

Garage Management System

Nandeeswar Alakunta
224g5a0213@srit.ac.in

Salesforce

Introduction:

Are you new to Salesforce? Not sure exactly what it is, or how to use it? Don't know where you should start on your learning journey? If you've answered yes to any of these questions, then you're in the right place. This module is for you.

Welcome to Salesforce! Salesforce is game-changing technology, with a host of productivity-boosting features, that will help you sell smarter and faster. As you work toward your badge for this module, we'll take you through these features and answer the question, "What is Salesforce, anyway?"

What Is Salesforce?

Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers.

Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud.

So what does that really mean? Well, before Salesforce, your contacts, emails, follow-up tasks, and prospective deals might have been organized something like this:

<https://youtu.be/r9EX3lGde5k>

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 :

Build enterprise-quality apps fast to bring your ideas to life

- Build apps fast with drag and drop tools
- Customize your data model with clicks
- Go further with Apex code
- Integrate with anything using powerful APIs
- Stay protected with enterprise-grade security
- Customize UI with clicks or any leading-edge web framework

Sign up for your Salesforce Developer Edition
A full-featured copy of the Platform, for free

Complete the form to start your free trial. Our team will be in touch to help you make the most of your trial.

First Name*

Last Name*

Email*

Role*

Company*

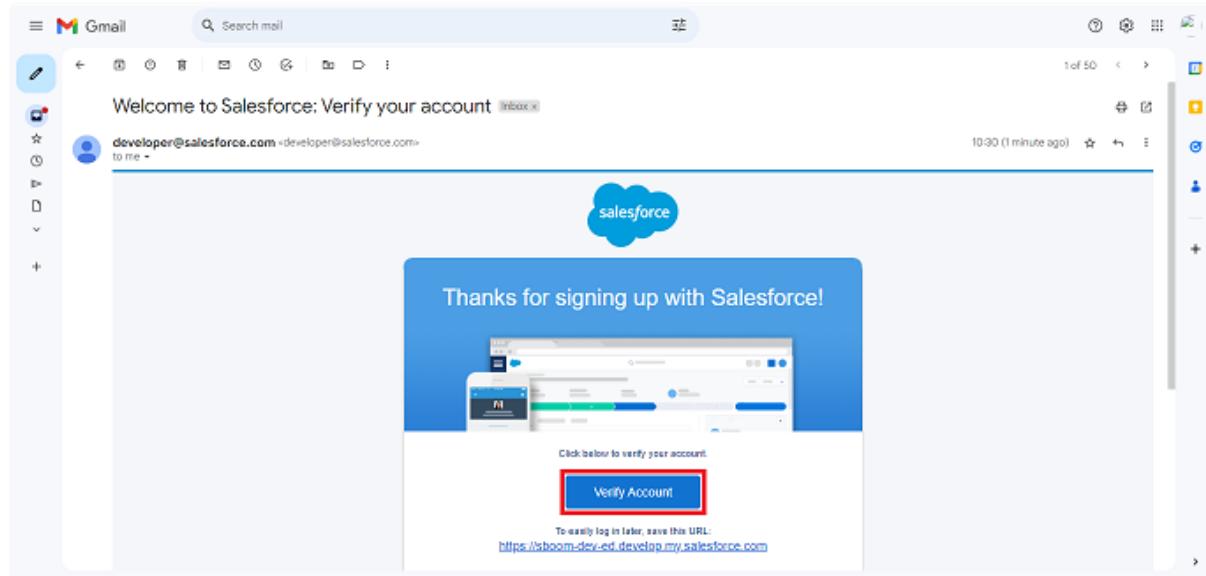
1. First name & Last name
2. Email
3. Role : Developer
4. Company : College Name
5. County : India
6. Postal Code : pin code
7. Username : should be a combination of your name and company

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.

Change Your Password

Enter a new password for lead@sb.oom.
Make sure to include at least:

- 8 characters
- 1 letter
- 1 number

* New Password
..... Good

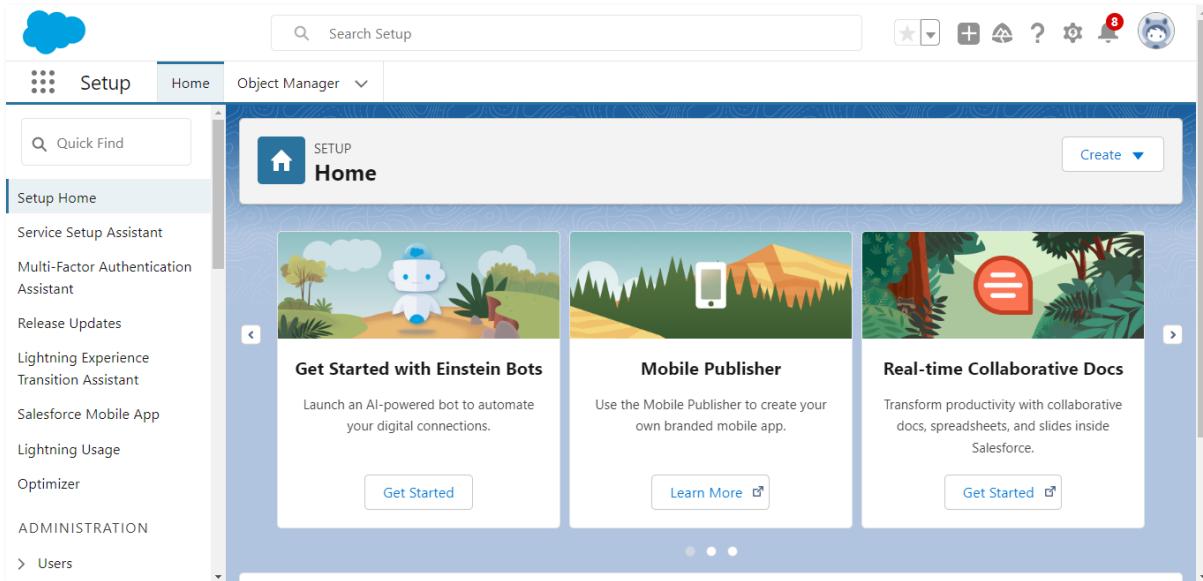
* Confirm New Password
..... Match

Security Question
▼ In what city were you born?

