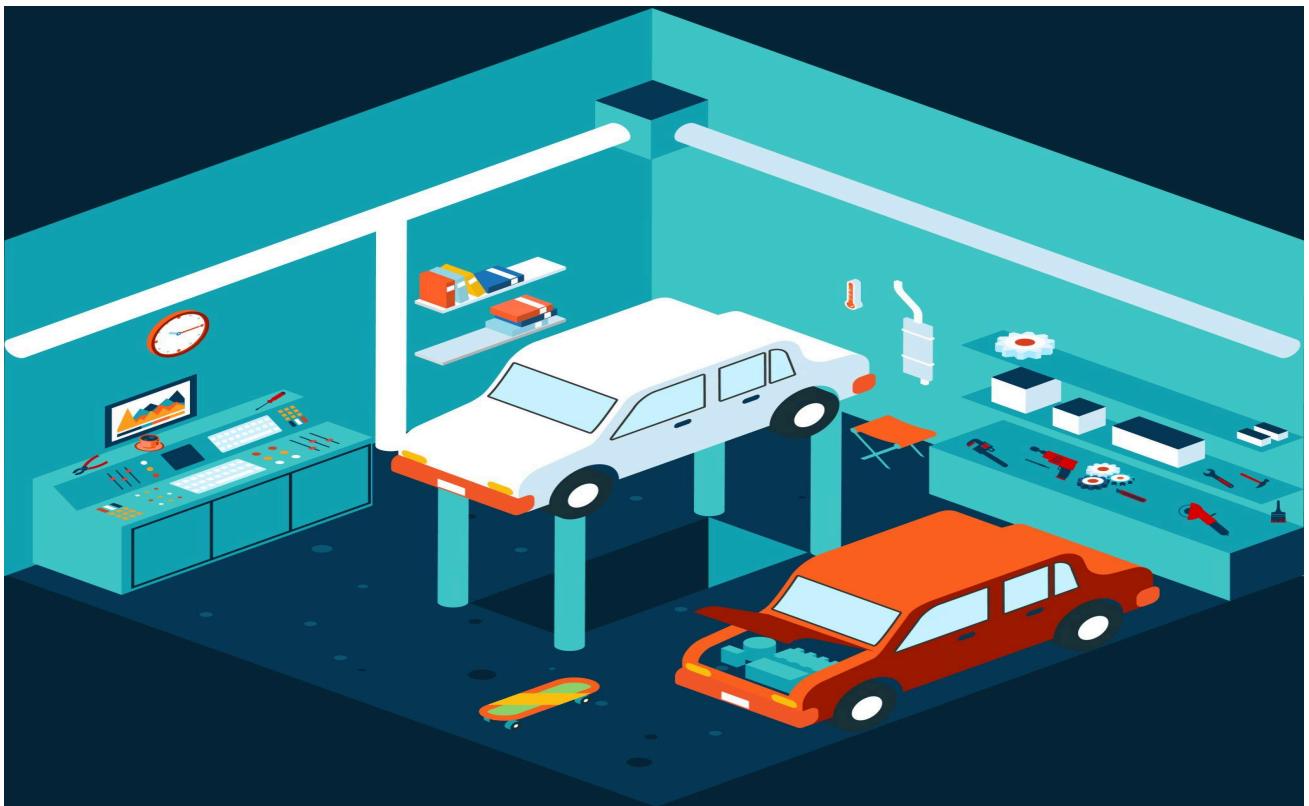


Garage Management system

Project Document

Prepared by
ARUNPRASANNA D
arunprasanna7123@gmail.com
Mahendra Engineering College
Namakkal



INDEX

Garage Management system

Milestone 1:Salesforce

- Creating Developer Account
- Account Activation

Milestone 2: Object

- Create Customer DetailsObject
- Create Appointment Object
- Create Service records Object
- Create Billing details and feedback Object

Milestone 3: Tabs

- Creating a Custom Tab
- Creating Remaining Tabs

Milestone 4: The Lightning App

- Create a Lightning App

Milestone 5: Fields

- Creation of fields for the Customer Details object
- Creation of Lookup Fields
- Creation of Checkbox Fields
- Creation of date Fields
- Creation of Currency Fields
- Creation of Text Fields
- Creation of Picklist Fields
- Creating Formula Field in Service records Object

Milestone 6: Validation rule

- To create a validation rule to an Appointment Object
- To create a validation rule to an Service records Object
- To create a validation rule to an Billing details and feedback Object

Milestone 7: Duplicate rule

- To create a matching rule to an Customer details Object
- To create a Duplicate rule to an Customer details Object

Milestone 8: Profiles

- Manager Profile
- Sales person Profile

Milestone 9: Role & Role Hierarchy

- Creating Manager Role
- Creating another roles

Milestone 10: Users

- Create User
- Creating another users

Milestone 11: Public groups

- Creating New Public Group

Milestone 12: Sharing Setting

- Creating Sharing settings

Milestone 13: Flows

- Create a Flow

Milestone 14: Apex Trigger

- Apex handler

Milestone 15: Reports

- Create a report folder
- Sharing a report folder
- Create Report Type
- Create Report

Milestone 16: Dashboards

- Create Dashboard Folder
- Create Dashboard

Garage Management System

A garage management system (GMS) is a comprehensive software solution designed to revolutionize the way auto repair shops and garages operate by integrating various aspects of their daily activities into a single, streamlined platform. This advanced system facilitates a range of functions critical to efficient garage management, including appointment scheduling, work order tracking, inventory control, customer relationship management (CRM), billing and invoicing, and comprehensive reporting and analytics.

At the core of a GMS is its **appointment scheduling** functionality, which allows customers to easily book, reschedule, or cancel service appointments either online or through the system. This feature not only enhances customer convenience but also significantly reduces administrative workload by automating appointment confirmations and reminders. Automated notifications help minimize no-shows and ensure that both customers and staff are kept informed of upcoming appointments, contributing to a smoother operational flow and more effective time management.

The **work order management** component is essential for maintaining operational efficiency. It allows for the creation, tracking, and management of detailed work orders that outline the specifics of each service request, including job descriptions, labor estimates, and required parts. This ensures that tasks are assigned accurately, progress is monitored in real time, and any issues are addressed promptly. By providing clear visibility into the status of each job, work order management helps streamline workflows, reduce delays, and maintain high standards of service quality.

The **inventory management** within a GMS is another crucial feature that supports effective operations. This component tracks the availability of parts and supplies, providing real-time insights into stock levels. It automatically alerts managers when inventory is low and can trigger reorder processes to prevent stockouts. This proactive approach to inventory management helps ensure that the garage is always equipped with the necessary parts to complete repairs efficiently, reducing downtime and minimizing disruptions to service delivery.

The **customer relationship management (CRM)** capabilities of a GMS enhance the way garages interact with their clients. By storing detailed customer information, including contact details, service history, and preferences, the system enables personalized service and targeted communication. This helps build stronger customer relationships, improve satisfaction, and foster loyalty. CRM features also support

marketing efforts by facilitating the creation of targeted promotions and follow-up communications based on customer data.

In terms of financial management, a GMS includes **billing and invoicing** functionalities that streamline the payment process. It generates accurate invoices, tracks payments, and manages accounts receivable. This automation reduces the risk of billing errors and simplifies financial management by providing clear and organized records of transactions. Integrated financial reporting tools also offer insights into revenue, expenses, and overall financial performance, helping garage owners make informed decisions and manage cash flow effectively.

Reporting and analytics are integral to a GMS, offering valuable insights into various aspects of garage operations. The system generates detailed reports on key performance indicators such as service efficiency, customer satisfaction, and revenue trends. These analytics help identify operational strengths and areas for improvement, enabling data-driven decision-making that supports business growth and optimization.

Furthermore, many modern GMS platforms provide **mobile access** and **customer portals**, enhancing overall convenience and accessibility. Mobile access allows staff to manage operations from anywhere, improving flexibility and responsiveness. Customer portals, on the other hand, enable clients to track the status of their service requests, view their service history, and make payments online, thereby enhancing their overall experience and engagement with the garage.

In summary, a garage management system is a powerful tool that integrates various operational functions into a single, cohesive platform. By enhancing appointment scheduling, work order management, inventory control, customer relationship management, billing, and reporting, a GMS significantly improves the efficiency and effectiveness of auto repair shops and garages. It not only streamlines daily operations but also provides valuable insights and tools to drive business growth, improve customer satisfaction, and ensure long-term success.

Salesforce

Introduction:

Salesforce is a prominent cloud-based customer relationship management (CRM) platform designed to help businesses streamline and optimize their customer interactions. Founded in 1999 by Marc Benioff, Salesforce has evolved into a comprehensive suite of tools that support sales, customer service, marketing, commerce, and analytics. Its core products include Sales Cloud, Service Cloud, Marketing Cloud, and Commerce Cloud, among others. By providing a unified platform that centralizes customer data and automates various processes, Salesforce enables organizations to improve customer satisfaction, drive sales growth, and enhance overall operational efficiency. Its extensive ecosystem, including the AppExchange marketplace, allows for further customization and integration with other business applications, making it a versatile solution for companies of all sizes.

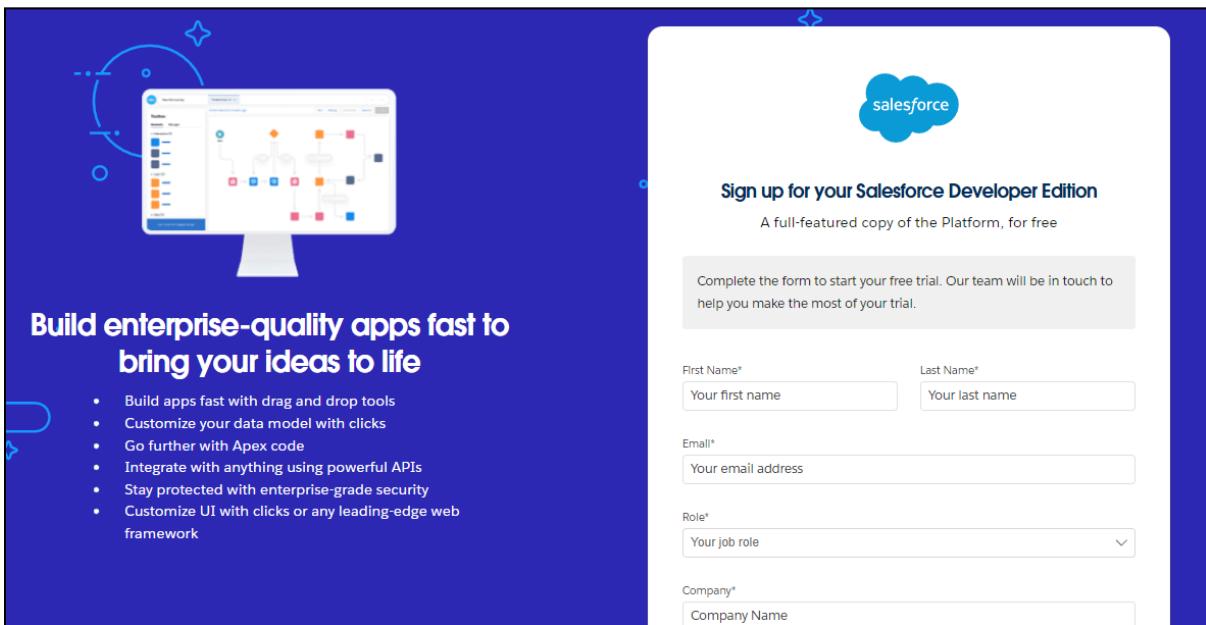
What Is Salesforce?

Salesforce is a leading cloud-based customer relationship management (CRM) platform that helps businesses manage their sales, customer service, marketing, and other critical functions. It offers a variety of tools and services, including Sales Cloud for managing sales operations, Service Cloud for customer support, Marketing Cloud for digital marketing automation, and Commerce Cloud for e-commerce solutions. Additionally, Salesforce provides a robust analytics platform and a marketplace called AppExchange for third-party integrations and applications. By centralizing customer information and automating various processes, Salesforce enables companies to enhance customer relationships, streamline workflows, and improve overall productivity.

Creating Developer Account:

Creating a developer org in salesforce.

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



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

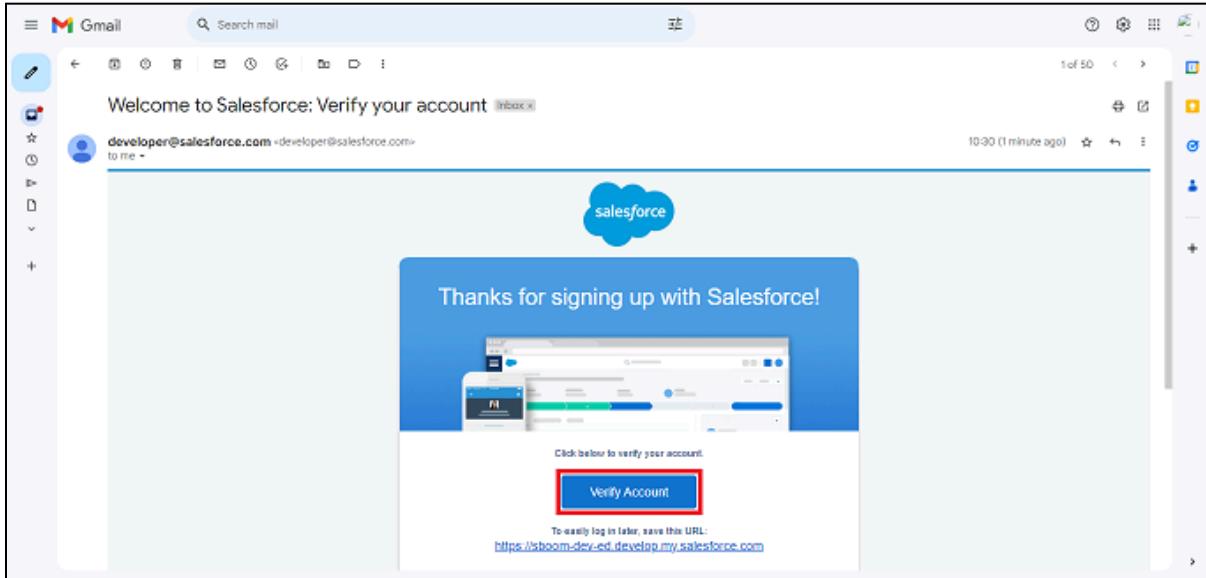
This need not be an actual email id, you can give anything in the format :

username@organization.com

Click on sign me up after filling these.

