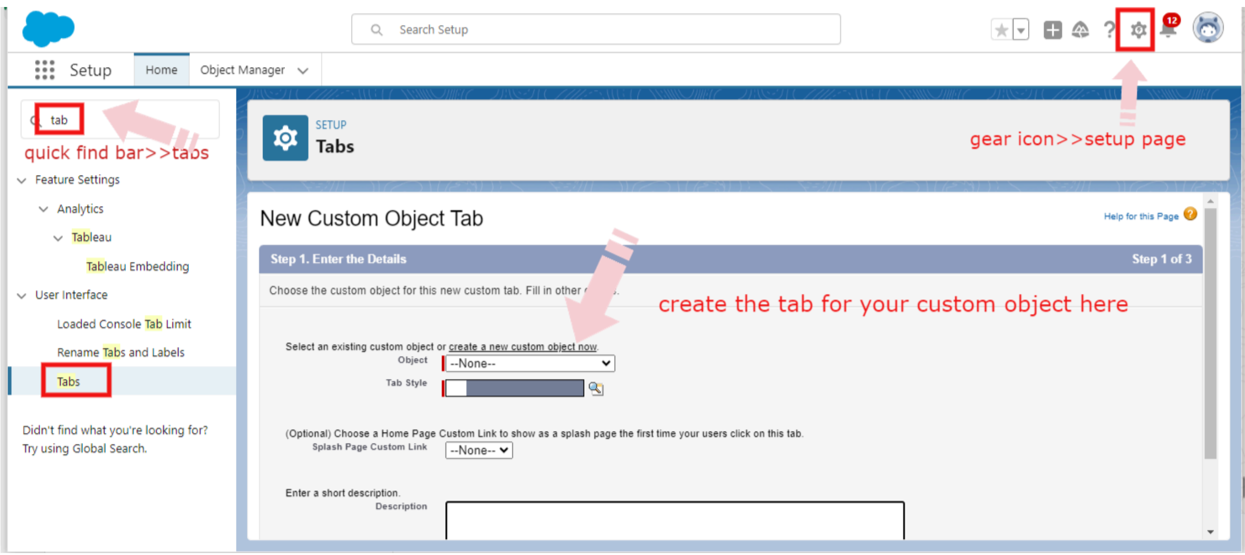
   Lightning Page Tabs let you add Lightning Pages to the mobile app navigation menu.

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.

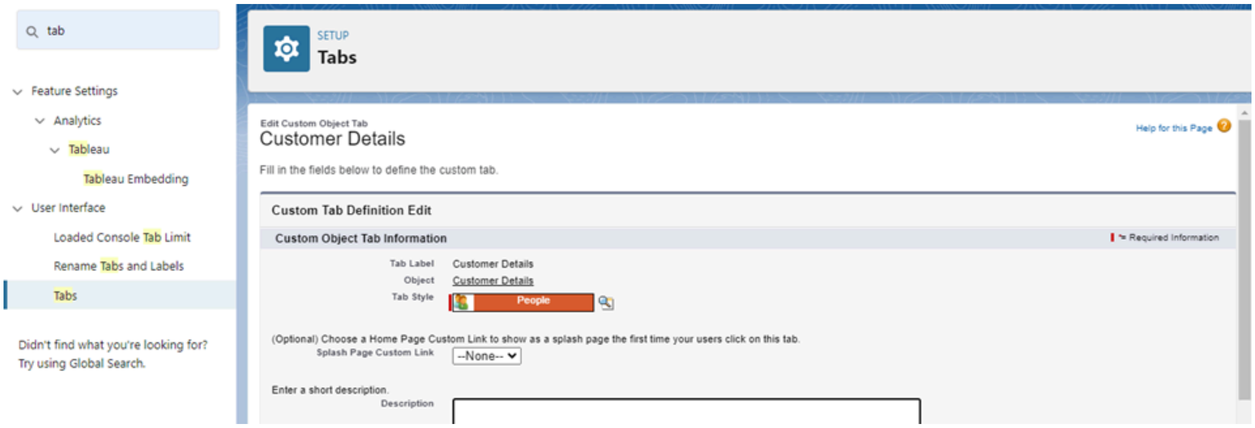
➤ Creating Custom Tab

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)

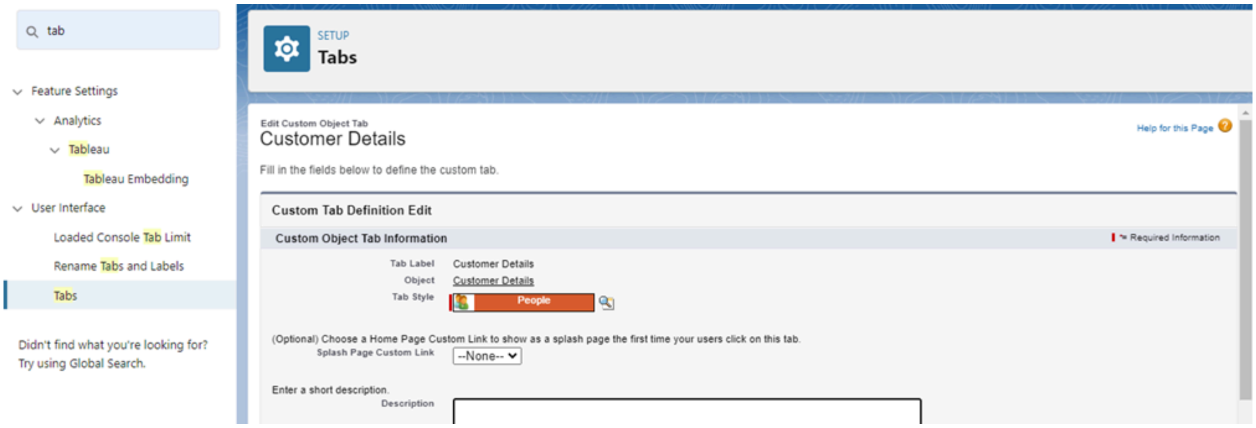


2. Select Object(Customer Details) >> Select the tab style >> Next (Add to profiles page) keep it as default >> Next (Add to Custom App) uncheck the include tab .



Tab Style: Selecting your own style for the tab

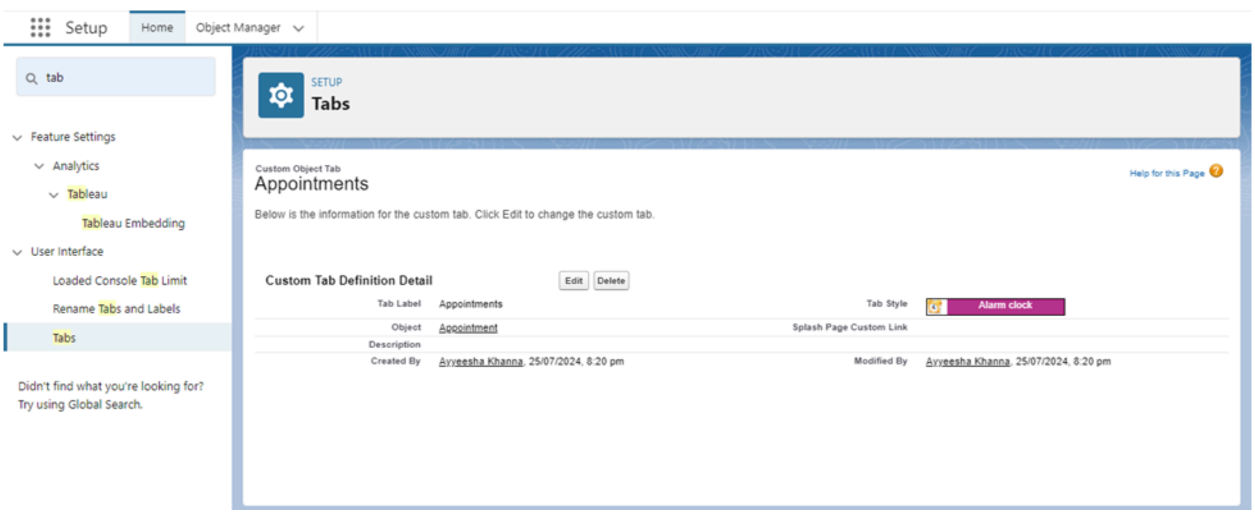
3. Make sure that the Append tab to users' existing personal customizations is checked.
4. Click save.



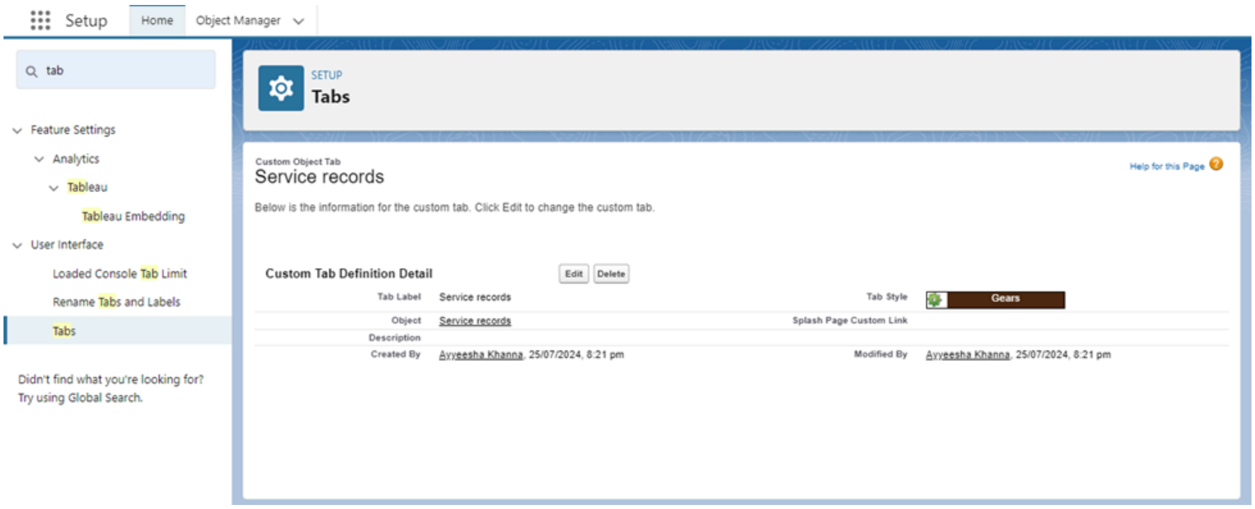
➢ Creating Remaining Tabs

1. Now create the Tabs for the remaining Objects, they are "Appointments, Service records, Billing details and feedback".
2. We need to follow the same steps as mentioned in above, while creating Customer Detail Tab.

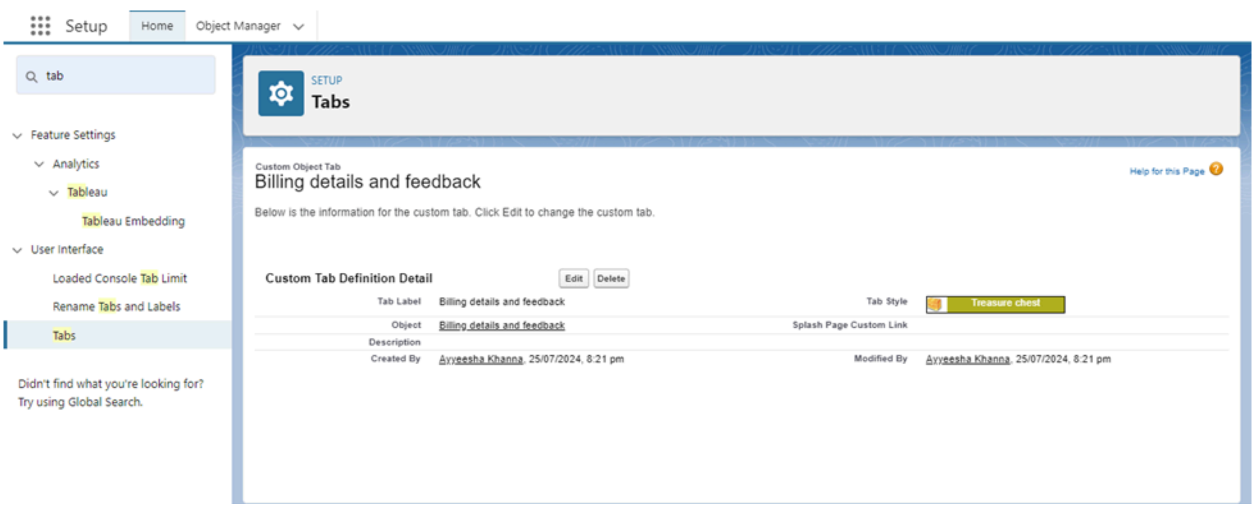
➢ Appointment Tab



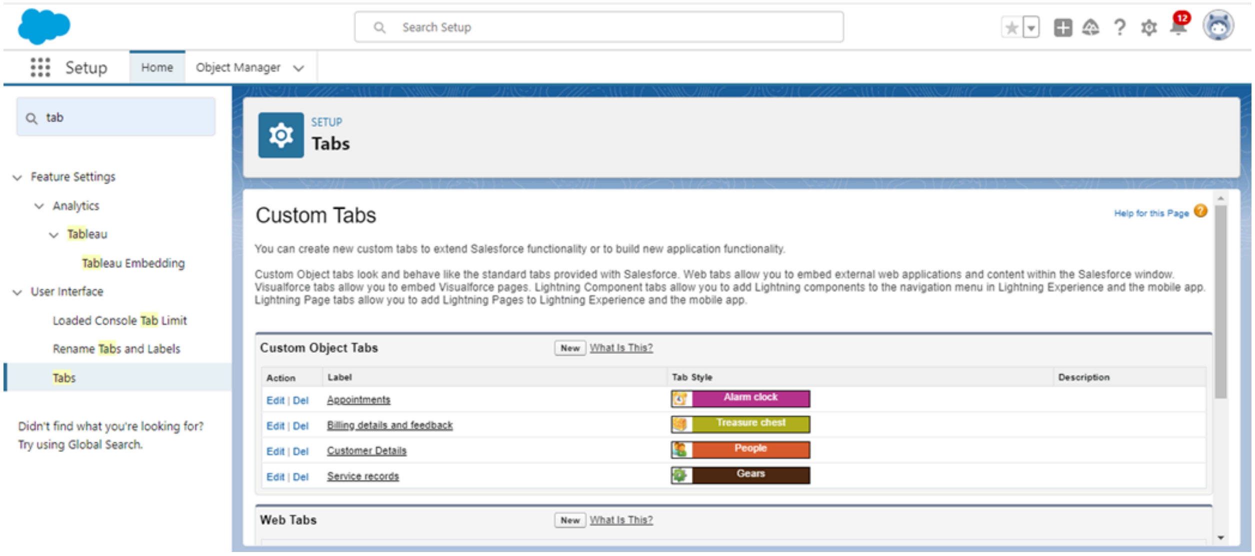
➤ Service Record Tab



➤ Billing Details & Feedback Tab



You have successfully created your tabs for all your custom objects


Search Setup
Setup
Home
Object Manager
tab
Feature Settings
Analytics
Tableau
Tableau Embedding
User Interface
Loaded Console Tab Limit
Rename Tabs and Labels
Tabs
Didn't find what you're looking for?
Try using Global Search.
SETUP
Tabs
Custom Tabs
Help for this Page
You can create new custom tabs to extend Salesforce functionality or to build new application functionality.
Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external web applications and content within the Salesforce window. Visualforce tabs allow you to embed Visualforce pages. Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app. Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.
Custom Object Tabs
New
What Is This?
Action
Label
Tab Style
Description
Edit | Del
Appointments
Alarm clock
Edit | Del
Billing details and feedback
Treasure chest
Edit | Del
Customer Details
People
Edit | Del
Service records
Gears
Web Tabs
New
What Is This?

CHAPTER-3

CREATING THE LIGHTNING APP

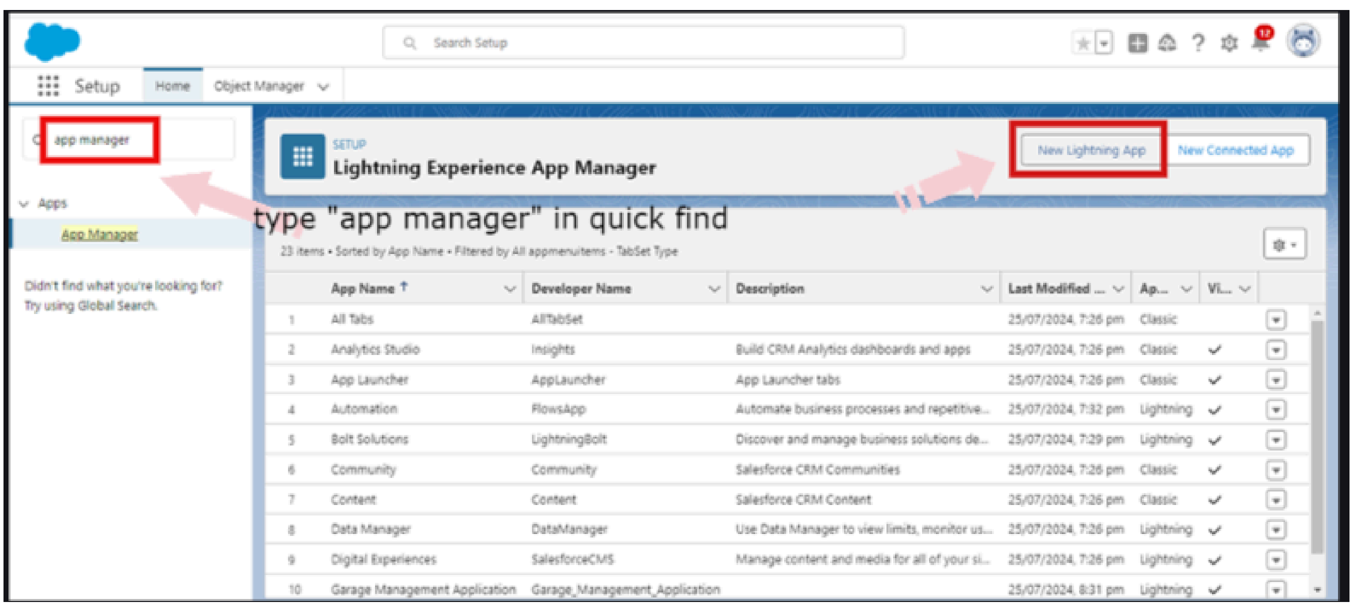
The Lightning App

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.

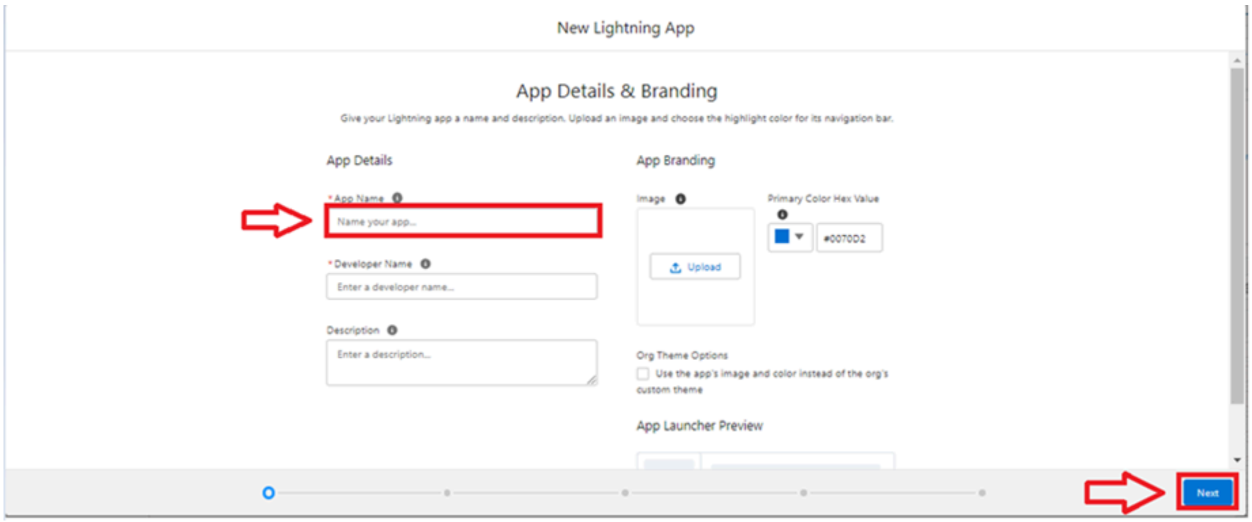
➤ Create a Lightning App

To create a lightning app page:

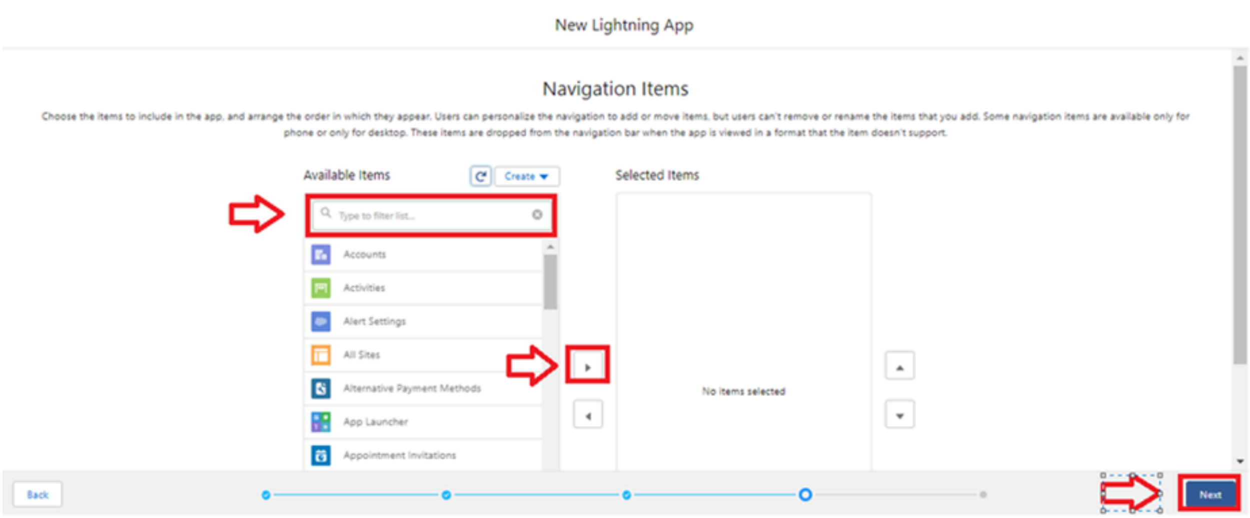
1. Go to setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.



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.

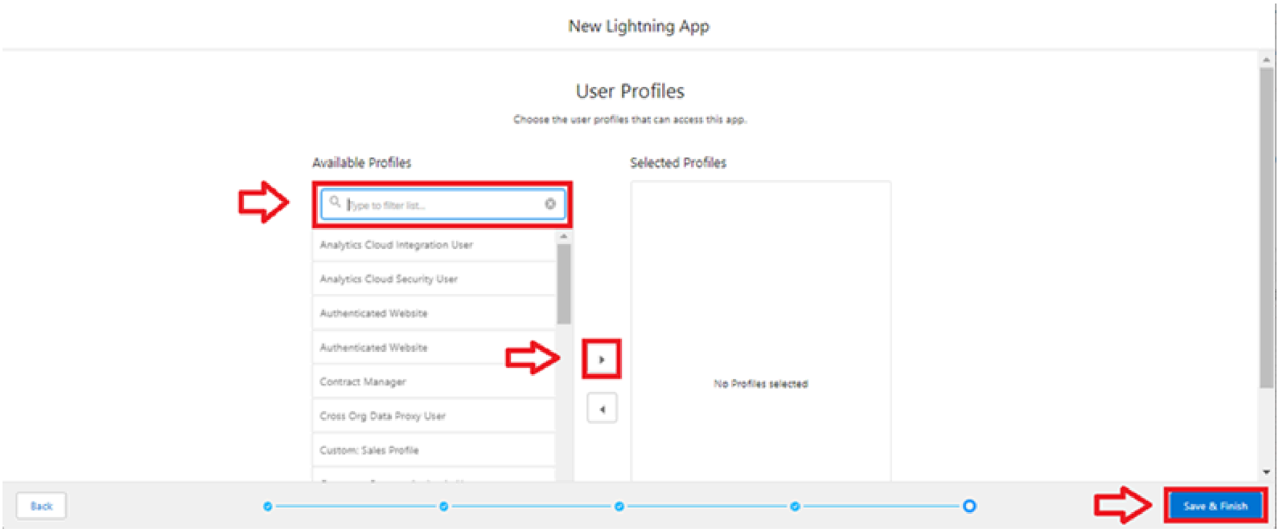


3. To Add Navigation Items:



4. 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.

5. To Add User Profiles:



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

CHAPTER- 4

CREATING FIELDS

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

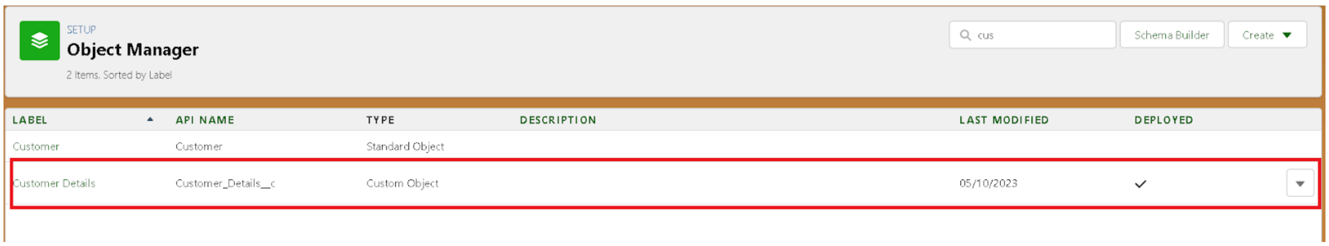
Custom Fields:

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.

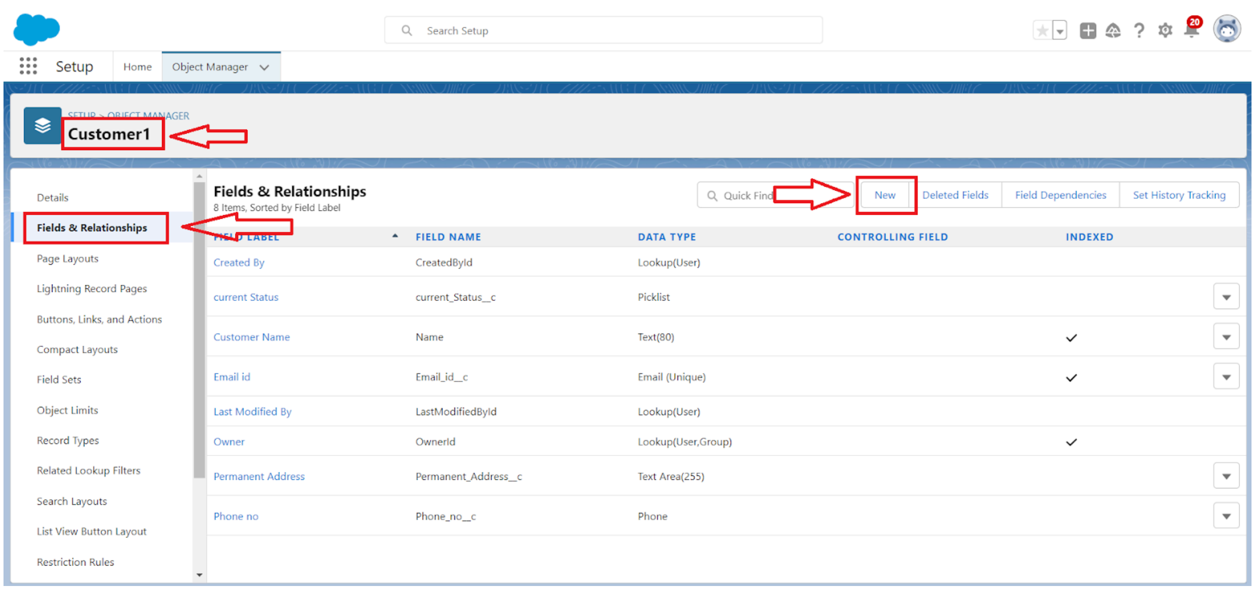
➤ Creation Of Fields For The Customer Details Object

To create fields in an object:

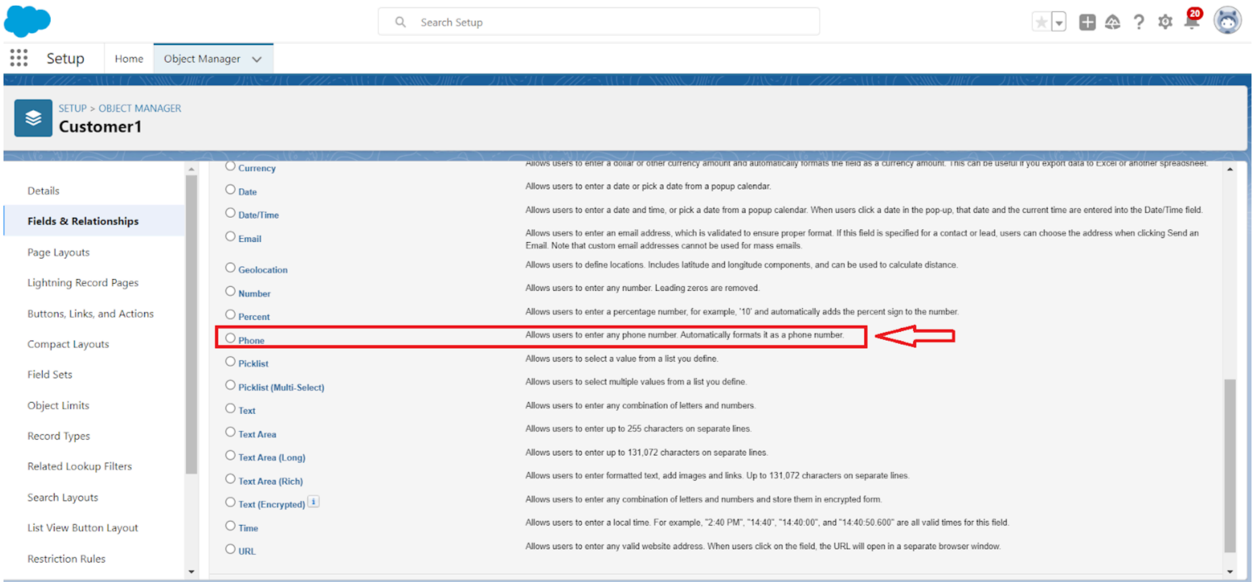
1. Go to setup >> click on Object Manager >> type object name(Customer Details) in search bar >> click on the object.