* Answer
asdfghjkl

Change Password

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



The screenshot shows the Salesforce Setup Home page. The top navigation bar includes a cloud icon, a search bar labeled "Search Setup", and various navigation icons. The main content area is titled "SETUP Home". It features three main cards: "Get Started with Einstein Bots" (Launch an AI-powered bot to automate your digital connections), "Mobile Publisher" (Use the Mobile Publisher to create your own branded mobile app), and "Real-time Collaborative Docs" (Transform productivity with collaborative docs, spreadsheets, and slides inside Salesforce). A sidebar on the left lists various setup options like "Setup Home", "Service Setup Assistant", and "ADMINISTRATION".

Object

What Is an Object?

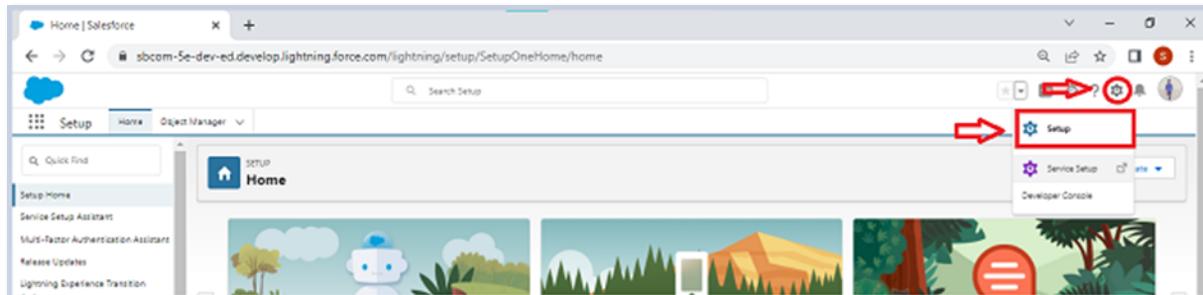
Salesforce objects are database tables that permit you to store data that is specific to an organization. What are the types of Salesforce objects

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

Custom Object Definition Edit

Custom Object Information

The singular and plural labels are used in lists, page layouts, and reports.

Label: Example: Account (arrow)

Plural Label: Example: Accounts (arrow)

Starts with novel sound:

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

Object Name: Example: Account (arrow)

Description:

Context-sensitive help setting: Open the standard Salesforce.com Help & Training window Open a window using a Visualforce page

Content Name: (arrow)

Enter Record Name Label and Format:

The Record Name appears in page layouts, key fields, 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 (arrow)

Optional Features:

- Allow Reports (arrow)
- 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 (arrow)

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

Buttons:

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

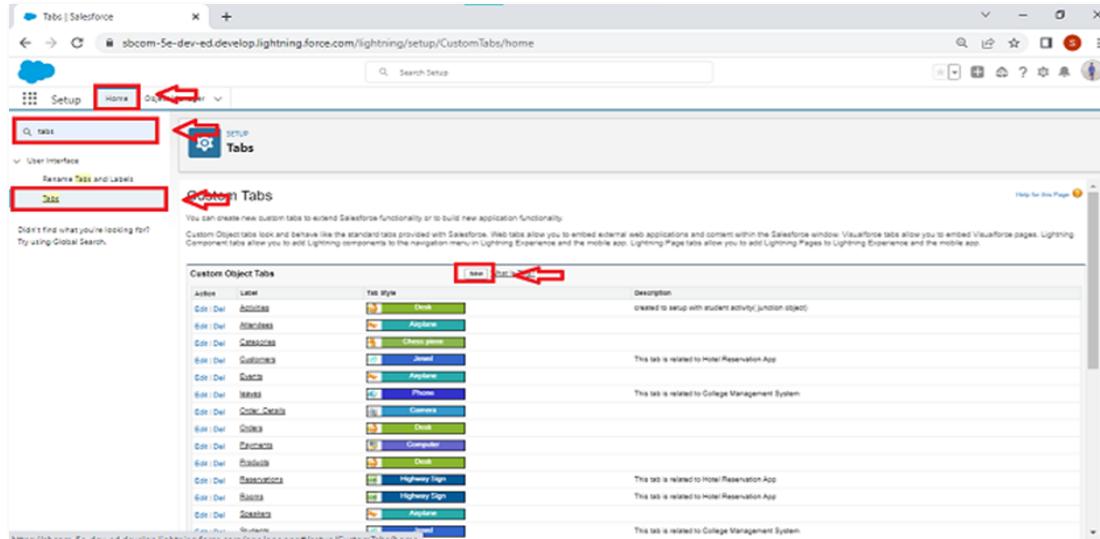
Lightning Page Tabs let you add Lightning Pages to the mobile app navigation menu. Lightning Page tabs don't work like other custom tabs. Once created, they don't show up on the All Tabs page when you click the Plus icon that appears to the right of your current tabs. Lightning Page tabs also don't show 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.

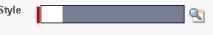
New Custom Object Tab

Step 1. Enter the Details Step 1 of 3

Choose the custom object for this new custom tab. Fill in other details.