Account Activation

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



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

The screenshot shows the 'Change Your Password' page in Salesforce. At the top, it says 'Enter a new password for lead@sb.oom.' Below that, it lists requirements: 'Make sure to include at least:' followed by three bullet points: '8 characters', '1 letter', and '1 number'. A red box highlights the password input fields. The first field is labeled '* New Password' and contains '*****' with a status 'Good'. The second field is labeled '* Confirm New Password' and contains '*****' with a status 'Match'. Below these is a 'Security Question' section with a dropdown menu showing 'In what city were you born?'. Underneath is an 'Answer' field containing 'asdfghjkl'. A red box also highlights the 'Change Password' button at the bottom.

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

The screenshot shows the Salesforce Setup Home page. The left sidebar includes links like 'Setup Home', 'Service Setup Assistant', 'Multi-Factor Authentication Assistant', 'Release Updates', 'Lightning Experience Transition Assistant', 'Salesforce Mobile App', 'Lightning Usage', 'Optimizer', and 'ADMINISTRATION > Users'. The main content area features a 'Home' card with sections for 'Get Started with Einstein Bots', 'Mobile Publisher', and 'Real-time Collaborative Docs', each with a 'Get Started' button. The top navigation bar includes 'Search Setup', 'Home', 'Object Manager', and various icons for account management.

Object

What Is an Object?

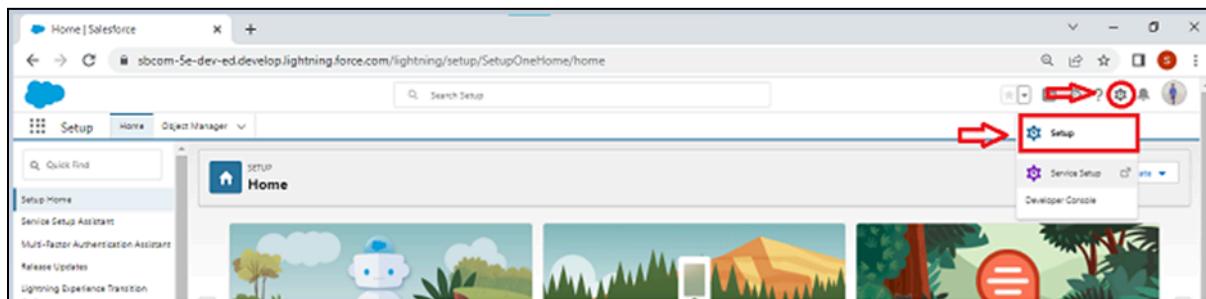
Salesforce objects are database tables that permit you to store data that is specific to an organization. What are the types of Salesforce objects

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.

New Custom Object | Salesforce

Custom Object Definition Edit

Custom Object Information

The singular and plural labels are used in tabs, page layouts, and records.

Label: Example: Account

Plural Label: Example: Accounts

Starts with account sound:

The Object Name is used when referencing the object via the API.

Object Name: Example: /sobjects/Account

Description:

Context sensitive help setting:

- Open the standard Salesforce.com Help & Training window
- Open a window using a Visualforce page

Content Name: Account

Enter Record Name Label and Format

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name: Example: Account Name

Data Type: Text

Optional Features

- Allow Reports
- Allow Activities
- Track Field History
- Allow in Chatter Groups
- Enable Licensing

Save | Save & New | Cancel

Optional Features

- Allow Reports
- Allow Activities
- Track Field History
- Allow in Chatter Groups
- Enable Licensing

Object Classification

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. Learn more.

- Allow Sharing
- Allow Bulk API Access
- Allow Streaming API Access

Deployment Status

- In Development
- Deployed

Search Status

When this setting is enabled, your users can find records of this object type when they search. Learn more.

- Allow Search

Object Creation Options (Available only when custom object is first created)

- Add Notes and Attachments related list to default page layout
- Launch New Custom Tab Wizard after saving this custom object

Save | Save & New | Cancel

4. Click on Save.

Create Customer Details 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 >> 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.

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.

Create Service records 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 >> Service records
 2. Plural label name >> Service records
 3. 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.

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.
 1. Enter the label name >> Billing details and feedback
 2. Plural label name >> Billing details and feedback
 3. 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.

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

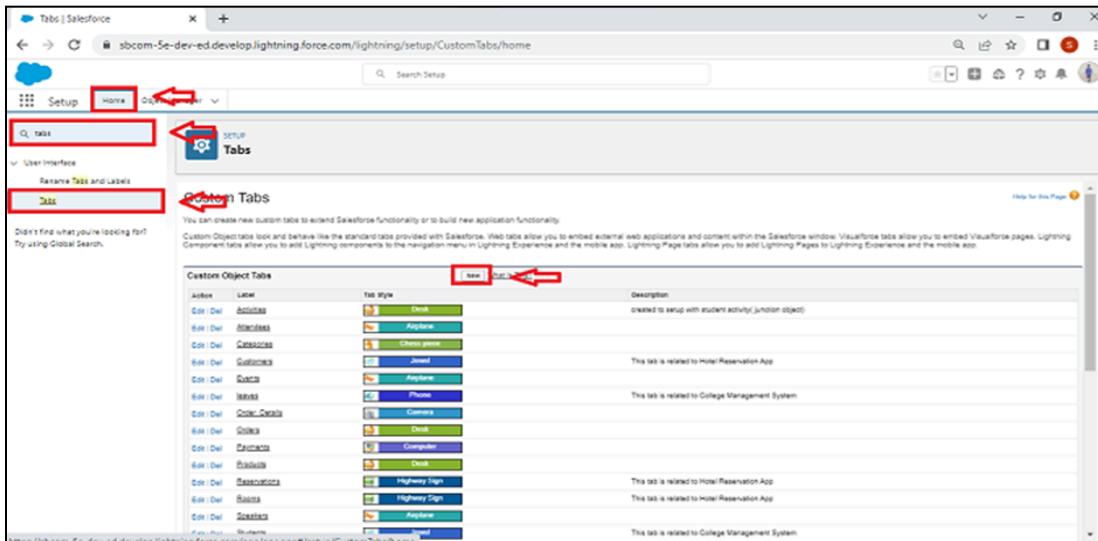
Lightning Page Tabs let you add Lightning Pages to the mobile app navigation menu.

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 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.

The screenshot shows the 'New Custom Object Tab' wizard, Step 1. The 'Object' dropdown is set to 'Customer Details'. The 'Tab Style' dropdown is set to 'Blank'. The 'Description' field is empty. The 'Next' button is visible at the bottom right.

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

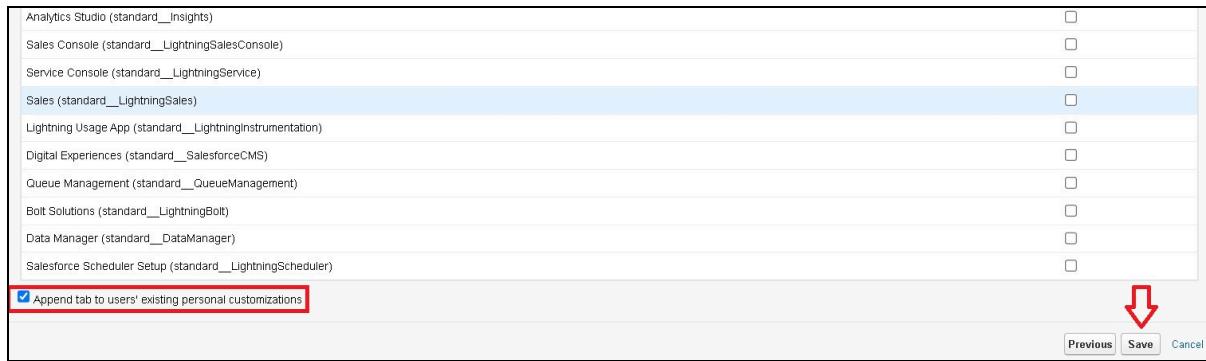
Save **Cancel**

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	 <input type="checkbox"/> Include Tab
Platform (standard__Platform)	<input type="checkbox"/>
Sales (standard__Sales)	<input type="checkbox"/>
Service (standard__Service)	<input type="checkbox"/>
Marketing (standard__Marketing)	<input type="checkbox"/>
Sample Console (standard__ServiceConsole)	<input type="checkbox"/>
High Volume Customer Portal User	<input type="checkbox"/>
Authenticated Website User	<input type="checkbox"/>
App Launcher (standard__AppLauncher)	<input type="checkbox"/>



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 .

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:

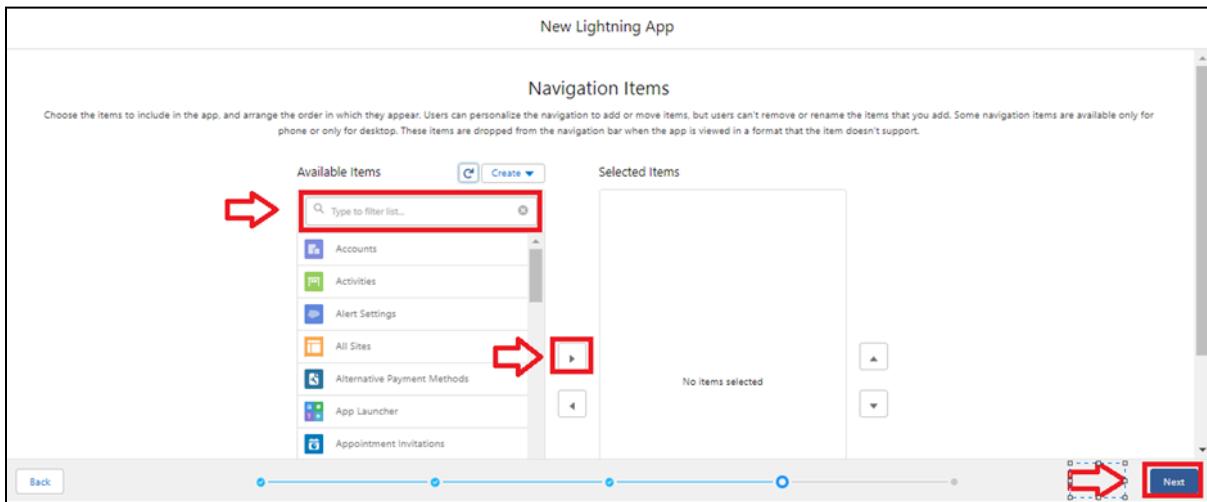
1. Go to setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.

The screenshot shows the Salesforce App Manager interface. At the top, there's a search bar with "app manager" typed in. To the right of the search bar is a "New Lightning App" button, which is highlighted with a red box and a red arrow pointing to it. Below the search bar, there's a "Clone Apps(Beta)" button, also highlighted with a red box and a red arrow pointing to it. The main area displays a list of existing apps, with columns for App Name, Developer Name, Description, Last Modified, App Type, and various edit icons. The list includes apps like All Tabs, Analytics Studio, App Launcher, and Data Manager.

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.

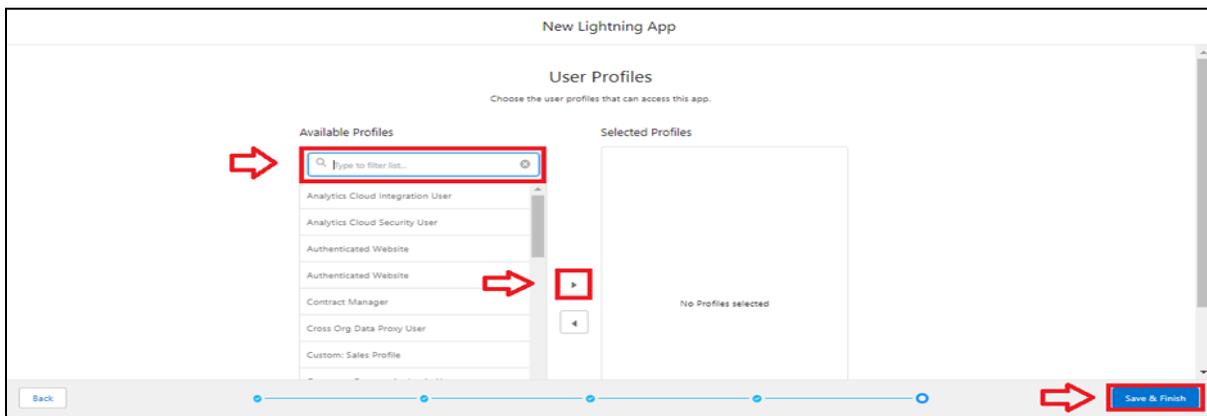
The screenshot shows the "New Lightning App" configuration page. The title is "New Lightning App". Under "App Details & Branding", there are sections for "App Details" and "App Branding". In the "App Details" section, there is an input field labeled "App Name" with the placeholder "Name your app...", which is highlighted with a red box and a red arrow pointing to it. In the "App Branding" section, there are fields for "Image" (with an "Upload" button) and "Primary Color Hex Value" (#0070D2). There are also "Org Theme Options" checkboxes. At the bottom, there's an "App Launcher Preview" section and a "Next" button, which is highlighted with a red box and a red arrow pointing to it.

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.

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

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.