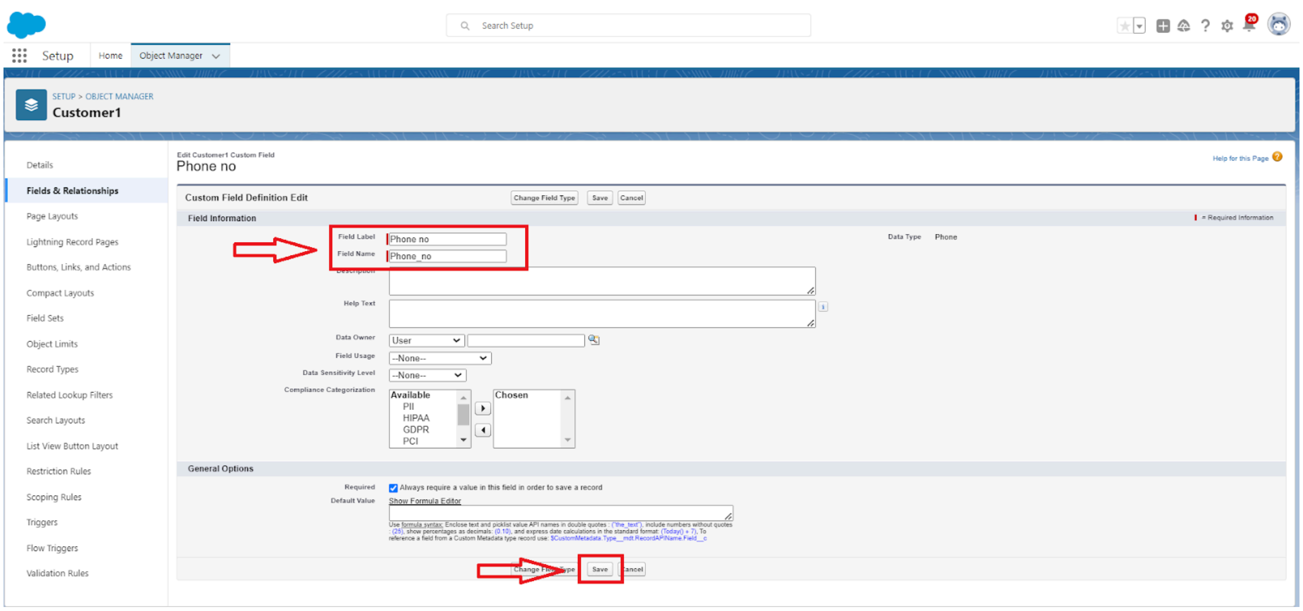
2. Now click on “Fields & Relationships” >> New



3. Select Data Type as a “Phone”



4. Click on next.



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.

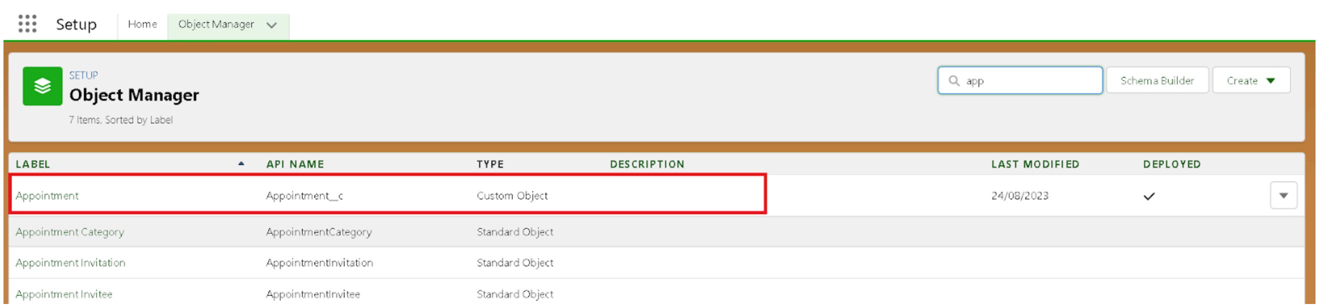
2. To create another fields in an object:

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.

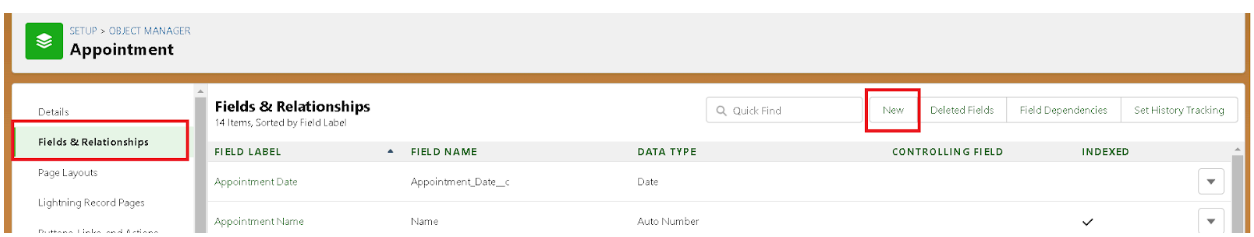
Creation Of Lookup Fields

➢ Creation of Lookup Field on Appointment Object :

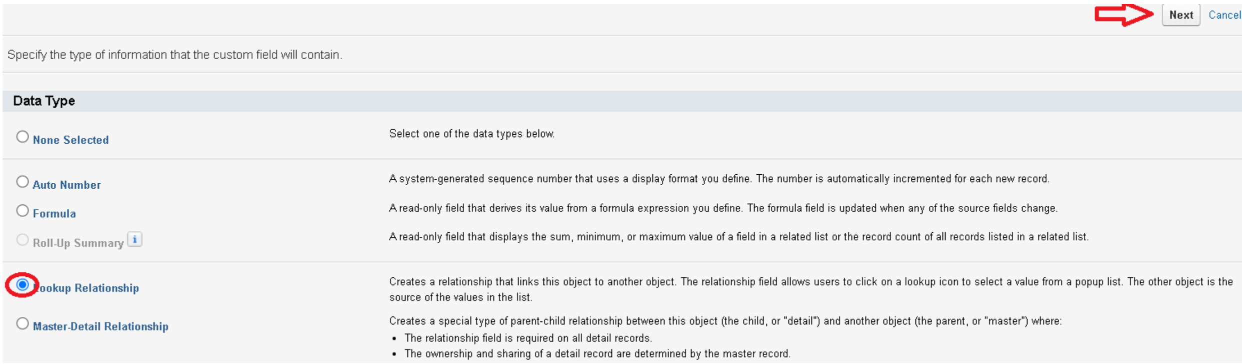
1. Go to setup >> click on Object Manager >> type object name (Appointment) in the search bar >> click on the object.

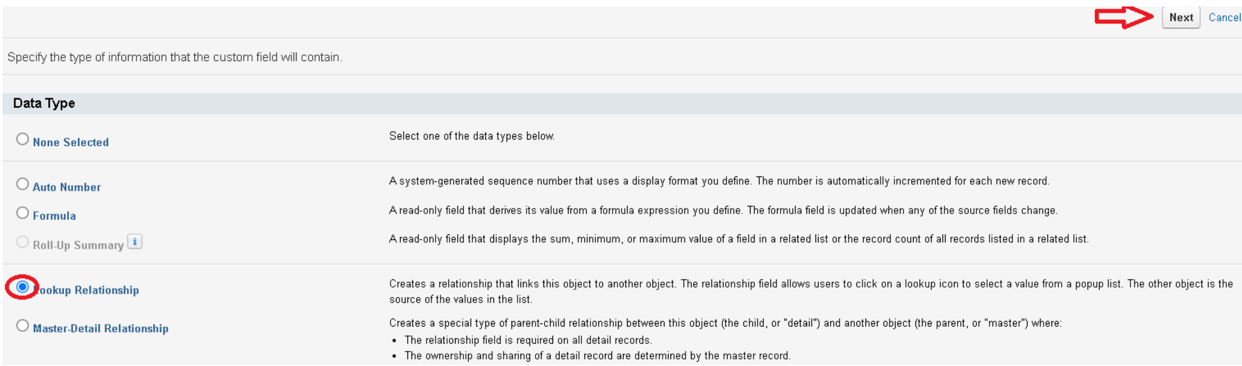


2. Now click on “Fields & Relationships” >> New



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



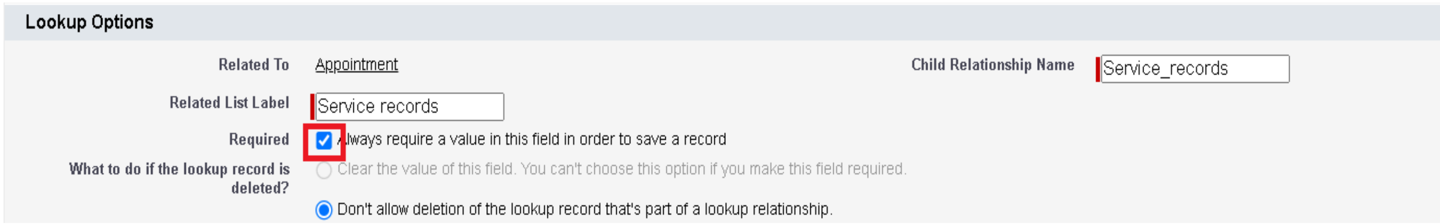


4. Select the related object “ Customer Details” and click next.
5. Next >> Next >> Save.

Note: Make sure you complete Activity 4 Before continuing.

➤ Creation of Lookup Field on Service records Object :

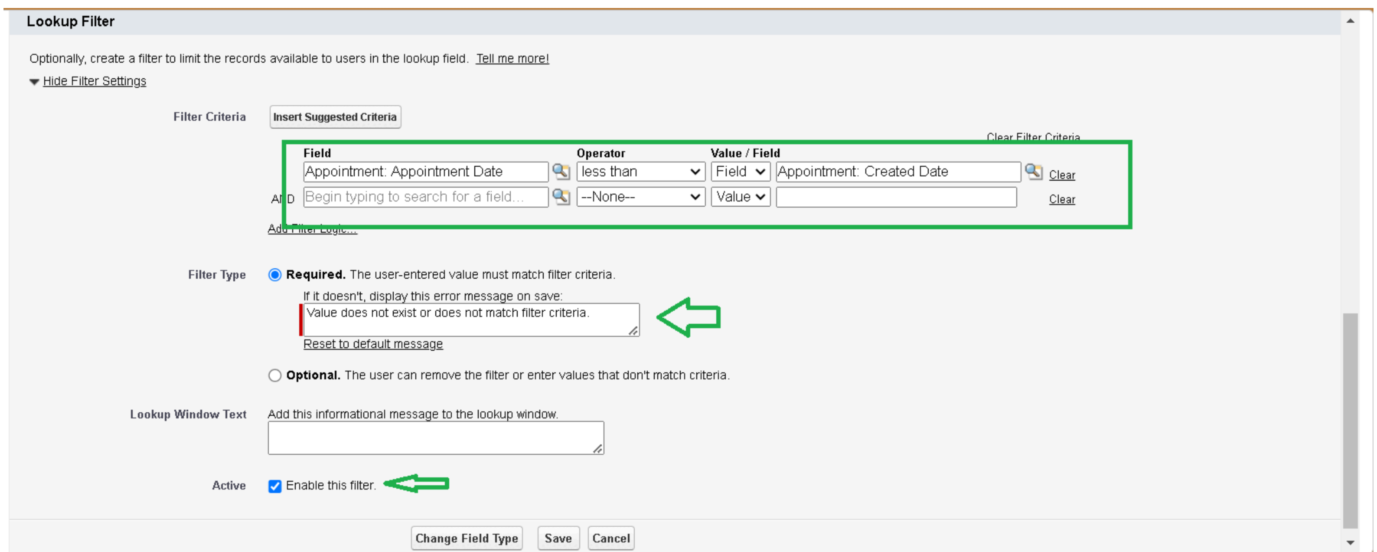
1. Go to setup >> click on Object Manager >> type object name(Service records) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select “Look-up relationship” as data type and click Next.
4. Select the related object “ Appointment ” and click next.
5. Make it a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

Note:- You'll have to create a field with data type as 'date" for the Appointment Date

9. Filter type should be Required.

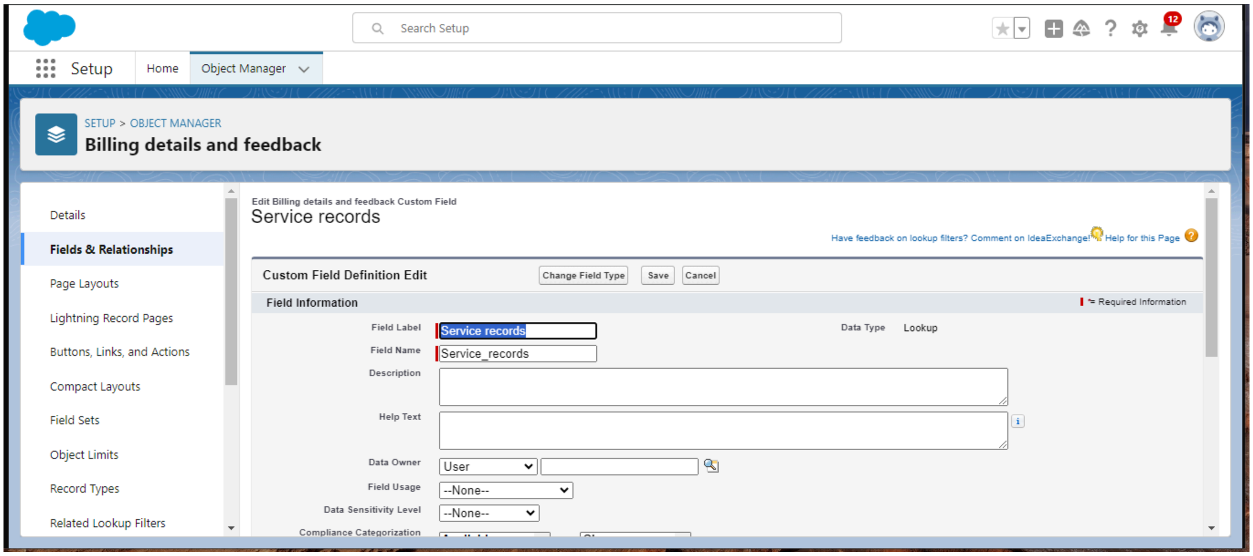


10. Error Message : Value does not match the criteria.
11. Enable the filter by click on Active.
12. Next >> Next >> Save.

➢ 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.

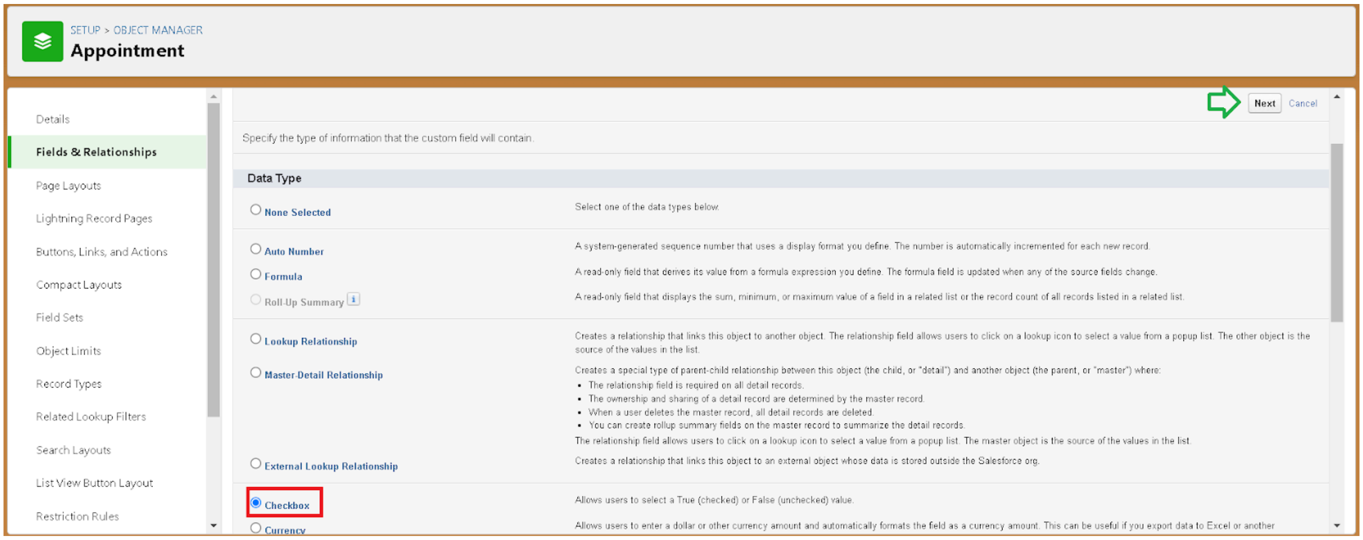
2. Now click on “Fields & Relationships” >> New.
3. Select “Look-up relationship” as data type and click Next.
4. Select the related object “ Service records” and click next.
5. Next >> Next >> Save & new.



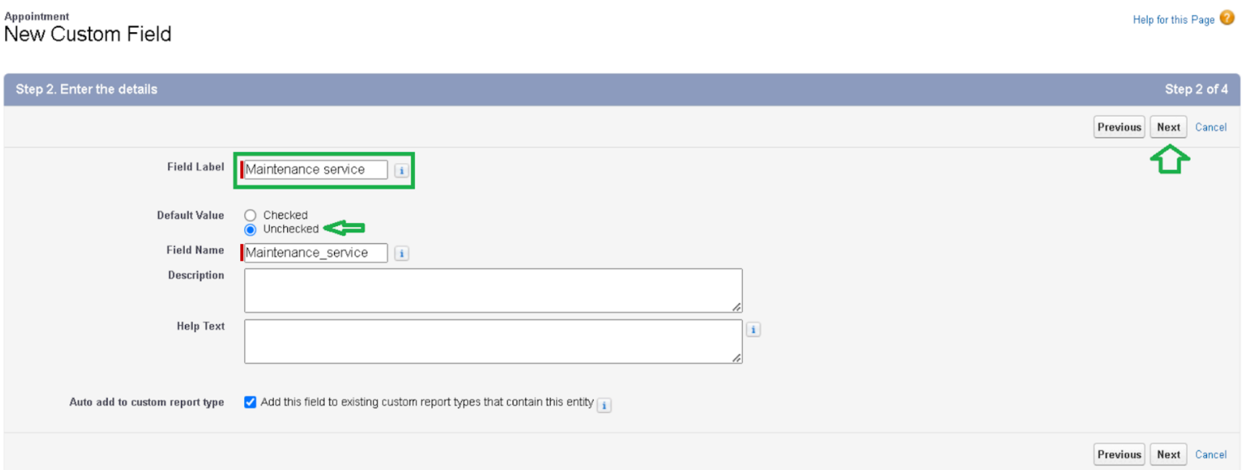
Creation Of Checkbox Fields

➢ 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.



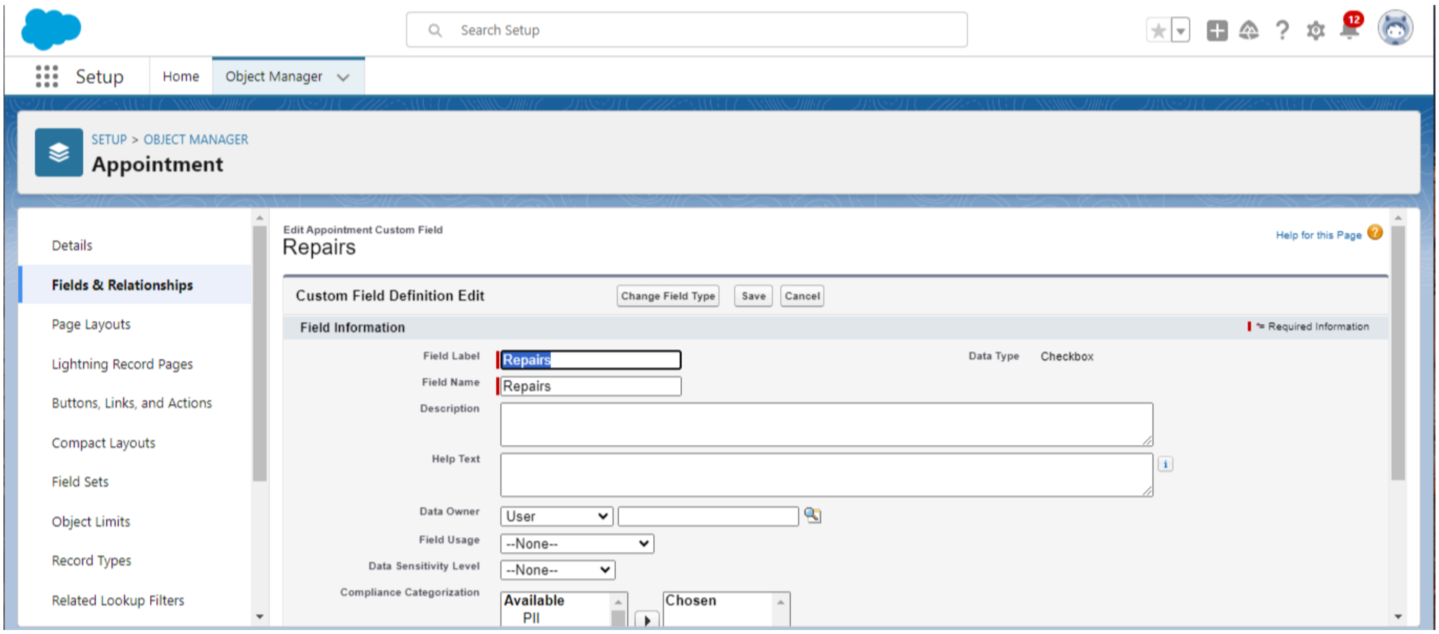
4. Give the Field Label : Maintenance service
5. Field Name : is auto populated
6. Default value : unchecked



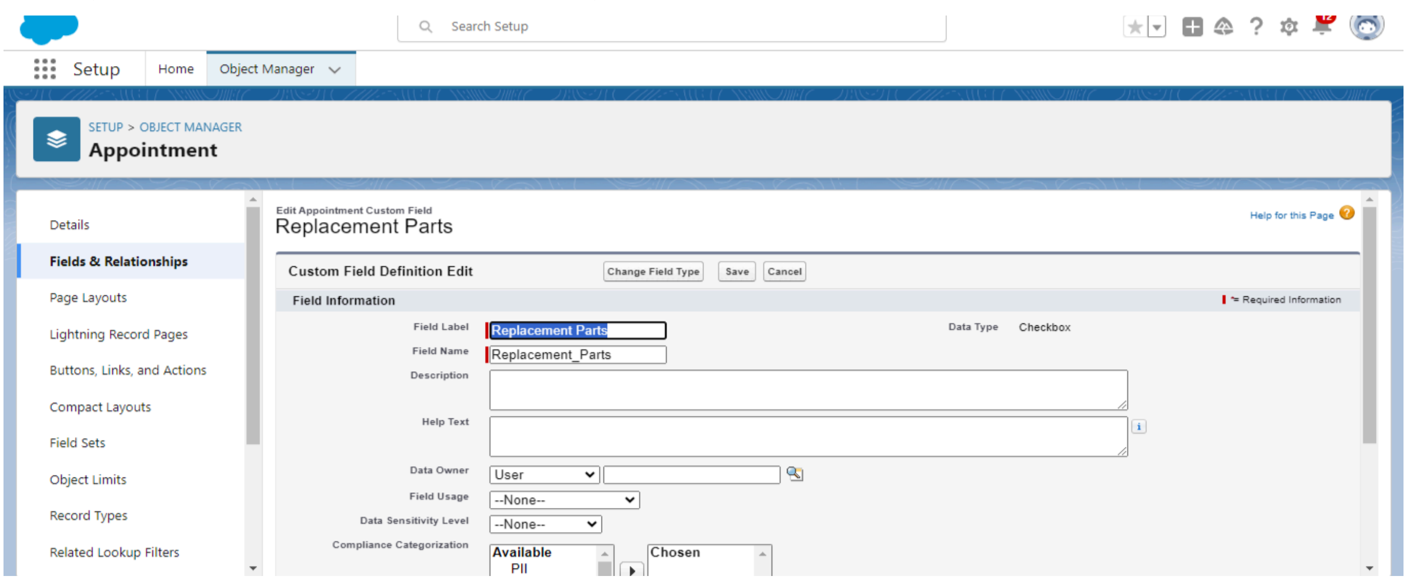
7. Click on next >> next >> save.

➤ Creation of Another Checkbox Field on Appointment Object :

1. Repeat the steps form 1 to 3.
2. Give the Field Label : Repairs
3. Field Nme : is auto populated
4. Default value : unchecked
5. Click on next >> next >> save

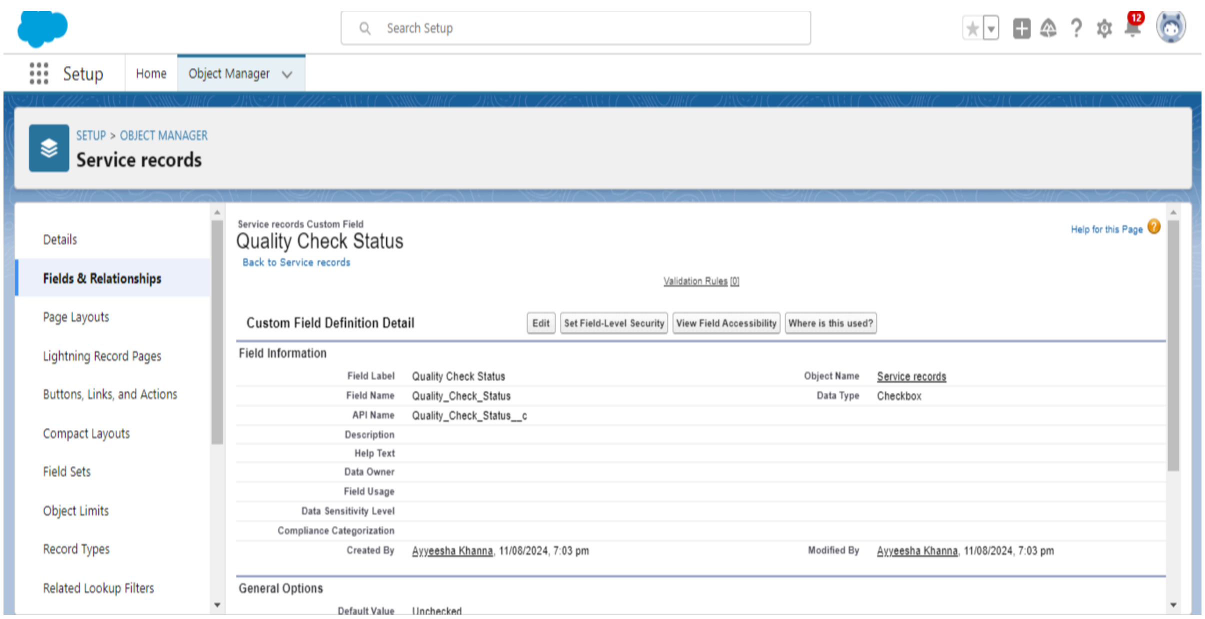


6. Follow the same and create another checkbox with given names
7. Give the Field Label : Replacement Parts
8. Field Nme : is auto populated
9. Default value : unchecked
10. Click on next >> next >> save.