Select an existing custom object or [create a new custom object now](#)

Object: Customer Details

Tab Style:  [Edit](#) [Delete](#)

(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.

Splash Page Custom Link: [--None--](#)

Enter a short description.

Description:

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

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](#)

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:

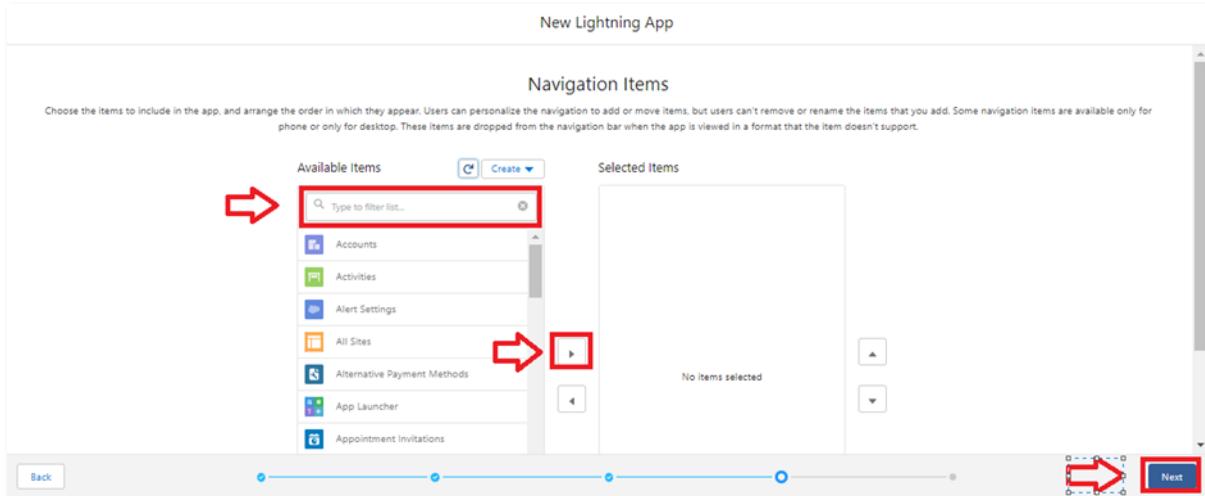
1. Go to setup page >> search “app manager” in quick find >> select “app manager”
>> click on New lightning App.

The screenshot shows the Salesforce App Manager interface. At the top, there is a search bar with "app manager" typed in. Below the search bar, there are navigation tabs for "Setup", "Home", and "Object Manager". A red box highlights the search bar. To the right of the search bar, there is a "New Lightning App" button, which is also highlighted with a red arrow. On the left side, there is a sidebar with sections for "Apps" and "Enable App Cloning (Beta)". A red box highlights the "Enable App Cloning" section. The main area displays a table of existing apps, with 13 items listed. The columns include "App Name", "Developer Name", "Description", "Last Modified", "App Type", and "V...". A red arrow points to the "New Lightning App" button at the top right of the page.

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.

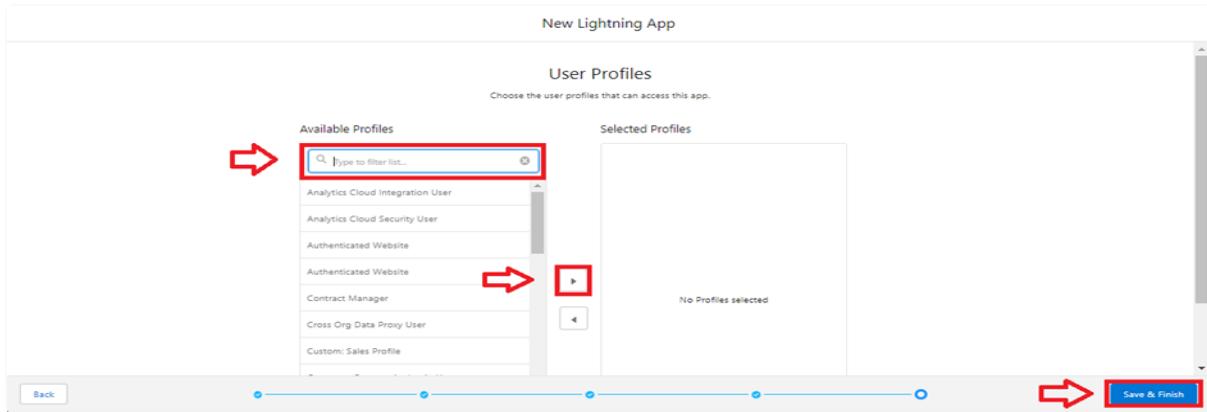
The screenshot shows the "New Lightning App" configuration page. The title is "New Lightning App". Under the heading "App Details & Branding", there is a sub-section titled "App Details". It contains fields for "App Name" (with a placeholder "Name your app..."), "Developer Name" (with a placeholder "Enter a developer name..."), and "Description" (with a placeholder "Enter a description..."). To the right of the "App Details" section is the "App Branding" section, which includes a "Primary Color Hex Value" field set to "#0070D2" and a "Upload" button for an "Image". Below the branding section is an "Org Theme Options" checkbox. At the bottom of the page is a "Next" button, which is highlighted with a red arrow.