The screenshot shows the Salesforce Object Manager interface. At the top, there's a search bar with 'cus' and buttons for 'Schema Builder' and 'Create'. Below the header, it says '2 Items, Sorted by Label'. A table lists two objects:

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Customer	Customer	Standard Object			
Customer Details	Customer_Details__c	Custom Object		05/10/2023	<input type="button" value="▼"/>

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

The screenshot shows the 'Fields & Relationships' page for the 'Customer' object. On the left, a sidebar lists various setup options like Page Layouts, Lightning Record Pages, etc. The main area shows a table of existing fields:

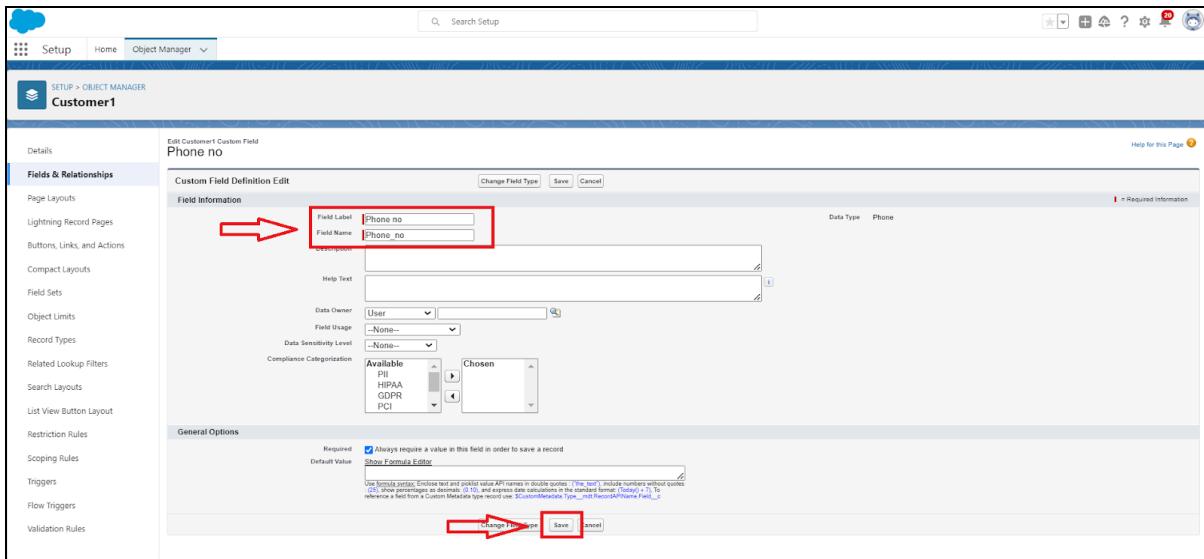
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedBy	Lookup(User)		
current Status	current_Status__c	Picklist		
Customer Name	Name	Text(80)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Email id	Email_id__c	Email (Unique)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Last Modified By	LastModifiedBy	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Permanent Address	Permanent_Address__c	Text Area(255)		
Phone no	Phone_no__c	Phone		

At the top right, there are buttons for 'New', 'Deleted Fields', 'Field Dependencies', and 'Set History Tracking'. A red arrow points from the 'Customer1' label in the sidebar to the 'Customer' object in the table. Another red arrow points from the 'New' button at the top right to the 'New' button in the table header.

3. Select Data Type as a “Phone”

The screenshot shows the 'Data Types' selection dialog for a new field. On the left, a list of data types includes Currency, Date, Date/Time, Email, Geolocation, Number, Percent, and Phone. The 'Phone' option is highlighted with a red box and a red arrow pointing to its description. The description for 'Phone' states: 'Allows users to enter any phone number. Automatically formats it as a phone number.' Other options like 'Text' and 'Text Area' are also listed with their descriptions.

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.

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.

The screenshot shows the Salesforce Setup interface with the Object Manager tab selected. A search bar at the top right contains the text "app". Below the search bar, there are buttons for "Schema Builder" and "Create". The main area displays a table of objects. The first row, "Appointment", is highlighted with a red border. The columns are labeled: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. The "Appointment" row has "Appointment_c" in the API Name column, "Custom Object" in the Type column, and "24/08/2023" in the Last Modified column.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Appointment	Appointment_c	Custom Object		24/08/2023	✓
Appointment Category	AppointmentCategory	Standard Object			
Appointment Invitation	AppointmentInvitation	Standard Object			
Appointment Invitee	AppointmentInvitee	Standard Object			

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

The screenshot shows the "Fields & Relationships" page for the Appointment object. The left sidebar has options for "Details", "Fields & Relationships" (which is selected and highlighted with a red box), "Page Layouts", and "Lightning Record Pages". The main area shows a table of fields. The first field, "Appointment Date", is listed with the API name "Appointment_Date__c" and data type "Date". The second field, "Appointment Name", is listed with the API name "Name" and data type "Auto Number". At the top right of the table, there are buttons for "New", "Deleted Fields", "Field Dependencies", and "Set History Tracking". The "New" button is highlighted with a red box.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Appointment Date	Appointment_Date__c	Date		
Appointment Name	Name	Auto Number		

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

The screenshot shows the "Data Type" selection step in the field creation wizard. It asks to specify the type of information the custom field will contain. The "Data Type" section has several options: "None Selected" (radio button), "Auto Number", "Formula", "Roll-Up Summary", "Lookup Relationship" (radio button selected and highlighted with a red circle), and "Master-Detail Relationship". The "Lookup Relationship" option is described as creating a relationship that links the object to another object. The "Next" button is highlighted with a red arrow pointing to it.

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.

The screenshot shows the "Lookup Options" configuration for the Service records object. It includes fields for "Related To" (Appointment), "Related List Label" (Service records), and "Child Relationship Name" (Service_records). Under the "Required" section, a checkbox is checked, indicating "Always require a value in this field in order to save a record". Below this, there are options for what to do if the lookup record is deleted: "Clear the value of this field. You can't choose this option if you make this field required." and "Don't allow deletion of the lookup record that's part of a lookup relationship." The "Don't allow deletion" option is selected.

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.

Lookup Filter

Optional, create a filter to limit the records available to users in the lookup field. [Tell me more!](#)

Hide Filter Settings

Filter Criteria

Field	Operator	Value / Field
Appointment: Appointment Date	less than	Field <input type="button" value="..."/> Appointment: Created Date <input type="button" value="Clear"/>
Add <input type="text" value="Begin typing to search for a field..."/>	--None--	<input type="button" value="Value"/> <input type="button" value="..."/>

Filter Type **Required.** The user-entered value must match filter criteria.
If it doesn't, display this error message on save:

Optional. The user can remove the filter or enter values that don't match criteria.

Lookup Window Text

Active Enable this filter.

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.

The screenshot shows the 'Object Manager' interface for the 'Appointment' object. The 'Fields & Relationships' tab is selected. On the right, under 'Data Type', the 'Checkbox' option is highlighted with a red box and selected, while other options like 'None Selected', 'Auto Number', 'Formula', 'Roll-up Summary', 'Lookup Relationship', 'Master Detail Relationship', and 'External Lookup Relationship' are shown with their respective descriptions.

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

The screenshot shows the 'Step 2. Enter the details' screen for creating a new custom field. The 'Field Label' is set to 'Maintenance service'. The 'Default Value' section shows 'Unchecked' selected (indicated by a green arrow). Other fields include 'Field Name' (auto-populated as 'Maintenance_service'), 'Description' (empty), and 'Help Text' (empty). At the bottom, there are checkboxes for 'Auto add to custom report type' and 'Add this field to existing custom report types that contain this entity'. Navigation buttons 'Previous', 'Next', and 'Cancel' are at the bottom right.

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 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 click on the Required option.
7. Click on next >> next >> save.

Appointment
New Custom Field

Step 2. Enter the details Step 2 of 4

Field Label

Field Name

Description

Help Text

Required Always require a value in this field in order to save a record

Auto add to custom report type Add this field to existing custom report types that contain this entity

Default Value Show Formula Editor

[Previous](#) [Next](#) [Cancel](#)

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

Step 2. Enter the details Step 2 of 4

Field Label

Please enter the length of the number and the number of decimal places. For example, a number with a length of 8 and 2 decimal places can accept values up to "12345678.90".

Length <input type="text" value="18"/>	Decimal Places <input type="text" value="0"/>
Number of digits to the left of the decimal point	Number of digits to the right of the decimal point

Field Name

Description

Help Text

Required Always require a value in this field in order to save a record

Auto add to custom report type Add this field to existing custom report types that contain this entity

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

Appointment
New Custom Field

Step 3. Establish field-level security Step 3 of 4

Field Label: Service Amounts
Data Type: Currency
Field Name: Service_Amounts
Description:

Select the profiles to which you want to grant edit access to this field via field-level security. The field will be hidden from all profiles if you do not add it to field-level security.

Field Level Security for Profile	Visible	Read-Only
Analytics Cloud Integration User	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Analytics Cloud Security User	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Authenticated Website	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Authenticated Website	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Contract Manager	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Cross Org Data Proxy User	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

- Click on next >> save.

Creation of Currency Field on Billing details and feedback Object :

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

Creation of Text Fields

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

Step 2. Enter the details Step 2 of 4

Previous Next Cancel

Field Label	Vehicle number plate	i
Please enter the maximum length for a text field below.		
Length	10	
Field Name	Vehicle_number_plate	i
Description		
Help Text		
Required	<input checked="" type="checkbox"/> Always require a value in this field in order to save a record	
Unique	<input checked="" type="checkbox"/> Do not allow duplicate values	
	<input checked="" type="radio"/> Treat "ABC" and "abc" as duplicate values (case insensitive)	
	<input type="radio"/> Treat "ABC" and "abc" as different values (case sensitive)	
External ID	<input type="checkbox"/> Set this field as the unique record identifier from an external system	
Auto add to custom report type	<input checked="" type="checkbox"/> Add this field to existing custom report types that contain this entity i	

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 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.

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.

Step 2. Choose output type

Step 2 of 5

Field Label 

Field Name 

Auto add to custom report type Add this field to existing custom report types that contain this entity 

Formula Return Type

None Selected Select one of the data types below

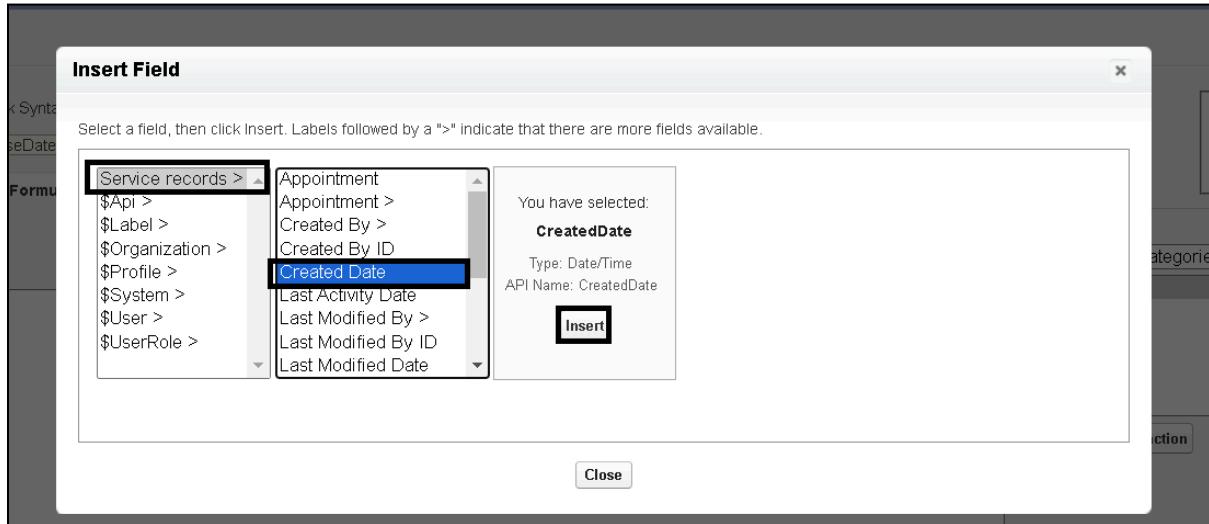
Checkbox Calculate a boolean value
Example: `(TODAY) > CloseDate`

Currency Calculate a dollar or other currency amount and automatically format the field as a currency amount.
Example: `Gross Margin = Amount - Cost_c`

Date Calculate a date, for example, by adding or subtracting days to other dates.
Example: `Reminder Date = CloseDate - 7`

Date/Time Calculate a datetime, for example, by adding a number of hours or days to another datetime.
Example: `Next = NOW() + 1`

5. Insert field formula should be : CreatedDate



Step 3. Enter formula

Step 3 of 5

Enter your formula and click Check Syntax to check for errors. Click the Advanced Formula subtab to use additional fields, operators, and functions.

Example: `Reminder Date = CloseDate - 7` [More Examples...](#)

Simple Formula Advanced Formula

service dates (Date) = 

Insert Operator 

Functions 

- Getting Started
- Operators & Functions

6. click "Check Syntax".
7. Click next >> next >> Save.

Validation rule

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

To create a validation rule to an 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.

The screenshot shows the Salesforce Object Manager for the 'Appointment' object. On the left, there's a sidebar with various configuration options like Buttons, Links, and Actions, Field Sets, Object Limits, Record Types, etc. At the bottom of this sidebar, the 'Validation Rules' option is highlighted with a green box. The main area is titled 'Validation Rules' and shows a table with one item: 'Vehicle'. The table has columns for RULE NAME, ERROR LOCATION, ERROR MESSAGE, ACTIVE, and MODIFIED BY. The 'Vehicle' row has 'Vehicle' in the RULE NAME column, 'Vehicle number plate' in the ERROR LOCATION column, 'Please enter valid number' in the ERROR MESSAGE column, an unchecked checkbox in the ACTIVE column, and 'project 2, 25/09/2023, 11:56 am' in the MODIFIED BY column. A 'New' button is located in the top right corner of the main area.

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Vehicle	Vehicle number plate	Please enter valid number	✓	project 2, 25/09/2023, 11:56 am

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}"))

Validation Rule Edit

Save Save & New Cancel

Rule Name: (highlighted by green box)

Active:

Description: vehicle

Error Condition Formula

Example: More Examples...
Display an error if Discount is more than 30%
If this formula expression is true, display the text defined in the Error Message area

(highlighted by green box)

Functions

-- All Function Categories -- ▾

- ABS
- ACOS
- ADDMONTHS
- AND
- ASCII
- ASIN

ABS(number)
Returns the absolute value of a number, a number without its sign
[Help on this function](#)

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

Error Message

Example: Discount percent cannot exceed 30%

This message will appear when Error Condition formula is true

Error Message: (highlighted by green box)

This error message can either appear at the top of the page or below a specific field on the page

Error Location: Top of Page Field (highlighted by green box) 

 Save Save & New Cancel

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"))

Validation Rule Edit

Rule Name: service_status_note

Active:

Description:

Error Condition Formula

Example: Discount_Percent_c>0.30 More Examples...