➢ 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



Creation of Date Field

➢ Creation of Date Field on Appointment Object :

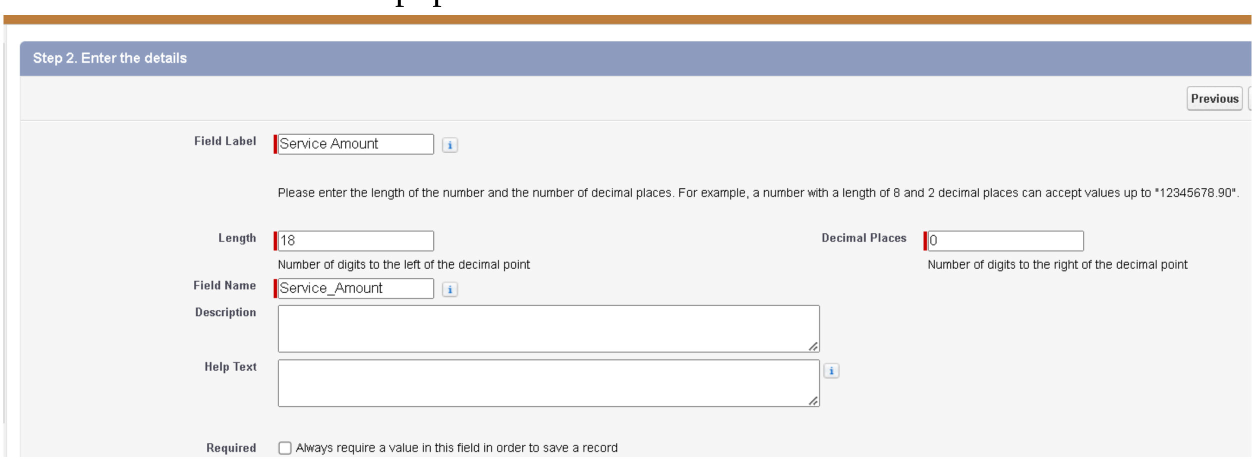
1. Go to setup >> click on Object Manager >> type object name(Appointment) in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New.
3. Select “Date” as data type and click Next.
4. Give the Field Label : Appointment Date

5. Field Nme : is auto populated
6. Make it as a Required field by click on the Required option.
7. Click on next >> save

Creation of Currency Field

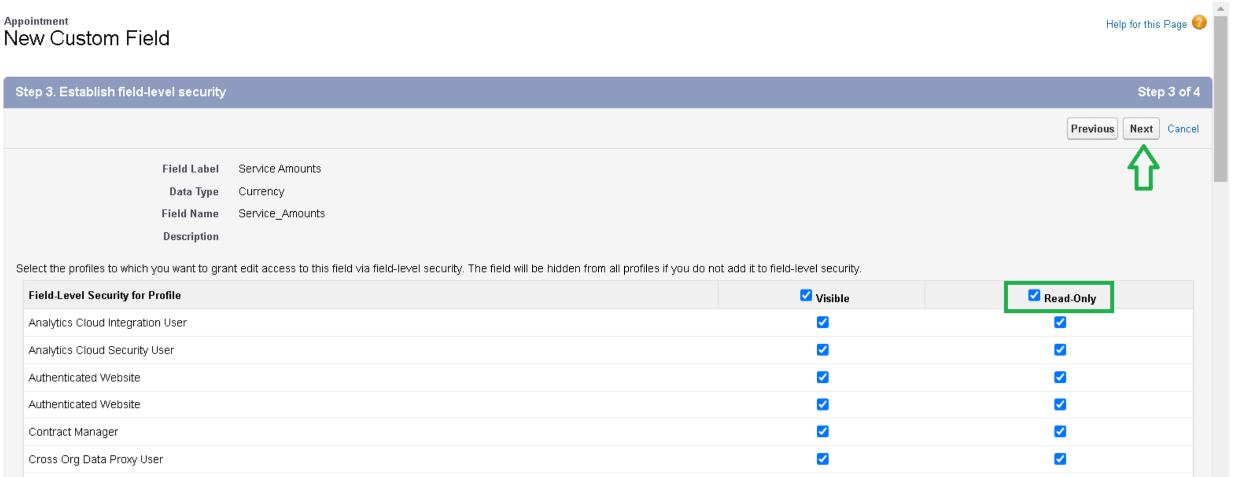
➤ Creation ofCurrency Field on Appointment Object :

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



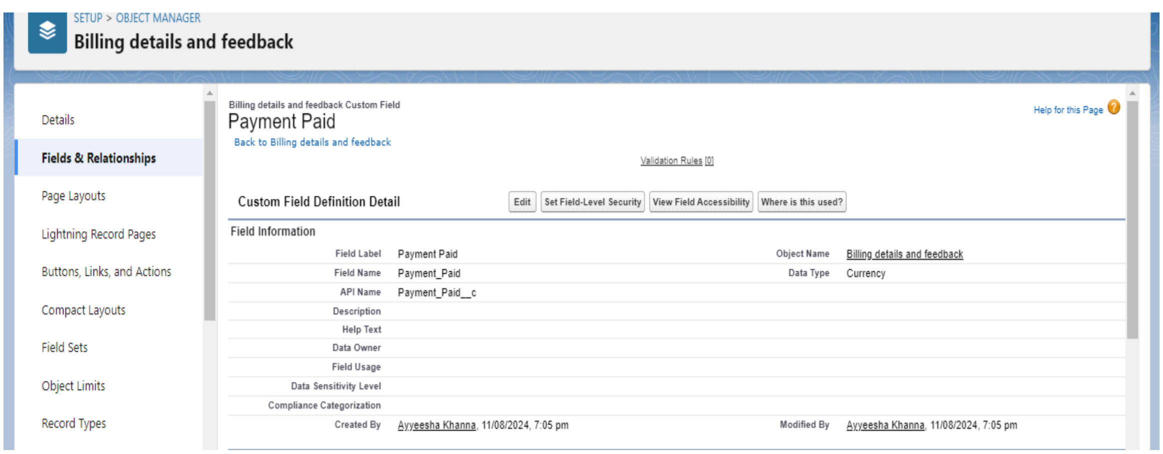
6. Click on next

7. Give read only for all the profiles in field level security for profile.



➢Creation of Currency Field on Billing details and feedback Object :

1. Follow the same steps as mentioned above inBilling details and feedbackObject.
2. Change the label name as mentioned.
3. Give the Field Label : Payment Paid
4. Field Nme : is auto populated



Creation of Text Field

➢ Creation of Text Field on Appointment Object

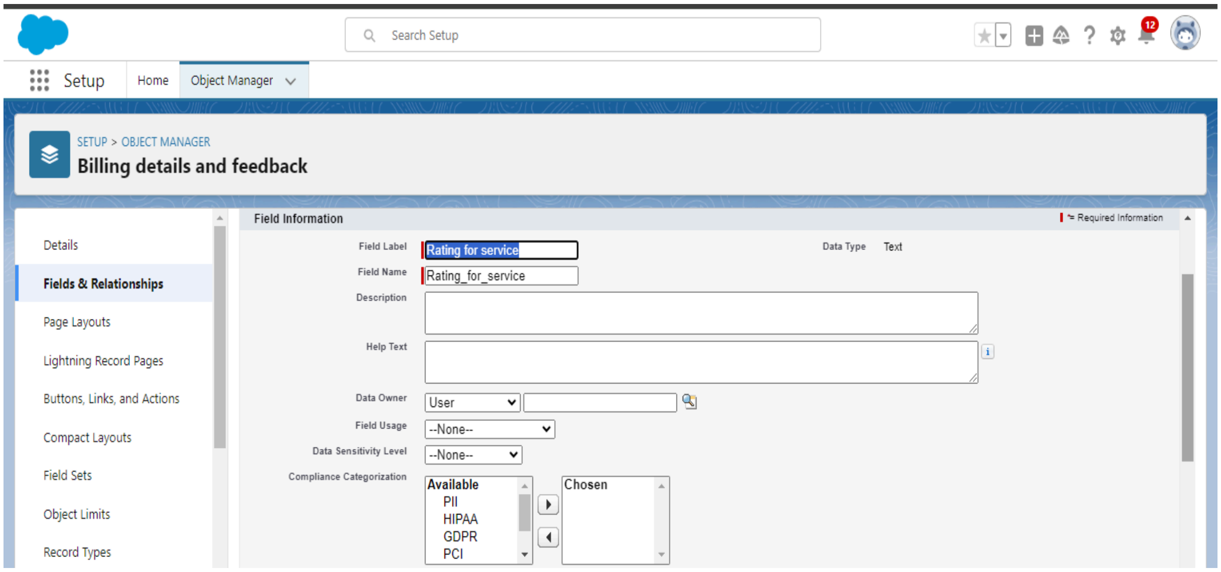
1. Go to setup >> click on Object Manager >> type object name(Appointment) in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New.
3. Select “Text” as data type and click Next.
4. Give the Field Label : Vehicle number plate
5. Field Name : is auto populated
6. Length : 10
7. Make field as Required and Unique.

8. Click on next >> next >> save.

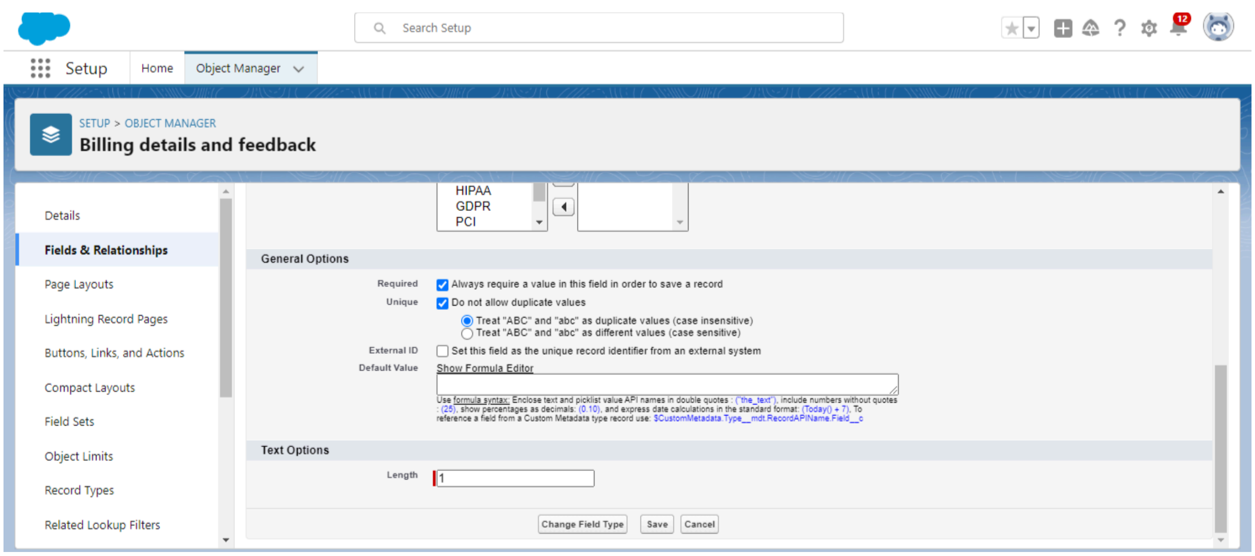
➢ Creation of Text Fields in Billing details and feedback object :

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 and Unique.
8. Click on next >> next >> save



Creation of Picklist Fields

➤ Creation of Picklist Fields inService records object :

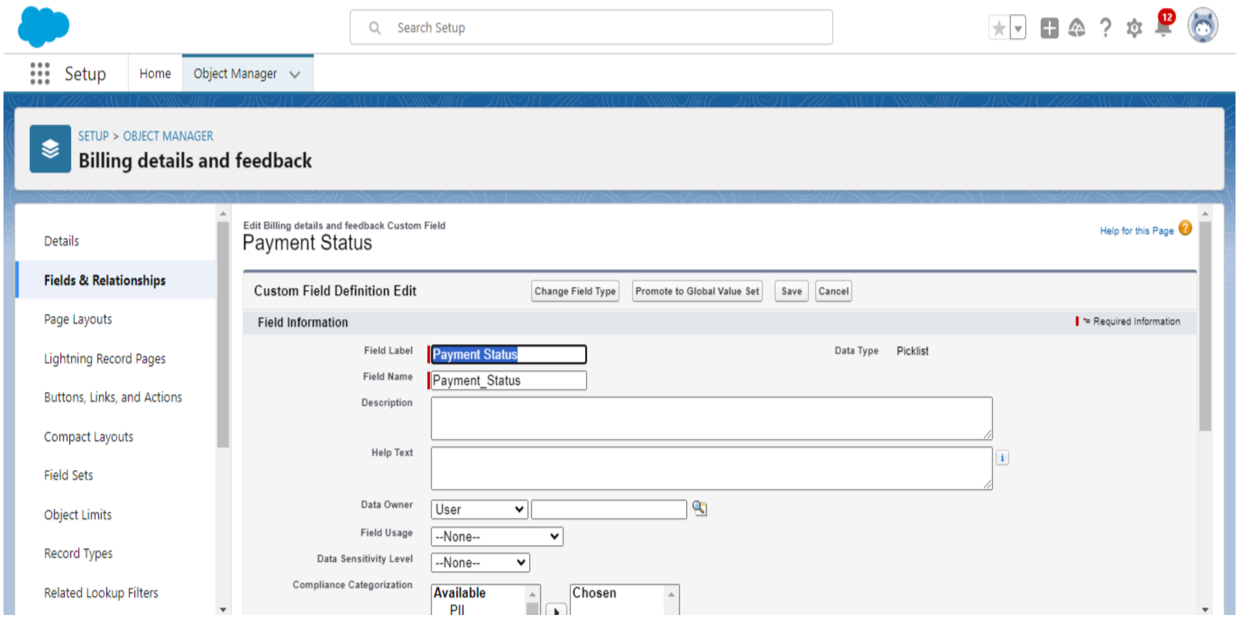
1. Go to setup >> click on Object Manager >> type object name(Service records)in search bar >> click on the object.

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 :

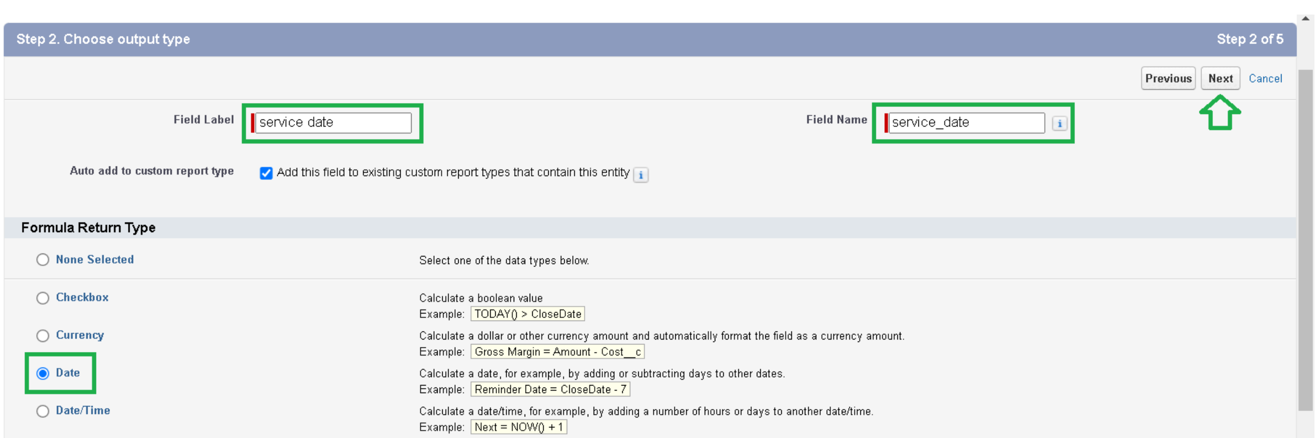
1. Go to setup >> click on Object Manager >> type object name(Billing details and feedback) in search bar >> click on the object.
2. Click on fields & relationship >> click on New.
3. Select Data type as “Picklist” and click Next.
4. Enter Field Label as “Payment Status”, under values select “Enter values, 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.



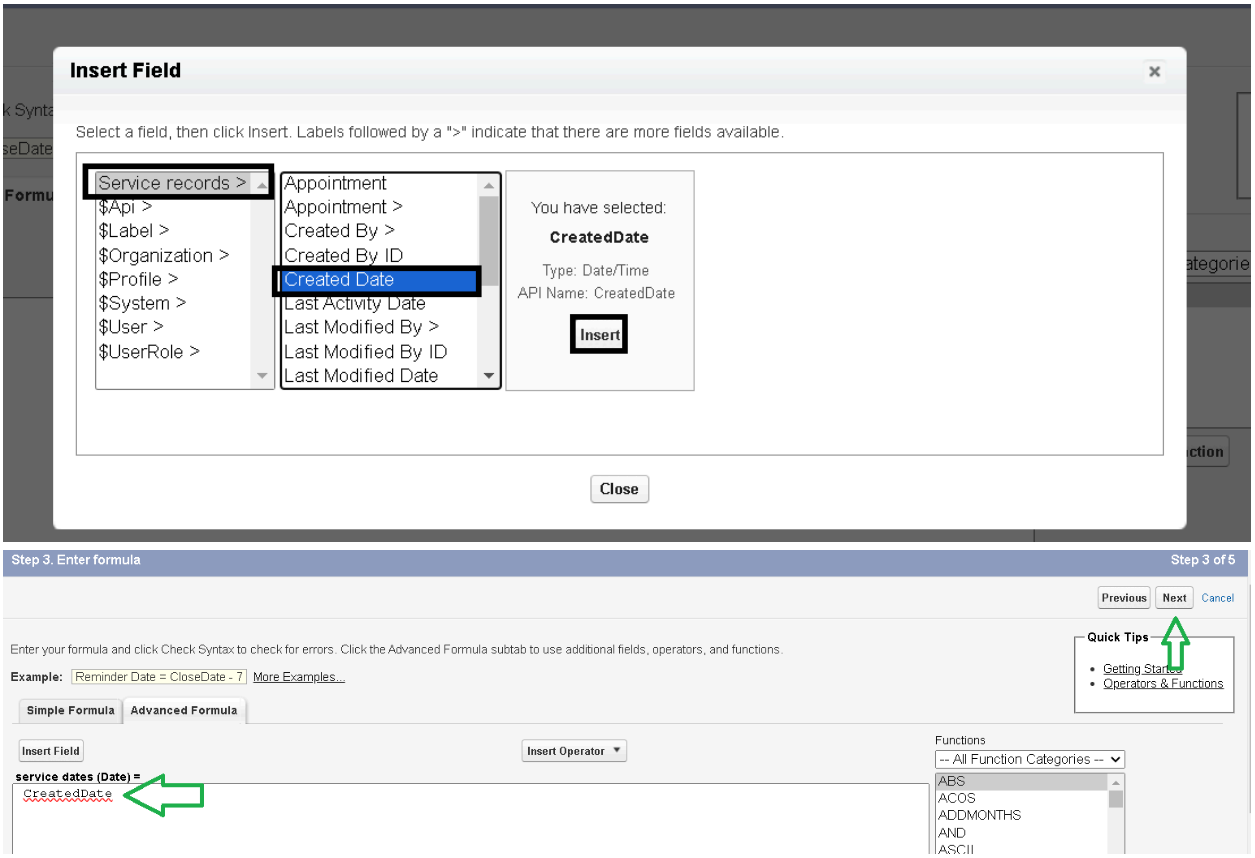
Creation of Formula Field

➢Creation of Formula Field on the Service Record 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” , if no syntax error then proceed.
7. Click next >> next >> Save.

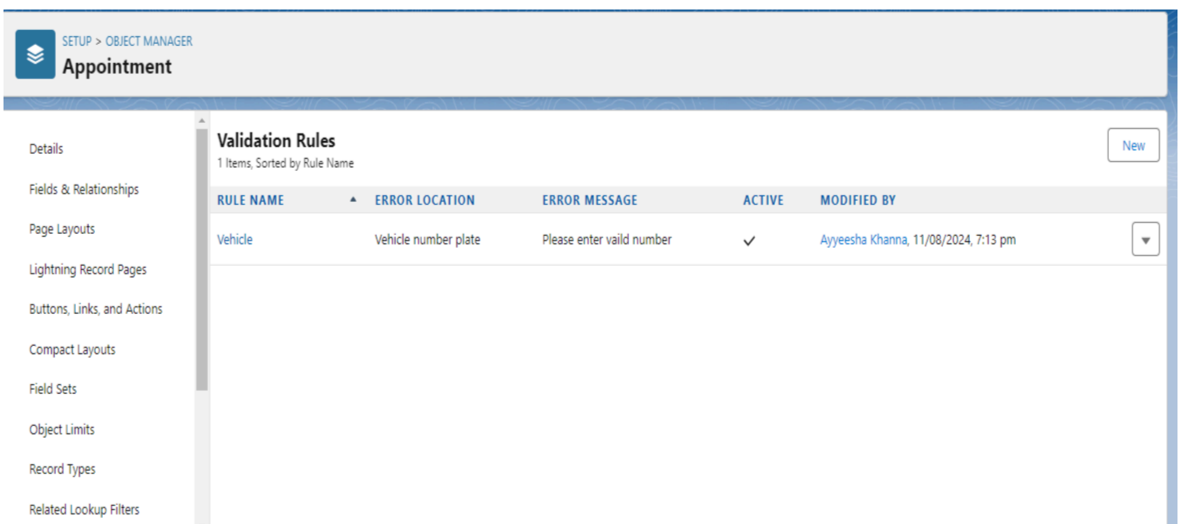
CHAPTER-5

VALIDATION RULES

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.

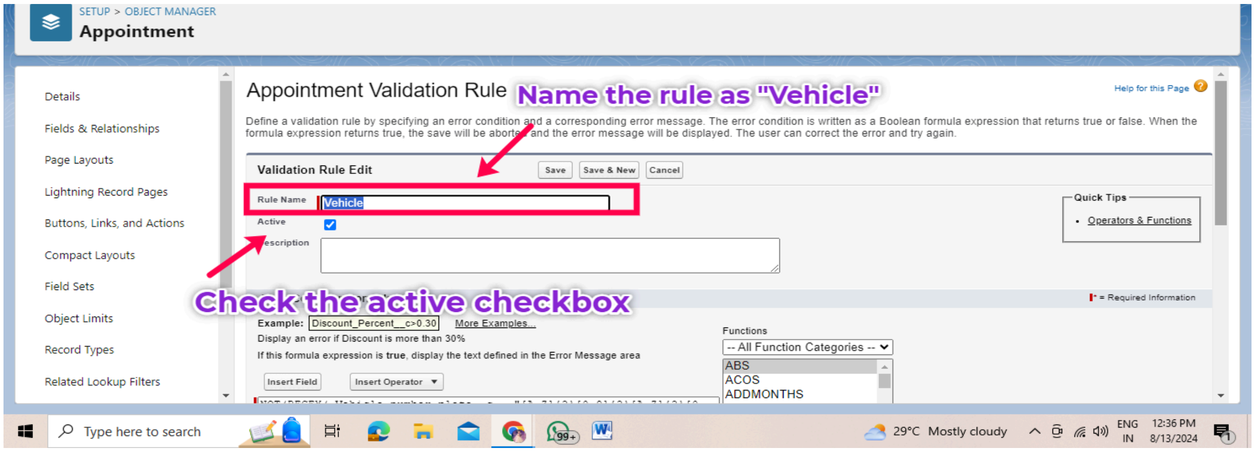
➤ To create a validation rule to an Appointment Object

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

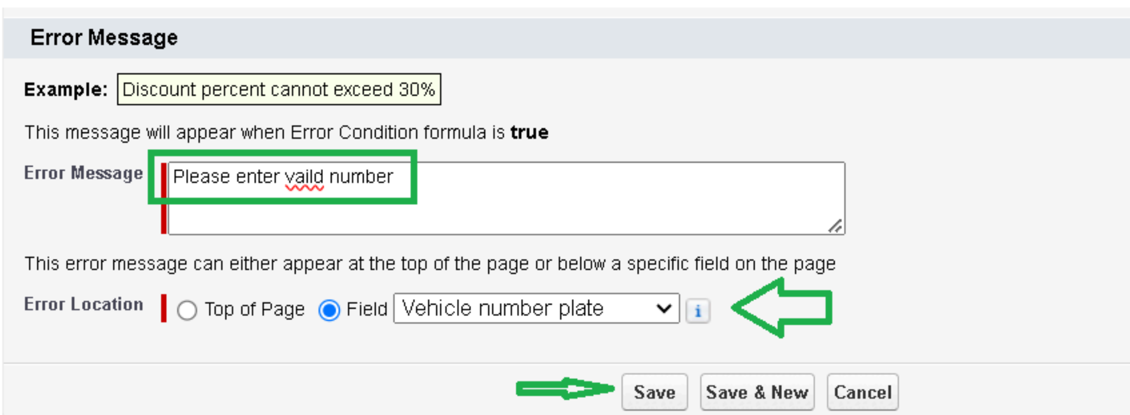


3. Enter the Rule name as “ Vehicle ”.
4. Insert the Error Condition Formula as : -

   NOT(REGEX(Vehicle_number_plate__c , "[A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{4}"))



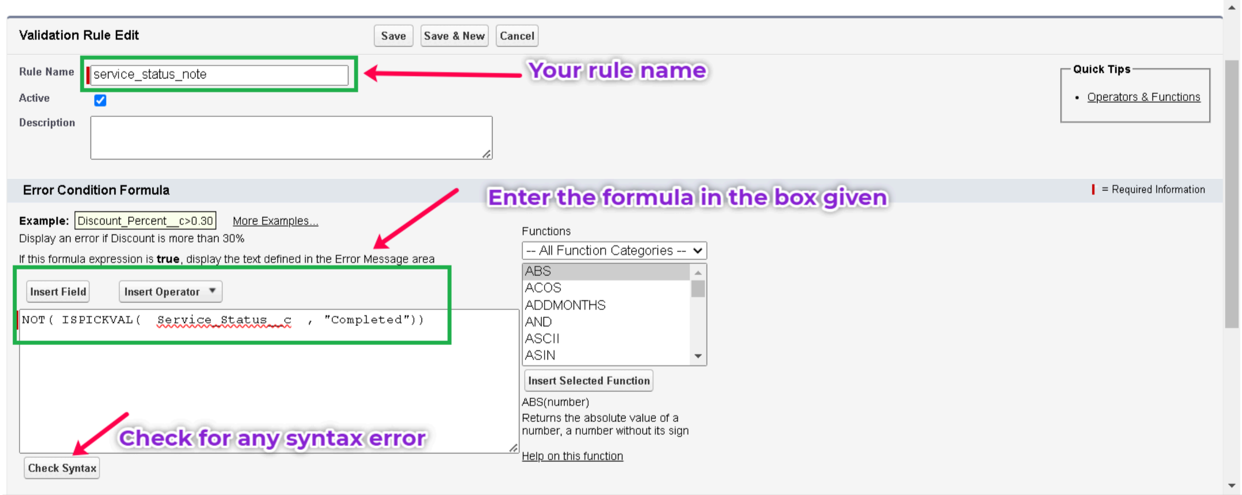
5. Enter the Error Message as “Please enter vaild number ”, select the Error location as Field and select the field as “Vehicle number plate”, and click Save.



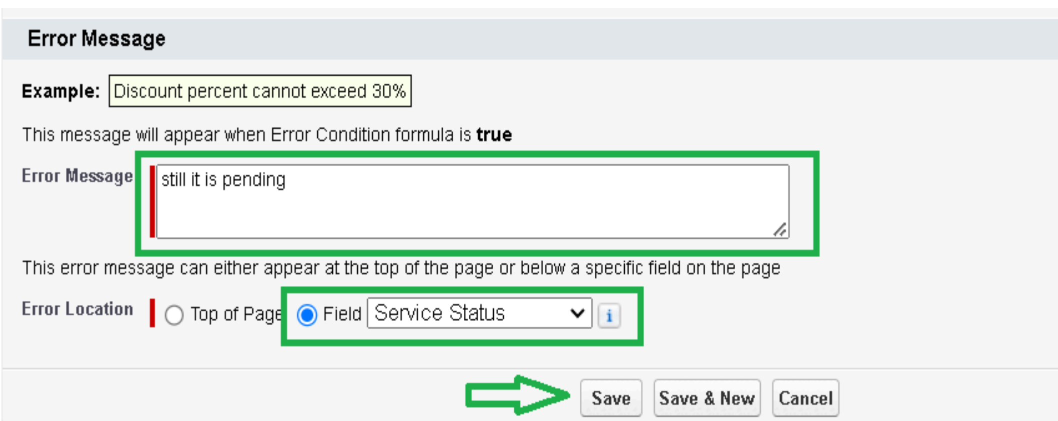
➢ To create a validation rule to an Service records Object

1. Go to the setup page >> click on object manager >> From drop down click edit for Service records object.
2. Click on the validation rule >> click New.
3. Enter the Rule name as “ service_status_note ”.
4. Insert the Error Condition Formula as : -

   NOT(ISPICKVAL(Service_Status__c , "Completed"))



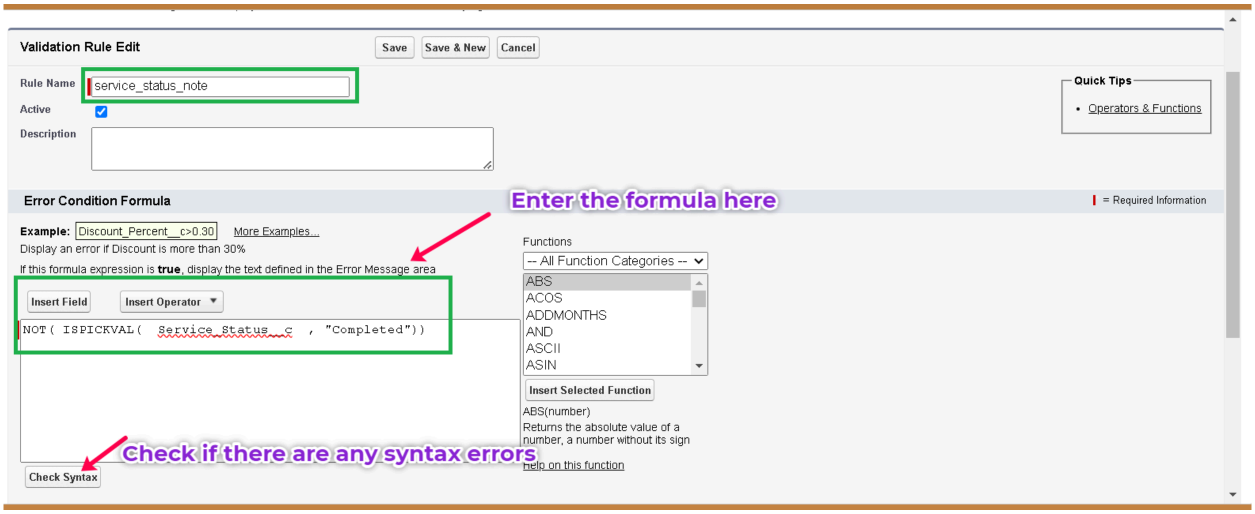
5. Enter the Error Message as “still it is pending”, select the Error location as Field and select the field as “Service status”, and click Save.