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' typed in, a 'Schema Builder' button, and a 'Create' button. Below the header, a message says '2 items. Sorted by Label'. The main area is a table with columns: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. There are two rows: one for 'Customer' (Standard Object) and one for 'Customer Details' (Custom Object). The 'Customer Details' row is highlighted with a red border. In the 'DEPLOYED' column for 'Customer Details', there's a dropdown menu with a red border around the open options.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Customer	Customer	Standard Object			
Customer Details	Customer_Details__c	Custom Object		05/10/2023	✓

2. Now click on "Fields & Relationships" >> New

SETUP > OBJECT MANAGER
Customer1

Fields & Relationships

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
current Status	current_Status__c	Picklist		
Customer Name	Name	Text(80)		
Email id	Email_id__c	Email (Unique)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		
Permanent Address	Permanent_Address__c	Text Area(255)		
Phone no	Phone_no__c	Phone		

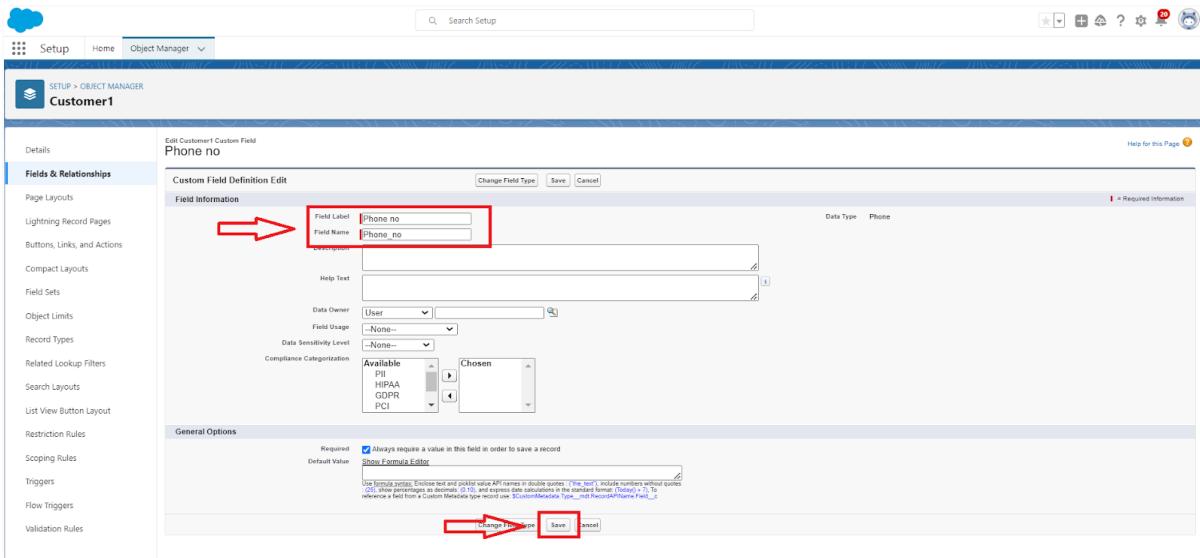
3. Select Data Type as a “Phone”

SETUP > OBJECT MANAGER
Customer1

Fields & Relationships

- Currency
- Date
- Date/Time
- Email
- Geolocation
- Number
- Percent
- Phone
- Picklist
- Picklist (Multi-Select)
- Text
- Text Area
- Text Area (Long)
- Text Area (Rich)
- Text (Encrypted) (1)
- Time
- URL

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 :

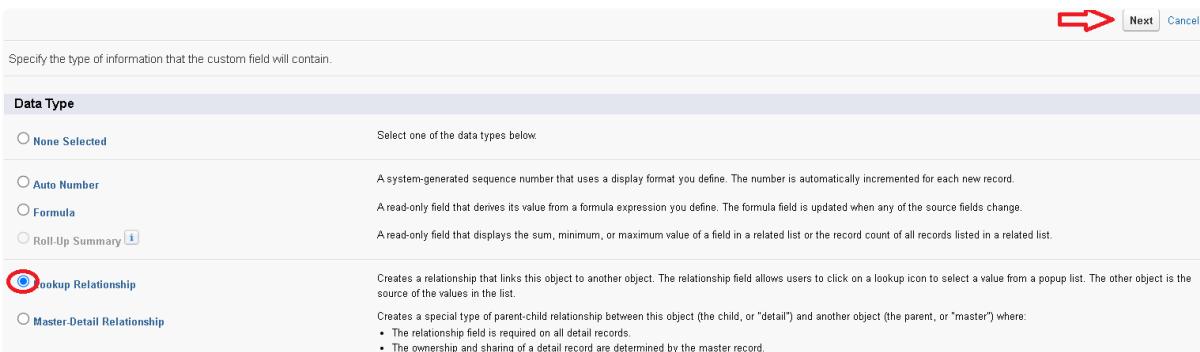
1. Go to setup >> click on Object Manager >> type object name(Appointment) in the search bar >> click on the object.

Object Manager					
LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Appointment	Appointment__c	Custom Object		24/06/2023	✓
Appointment Category	AppointmentCategory	Standard Object			
Appointment Invitation	AppointmentInvitation	Standard Object			
Appointment Invitee	AppointmentInvitee	Standard Object			

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

Fields & Relationships					
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.



Specify the type of information that the custom field will contain.

Data Type

None Selected Select one of the data types below.

Auto Number A system-generated sequence number that uses a display format you define. The number is automatically incremented for each new record.

Formula A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.

Roll-Up Summary A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.

Lookup Relationship Creates a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other object is the source of the values in the list.

Master-Detail Relationship Creates a special type of parent-child relationship between this object (the child, or “detail”) and another object (the parent, or “master”) where:

- The relationship field is required on all detail records.
- The ownership and sharing of a detail record are determined by the master record.

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.



Lookup Options

Related To: Appointment

Related List Label: Service records

Child Relationship Name: Service_records

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

What 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.
- Don't allow deletion of the lookup record that's part of a lookup relationship.