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

NOT(ISPIKVAL(Service_Status__c , "Completed"))

Functions

- All Function Categories --
- ABS
- ACOS
- ADDMONTHS
- AND
- ASCII
- ASIN

Insert Selected Function

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

Check Syntax

- 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.

Error Message

Example: Discount percent cannot exceed 30%

This message will appear when Error Condition formula is true

Error Message: still it is pending

This error message can either appear at the top of the page or below a specific field on the page

Error Location: Top of Page Field Service Status

Save Save & New Cancel

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

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

Validation Rule Edit

Save Save & New Cancel

Rule Name: **service_status_note** (highlighted with a green box)

Active:

Description:

Error Condition Formula

Example: [Discount_Percent_c>0.30](#) More Examples...

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

NOT (ISPICKVAL(Service_Status__c , "Completed"))

Functions

-- All Function Categories --

- ABS
- ACOS
- ADDMONTHS
- AND
- ASCII
- ASIN

Insert Selected Function

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

Check Syntax

Quick Tips
Operators & Functions

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.

Error Message

Example: Discount percent cannot exceed 30%

This message will appear when Error Condition formula is true

Error Message: still it is pending (highlighted with a green box)

This error message can either appear at the top of the page or below a specific field on the page

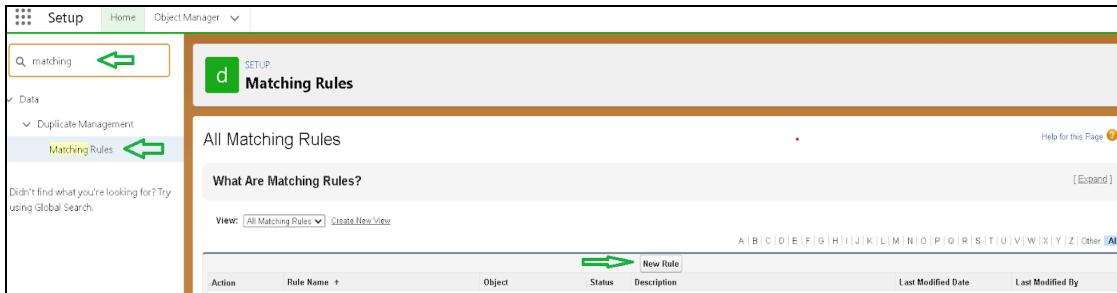
Error Location: Top of Page Field Service Status (highlighted with a green box)

Save Save & New Cancel

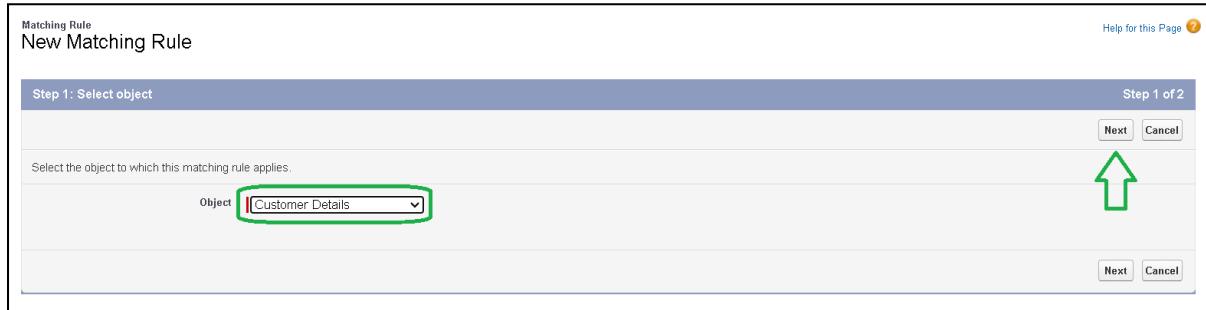
Duplicate rule

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.

Rule Details

Object	Customer Details	Save Cancel
Rule Name	matching Customer data	 = Required Information
Unique Name	matching_Customer_det	
Description	<div style="border: 1px solid #ccc; height: 50px; width: 100%;"></div>	

Matching Criteria

Tell the rule which fields to compare and how.

Field	Matching Method	Match Blank Fields	
Gmail	Exact	<input type="checkbox"/>	AND
Phone Number	Exact	<input type="checkbox"/>	AND
--None--	Exact	<input type="checkbox"/>	AND
--None--	Exact	<input type="checkbox"/>	AND
--None--	Exact	<input type="checkbox"/>	

[Add Filter Logic...](#)

| Save | Cancel

Matching Rule
matching Customer details

Matching Rule Detail

Object	Customer Details
Rule Name	matching Customer details
Unique Name	matching_Customer_details
Description	(Customer Details: Email EXACT MatchBlank = FALSE) AND (Customer Details: Phone_Number EXACT MatchBlank = FALSE)
Matching Criteria	(Customer Details: Email EXACT MatchBlank = FALSE) AND (Customer Details: Phone_Number EXACT MatchBlank = FALSE)
Status	Inactive
Created By	project2, 25/09/2023, 10:15 am
Modified By	project2, 10/10/2023, 3:32 pm

Help for this Page ⓘ

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.

Duplicate Rules

All Duplicate Rules

What Are Duplicate Rules?

View: All Duplicate Rules

Role Name	Description	Matching Rule	Active	Last Modified By	Last Modified Date
Customer Detail duplicate	Identify accounts that duplicate other accounts	matching_Customer_details	<input type="checkbox"/>	q2	10/10/2023
Standard Account Duplicate Rule	Identify contacts that duplicate other contacts and leads	Standard Account Matching Rule	<input checked="" type="checkbox"/>	q2	24/09/2023
Standard Contact Duplicate Rule	Identify leads that duplicate other leads and contacts	Standard Lead Matching Rule	<input checked="" type="checkbox"/>	q2	24/09/2023
Standard Lead Duplicate Rule	Identify accounts that duplicate other accounts	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	q2	24/09/2023

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.

The screenshot shows the 'Edit Duplicate Rule' screen for a rule named 'Customer Detail duplicate'. The rule details include:

- Rule Name:** Customer Detail duplicate
- Description:** Customer Details
- Object:** Customer Details
- Record-Level Security:** Enforce sharing rules (selected)

Actions:

- Action On Create: Allow (selected), Alert (checked), Report (unchecked)
- Action On Edit: Allow (selected), Alert (unchecked), Report (unchecked)
- Alert Text: Use one of these records?

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

Conditions:

Optionally, specify the conditions a record must meet for the rule to run.

Field	Operator	Value	AND
[-None-]	[-None-]		AND
[-None-]	[-None-]		

Buttons:

- Add Rule
- Remove Rule
- Save (highlighted with a green arrow)
- Save & New
- Cancel

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 delete 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.

Data Manager (standard_DataManager)	<input checked="" type="checkbox"/>	<input type="radio"/>	[standard_ServiceConsole]	<input type="checkbox"/>	<input type="radio"/>
Digital Experiences (standard_SalesforceCMS)	<input checked="" type="checkbox"/>	<input type="radio"/>	Service (standard_Service)	<input checked="" type="checkbox"/>	<input type="radio"/>
Garage Management Application (Garage_Management_Application)	<input type="checkbox"/>	<input checked="" type="radio"/>	Service Console (standard_LightningService)	<input checked="" type="checkbox"/>	<input type="radio"/>
Laptop Hub (Laptop_Hub)	<input type="checkbox"/>	<input type="radio"/>	Site.com (standard_Sites)	<input checked="" type="checkbox"/>	<input type="radio"/>
			Subscription Management (standard_RevenueCloudConsole)	<input checked="" type="checkbox"/>	<input type="radio"/>

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.

Custom Object Permissions						
	Basic Access				Data Administration	
	Read	Create	Edit	Delete	View All	Modify All
Appointments	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Billing details and feedback	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Customer Details	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Environments	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Laptops	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Service records	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
SessionData	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

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 “

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.

Custom Object Permissions						
	Basic Access				Data Administration	
	Read	Create	Edit	Delete	View All	Modify All
Appointments	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Billing details and feedback	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Customer Details	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Environments	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Laptops	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Service records	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SessionData	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

5. And click save.

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.

The screenshot shows the Salesforce Setup interface. In the left sidebar, under 'Users', the 'Roles' link is highlighted with a red box. The main content area displays a 'Sample Role Hierarchy' diagram titled 'Territory-based Sample'. The hierarchy is as follows:

- Executive Staff**: CEO_President, CFO_VP_Sales, Director of Exec Staff
- Western Sales Director**: Director of Western Sales, Western Sales Rep (CA Sales Rep, OR Sales Rep)
- Eastern Sales Director**: Director of Eastern Sales, Eastern Sales Rep (NY Sales Rep, MA Sales Rep)
- International Sales Director**: Director of International Sales, International Sales Rep (Asian Sales Rep, European Sales Rep)

Each role has associated permissions listed to its right, such as 'View & edit data, roll up forecasts, & generate reports for this level' and 'Can't access data of other Executive Staff'.

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

The screenshot shows the 'Your Organization's Role Hierarchy' page. The tree structure is as follows:

- Nick Enterprises**
 - CFO**: Edit | Del | Assign
 - Manager**: Edit | Del | Assign
 - On Site Emp**: Edit | Del | Assign
 - Remote Emp**: Edit | Del | Assign
 - Add Role**
 - HR**: Edit | Del | Assign
 - Add Role**
 - Add Role**

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

Role Edit

Label: Manger ←

Role Name: Manger i

This role reports to: CEO 🔍

Role Name as displayed on reports:

← Save Save & New Cancel

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.

Collapse All Expand All

- TheSmartBridge
 - + Add Role
 - + CEO [Edit](#) | [Del](#) | [Assign](#)
 - + Add Role
 - + CFO [Edit](#) | [Del](#) | [Assign](#)
 - + Add Role
 - + COO [Edit](#) | [Del](#) | [Assign](#)
 - + Add Role
 - + Manager [Edit](#) | [Del](#) | [Assign](#)
 - Add Role**
 - + SVP, Customer Service & Support [Edit](#) | [Del](#) | [Assign](#)
 - + Add Role
 - + SVP, Human Resources [Edit](#) | [Del](#) | [Assign](#)
 - + Add Role
 - + SVP, Sales & Marketing [Edit](#) | [Del](#) | [Assign](#)
 - + Add Role

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

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
 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

New User

User Edit Save Save & New Cancel

General Information

First Name: Niklaus
Last Name: Mikaelson
Alias: Inmika
Email:
Username: Mikaelson@Niklaus
Nickname: Nik
Title:
Company:
Department:
Division:

Role: Manager
User License: Salesforce
Profile: Manager
Active: ✓

Marketing User
Offline User
Knowledge User
Flow User
Service Cloud User
Site.com Contributor User
Site.com Publisher User
WDC User
Data.com User Type: --None--

3. Save.

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.

Public groups

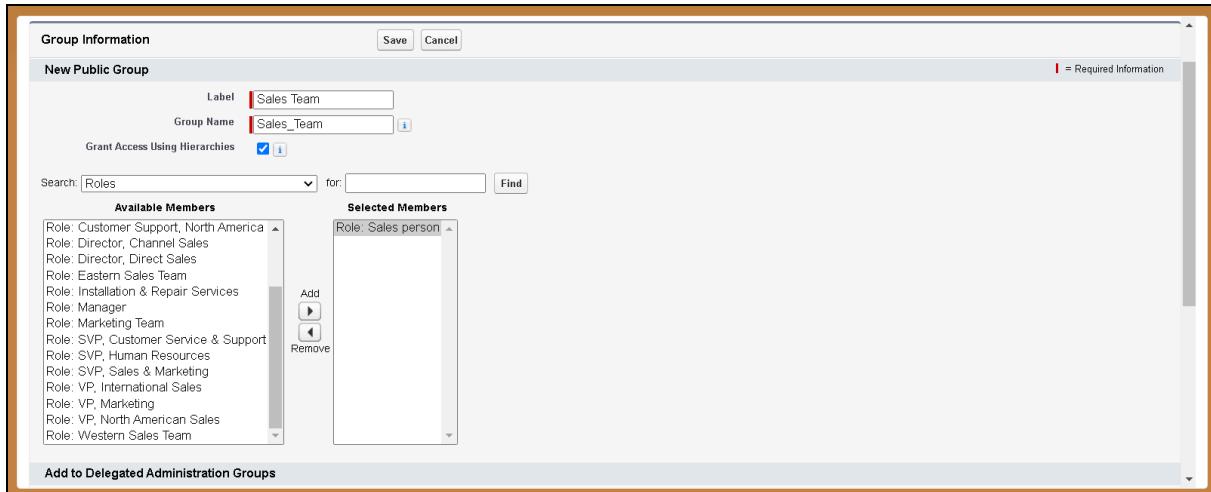
Public groups are a valuable tool for Salesforce administrators and developers to streamline user management, data access, and security settings. By creating and using public groups effectively, you can maintain a secure and organized Salesforce environment while ensuring that users have appropriate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

The screenshot shows the Salesforce Setup page for 'Public Groups'. In the top left, there's a search bar with 'public groups' and a sidebar with 'Users' and 'Public Groups' selected. The main area has a title 'Public Groups' with a sub-instruction: 'A public group is a set of users. It can contain individual users, other groups, the users in a particular role or territory, or the users in a role or territory plus all of the users below that role or territory in the hierarchy.' Below this is a 'View' dropdown set to 'All', a 'Create New View' button, and a 'Help for this Page' link. A navigation bar at the bottom includes letters from A to Z and an 'Other' option. The main table has columns for 'Label', 'Group Name', 'Created By', and 'Created Date'. A 'New' button is at the top right of the table. The table currently shows 'No records to display'.

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.



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.

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.