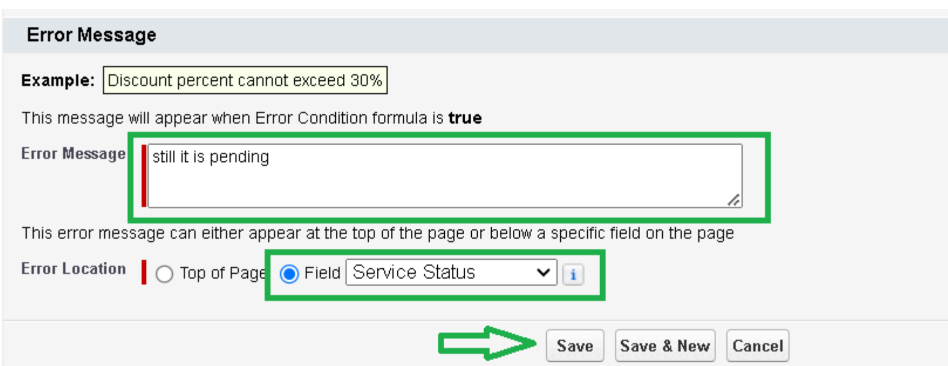
➢ To create a validation rule to an Billing details and feedback Object:

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.



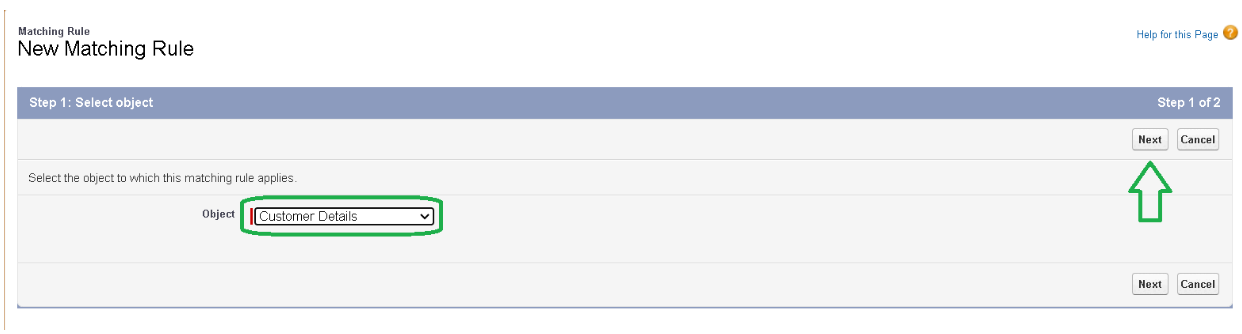
<u>CHAPTER-6</u>

DUPLICATE RULES

➢ 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.

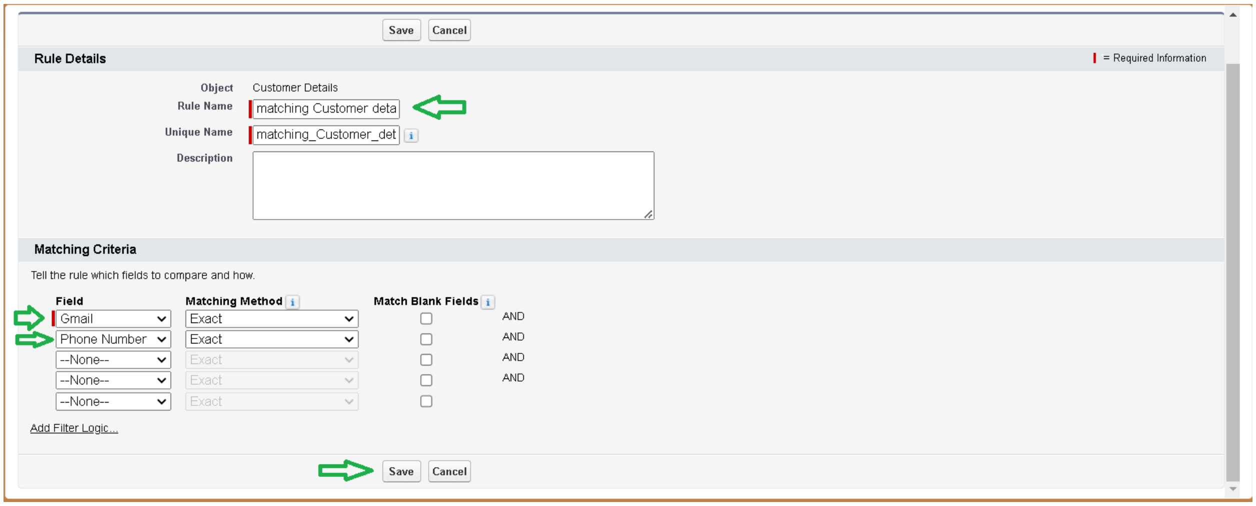


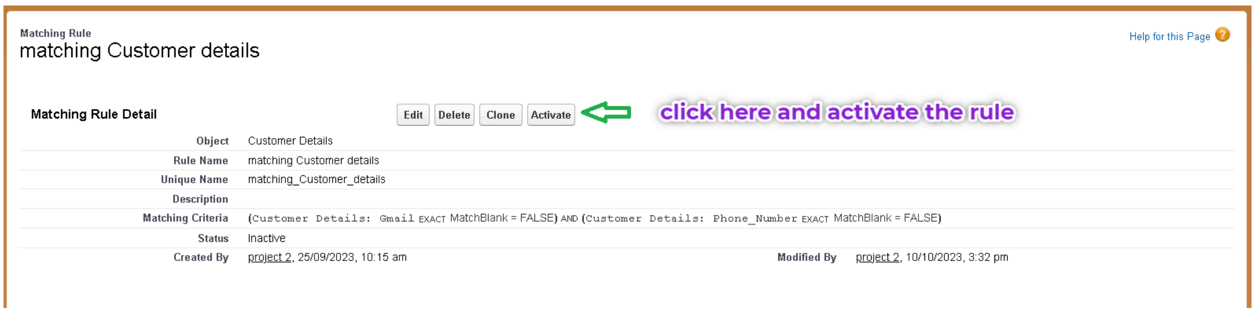
4. Give the Rule name : Matching customer details
5. Unique name : is auto populated
6. Define the matching criteria as
7. Field — Matching Method

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

8. Click save.

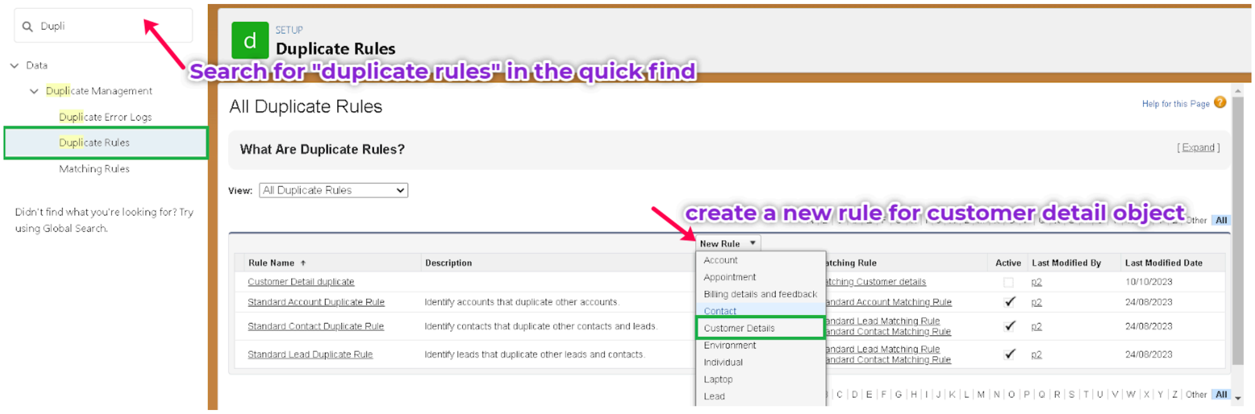
9. After Saving Click on Activate.



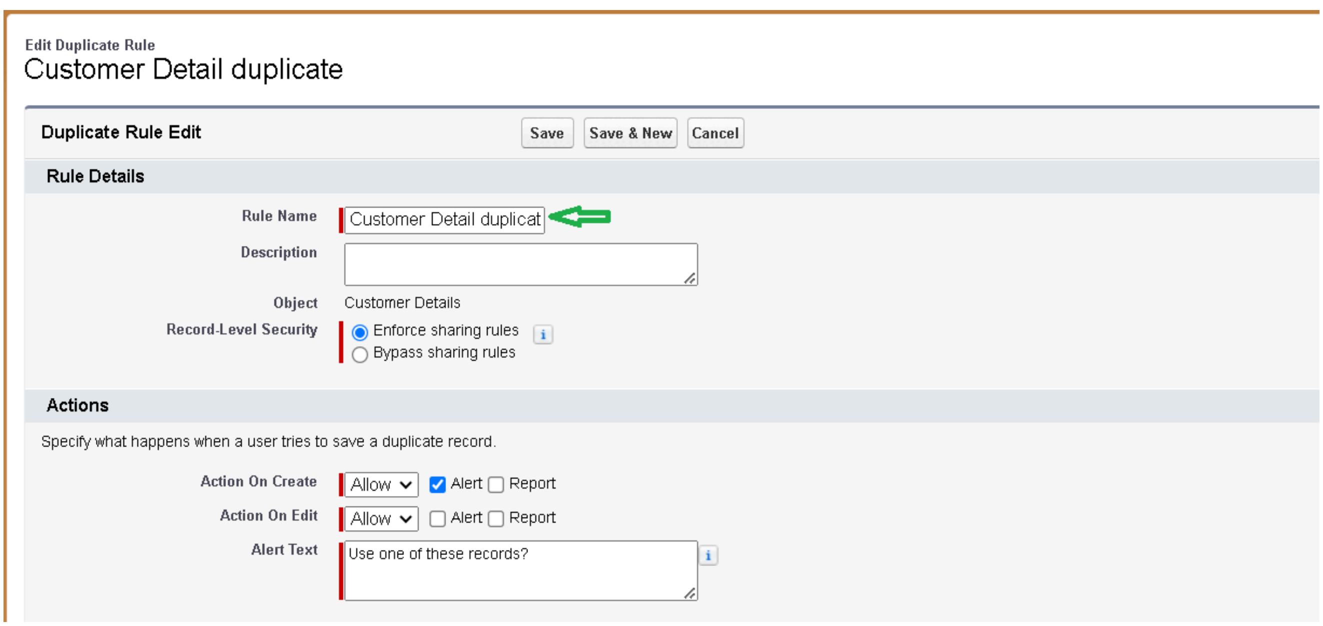


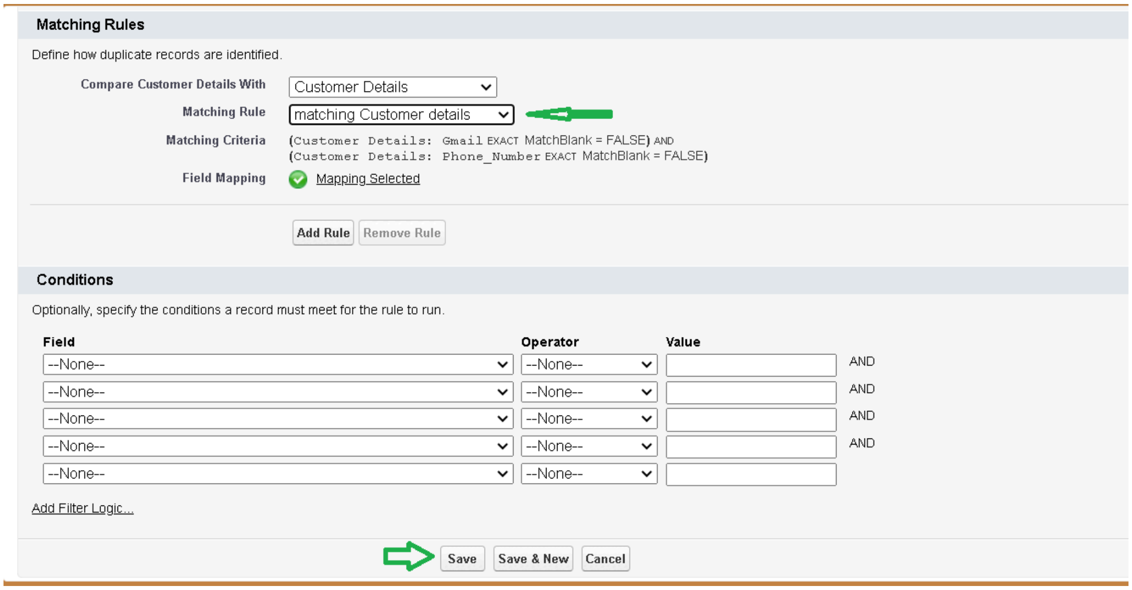
➤ To create a Duplicate rule to an Customer details Object

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.




Matching Rules
Define how duplicate records are identified.
Compare Customer Details With
Customer Details
Matching Rule
matching Customer details
Matching Criteria
(Customer Details: Gmail EXACT MatchBlank = FALSE) AND
(Customer Details: Phone_Number EXACT MatchBlank = FALSE)
Field Mapping
Mapping Selected
Add Rule
Remove Rule
Conditions
Optionally, specify the conditions a record must meet for the rule to run.
Field
Operator
Value
--None--
AND
Add Filter Logic...
Save
Save & New
Cancel

CHAPTER-7

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:**

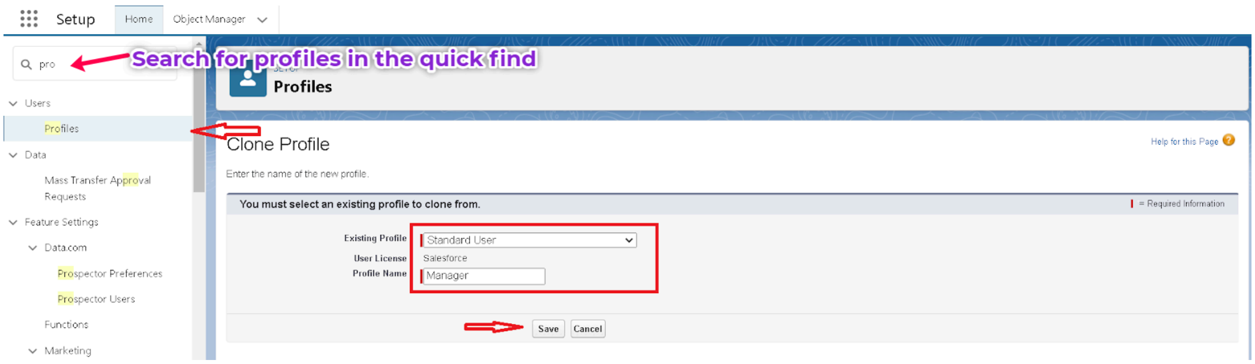
   Custom ones defined by us.

   They can be deleted if there are no users assigned with that particular one.

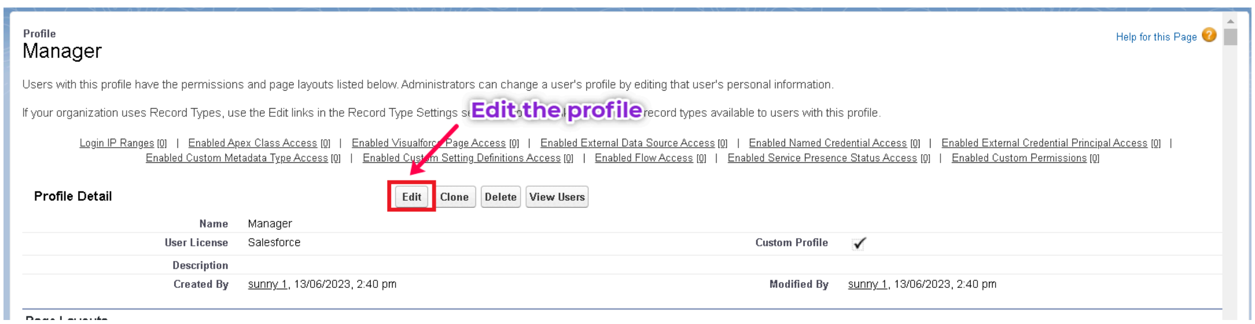
➢ Manager Profile

To create a new profile:

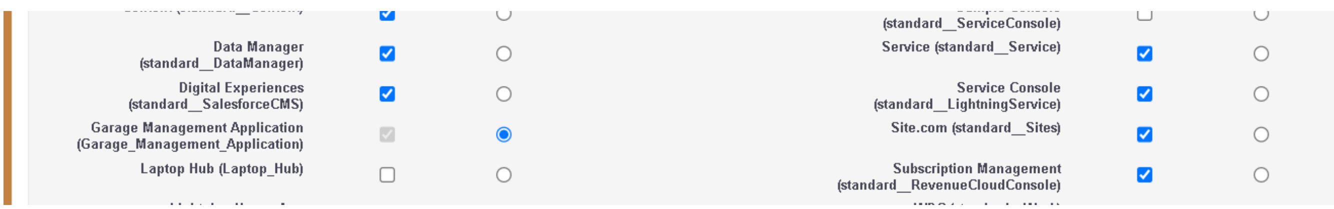
1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Standard User) >> enter profile name (Manager) >> Save.



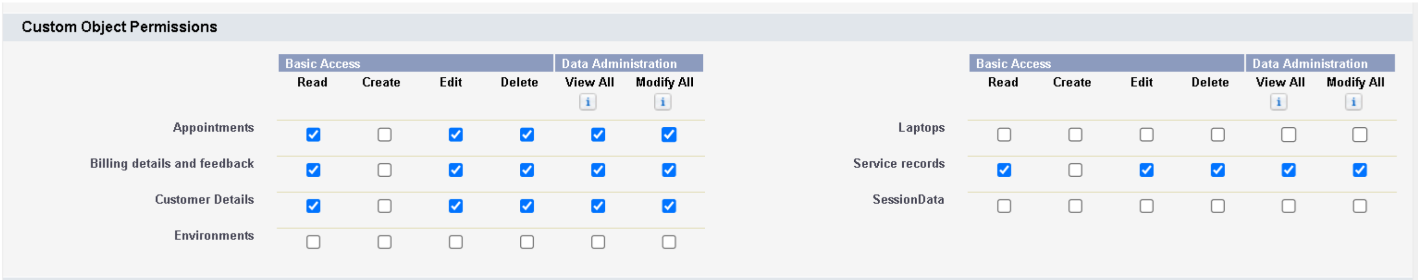
2. While still on the profile page, then click Edit.



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



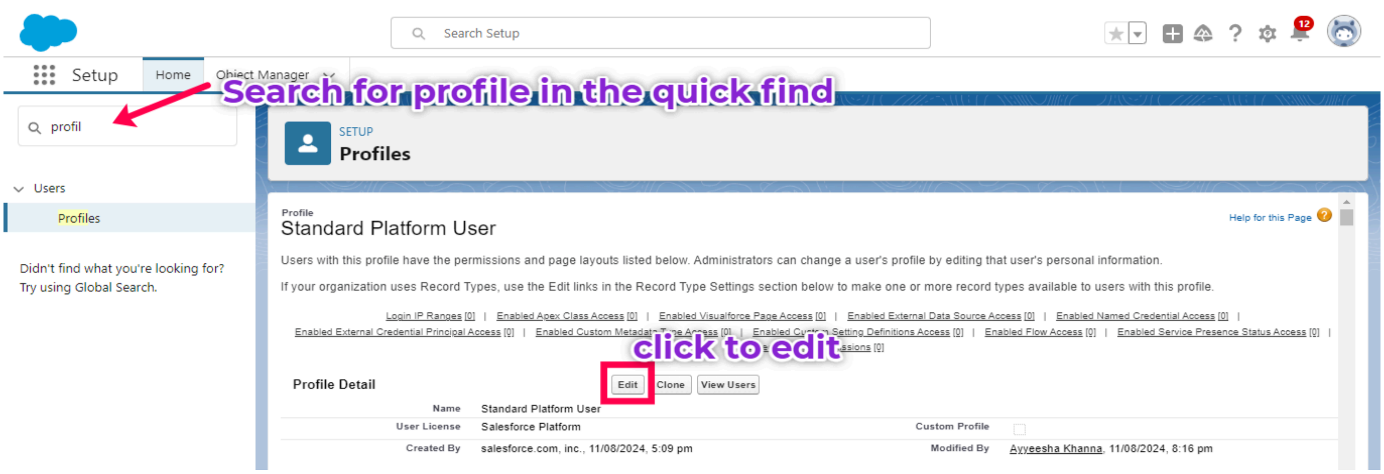
4. Scroll down to Custom Object Permissions and Give access permissions for Appointments,Billing details and feedback , service records and customer details objects as mentioned in the below diagram.



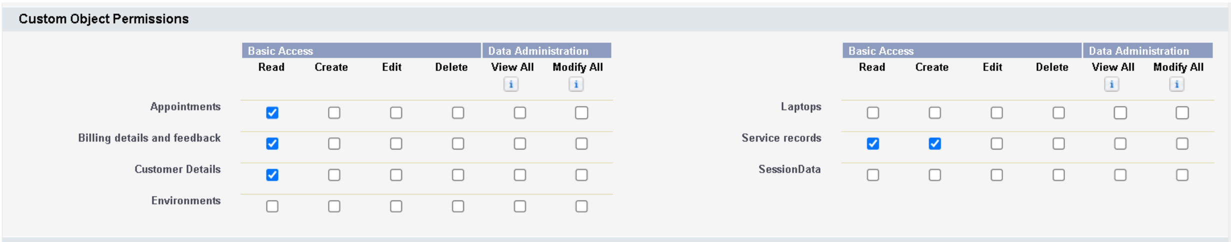
5. Changing the session times out after should be “ 8 hours of inactivity”.
6. Change the password policies as mentioned :
7. User passwords expire in should be “ never expires ”.
8. Minimum password length should be “ 8 ”, and click save.

➤ 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.

CHAPTER-8

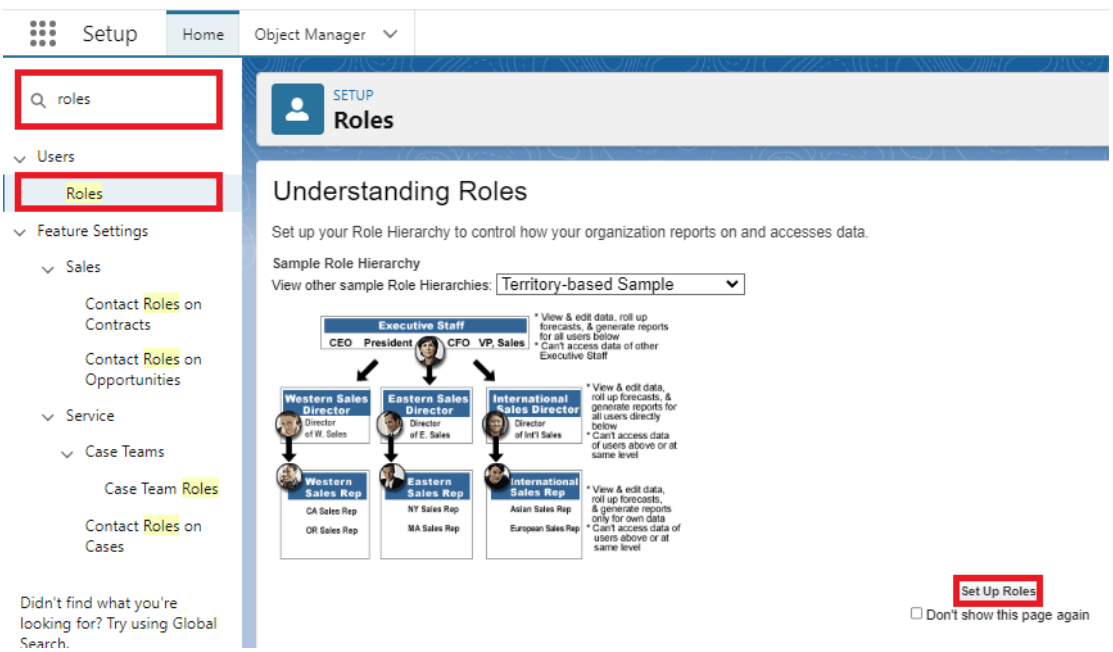
ROLE & ROLE HIERARCHY

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.

➤ Creating Manager Role

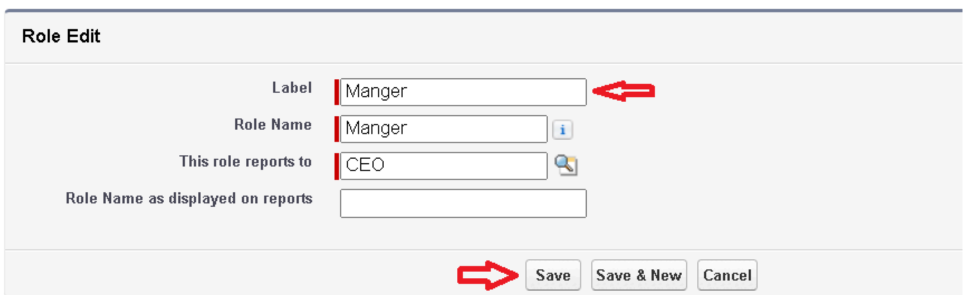
Creating Manager Role:

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.



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.

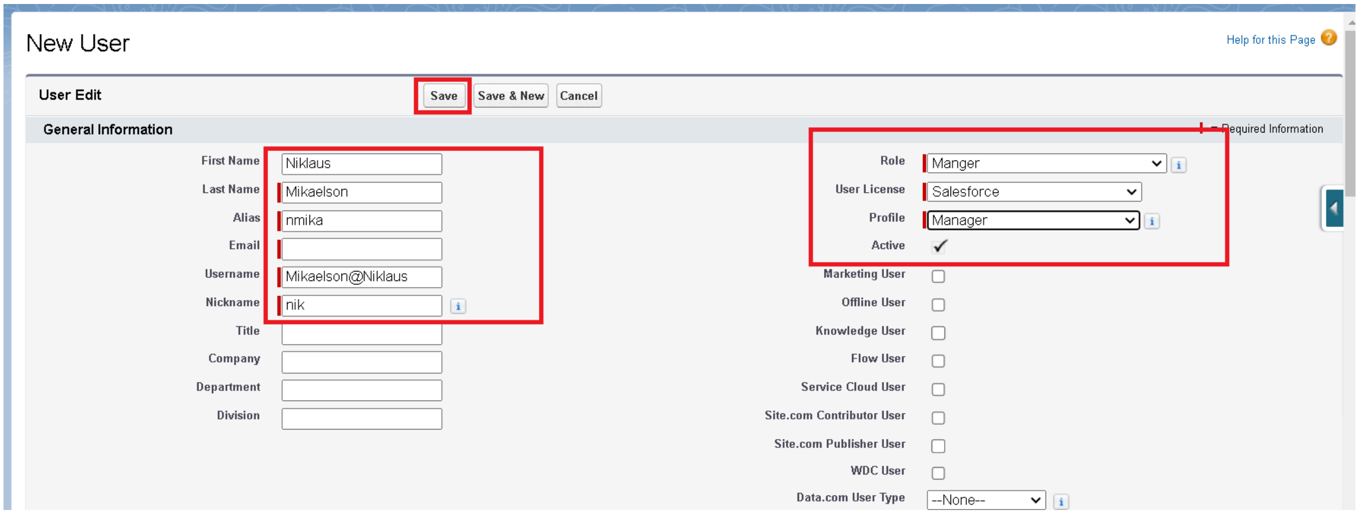
<u>CHAPTER-9</u>

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.

➢ Create User

1. Go to setup >> type users in quick find box >> select users >> click New user.
2. Fill in the fields
   i. First Name : Niklaus
   ii. Last Name : Mikaelson
   iii. Alias : Give a Alias Name
   iv. Email id : Give your Personal Email id
   v. Username : Username should be in this form: text@text.text
   vi. Nick Name : Give a Nickname
   vii. Role : Manager
   viii. User licence : Salesforce
   ix. Profiles : Manager

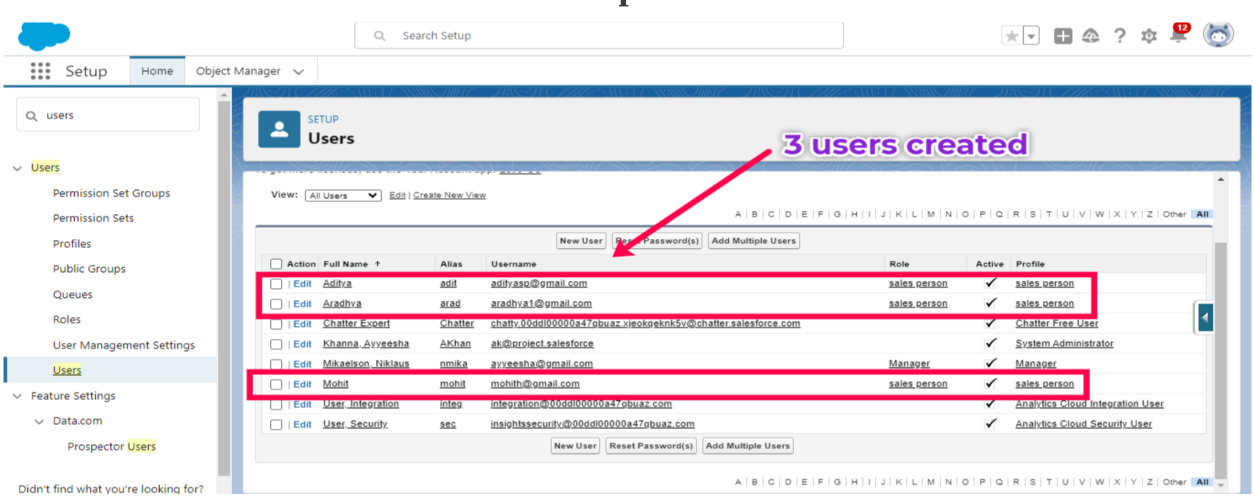


3. Save.

➢ Creating Another Users

1. Repeat the steps and create another user using
   1. Role : sales person
   2. User licence : Salesforce Platform
   3. Profile : sales person

Note : create atleast 3 users with these permissions.