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!](#)

Filter Criteria **Insert Suggested Criteria** **Clear Filter Criteria**

Field	Operator	Value / Field
Appointment: Appointment Date	less than	Field <input style="width: 100px; height: 20px;" type="button" value="..."/> Appointment: Created Date <input style="width: 20px; height: 20px;" type="button" value="..."/> Clear
AND <input type="text" value="Begin typing to search for a field..."/> <input style="width: 20px; height: 20px;" type="button" value="..."/> --None-- <input style="width: 20px; height: 20px;" type="button" value="..."/> Value <input style="width: 20px; height: 20px;" type="button" value="..."/> Clear		

Add filter logic...

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

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

Lookup Window Text Add this informational message to the lookup window.

Active Enable this filter.

Buttons: Change Field Type | Save | Cancel

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.

Setup > Object Manager

Appointment

Details

Fields & Relationships Next Cancel

- Page Layouts
- Lightning Record Pages
- Buttons, Links, and Actions
- Compact Layouts
- Field Sets
- Object Limits
- Record Types
- Related Lookup Filters
- Search Layouts
- List View Button Layout
- Restriction Rules

Data Type

- None Selected Select one of the data types below.
- Auto Number A system-generated sequence number that uses a display format you define. The number is automatically incremented for each new record.
- Formula A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.
- Roll-Up Summary A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.
- Lookup Relationship Creates a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other object is the source of the values in the list.
 - The relationship field is required on all detail records.
 - The ownership and sharing of a detail record are determined by the master record.
 - When a user deletes the master record, all detail records are deleted.
 - You can create roll-up summary fields on the master record to summarize the detail records.
- Master-Detail Relationship The relationship field allows users to click on a lookup icon to select a value from a popup list. The master object is the source of the values in the list.
- External Lookup Relationship Creates a relationship that links this object to an external object whose data is stored outside the Salesforce org.
- Checkbox Allows users to select a True (checked) or False (unchecked) value.
- Currency Allows users to enter a dollar or other currency amount and automatically formats the field as a currency amount. This can be useful if you export data to Excel or another application.

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

Appointment
New Custom Field

Help for this Page ?

Step 2. Enter the details Step 2 of 4

Previous Next Cancel

Field Label i

Default Value Checked Unchecked ←

Field Name i

Description

Help Text

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

Previous Next Cancel

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: [i]

Field Name: [i]

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 [i]

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

This screenshot shows the 'Step 2. Enter the details' screen for creating a new field. The field label is set to 'Service Amount'. The length is specified as 18 digits, and the decimal places are set to 0. The field name is 'Service_Amount'. There is a large empty text area for the description and another for help text. At the bottom, there are checkboxes for 'Required' (unchecked), 'Always require a value in this field in order to save a record', 'Auto add to custom report type' (unchecked), and 'Add this field to existing custom report types that contain this entity' (checked). Navigation buttons at the top right include 'Previous', 'Next', and 'Cancel'.

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

This screenshot shows the 'Step 3. Establish field-level security' screen. It displays the field details: Field Label: Service Amounts, Data Type: Currency, Field Name: Service_Amounts, and Description. Below this, a table lists various profiles with checkboxes for 'Visible' and 'Read-Only'. A green arrow points to the 'Read-Only' column, which is checked for all profiles. Navigation buttons at the top right include 'Previous', 'Next', and 'Cancel'.

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"/>

8. Click on next >> save.

Creation of Currency Field on Billing details and feedback Object :

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

Creation of Text Fields

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

Step 2. Enter the details Step 2 of 4

Field Label: Vehicle number plate

Length: 10

Field Name: Vehicle_number_plate

Description:

Help Text:

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

Unique: Do not allow duplicate values
 Treat "ABC" and "abc" as duplicate values (case insensitive)
 Treat "ABC" and "abc" as different values (case sensitive)

External ID: Set this field as the unique record identifier from an external system

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

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.

New Custom Field

Step 2. Enter the details Step 2 of 4

Field Label: Service Status

Values:

- Use global picklist value set
- Enter values, with each value separated by a new line

Started
Completed

Display values alphabetically, not in the order entered

Use first value as default value

Restrict picklist to the values defined in the value set

Field Name: Service_Status

Description:

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

Field Label: Field Name:

Auto add to custom report type Add this field to existing custom report types that contain this entity [i](#)

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) / Cost`

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

Select a field, then click Insert. Labels followed by a ">" indicate that there are more fields available.

Insert Field

Service records >

- \$Api >
- \$Label >
- \$Organization >
- \$Profile >
- \$System >
- \$User >
- \$UserRole >

Appointment >

- Appointment
- Created By >
- Created By ID
- Created Date**
- Last Activity Date
- Last Modified By >
- Last Modified By ID
- Last Modified Date