The screenshot shows the 'Sharing Settings' page in the Salesforce Setup. The main table lists various objects with their OWD (Object-Level Sharing) settings. The 'Service records' row is highlighted with a red box around its OWD dropdown, which is set to 'Private'. The 'Save' button at the bottom left of the form is also highlighted with a red box.

Object	OWD	Actions
Work Plan Template	Private	
Work Step Template	Private	
Work Type	Private	
Work Type Group	Public Read/Write	
Appointment	Public Read/Write	
Billing details and feedback	Public Read/Write	
Customer Details	Public Read/Write	
Environment	Public Read/Write	
Laptop	Public Read/Write	
Service records	Private	
SessionData	Public Read/Write	

User Visibility Settings

Portal User Visibility Site User Visibility

Other Settings

Standard Report Visibility Manual User Record Sharing Manager Groups

Minimize the number of roles created, which improves performance by cutting down processing loads

Grant site users access to related cases Secure guest user record access Require permission to view record names in lookup fields

Buttons

Save Cancel

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

Service records Sharing Rules

New Recalculate

No sharing rules specified.

Service records Sharing Rules Help ?

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.

Sharing Settings

You can use sharing rules only to grant wider access to data, NOT TO RESTRICT ACCESS.

Step 1: Rule Name

Label: sharing settings
Rule Name: sharing_settings

Step 2: Select your rule type

Rule Type: Based on record owner

Step 3: Select which records to be shared

Service records: owned by members of: Roles (highlighted)
Sales person (highlighted)

Step 4: Select the users to share with

Share with: Roles (highlighted)
Manager (highlighted)

Step 5: Select the level of access for the users

Access Level: Read/Write (highlighted)

Save Cancel

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.

The screenshot shows the Salesforce Setup interface. In the left sidebar, under 'Process Automation', the 'Flows' item is selected and highlighted with a red box, labeled '2'. A search bar at the top contains the text 'flows', also highlighted with a red box, labeled '1'. On the right, the 'Flows' page is displayed with a table of flow definitions. A red box highlights the 'New Flow' button in the top right corner of the page area, labeled '3'.

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

The screenshot shows the 'New Flow' creation page. Under the 'Core' tab, there are several flow types listed: 'Screen Flow', 'Record-Triggered Flow', 'Schedule-Triggered Flow', 'Platform Event—Triggered Flow', 'Autolaunched Flow (No Trigger)', and 'Record-Triggered Orchestration'. The 'Record-Triggered Flow' option is highlighted with a red box and labeled '1'. At the bottom right of the page, a blue 'Create' button is highlighted with a red box and labeled '2'.

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.

Configure Start

Select Object

Select the object whose records trigger the flow when they're created, updated, or deleted.

*Object
Billing details and feedback

Configure Trigger

*Trigger the Flow When:

- A record is created
- A record is updated
- A record is created or updated
- A record is deleted



Set Entry Conditions

Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

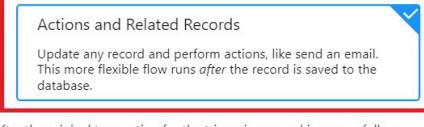
If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Condition Requirements
None

*Optimize the Flow for:

Fast Field Updates
Update fields on the record that triggers the flow to run. This high-performance flow runs *before* the record is saved to the database.

Actions and Related Records
Update any record and perform actions, like send an email. This more flexible flow runs *after* the record is saved to the database.



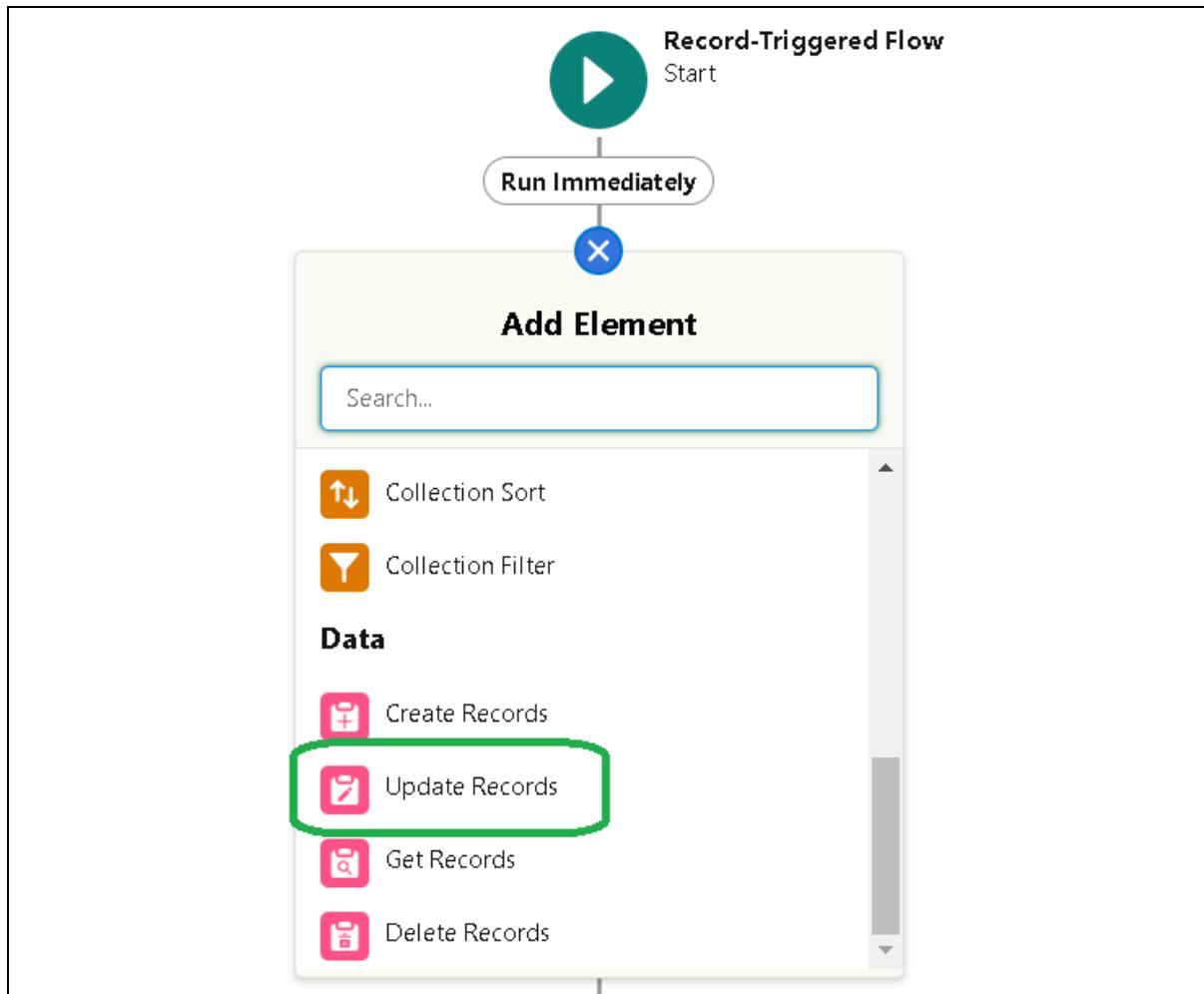

3

4

Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed

Cancel 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	*API Name
Amount Update	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	Operator	Value
Payment_Status__c	Equals	Completed

+ Add Condition

Set Field Values for the Billing details and feedback Record

Field	Value
Payment_Paid__c	\$Record > Service records > Appointment > Service A...

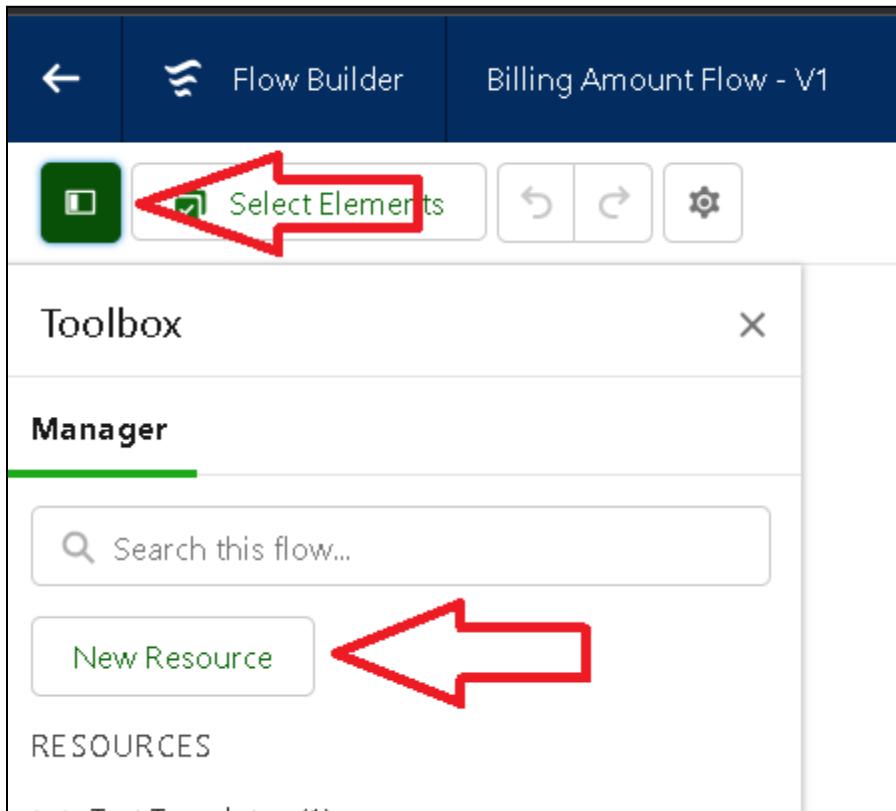
+ Add Field

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.



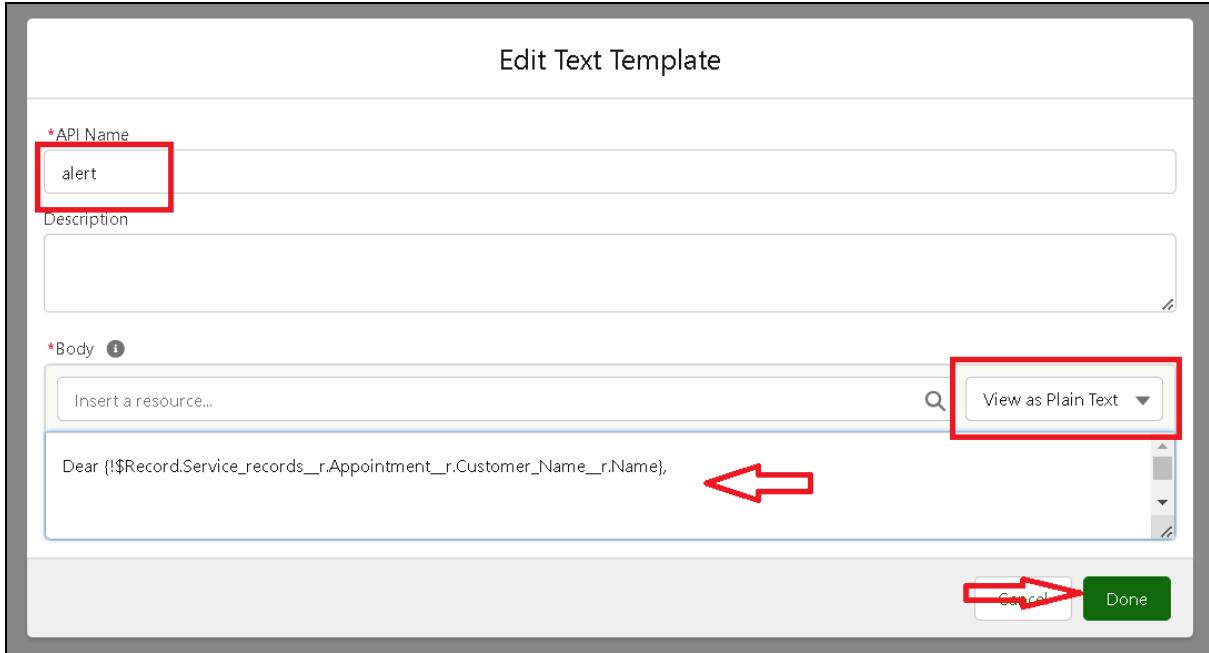
18. Click on the New Resource, And select Variable.
 19. Select the resource type as text template.
 20. Enter the API name as " alert".
 21. Change the view as Rich Text ? View to Plain Text.
 22. In body field paste the syntax that given below.
- Dear {\$Record.Service_records__r.Appointment__r.Customer_Name__r.Name},

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.