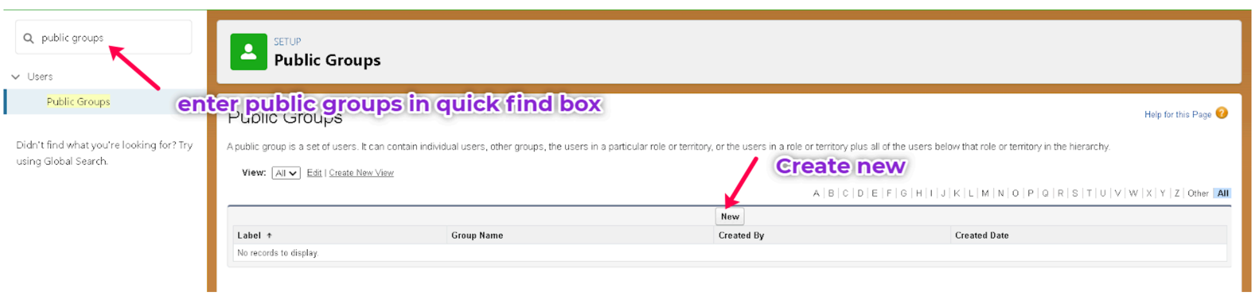
CHAPTER-10

PUBLIC GROUPS

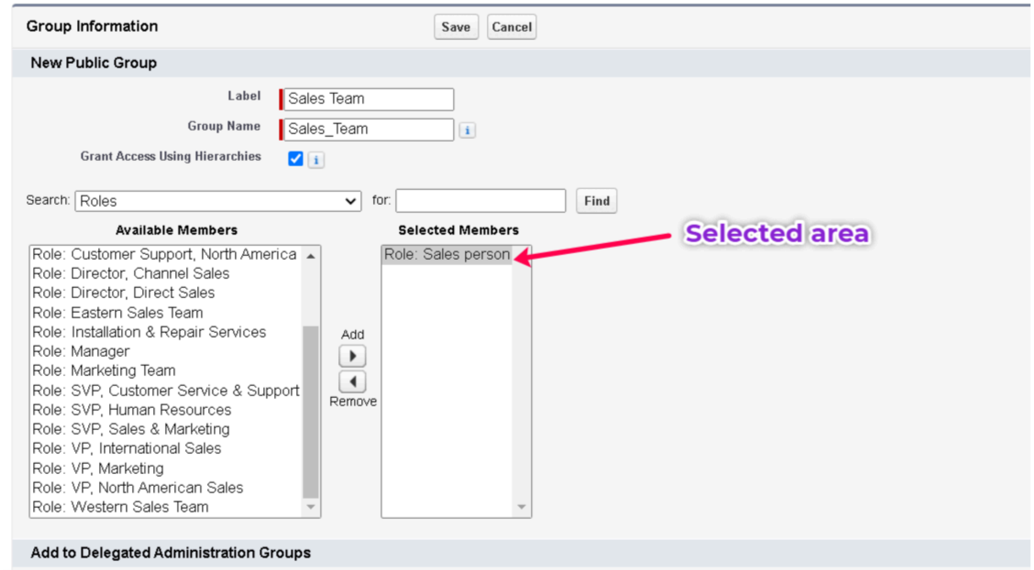
Public groups are a valuable tool for Salesforce administrators and developers to streamline user management, data access, and security settings. By creating and using public groups effectively, you can maintain a secure and organized Salesforce environment while ensuring that users have appropriate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

➤ 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.


Group Information
Save
Cancel
New Public Group
Label
Sales Team
Group Name
Sales_Team
Grant Access Using Hierarchies
Search:
Roles
for:
Find
Available Members
Role: Customer Support, North America
Role: Director, Channel Sales
Role: Director, Direct Sales
Role: Eastern Sales Team
Role: Installation & Repair Services
Role: Manager
Role: Marketing Team
Role: SVP, Customer Service & Support
Role: SVP, Human Resources
Role: SVP, Sales & Marketing
Role: VP, International Sales
Role: VP, Marketing
Role: VP, North American Sales
Role: Western Sales Team
Add
Remove
Selected Members
Role: Sales person
Selected area
Add to Delegated Administration Groups

CHAPTER-11

SHARING SETTINGS

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.

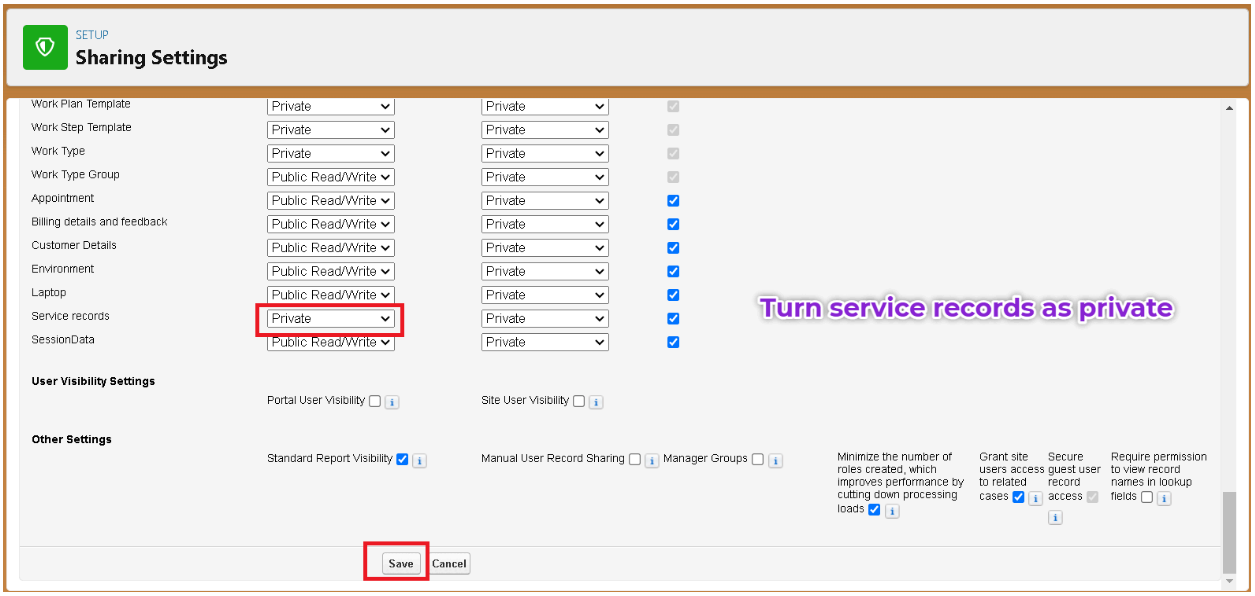
Sharing Rules:

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:

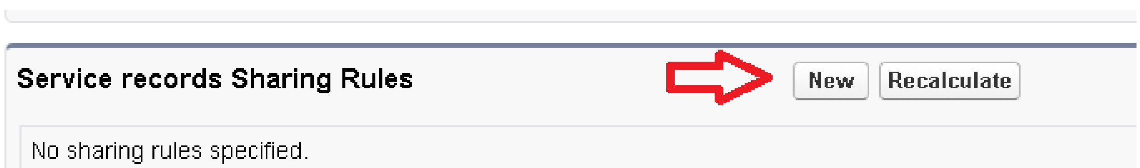
Administrators and record owners can manually share specific records with other users or groups.

➢ Creating Sharing settings

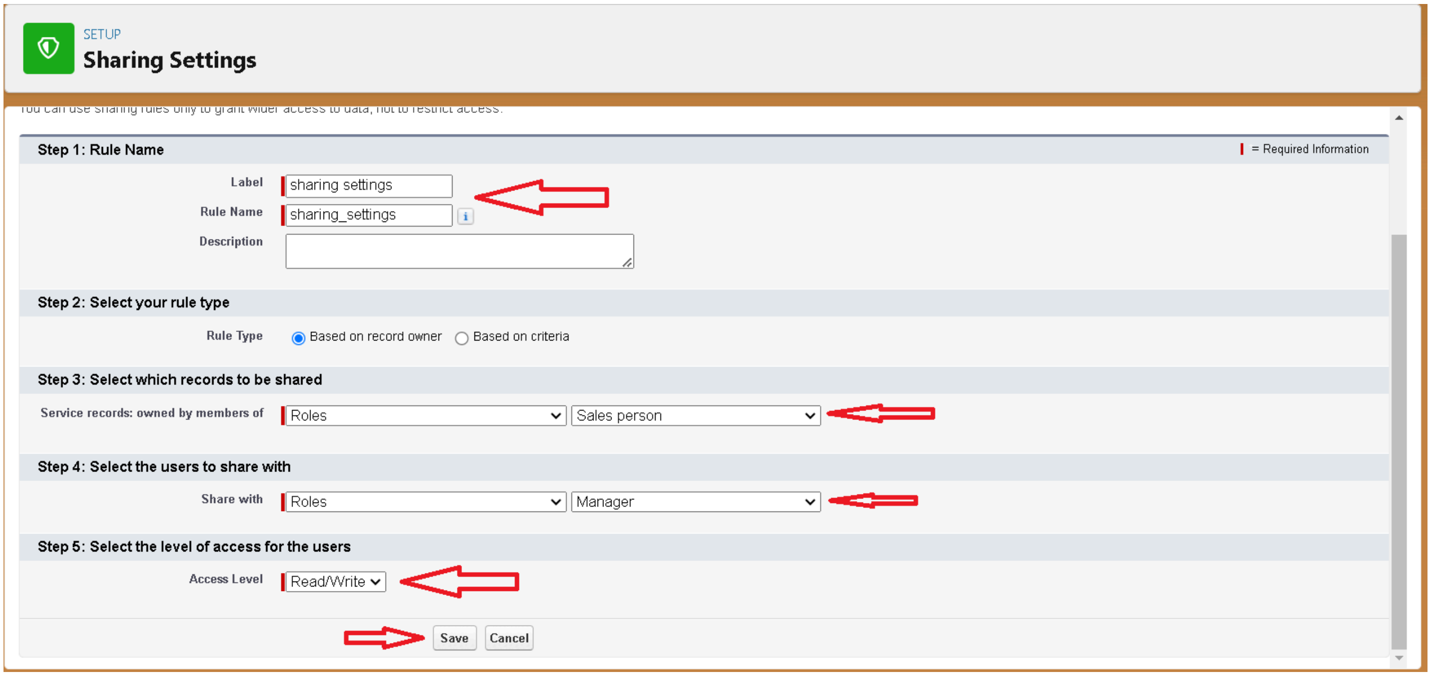
1. Go to setup >> type users in quick find box >> select Sharing Settings >> click Edit.
2. Change the OWD setting of the Service records Object to private as shown in fig.



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



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


SETUP
Sharing Settings
You can use sharing rules only to grant wider access to data, not to restrict access.
Step 1: Rule Name
= Required Information
Label
sharing settings
Rule Name
sharing_settings
Description
Step 2: Select your rule type
Rule Type
Based on record owner
Based on criteria
Step 3: Select which records to be shared
Service records: owned by members of
Roles
Sales person
Step 4: Select the users to share with
Share with
Roles
Manager
Step 5: Select the level of access for the users
Access Level
Read/Write
Save
Cancel

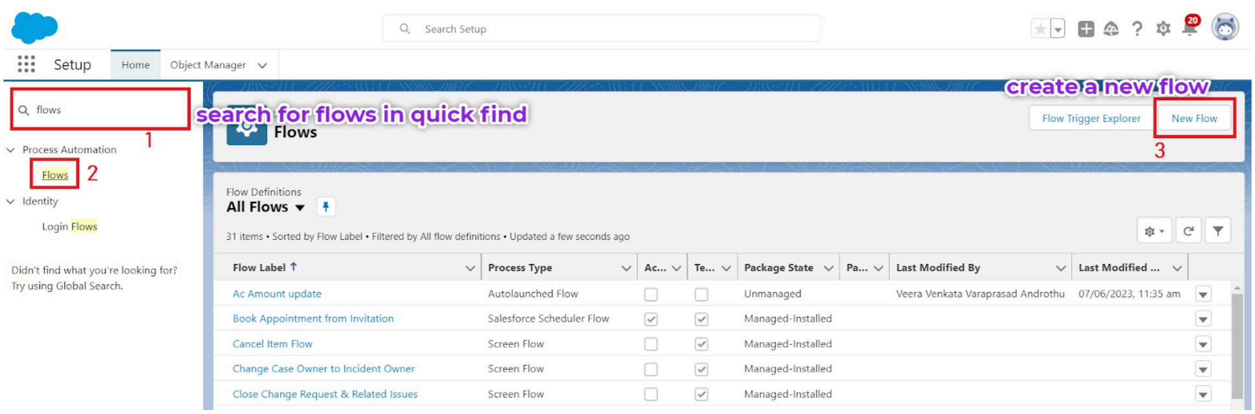
CHAPTER-12

FLOWS

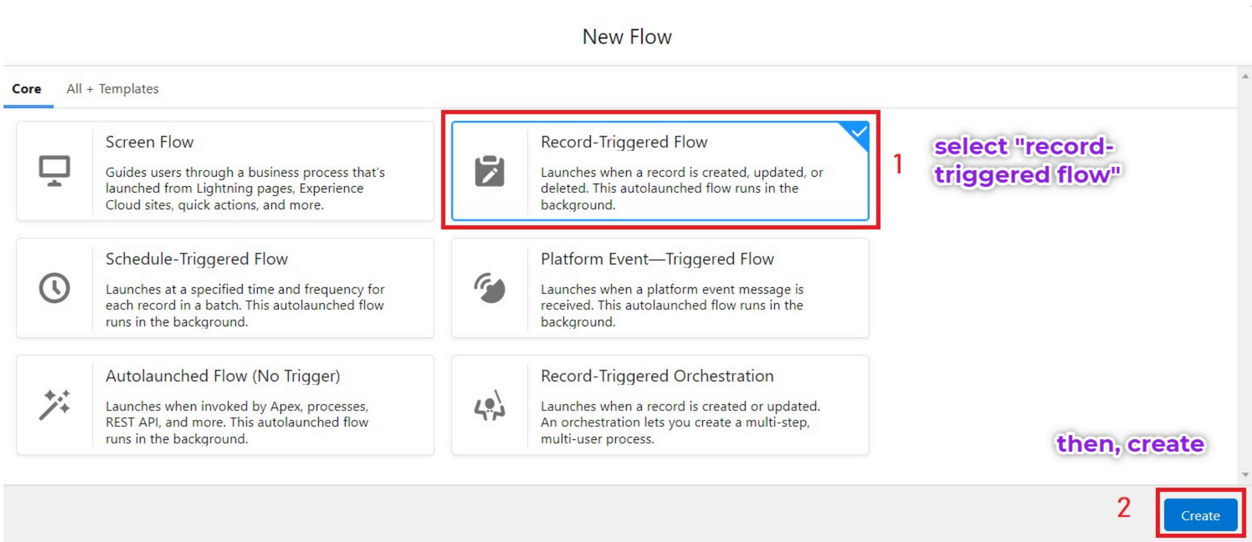
In Salesforce, a flow is a powerful tool that allows you to automate business processes, collect and update data, and guide users through a series of screens or steps. Flows are built using a visual interface and can be created without any coding knowledge.

➢ Create a Flow

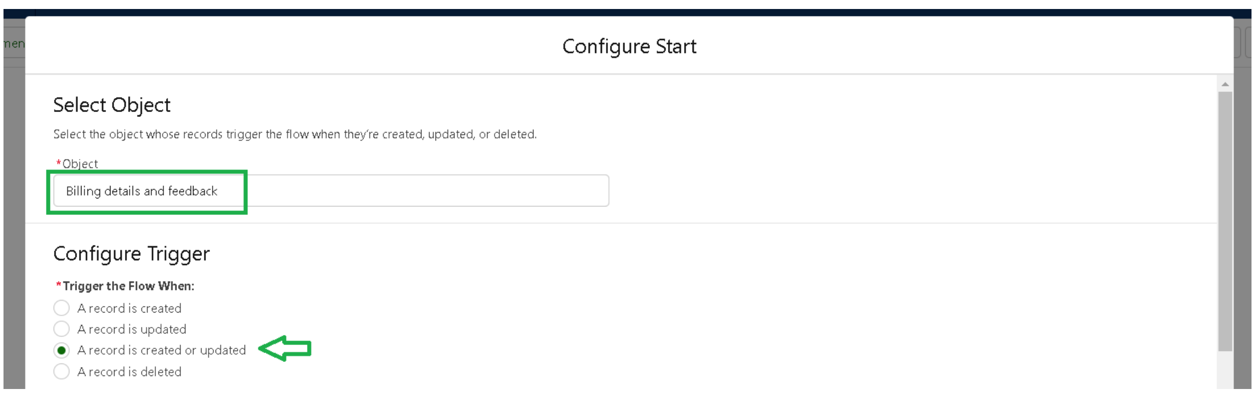
1. Go to setup >> type Flow in quick find box >> Click on the Flow and Select the New Flow.

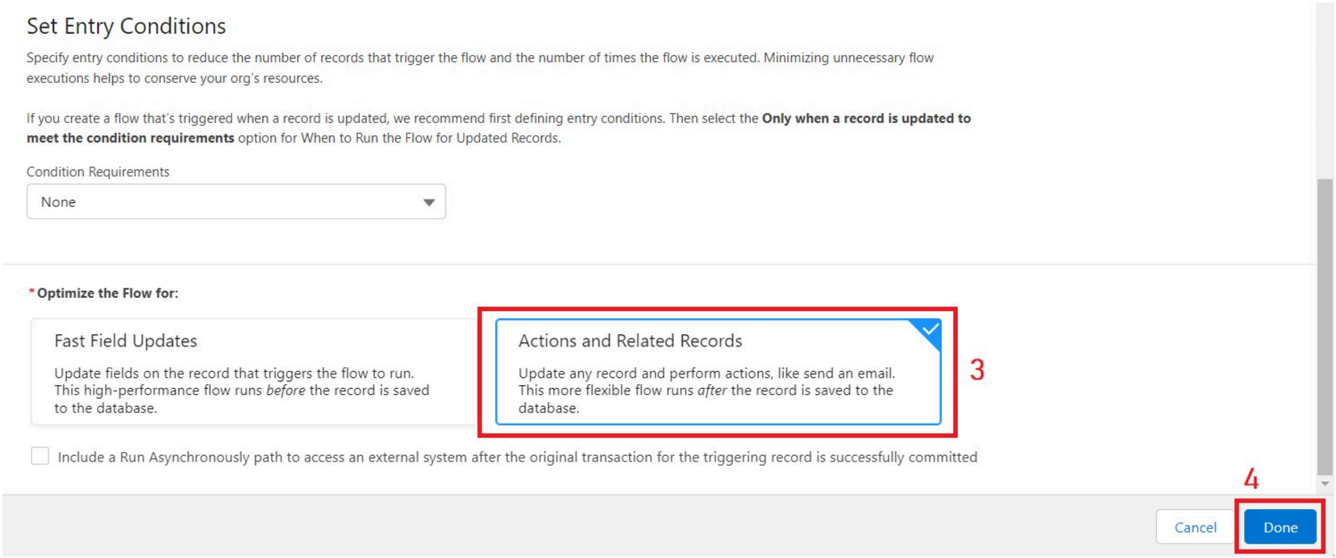


2. Select the Record-triggered flow and Click on Create.

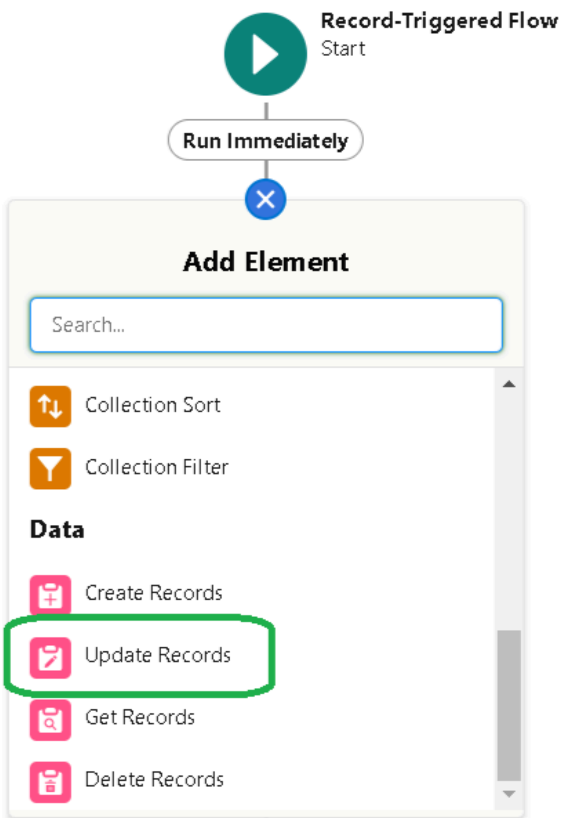


3. Select the Object as “Billing details and feedback”in the Drop down list.
4. Select the Trigger Flow when: “A record is Created or Updated”.
5. Select the Optimize the flow for: “Actions and Related Records” and Click on Done.

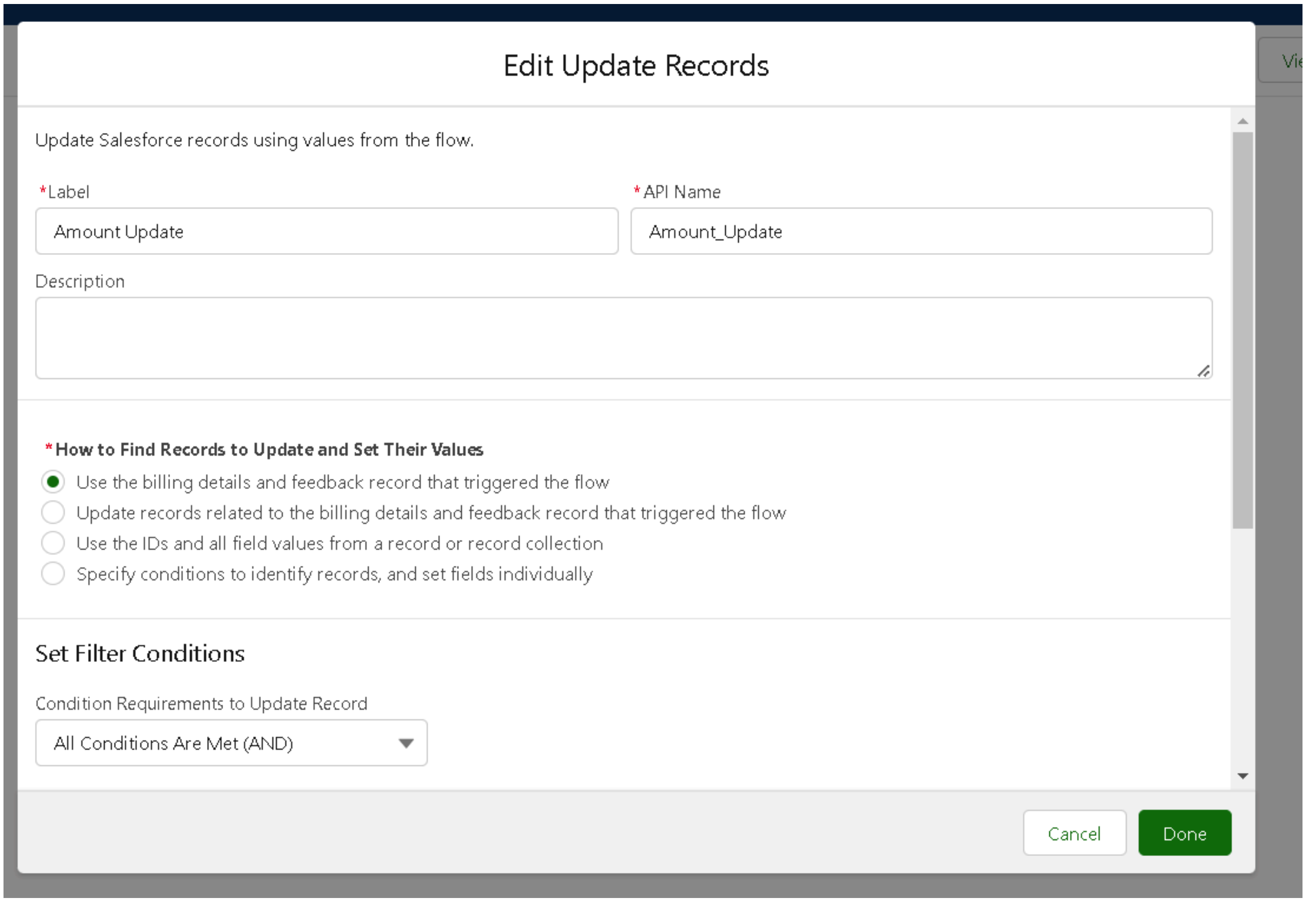


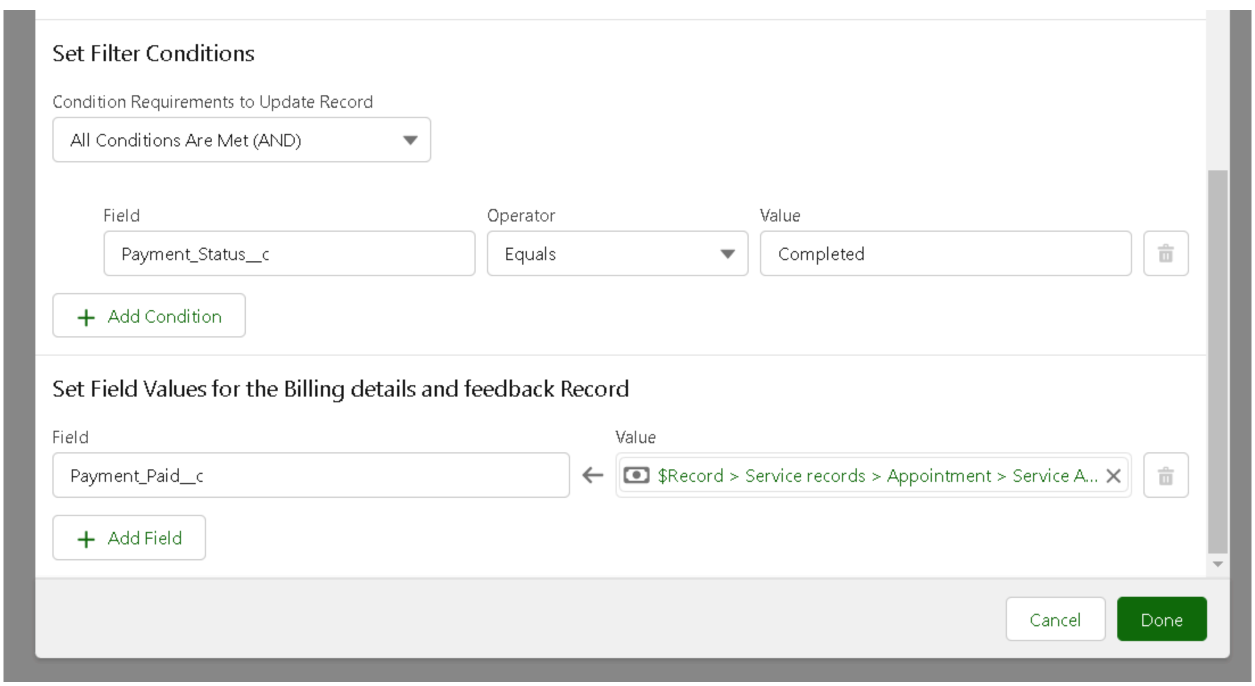


6. Under the Record-triggered Flow Click on "+" Symbol and In the Drop down List select the "Update records Element".



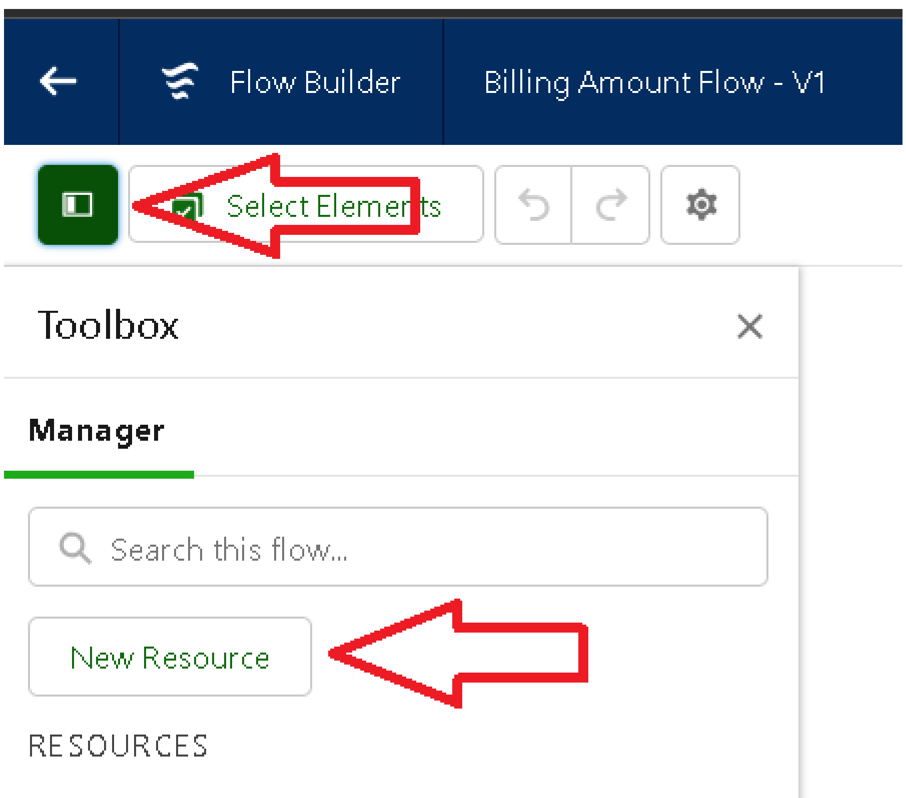
7. Give the Label Name : Amount Update
8. Api name : is auto populated


Edit Update Records
Update Salesforce records using values from the flow.
*Label
Amount Update
*API Name
Amount_Update
Description
*How to Find Records to Update and Set Their Values
Use the billing details and feedback record that triggered the flow
Update records related to the billing details and feedback record that triggered the flow
Use the IDs and all field values from a record or record collection
Specify conditions to identify records, and set fields individually
Set Filter Conditions
Condition Requirements to Update Record
All Conditions Are Met (AND)
Cancel
Done


Set Filter Conditions
Condition Requirements to Update Record
All Conditions Are Met (AND)
Field
Payment_Status__c
Operator
Equals
Value
Completed
Add Condition
Set Field Values for the Billing details and feedback Record
Field
Payment_Paid__c
Value
$Record > Service records > Appointment > Service A...
Add Field
Cancel
Done

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.