You have selected:
CreatedDate
Type: Date/Time
API Name: CreatedDate

Insert

Close

Step 3. Enter formula

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)= **CreatedDate**

Insert Operator

Functions

-- All Function Categories --

- ABS
- ACOS
- ADDMONTHS
- AND
- ASCII

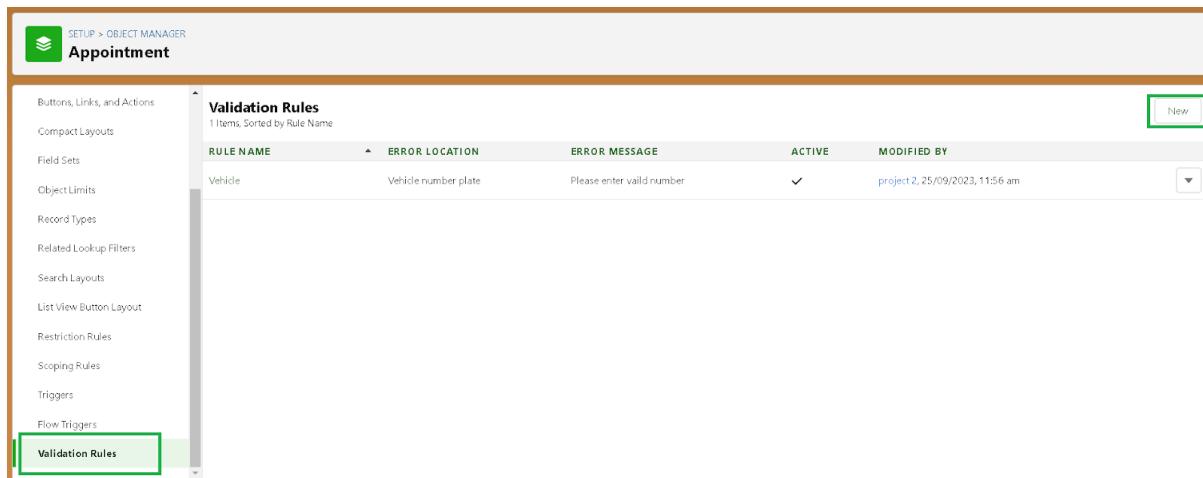
6. click "Check Syntax".
7. Click next >> next >> Save.

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. The left sidebar contains various setup options like Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules, Triggers, and Flow Triggers. The 'Validation Rules' link is highlighted with a green box. The main area displays a table titled 'Validation Rules' with one item listed:

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

A green box also highlights the 'New' button in the top right corner of the table header.

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 Vehicle

Active

Description

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

Insert Field Insert Operator ▾

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

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
[Help on this function](#)

Check Syntax

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:

This message will appear when Error Condition formula is true

Error Message Please enter valid number

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 Vehicle number plate ←

→ Save Save & New Cancel

To create a validation rule to an Service records obj.

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 :-

$$\text{NOT(ISPICKVAL(Service_Status__c , "Completed"))}$$

Validation Rule Edit

Save Save & New Cancel

Rule Name: service_status_note

Active:

Description:

Error Condition Formula

Example: Discount_Percent_c>0.30 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

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
Help on this function

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

Rule Name: service_status_note

Active:

Description:

Error Condition Formula

Example: Discount_Percent_c>0.30 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

NOT (ISPICKVAL(Service_Status__c , "Completed"))

Functions

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

Insert Selected Function
ABS(number)
Returns the absolute value of a number, a number without its sign
Help on this function

Check Syntax

- 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

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

Duplicate rule

To create a matching rule to an Customer details Object

- Go to quick find box in setup and search for matching Rule.
- Click on matching rule >> click on New Rule.

Setup Home Object Manager

Q: matching

SETUP Matching Rules

All Matching Rules

What Are Matching Rules?

View: All Matching Rules Create New View

Action	Rule Name	Object	Status	Description	Last Modified Date	Last Modified By
	New Rule					

- Select the object as Customer details and click Next.

Matching Rule
New Matching Rule

Step 1: Select object Step 1 of 2

Select the object to which this matching rule applies.

Object: Customer Details

Next Cancel

Up arrow icon

Next Cancel

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.

Save Cancel

Rule Details

Object: Customer Details
Rule Name: matching Customer data
Unique Name: matching_Customer_det
Description:

Matching Criteria

Tell the rule which fields to compare and how.

Field: Gmail	Matching Method: Exact	Match Blank Fields: AND
Field: Phone Number	Matching Method: Exact	Match Blank Fields: AND
--None--	Exact	AND
--None--	Exact	AND
--None--	Exact	AND

Add Filter Logic... Save Cancel

Matching Rule
matching Customer details

Help for this Page

Matching Rule Detail

Edit Delete Clone Activate

Object: Customer Details	Created By: project_2, 25/09/2023, 10:15 am	Modified By: project_2, 10/10/2023, 3:32 pm
Rule Name: matching Customer details		
Unique Name: matching_Customer_details		
Description:		
Matching Criteria: (Customer_Details: Gmail EXACT MatchBlank = FALSE) AND (Customer_Details: Phone_Number EXACT MatchBlank = FALSE)		
Status: Inactive		

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.

The screenshot shows the 'Duplicate Rules' page in the Salesforce Setup. The left sidebar has a search bar and navigation links for Data, Duplicate Management (with 'Duplicate Rules' highlighted), Duplicate Error Logs, Matching Rules, and Global Search. The main area is titled 'All Duplicate Rules' and contains a section 'What Are Duplicate Rules?' with a 'View' dropdown set to 'All Duplicate Rules'. Below is a table of existing rules:

Rule 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		Standard Contact Matching Rule	<input checked="" type="checkbox"/>	q2	24/09/2023
Environment		Standard Lead Matching Rule	<input checked="" type="checkbox"/>	q2	24/09/2023
Individual		Standard Contact Matching Rule	<input checked="" type="checkbox"/>	q2	24/09/2023
Laptop					
Lead					

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' page for the rule named 'Customer Detail duplicate'. The page has tabs for 'Rule Details' and 'Actions'. In 'Rule Details', the 'Rule Name' is 'Customer Detail duplicate' (highlighted with a green arrow), 'Description' is empty, 'Object' is 'Customer Details', and 'Record-Level Security' is set to 'Enforce sharing rules'. In the 'Actions' section, there are fields for 'Action On Create' (Allow, Alert checked, Report unchecked), 'Action On Edit' (Allow, Alert unchecked, Report unchecked), and '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

Add Rule Remove Rule

Conditions

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

Field	Operator	Value	AND
--None--	--None--		AND
--None--	--None--		

Add Filter Logic

Save 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 deleted standard ones

Each of these standard ones includes a default set of permissions for all of the standard objects available on the platform.