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

Aa Body i

{!alert}

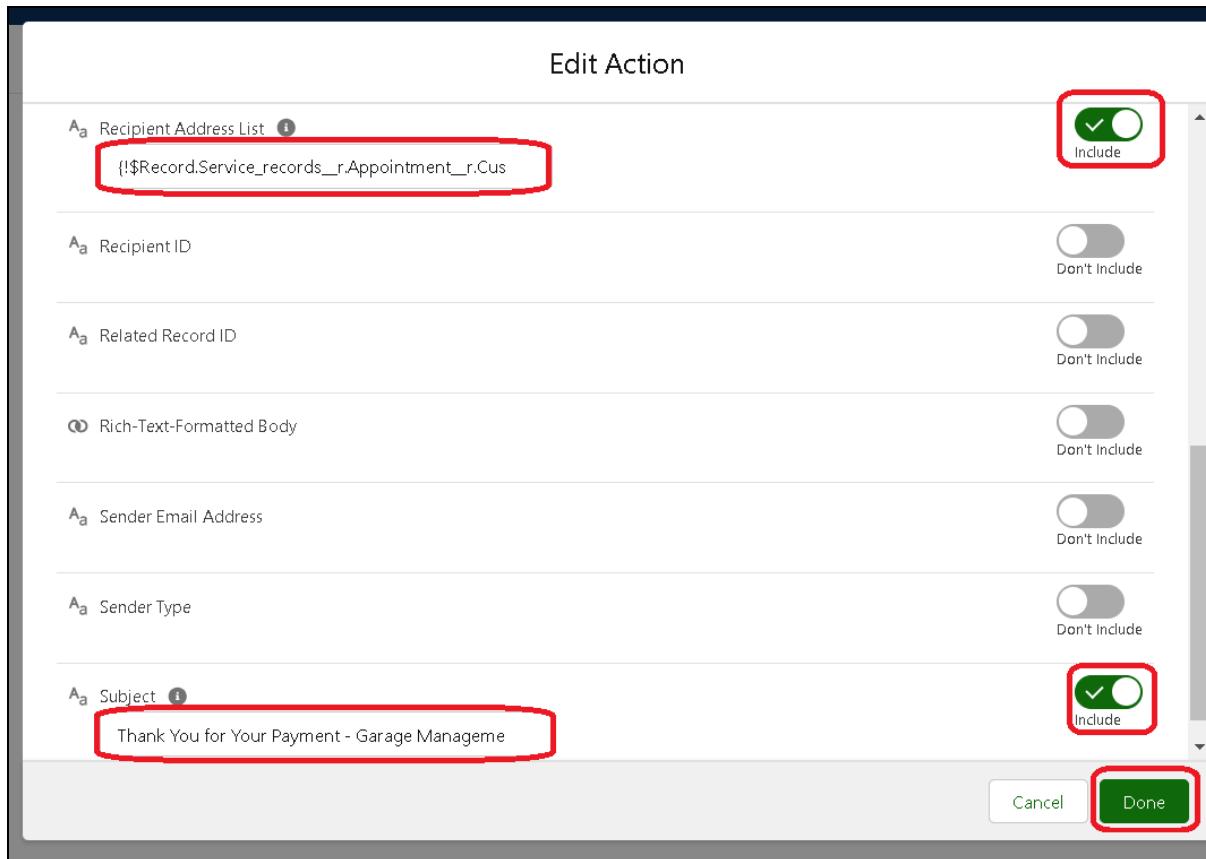


Aa Email Template ID



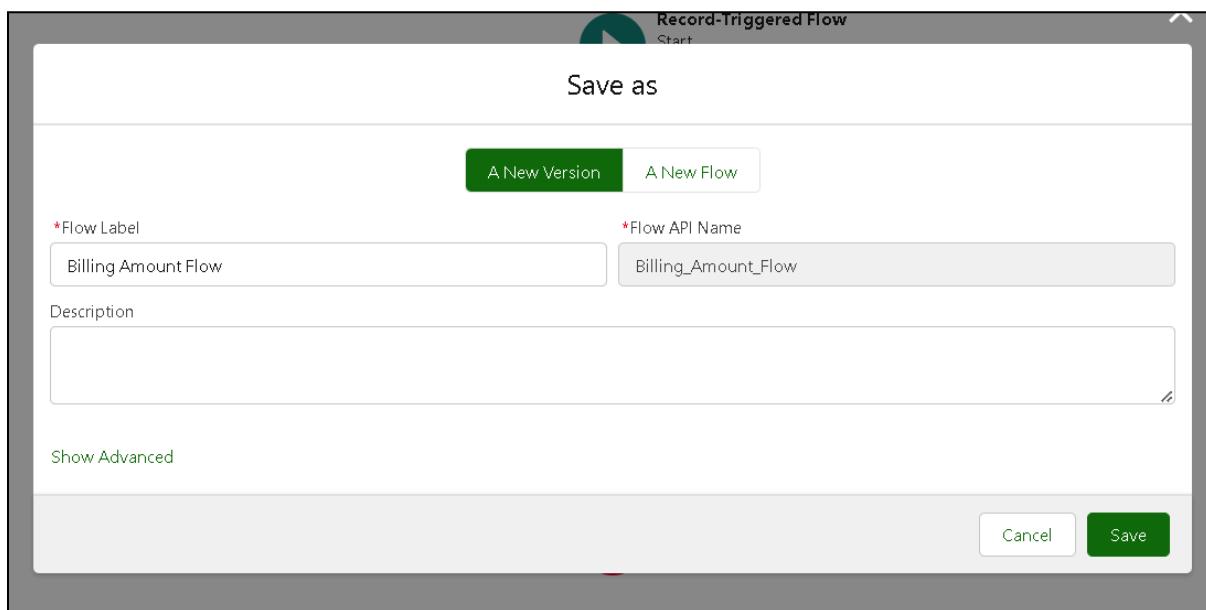
QD Log Email on Send

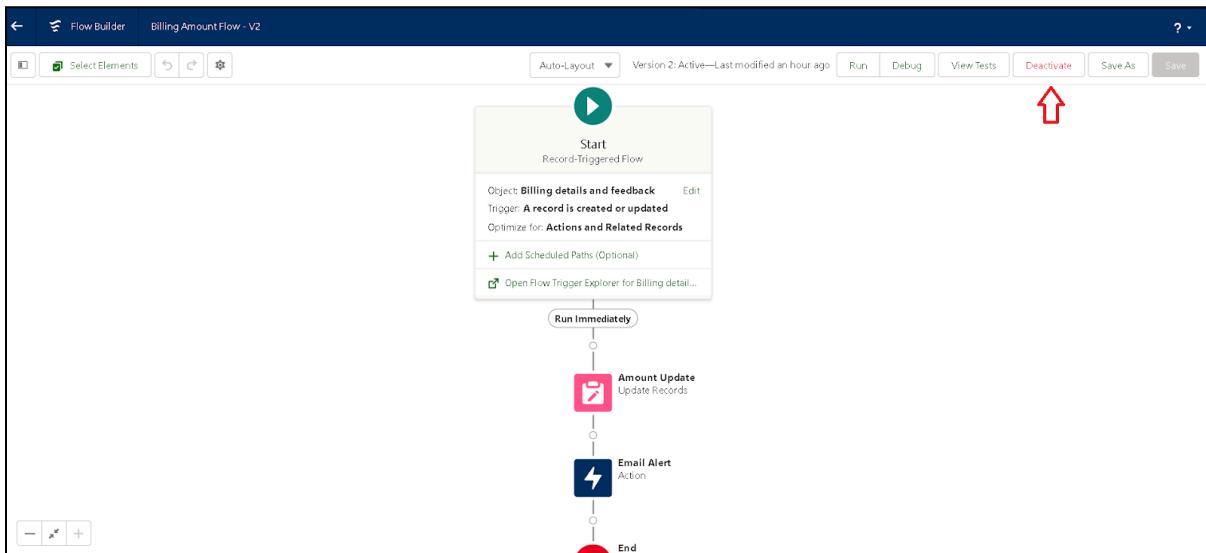




34. Click on save. Give the Flow label , Flow Api name will be autopopulated.

35. And click save, and click on activate.





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:

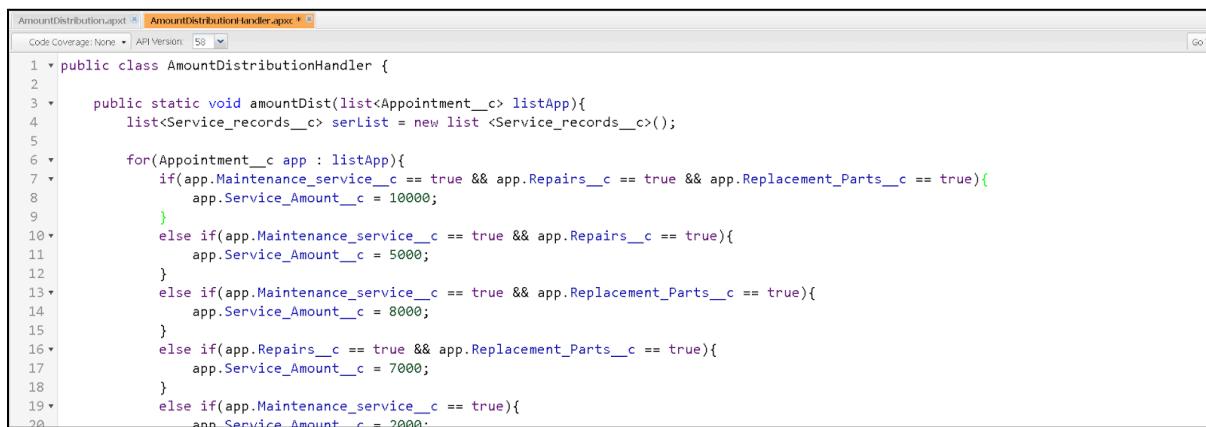
Before Trigger: This type of trigger in Salesforce is used either to update or validate the values of a record before they can be saved into the database. So, basically, the before trigger validates the record first and then saves it. Some criteria or code can be set to check data before it gets ready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

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.apc  AmountDistributionHandler.apc
Code Coverage: None | API Version: 58 | Go To
1 * public class AmountDistributionHandler {
2
3     public static void amountDist(list<Appointment__c> listApp){
4         list<Service_records__c> serlist = new list <Service_records__c>();
5
6         for(Appointment__c app : listApp){
7             if(app.Maintenance_service__c == true && app.Repairs__c == true && app.Replacement_Parts__c == true){
8                 app.Service_Amount__c = 10000;
9             }
10            else if(app.Maintenance_service__c == true && app.Repairs__c == true){
11                app.Service_Amount__c = 5000;
12            }
13            else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
14                app.Service_Amount__c = 8000;
15            }
16            else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
17                app.Service_Amount__c = 7000;
18            }
19            else if(app.Maintenance_service__c == true){
20                app.Service_Amount__c = 2000;
21            }
22        }
23    }
24}
```

```
12     }
13     else if(app.Maintenance_service__c == true && app.Replacement_Parts__c == true){
14         app.Service_Amount__c = 8000;
15     }
16     else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
17         app.Service_Amount__c = 7000;
18     }
19     else if(app.Maintenance_service__c == true){
20         app.Service_Amount__c = 2000;
21     }
22     else if(app.Repairs__c == true){
23         app.Service_Amount__c = 3000;
24     }
25     else if(app.Replacement_Parts__c == true){
26         app.Service_Amount__c = 5000;
27     }
28
29 }
30
31 }
```

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;
    }
}

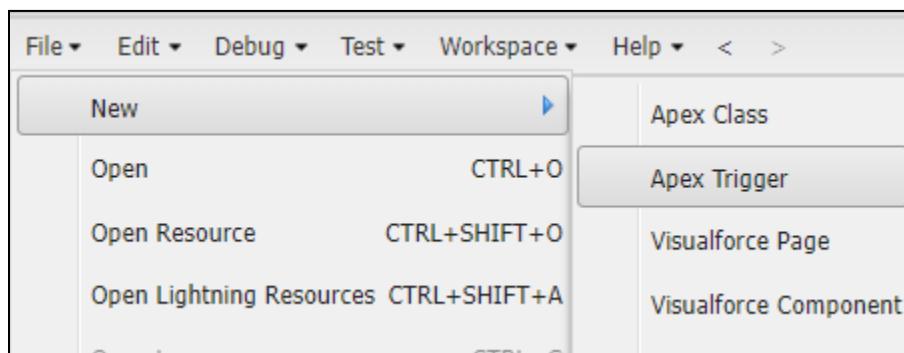
}
}
}

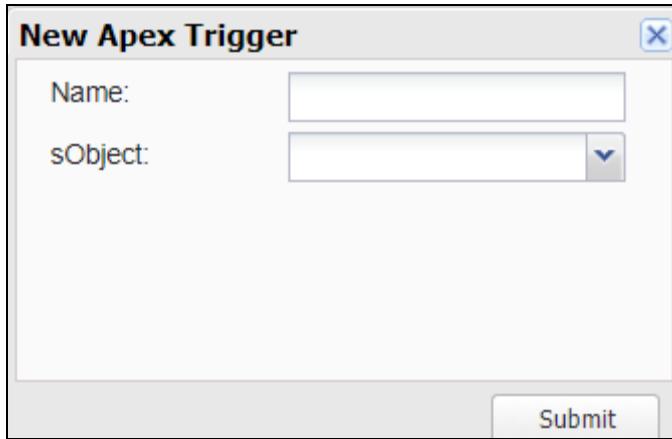
```

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

```
File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >  
AmountDistribution.apxt AmountDistributionHandler.apxc *  
Code Coverage: None API Version: 58  
1 trigger AmountDistribution on Appointment__c (before insert, before update) {  
2  
3     if(trigger.isbefore && trigger.isinsert || trigger.isupdate){  
4         AmountDistributionHandler.amountDist(trigger.new);  
5     }  
6  
7 }  
8 }
```

Code:

```

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

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

    }

}

```

Reports

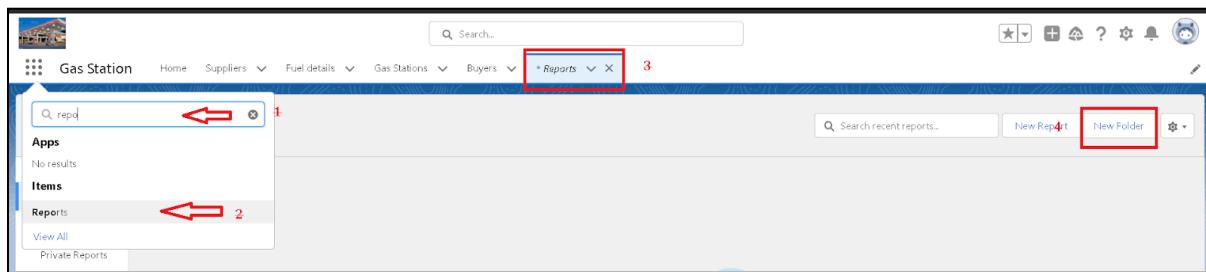
Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics.