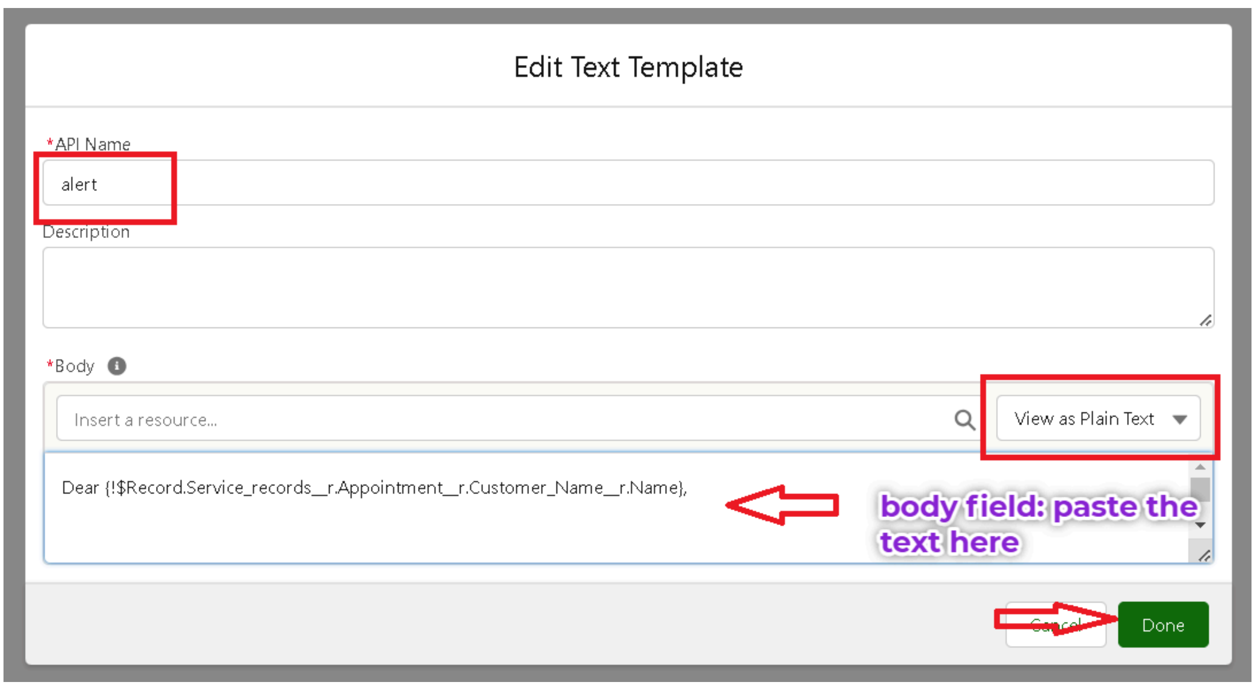
Dear {!$Record.Service_records__r.Appointment__r.Customer_Name__r.Name},

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.

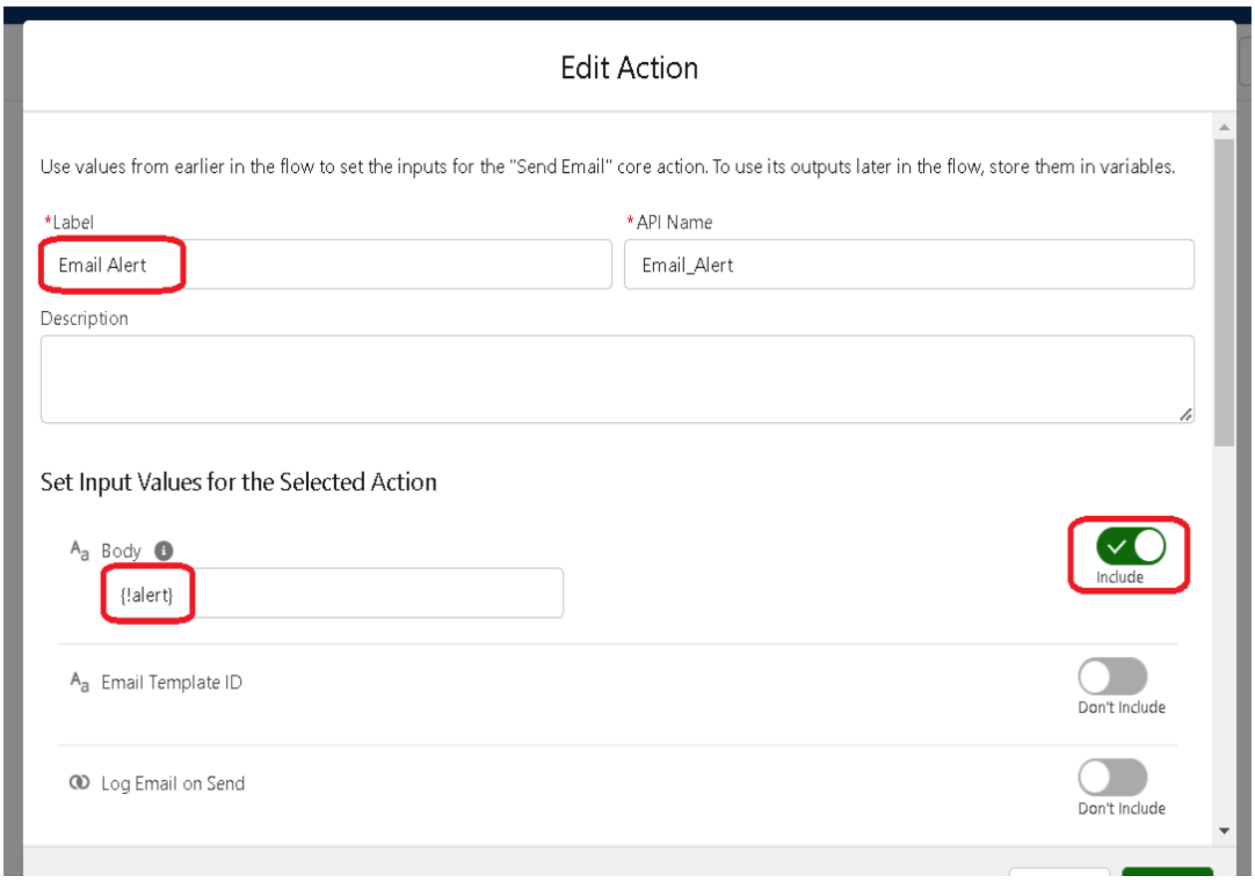
29.Select the text template that created , Body : {!alert}

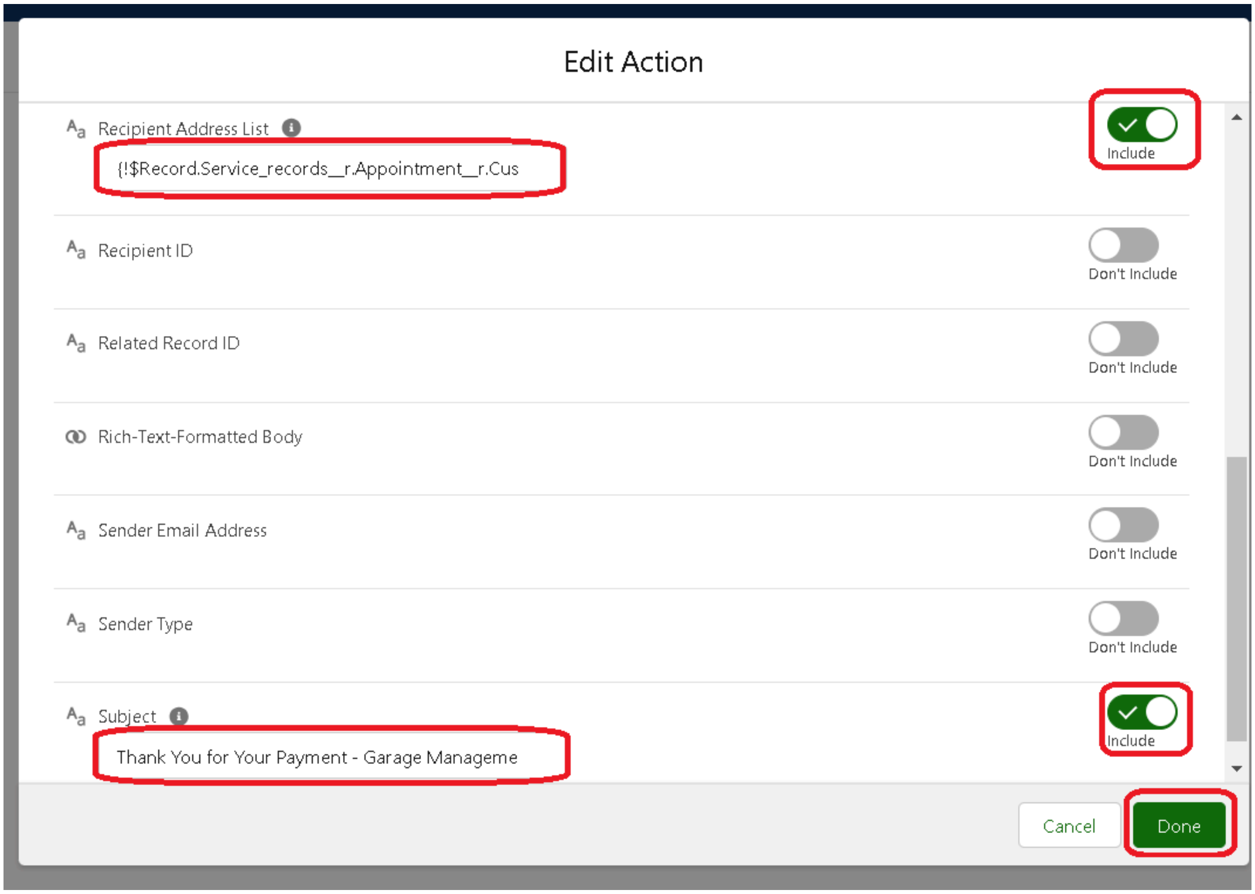
30.Include recipient address list select the email form the record.

31.RecipientAddressList:
{!$Record.Service_records__r.Appointment__r.Customer_Name__r.Gmail__c}

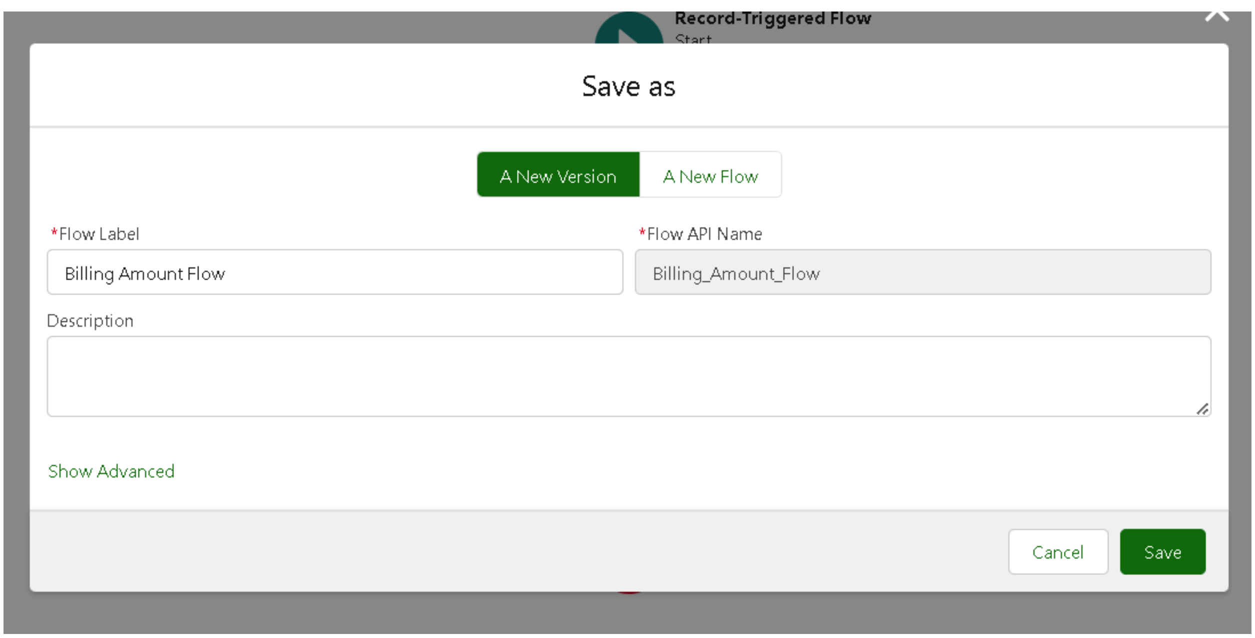
32.Include subject as " Thank You for Your Payment - Garage Management".

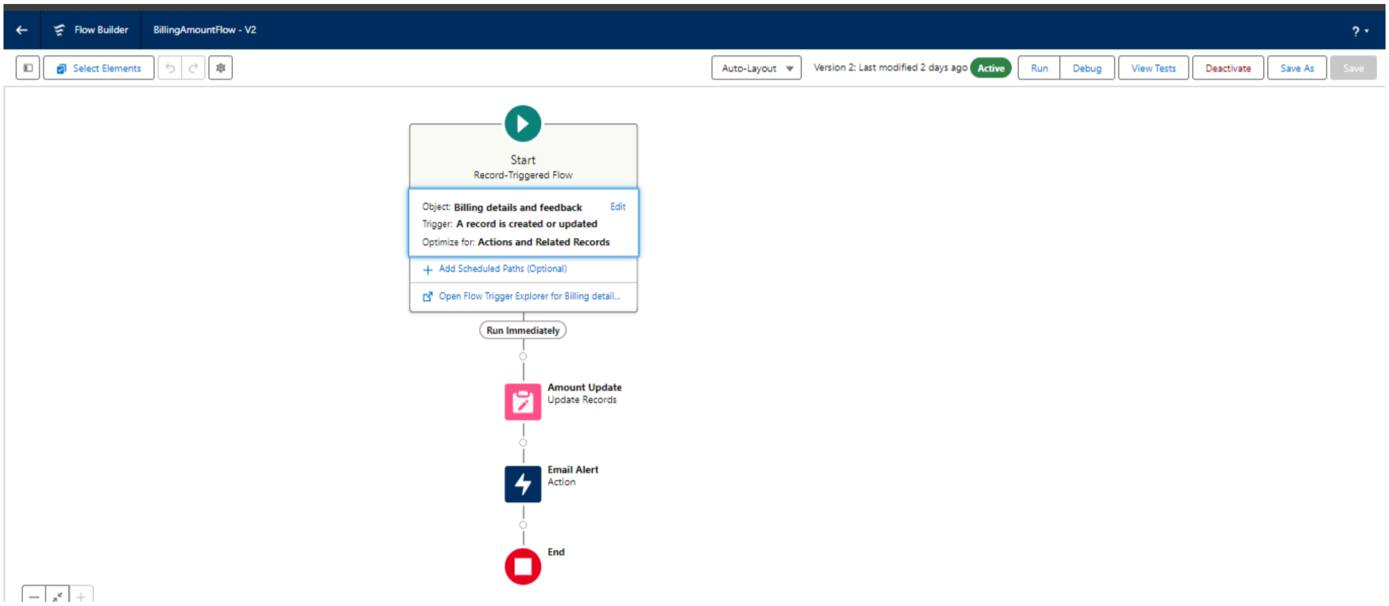
33.Click done.





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





Flow Builder
BillingAmountFlow - V2
Select Elements
Auto-Layout
Version 2: Last modified 2 days ago
Active
Run
Debug
View Tests
Deactivate
Save As
Save
Start
Record-Triggered Flow
Object: Billing details and feedback
Edit
Trigger: A record is created or updated
Optimize for: Actions and Related Records
Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Billing detail...
Run Immediately
Amount Update
Update Records
Email Alert
Action
End

CHAPTER-13

APEX TRIGGERS

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:

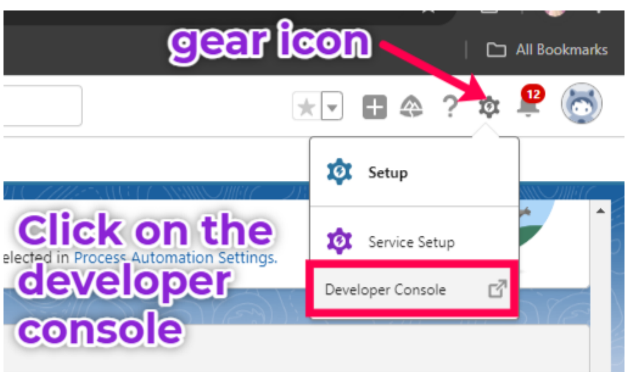
Before Trigger: This type of trigger in Salesforce is used either to update or validate the values of a record before they can be saved into the database. So, basically, the before trigger validates the record first and then saves it. Some criteria or code can be set to check data before it gets ready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

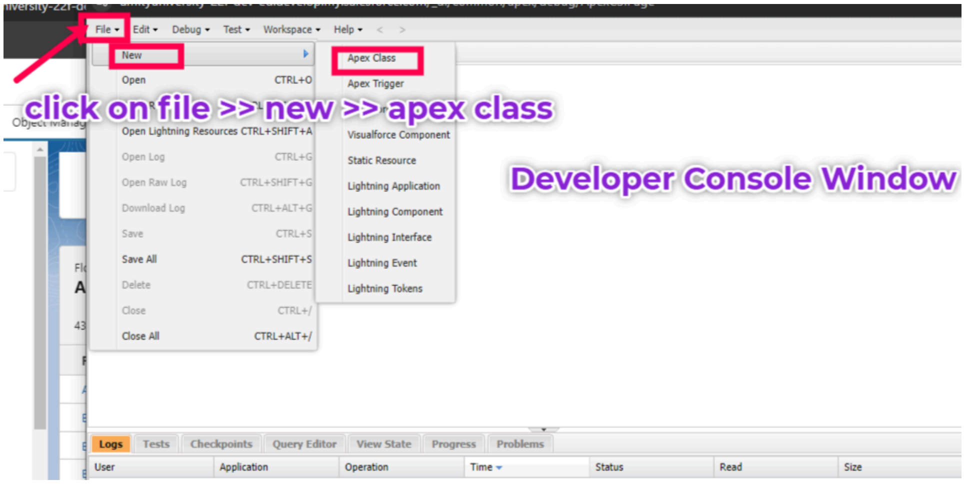
➤ Apex Handler

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

```
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;
```

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:

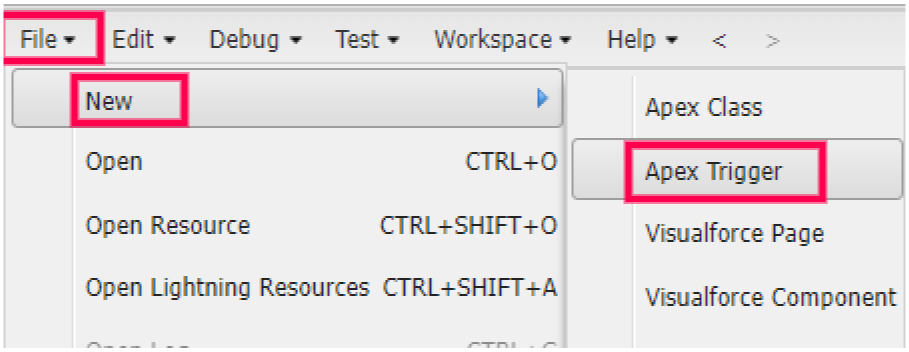
```
public class AmountDistributionHandler {

    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;
```

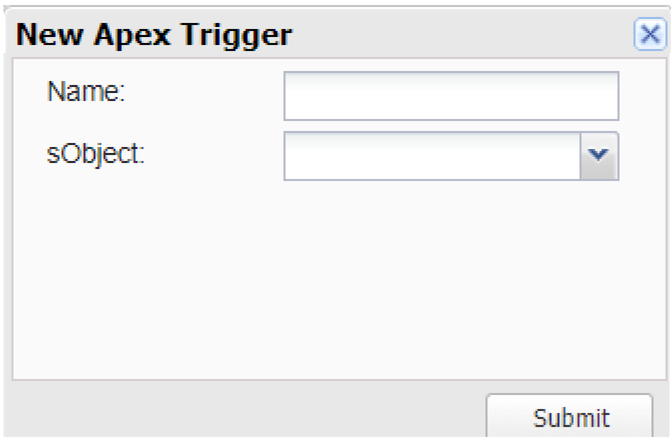
```
        }
        else if(app.Replacement_Parts__c == true){
            app.Service_Amount__c = 5000;
        }

    }
    }
}
```

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 :

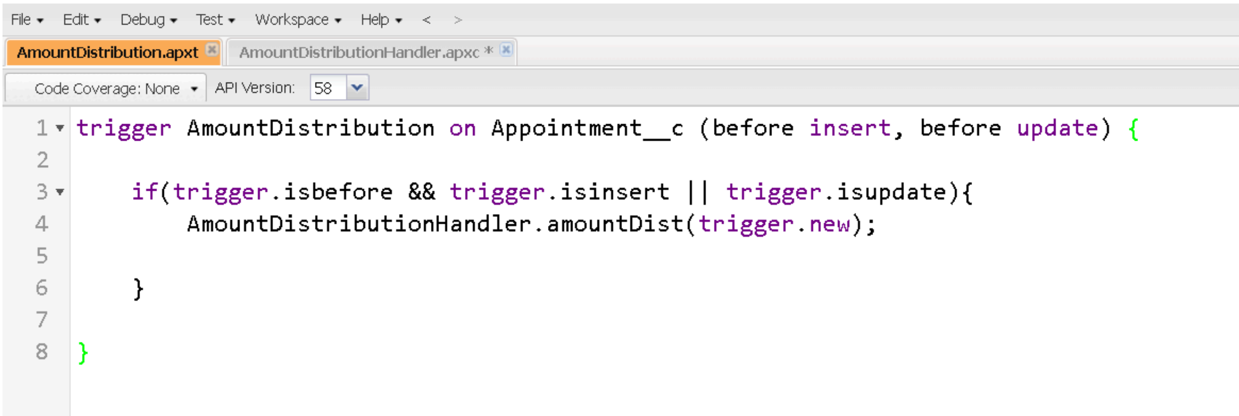
Trigger [trigger name] on [object name](Before/After event)

{

}

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);