2. Custom Profiles:

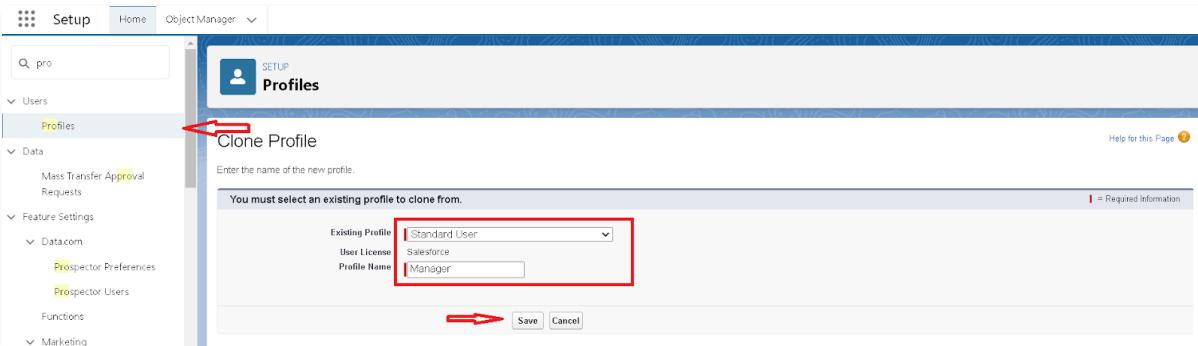
Custom ones defined by us.

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

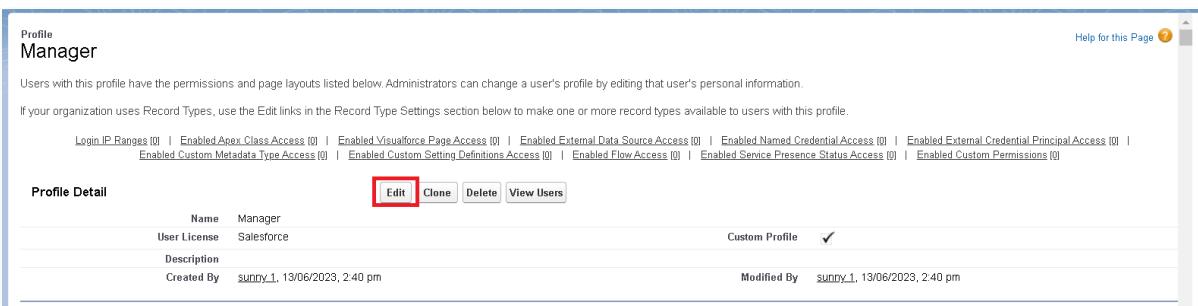
Manager Profile

To create a new profile:

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



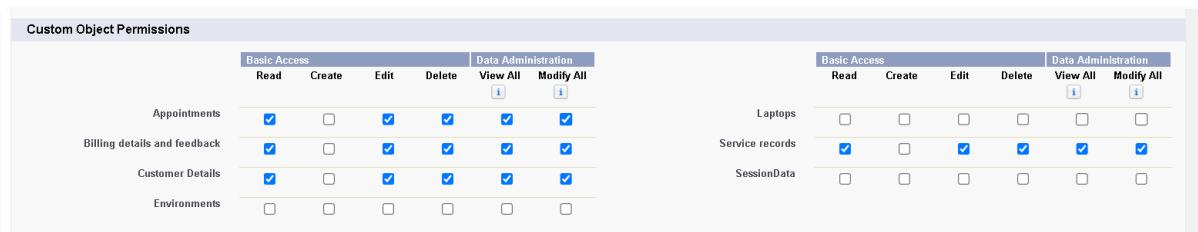
2. While still on the profile page, then click Edit.



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



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



5. Changing the session times out after should be “ 8 hours of inactivity”.

6. Change the password policies as mentioned :
7. User passwords expire in should be “ never expires ”.
8. Minimum password length should be “ 8 ”, and click save.

sales person Profile

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

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.

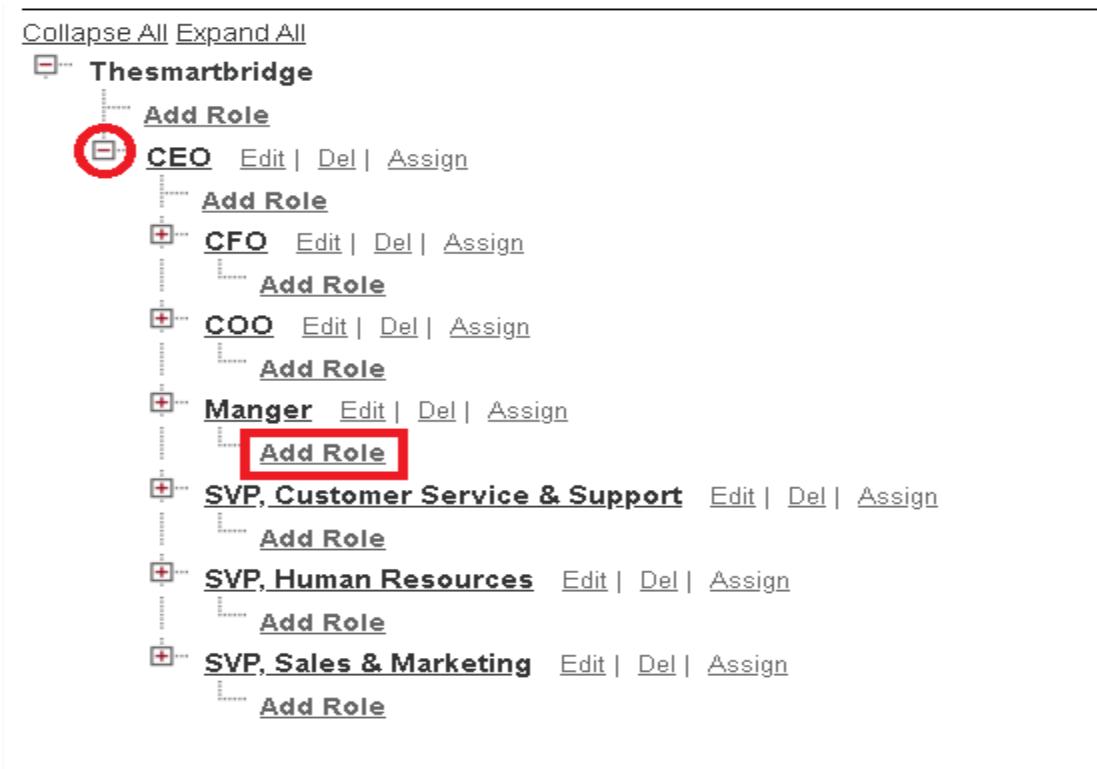
2. Click on Expand All and click on add role under whom this role works.