Types of Reports in Salesforce

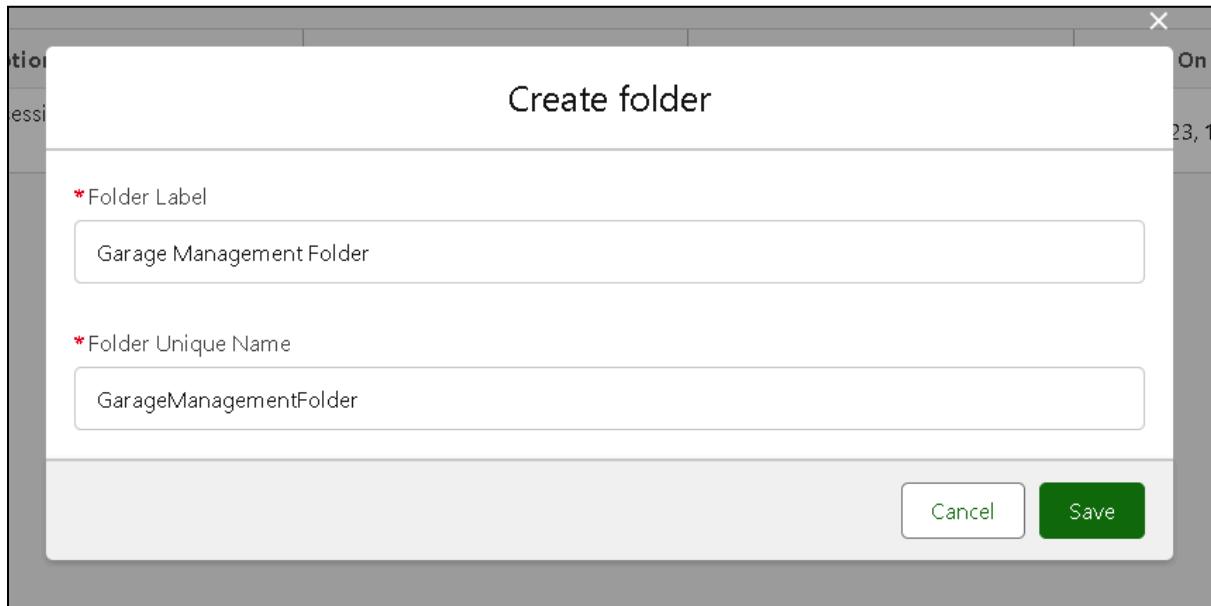
1. Tabular
2. Summary
3. Matrix
4. Joined Reports

create a report folder

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

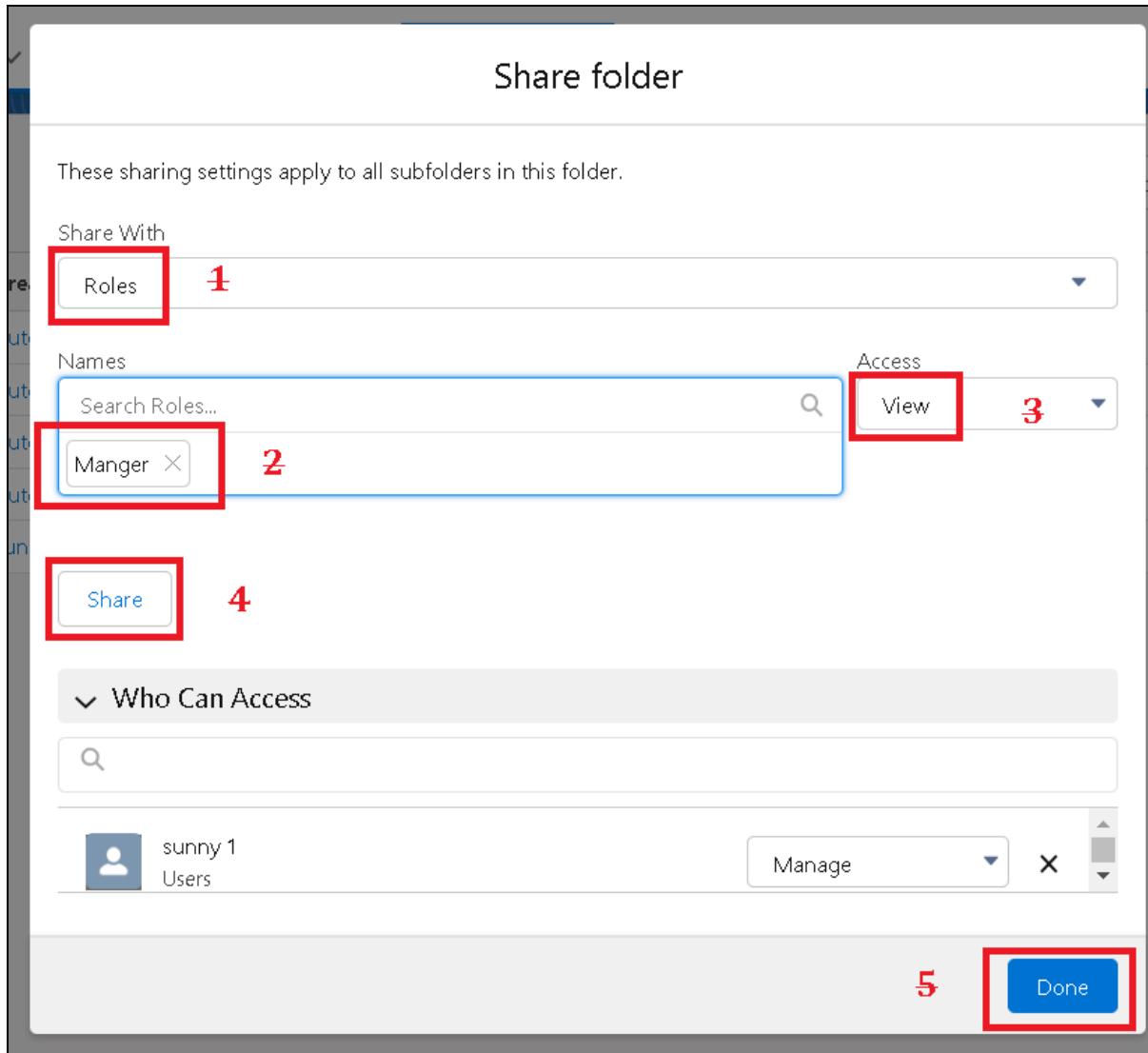


3. Give the Folder label as “Garage Management Folder”, Folder unique name will be auto populated.
4. Click save.



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.

Setup Home Object Manager ▾

Q report type

Feature Settings

Analytics

Reports & Dashboards

Report Types

All Custom Report Types

With custom report types, you can enable users to create reports from the predefined objects, object relationships, and fields that you specify.

View: All Custom Report Types ▾ Edit | Create New View

Action	Label +	Description	Category	Deployed	Created By Alias	Created Date
Edit Del	Bot Metrics Daily Summer '23	Einstein Bot metrics aggregated by day	Other Reports	✓	autopro	28/09/2023
Edit Del	Bot Metrics Hourly Summer '23	Einstein Bot metrics aggregated by hour	Other Reports	✓	autopro	28/09/2023
Edit Del	Screen Flows	Find out which flows get executed and how long users take to complete each flow screen.	Other Reports	✓	autopro	24/09/2023
Edit Del	Session Metrics Summer '23	Einstein Bot session metrics	Other Reports	✓	autopro	28/09/2023

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All

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 “ Deployed ” , click on Next.

SETUP Report Types

Report Type Focus

Specify what type of records (rows) will be the focus of reports generated by this report type.

Example: If reporting on "Contacts with Opportunities with Partners," select "Contacts" as the primary object.

Primary Object Customer Details

Identification

Report Type Label Service information

Report Type Name Service_information

Description Service Information

Store in Category Other Reports

Deployment

A report type with deployed status is available for use in the report wizard. While in development, report types are visible only to authorized administrators and their delegates.

Deployment Status In Development Deployed

Next Cancel

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

New Custom Report Type
Service information

Help for this Page 

Step 2. Define Report Records Set Step 2 of 2

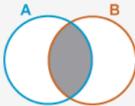
Previous Save Cancel

This report type will generate reports about Customer Details. You may define which related records from other objects are returned in report results by choosing a relationship to another object.

A Customer Details
Primary Object

B Select Object:

Appointments  at least one related "B" record.
Activities at least one related "B" records.
Duplicate Record Items





Previous Save Cancel

Step 2. Define Report Records Set

This report type will generate reports about Customer Details. You may define which related records from other objects are returned in report results by choosing a relationship to another object.

A Customer Details
Primary Object

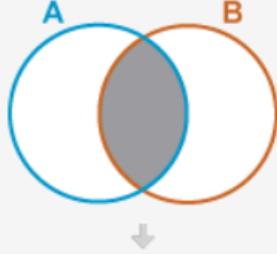
B Appointments

A to B Relationship:

Each "A" record must have at least one related "B" record.
 "A" records may or may not have related "B" records.

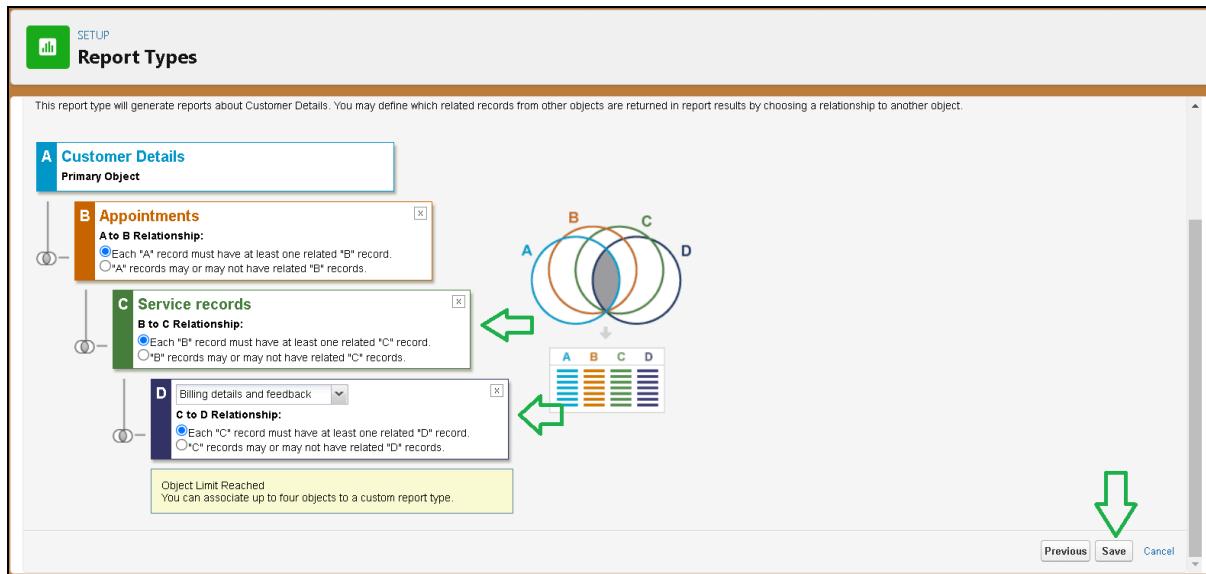








11. Again Click to relate another object.
12. And select the related object as “ service records”.
13. Repeat the process and select the related object as “ Billing details and feedback”.
14. And click on save.



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.

REPORTS	Report Name	Description	Folder	Created By	Created On	Subscribed
Recent	Employee's working on projects report		Private Reports	Employee Project	5/6/2023, 9:33 am	
Created by Me	Assets assigned to Employees		Private Reports	Employee Project	5/6/2023, 9:36 am	

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

The screenshot shows the 'Create Report' interface. On the left, there's a sidebar with categories like Leads, Campaigns, Activities, Contracts and Orders, etc., and a 'Other Reports' section highlighted in green. The main area is titled 'Select a Report Type' with a search bar. It lists report types categorized by 'Category': Standard (Service records, Service records with Appointment, Service records History, Billing details and feedback with Service records) and Custom (Service information). The 'Service information' report type is selected. To the right, a panel titled 'Service information' shows its details: 'Custom Report Type', 'Start Report' button, 'Details' section (with a green arrow pointing to it), 'Description' (Service information), 'Created By You' (No Reports Yet), and 'Created By Others' (No Reports Yet). A green double-headed arrow is positioned between the 'Category' dropdown in the report type list and the 'Category' dropdown in the 'Service information' details panel.

4. Their outline pane is opened already, select the fields that mentioned below in 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.

The screenshot shows the report preview interface for 'New Service information Report'. At the top, there are buttons for Save & Run, Save, Close, and Run. The main area has sections for 'Fields' (Groups, Rating for service, Payment Status) and 'Columns' (Customer Name, Appointment Date, Service Status, # Payment Paid). A table titled 'Rating for service' shows data for 'Completed' status with columns for Total, Sum of Payment Paid, and Record Count. To the right, a line chart titled 'Sum of Payment Paid' shows a single blue line starting at approximately 18k for rating 4 and ending at approximately 8k for rating 5. Below the table is a detailed view of the rows, showing specific values for each record.

	Rating for service	Payment Status	Completed	Total
4	Sum of Payment Paid	Record Count	₹15,000	₹15,000
4	Sum of Payment Paid	Record Count	4	4
5	Sum of Payment Paid	Record Count	₹5,000	₹5,000
5	Sum of Payment Paid	Record Count	2	2
Total	Sum of Payment Paid	Record Count	₹20,000	₹20,000
			6	6

	Customer Name	Appointment Date	Service Status	Payment Paid
1	meghana	11/10/2023	Completed	₹8,000
2	rushi	08/09/2023	Completed	₹3,000
3	shivam	12/10/2023	Completed	₹2,000
4	shivam	12/10/2023	Completed	₹2,000
5	rushi	08/09/2023	Completed	₹3,000

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

x

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.