    }

}
```

<u>CHAPTER-14</u>

REPORTS

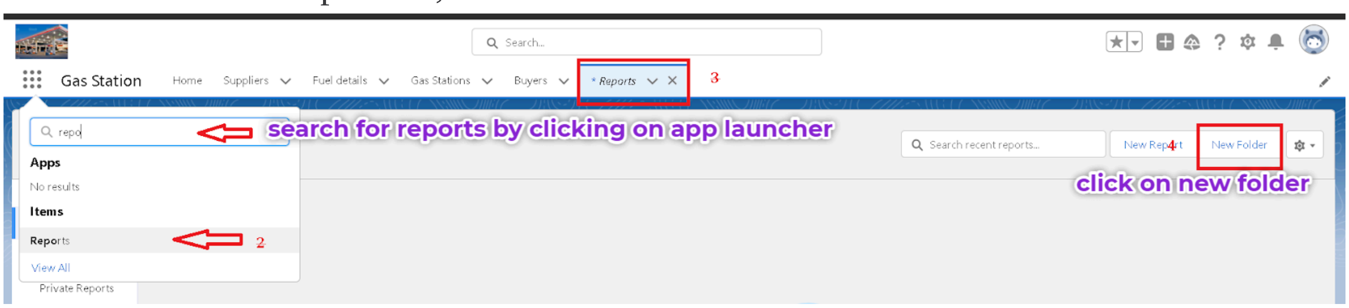
Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics.

Types of Reports in Salesforce

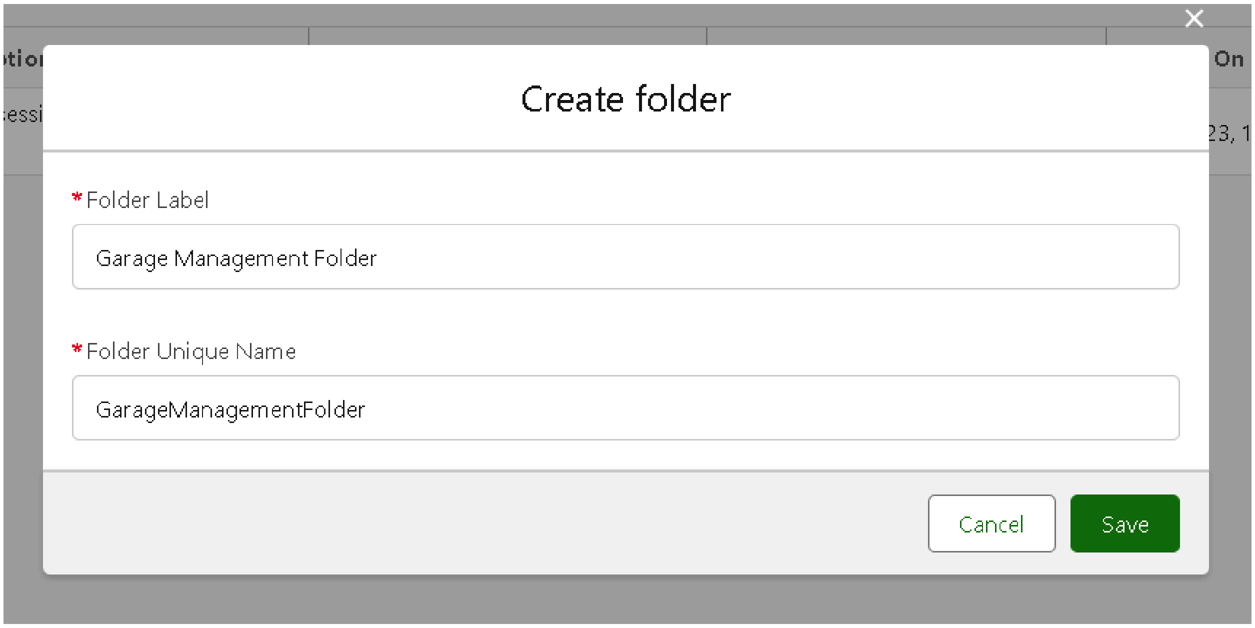
1. Tabular
2. Summary
3. Matrix
4. Joined Reports

➢ Create a Report Folder

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

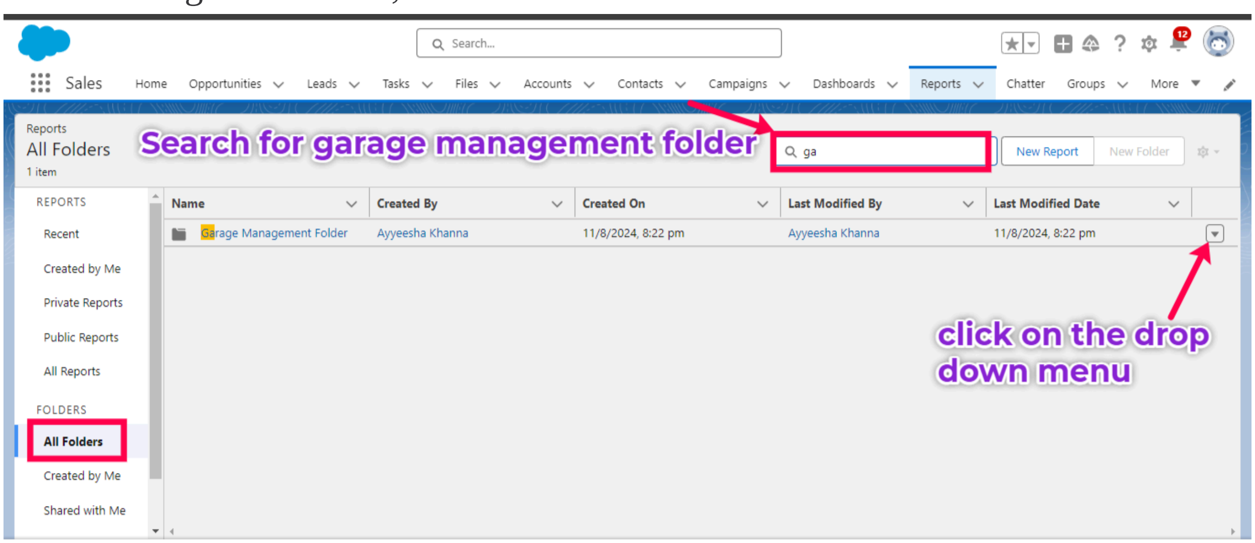


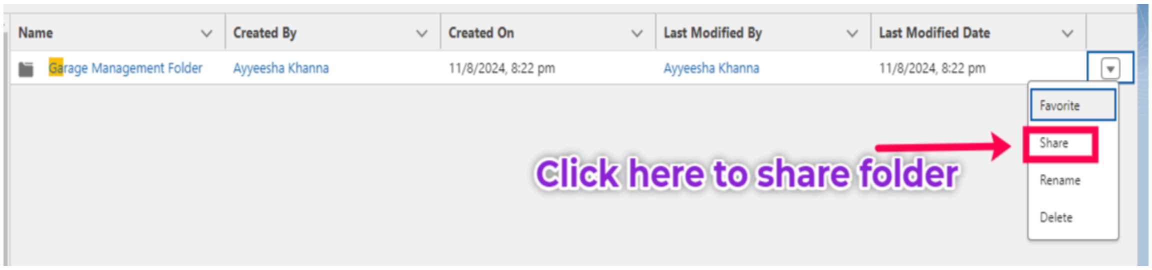
3. Give the Folder label as “Garage Management Folder”, Folder unique name will be auto populated.
4. Click save.



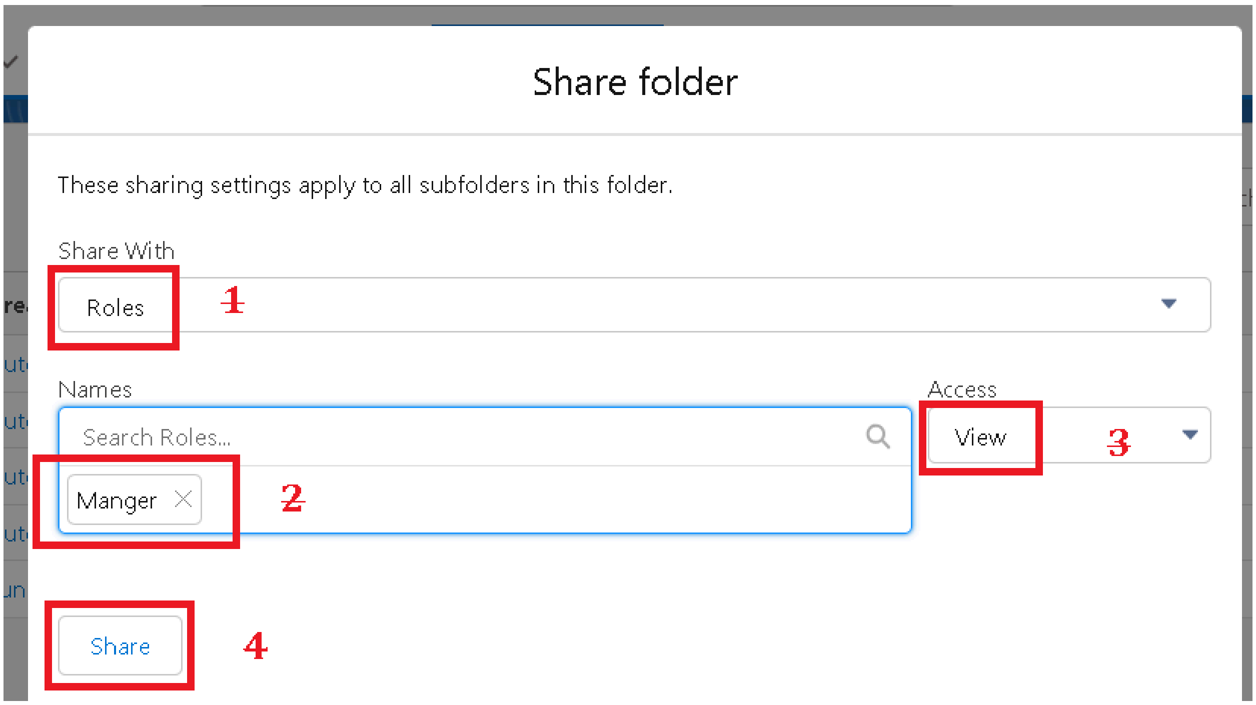
➤ Sharing a report folder

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.



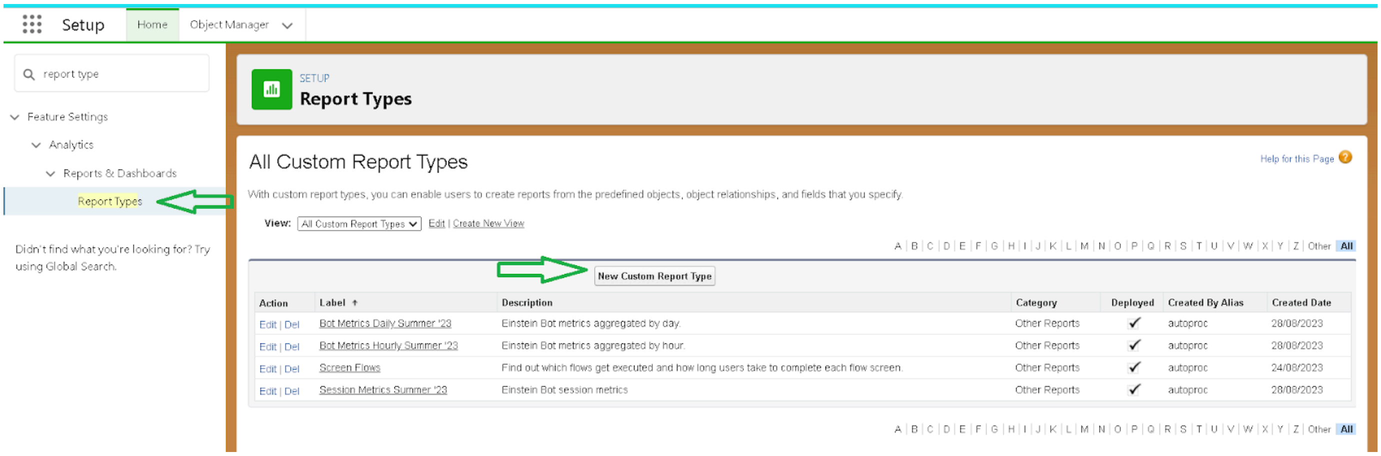


3. Select the share with as “roles”, in name field search for “manager”, give “view” as access for that role.
4. Then click share, and click on Done.

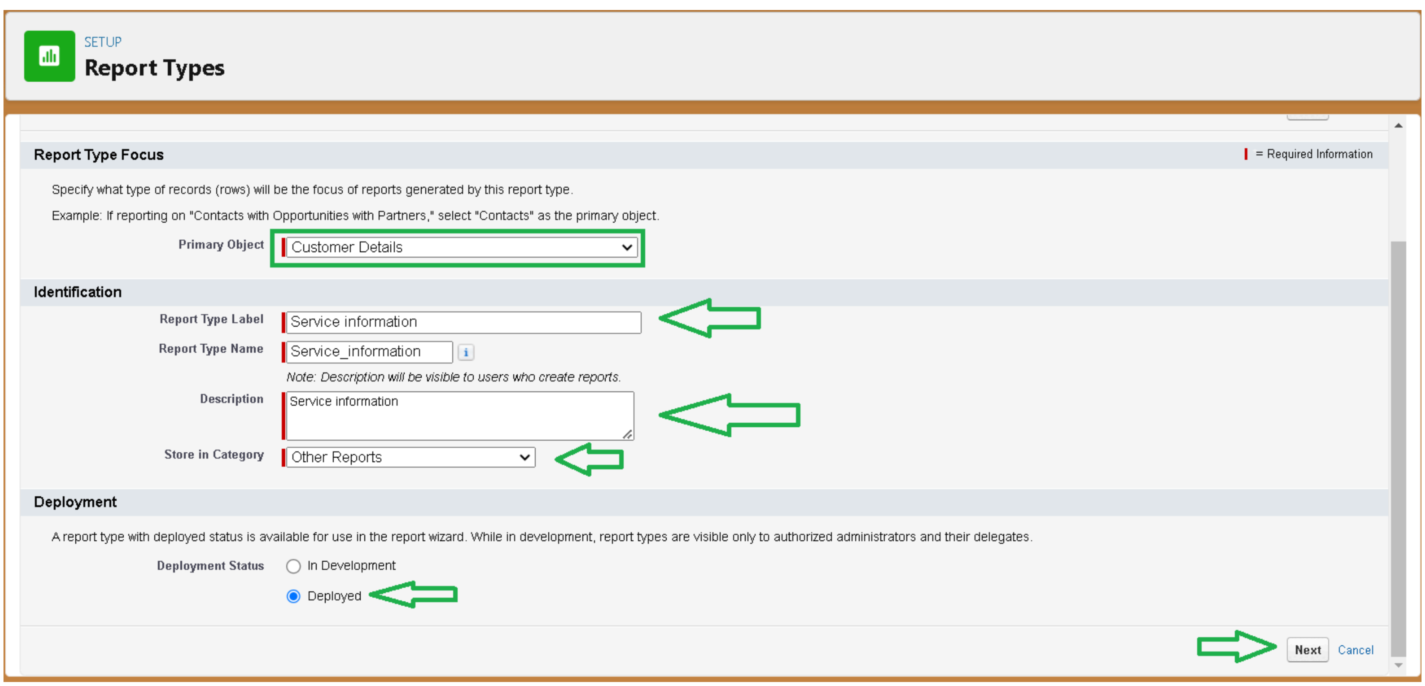


➤ Create Report Type

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

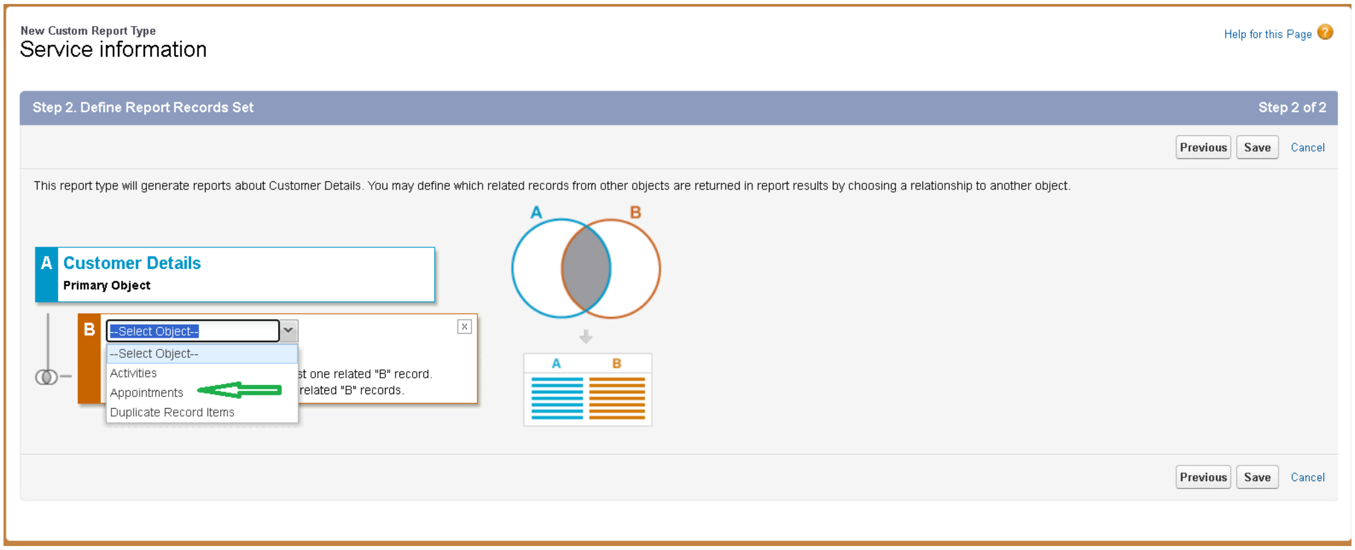


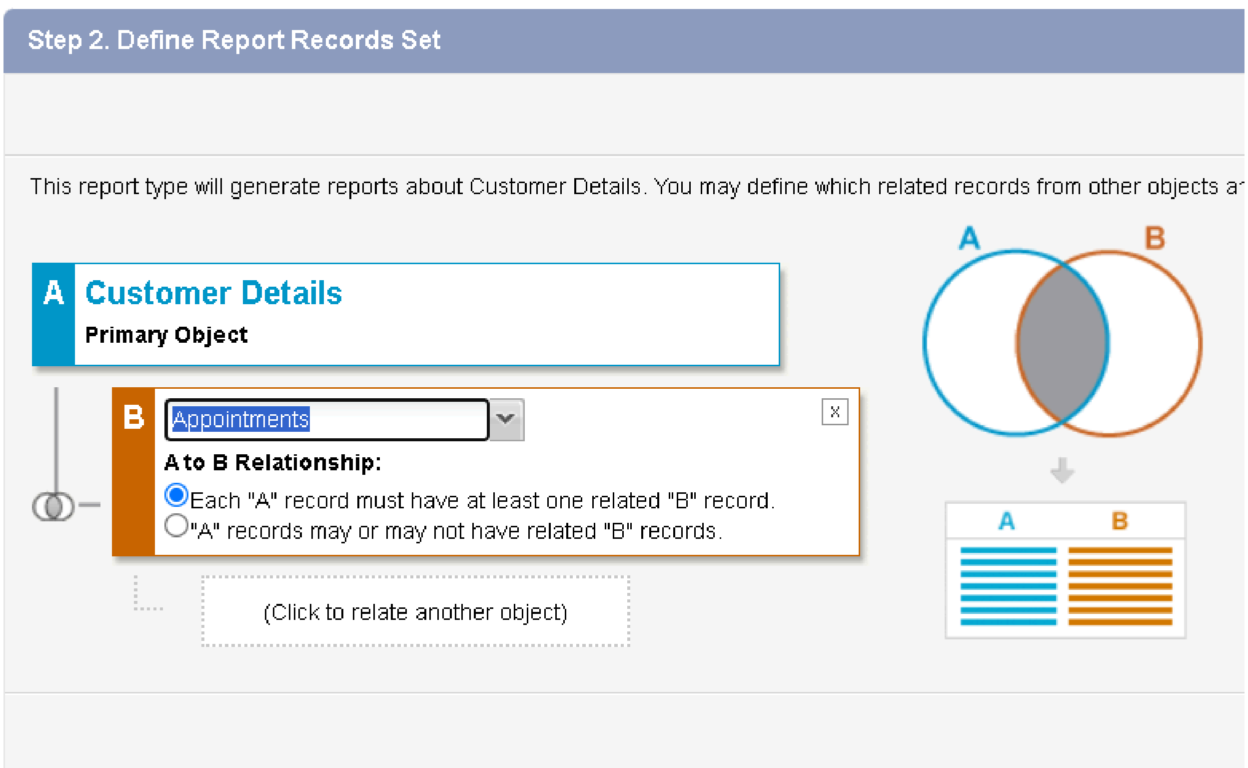
3. Select the Primary object as “ Customer details” .
4. Give the Report type Label as “ Service information ”
5. Report type Name is autopopulated.
6. Keep the Description as same.
7. Select Store in Category as “ other Reports ”
8. Select the deployment status as “ Depolyed ”, click on Next.



9. Now , Click on Related object box.

10.Click on Select Object, choose Appointment Object as shown in fig.



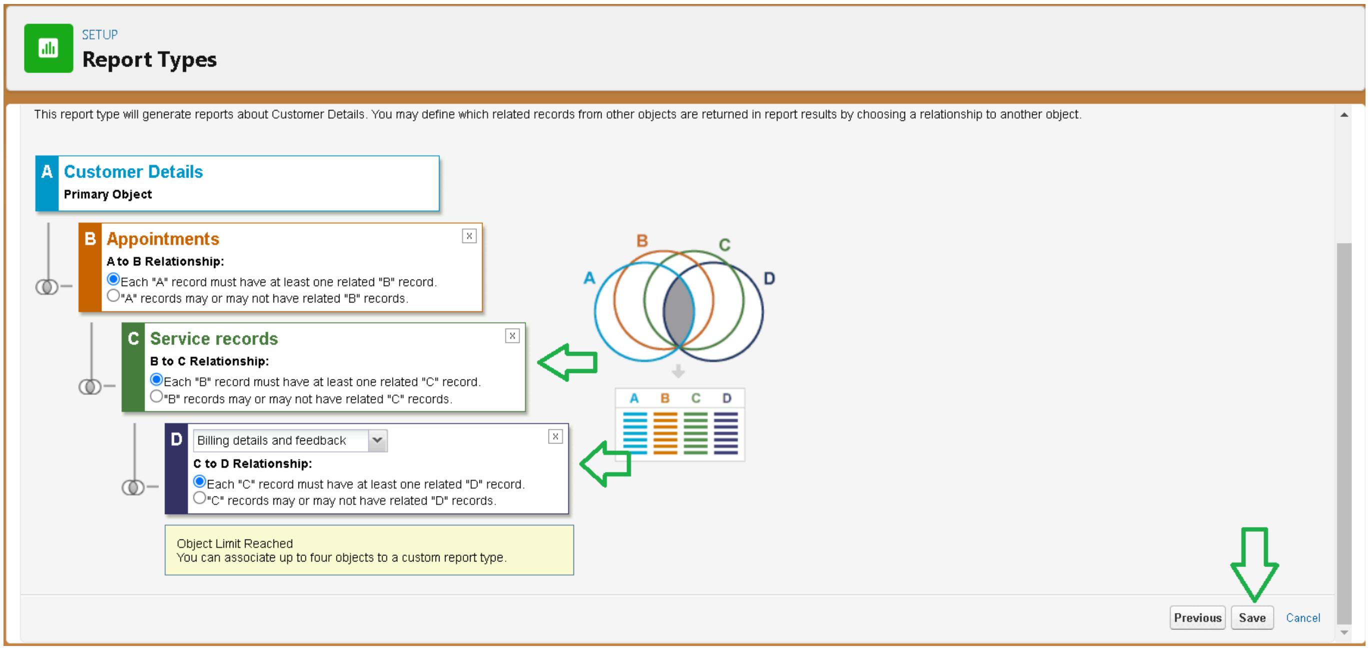


11.Again Click to relate another object.

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.



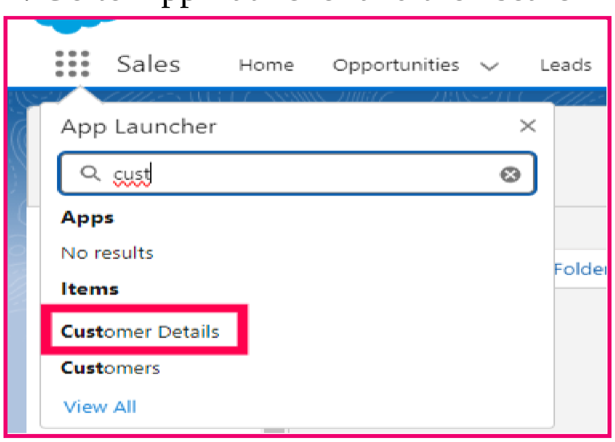
Note : Before creating report, create latest “10” records in every object.

Try to fill every field in each record for better experience.

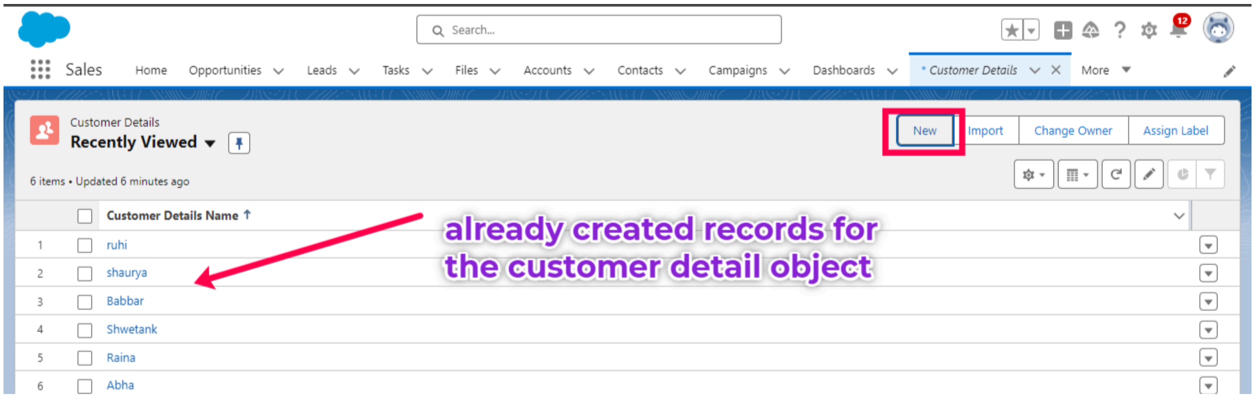
Lets start with creating records for every object first.

➢ Records for Customer Detail Object

1. Go to App Launcher and then search for the object "Customer Details"



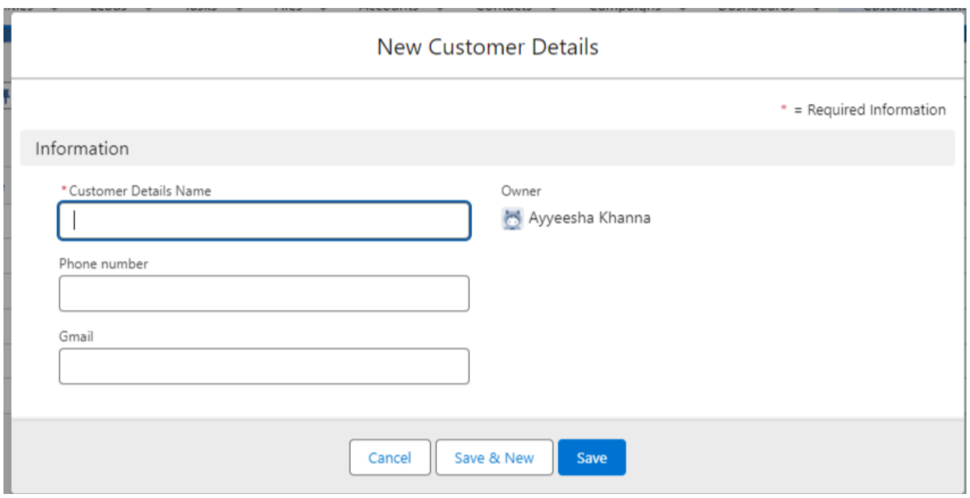
2. Click on the Customer Details and then click on new



3. Then, enter all the details:- name, phone no. and gmail

(Create any user of your choice)

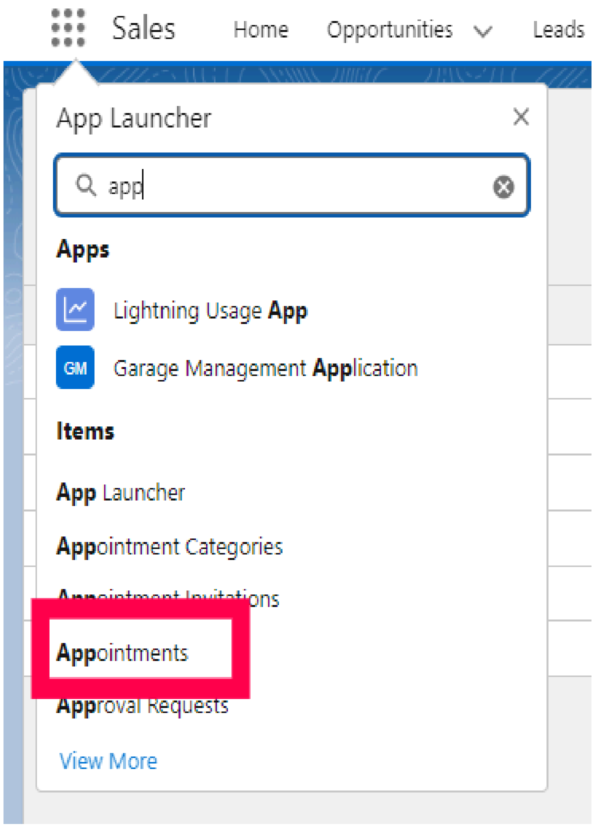
4. After entering all details, click on save



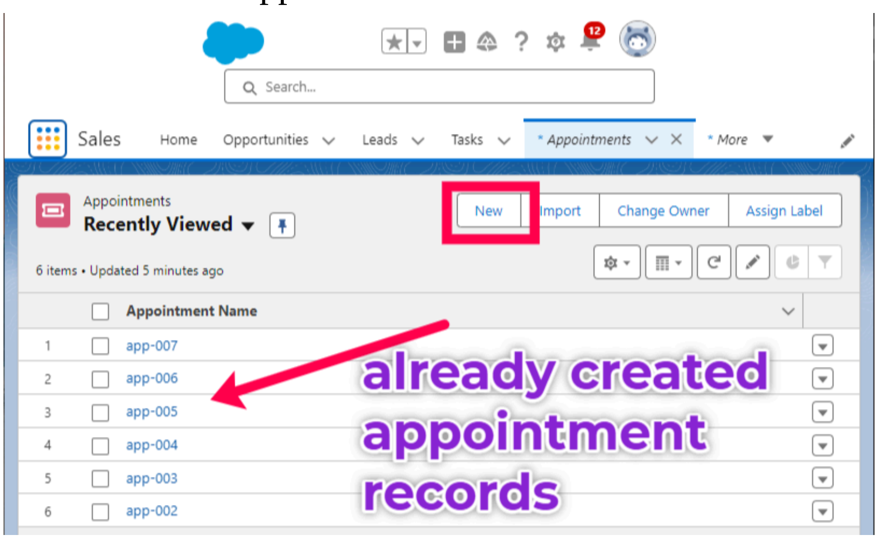
5. Now, Repeat the same for all the remaining objects (Appointments, Service records,Billing details and feedback)

➢ Records for Appointments Object

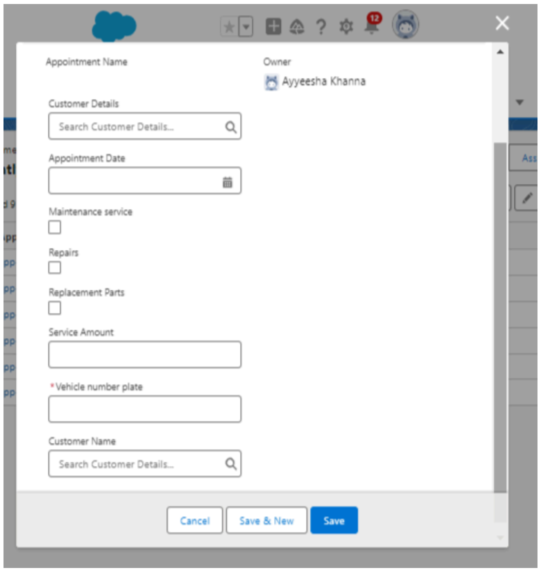
1. Go to App Launcher and then search for the object "Appointments"



2. Click on the Appointments and then click on new



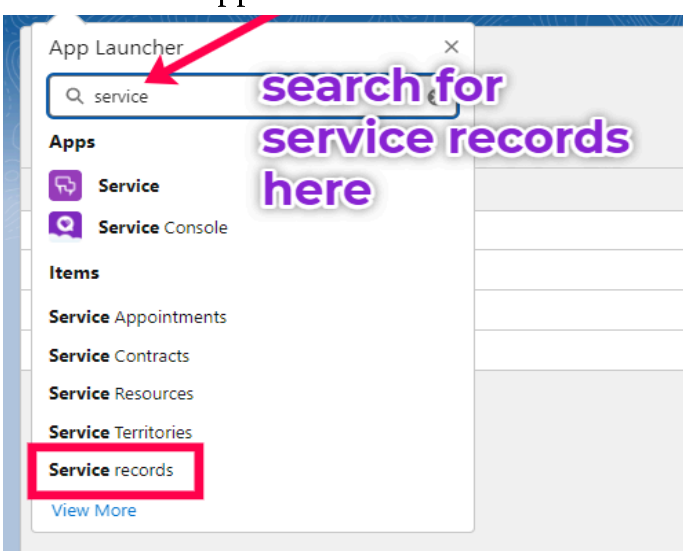
3. Enter all details in the fields mentioned below



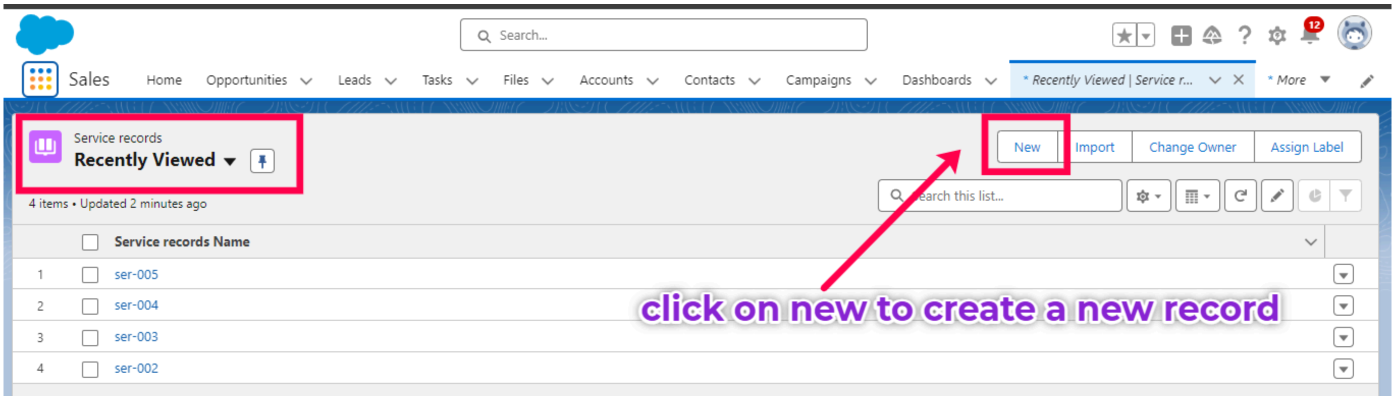
4. Select the customer in Customer details
5. Choose an Appointment date before the current date(Suppose the current date is 14/08/24 then choose any date before the 14th of Aug 2024)
6. Check atmost one from the given options Maintenance service, Replacement Parts n Repairs.
7. Enter the Service Amount
8. Enter a unique number for the Vehicle Number Plate.
9. Now, click on save.

➤ **Records for Service Record Object**

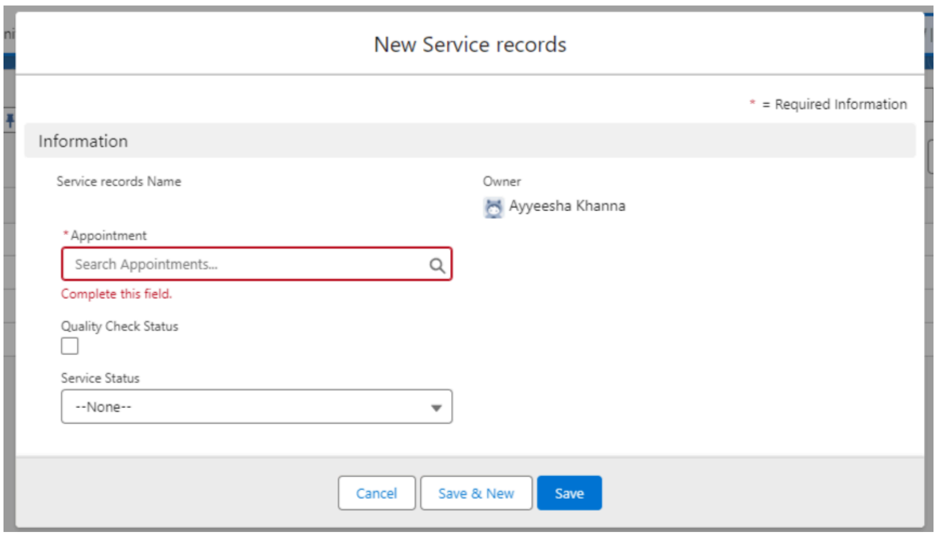
1. Go to App Launcher and then search for the object "Service Record"



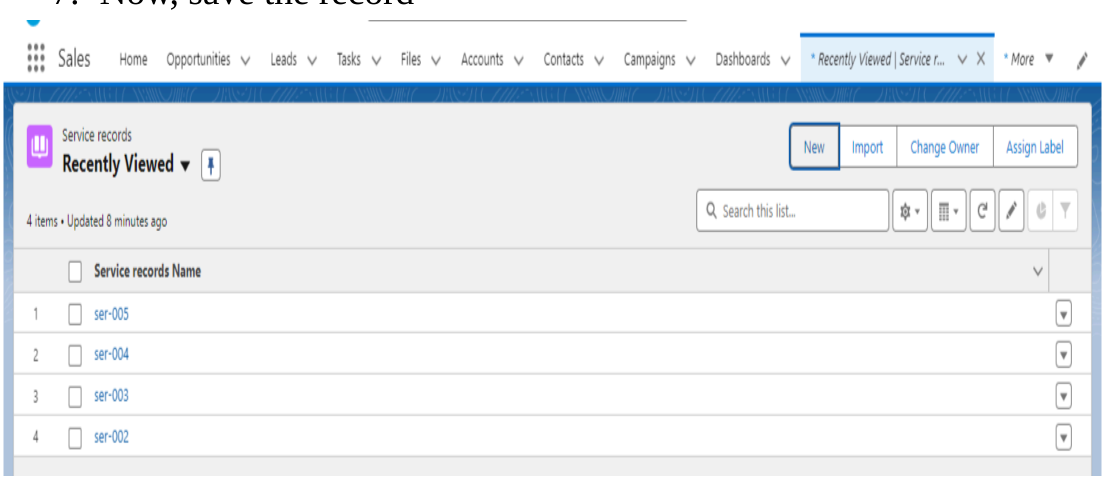
2. Click on the Service Records and then click on new



3. Enter all details in the fields mentioned below

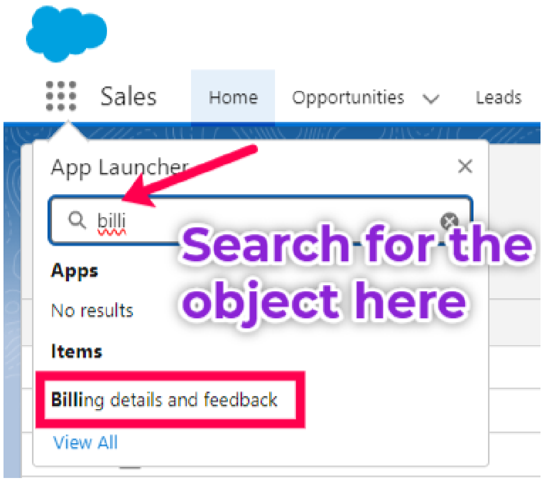


4. Select the appointment record
5. Check the uality Check Status.
6. Now, select the service startus from as pending or complete.
7. Now, save the record

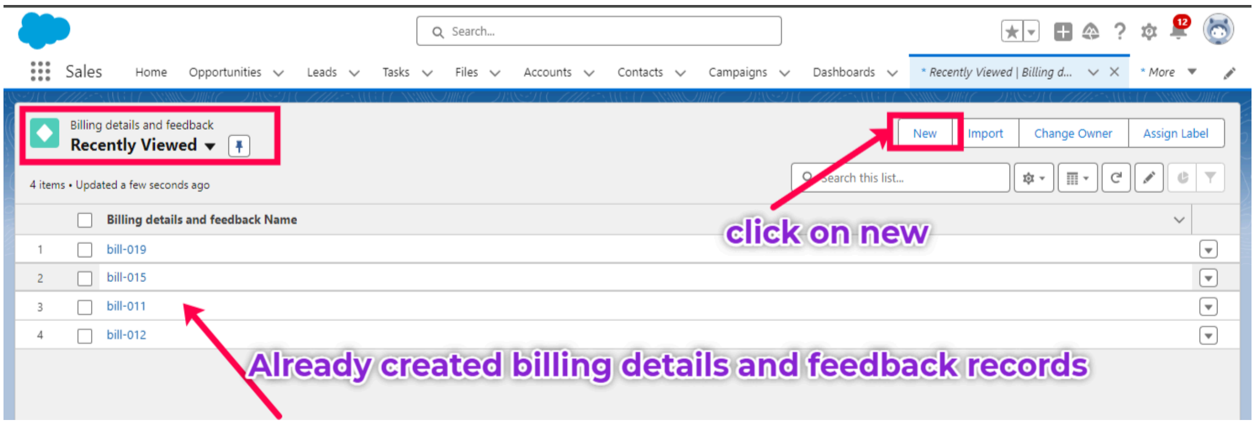


➢ **Records for Billing Details and Feedaback Object**

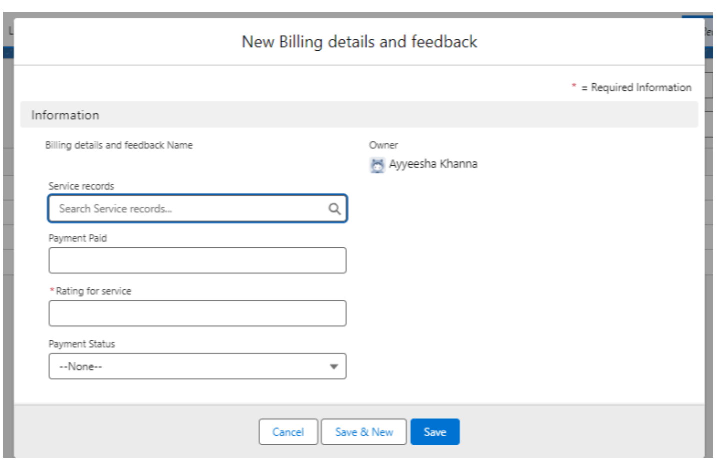
1. Go to App Launcher and then search for the object "Billing Detail and feedback " object



2. Click on theBilling Detail and feedback and then click on new

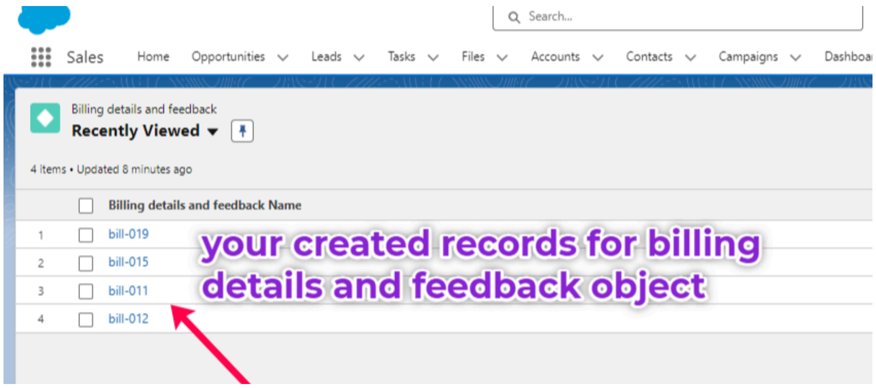


3. Enter all details in the fields mentioned below



4. Entrer the service records

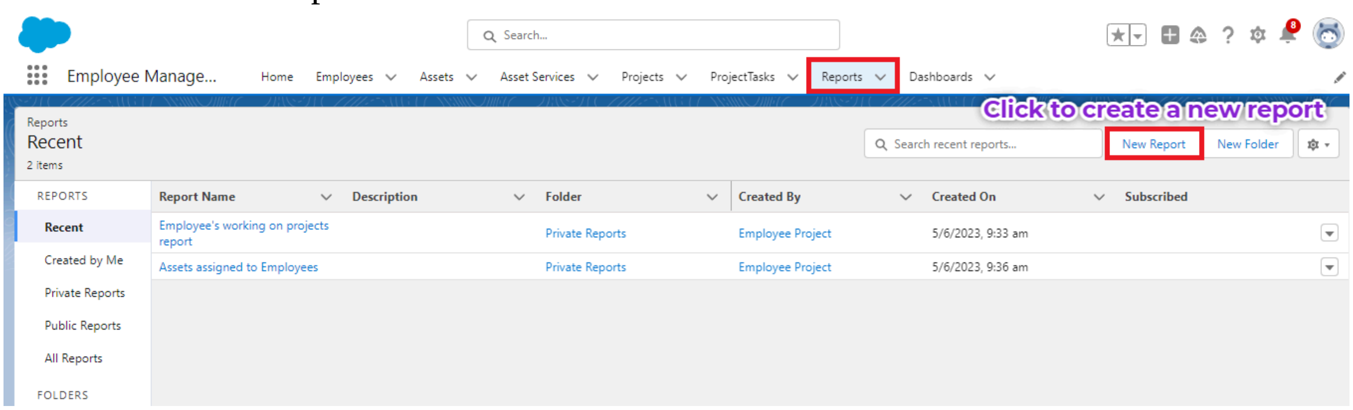
5. Enter the same amount as you did while creating the Appointment record for that particular customer
6. Select the rating for the service ranging between 0-5
7. Select the payment status as either pending or completed.



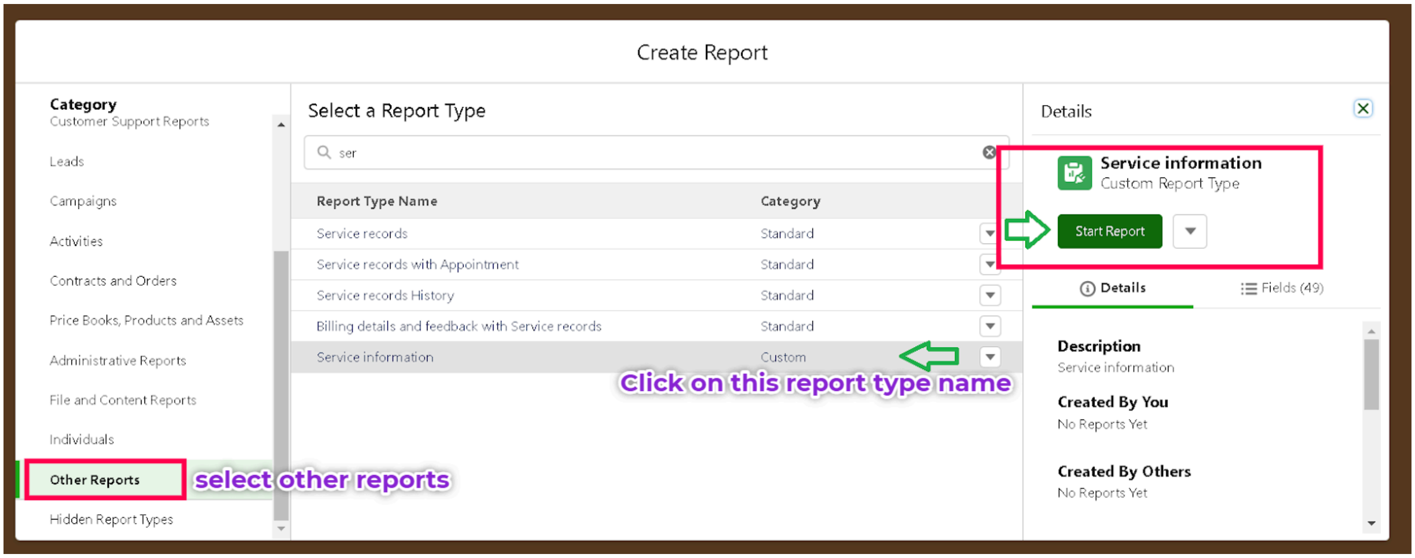
You have successfuly made records for each object.
Now, let's start by creating the report

➤ Create Report

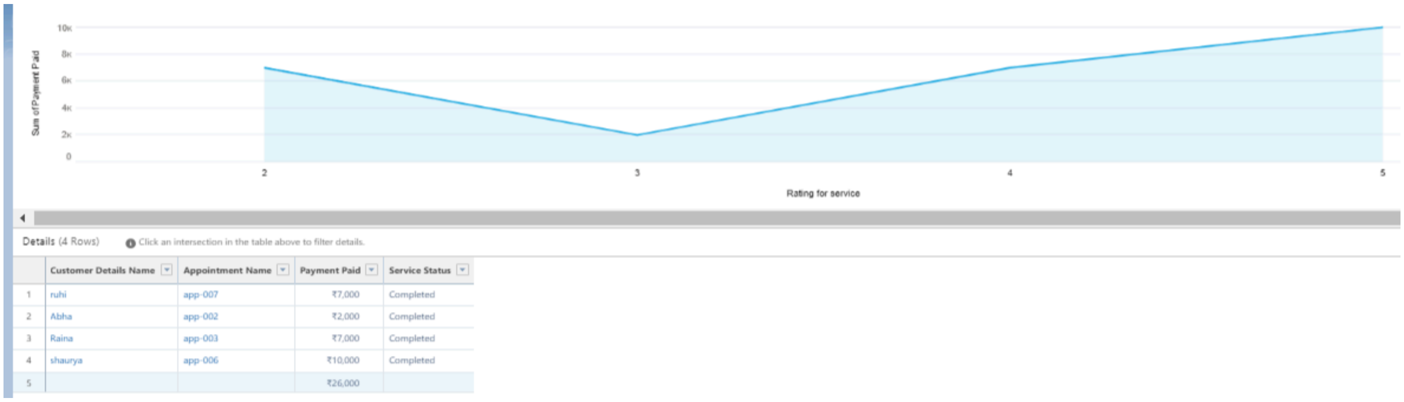
1. Go to the app >> click on the reports tab
2. Click New Report.



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 alredy, select the fields that mentioned below in column section.
   - Customer name
   - Appointment Date
   - Service Status
   - Payment paid
5. Remove the unnecessary fields.
6. Select the fields that mentioned below in GROUP ROWS section.
   - Rating for Service
7. Select the fields that mentioned below in GROUP COLUMN section.
   - 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.


Sum of Payment Paid
10k
8k
6k
4k
2k
0
2
3
4
5
Rating for service
Details (4 Rows)
Click an intersection in the table above to filter details.
Customer Details Name
Appointment Name
Payment Paid
Service Status
1 ruhi app-007 ₹7,000 Completed
2 Abha app-002 ₹2,000 Completed
3 Raina app-003 ₹7,000 Completed
4 shaurya app-006 ₹10,000 Completed
5 ₹26,000


Save Report
* Report Name
New Service information Report
Report Unique Name
New_Service_information_Report_oVu
Report Description
Folder
Garage Management Folder
Select Folder
Cancel
Save

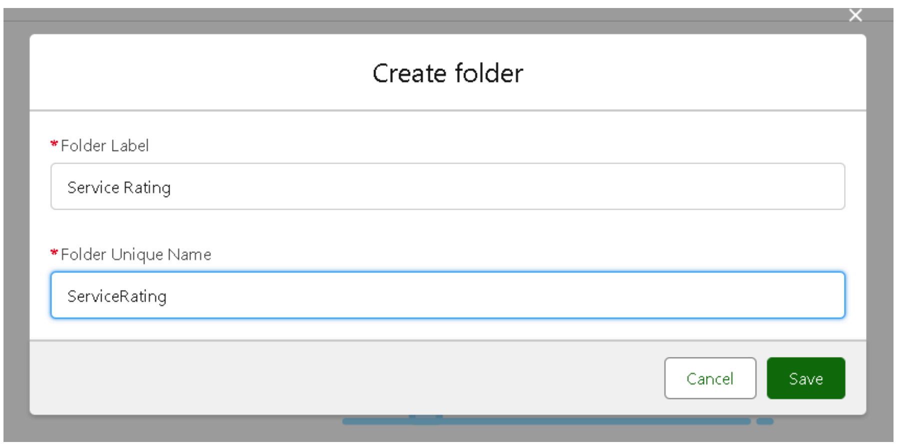
CHAPTER- 15

DASHBOARD

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.

➤ Create Dashboard Folder

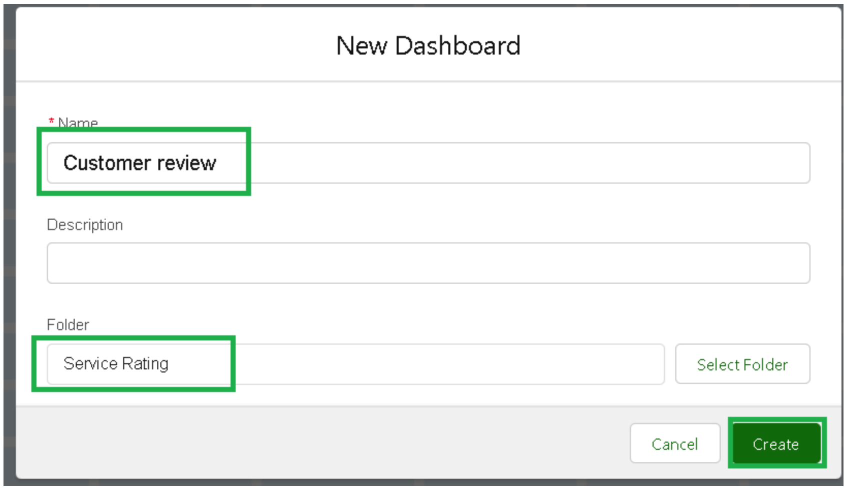
1. Click on the app launcher and search for dashboard.
2. Click on dashboard tab.
3. Click new folder, give the folder label as " Service Rating dashboard".
4. Folder unique name will be auto populated.
5. Click save.



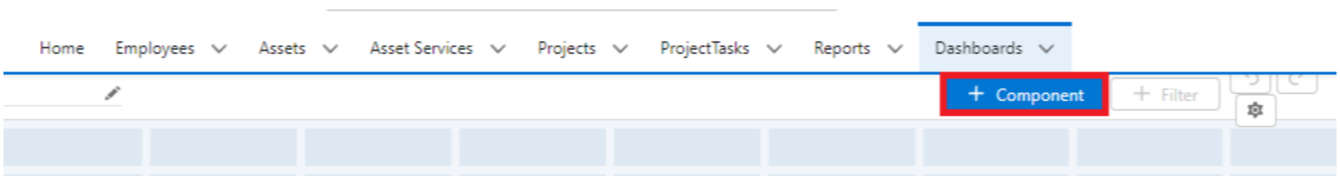
6. Follow the same steps, form milestone 14(i.e,. Repprt --> Sharing a Report) , and activity 2, and provide the sharing settings for the folder that just created.

➤ Create Dashboard

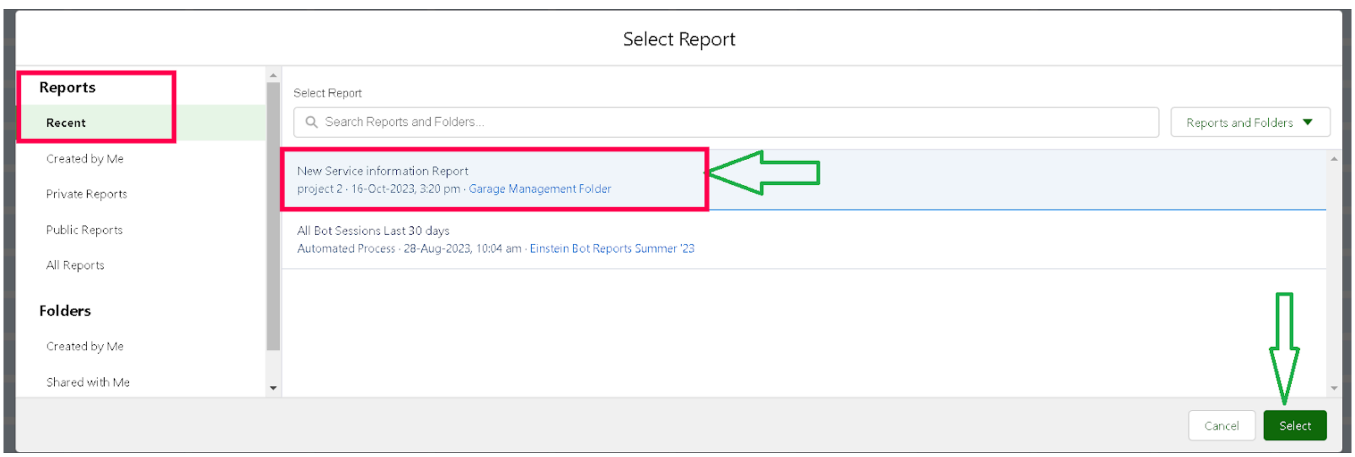
1. Go to the app >> click on the Dashboards tabs.
2. Give a Name and select the folder that created, and click on create.



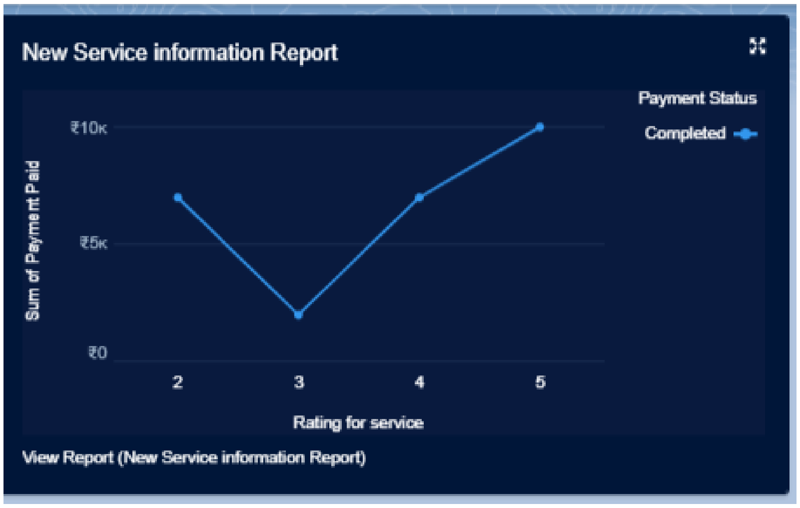
3. Select add component.



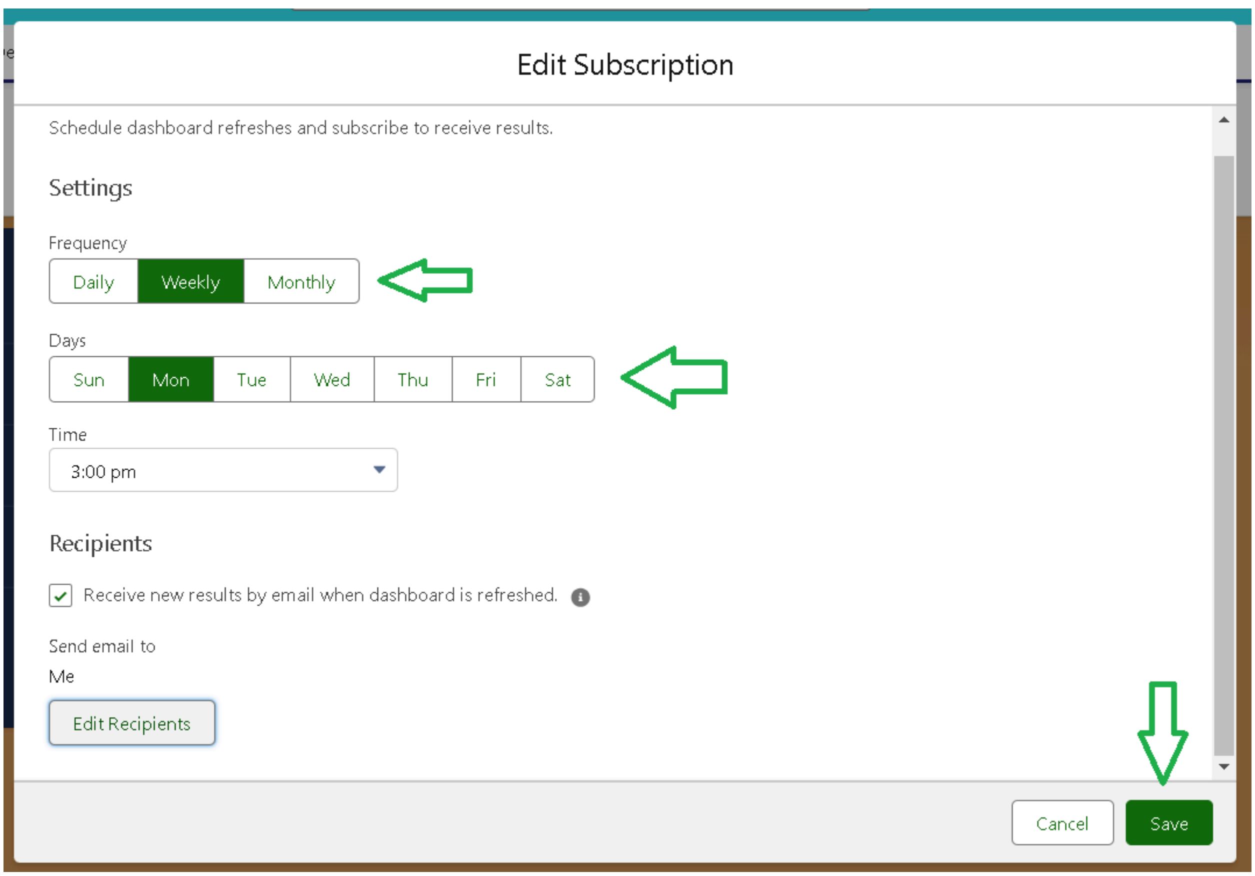
4. Select a Report and click on select.

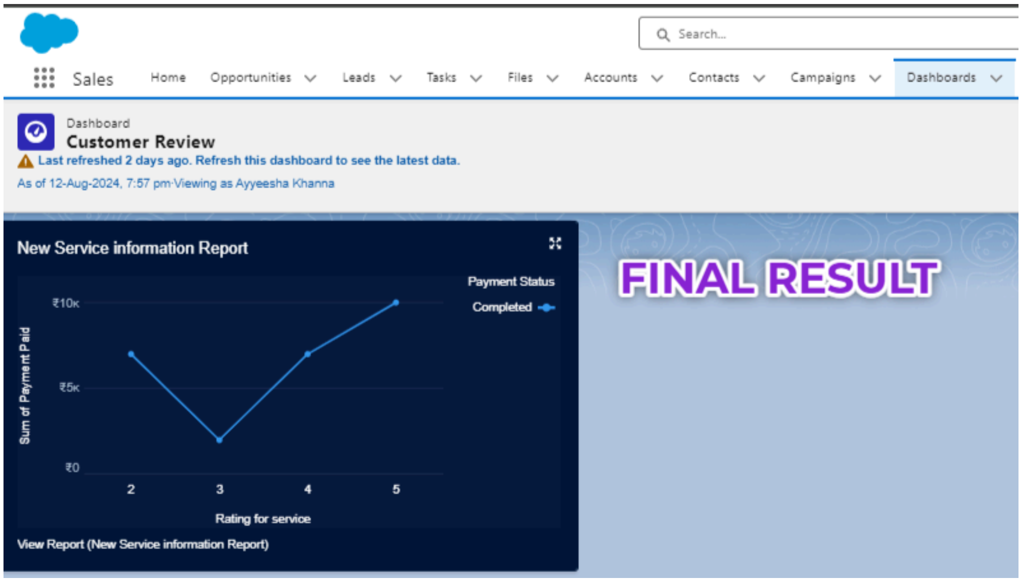


5. Select the Line Chart. Change the theme.
6. Click Add then click on Save and then click on Done.
7. Preview is shown below.



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.




Search...
Sales
Home
Opportunities
Leads
Tasks
Files
Accounts
Contacts
Campaigns
Dashboards
Dashboard
Customer Review
Last refreshed 2 days ago. Refresh this dashboard to see the latest data.
As of 12-Aug-2024, 7:57 pm·Viewing as Ayyeesha Khanna
New Service information Report
FINAL RESULT
Payment Status
Completed
₹10K
₹5K
₹0
Sum of Payment Paid
2
3
4
5
Rating for service
View Report (New Service information Report)