Create Dashboard Folder

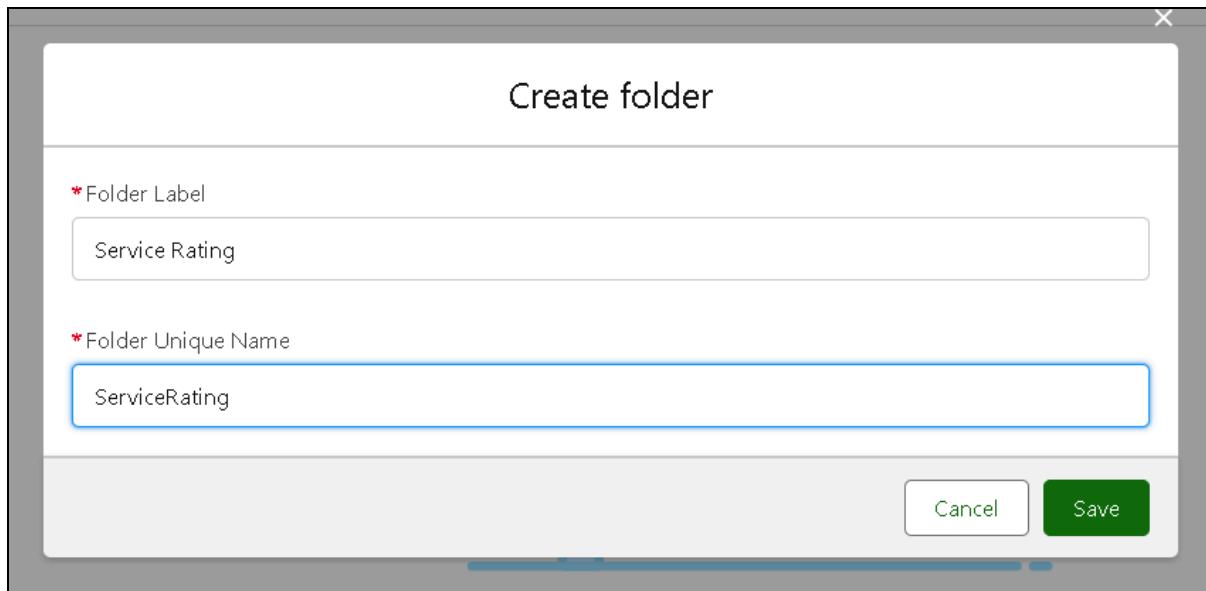
1. Click on the app launcher and search for dashboard.
2. Click on dashboard tab.
3. Click new folder, give the folder label as "Service Rating dashboard".
4. Folder unique name will be auto populated.
5. Click save.

Create folder

* Folder Label
Service Rating

* Folder Unique Name
ServiceRating

Cancel **Save**



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

Create Dashboard

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

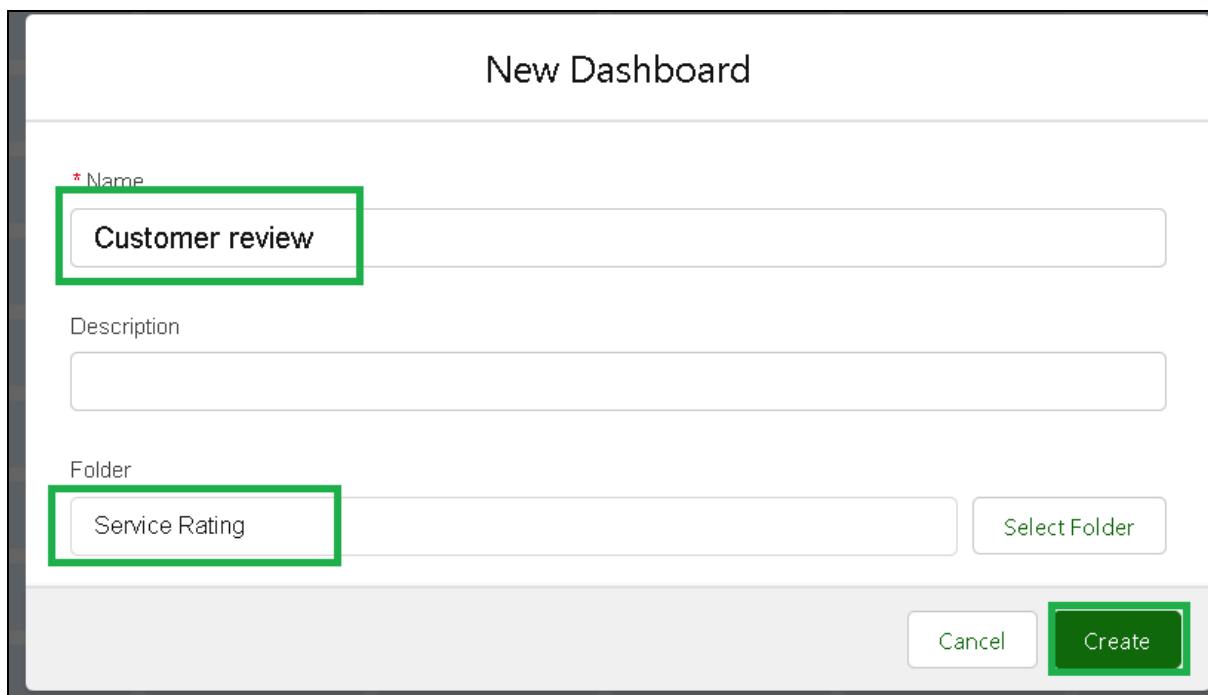
New Dashboard

* Name
Customer review

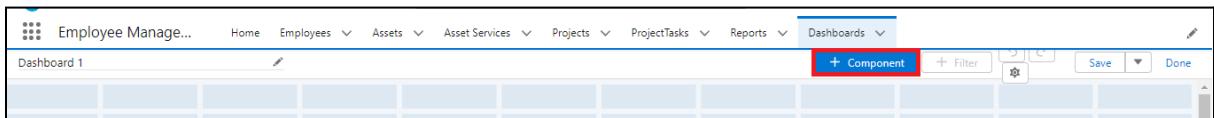
Description

Folder
Service Rating **Select Folder**

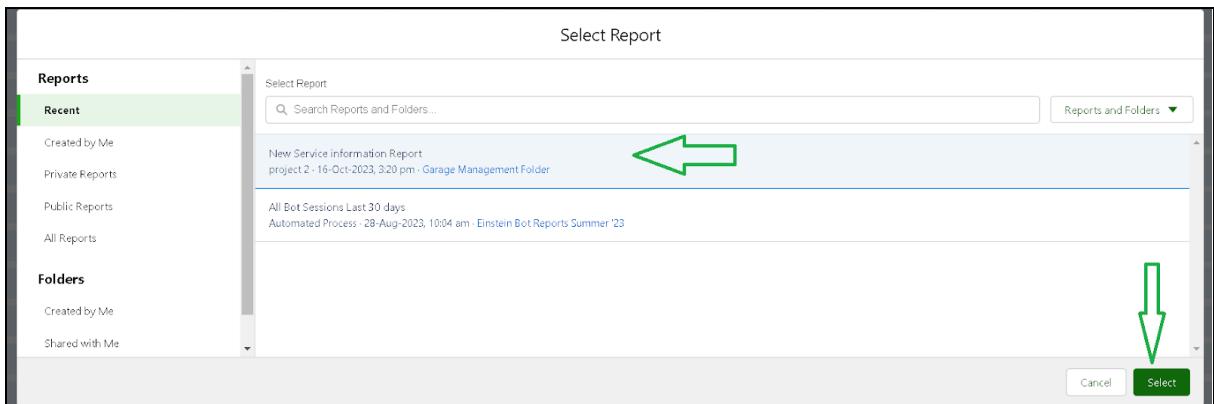
Cancel **Create**



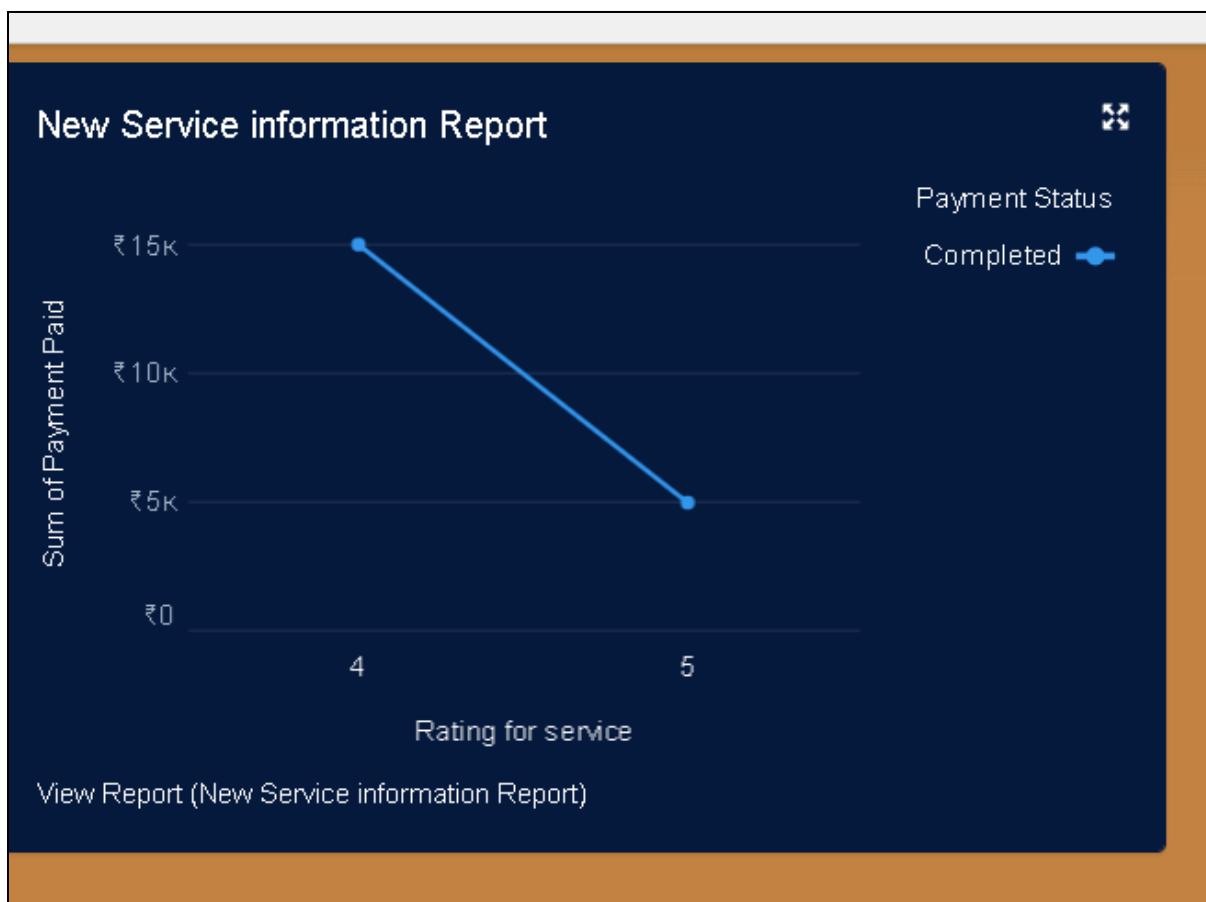
3. Select add component.



4. Select a Report and click on select.

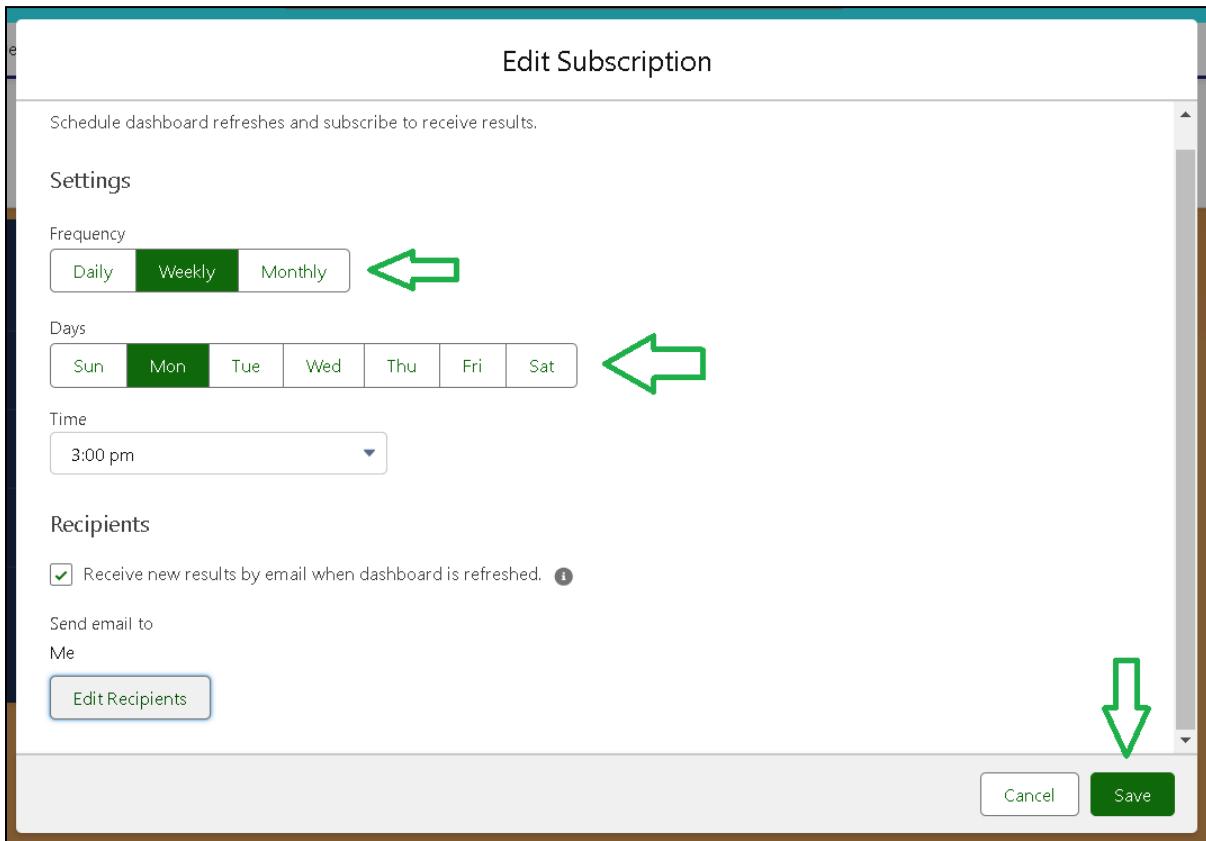


5. Select the Line Chart. Change the theme.
6. Click Add then click on Save and then click on Done.
7. Preview is shown below.



8. After that Click on Subscribe on top right.

9. Set the Frequency as " weekly ".
10. Set a day as monday.
11. And Click on save.



THANK YOU