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

Creating another roles

Creating another two roles under manager

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



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

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	nmika
Email	(redacted)
Username	Mikaelson@Niklaus
Nickname	nik
Title	(redacted)
Company	(redacted)
Department	(redacted)
Division	(redacted)

Required Information

Role	Manager
User License	Salesforce
Profile	Manager
Active	<input checked="" type="checkbox"/>
Marketing User	<input type="checkbox"/>
Offline User	<input type="checkbox"/>
Knowledge User	<input type="checkbox"/>
Flow User	<input type="checkbox"/>
Service Cloud User	<input type="checkbox"/>
Site.com Contributor User	<input type="checkbox"/>
Site.com Publisher User	<input type="checkbox"/>
WDC User	<input type="checkbox"/>
Data.com User Type	--None--

3. Save.

creating another users

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

Note : create atleast 3 users with these permissions.

Public groups

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

Creating New Public Group

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

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

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.

SETUP

Sharing Settings

Work Plan Template	Private	Private	<input checked="" type="checkbox"/>
Work Step Template	Private	Private	<input checked="" type="checkbox"/>
Work Type	Private	Private	<input checked="" type="checkbox"/>
Work Type Group	Public Read/Write	Private	<input checked="" type="checkbox"/>
Appointment	Public Read/Write	Private	<input checked="" type="checkbox"/>
Billing details and feedback	Public Read/Write	Private	<input checked="" type="checkbox"/>
Customer Details	Public Read/Write	Private	<input checked="" type="checkbox"/>
Environment	Public Read/Write	Private	<input checked="" type="checkbox"/>
Laptop	Public Read/Write	Private	<input checked="" type="checkbox"/>
Service records	Private	Private	<input checked="" type="checkbox"/>
SessionData	Public Read/Write	Private	<input checked="" type="checkbox"/>

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 record 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 Service records Sharing Rules Help ?

No sharing rules specified.

 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.

The screenshot shows the 'Sharing Settings' wizard in Salesforce. It consists of five steps:

- Step 1: Rule Name**: Shows fields for Label (sharing settings), Rule Name (sharing_settings), and Description. A red arrow points to the 'sharing settings' label.
- Step 2: Select your rule type**: Shows two radio buttons: 'Based on record owner' (selected) and 'Based on criteria'. A red arrow points to the selected radio button.
- Step 3: Select which records to be shared**: Shows dropdowns for 'Service records: owned by members of' (Roles) and 'Sales person'. A red arrow points to the 'Sales person' dropdown.
- Step 4: Select the users to share with**: Shows dropdowns for 'Share with' (Roles) and 'Manager'. A red arrow points to the 'Manager' dropdown.
- Step 5: Select the level of access for the users**: Shows a dropdown for 'Access Level' (Read/Write). A red arrow points to the 'Read/Write' dropdown.

At the bottom are 'Save' and 'Cancel' buttons, with a red arrow pointing to the 'Save' button.

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. On the left, there's a sidebar with a search bar containing 'flows' (1). Below it, under 'Process Automation', is a 'Flows' link (2). The main area is titled 'Flows' and shows a list of flow definitions. At the top right of this list is a 'New Flow' button (3).

Flow Label	Process Type	Ac...	Te...	Package State	Pa...	Last Modified By	Last Modified ...
Ac Amount update	Autlaunched Flow	<input type="checkbox"/>	<input type="checkbox"/>	Unmanaged		Veera Venkata Varaprasad Androthu	07/06/2023, 11:35 am
Book Appointment from Invitation	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Cancel Item Flow	Screen Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Change Case Owner to Incident Owner	Screen Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Close Change Request & Related Issues	Screen Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			

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

New Flow

Core All + Templates

Screen Flow Guides users through a business process that's launched from Lightning pages, Experience Cloud sites, quick actions, and more.	Record-Triggered Flow Launches when a record is created, updated, or deleted. This autolaunched flow runs in the background.
Schedule-Triggered Flow Launches at a specified time and frequency for each record in a batch. This autolaunched flow runs in the background.	Platform Event—Triggered Flow Launches when a platform event message is received. This autolaunched flow runs in the background.
Autolaunched Flow (No Trigger) Launches when invoked by Apex, processes, REST API, and more. This autolaunched flow runs in the background.	Record-Triggered Orchestration Launches when a record is created or updated. An orchestration lets you create a multi-step, multi-user process.

1

2 Create

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

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

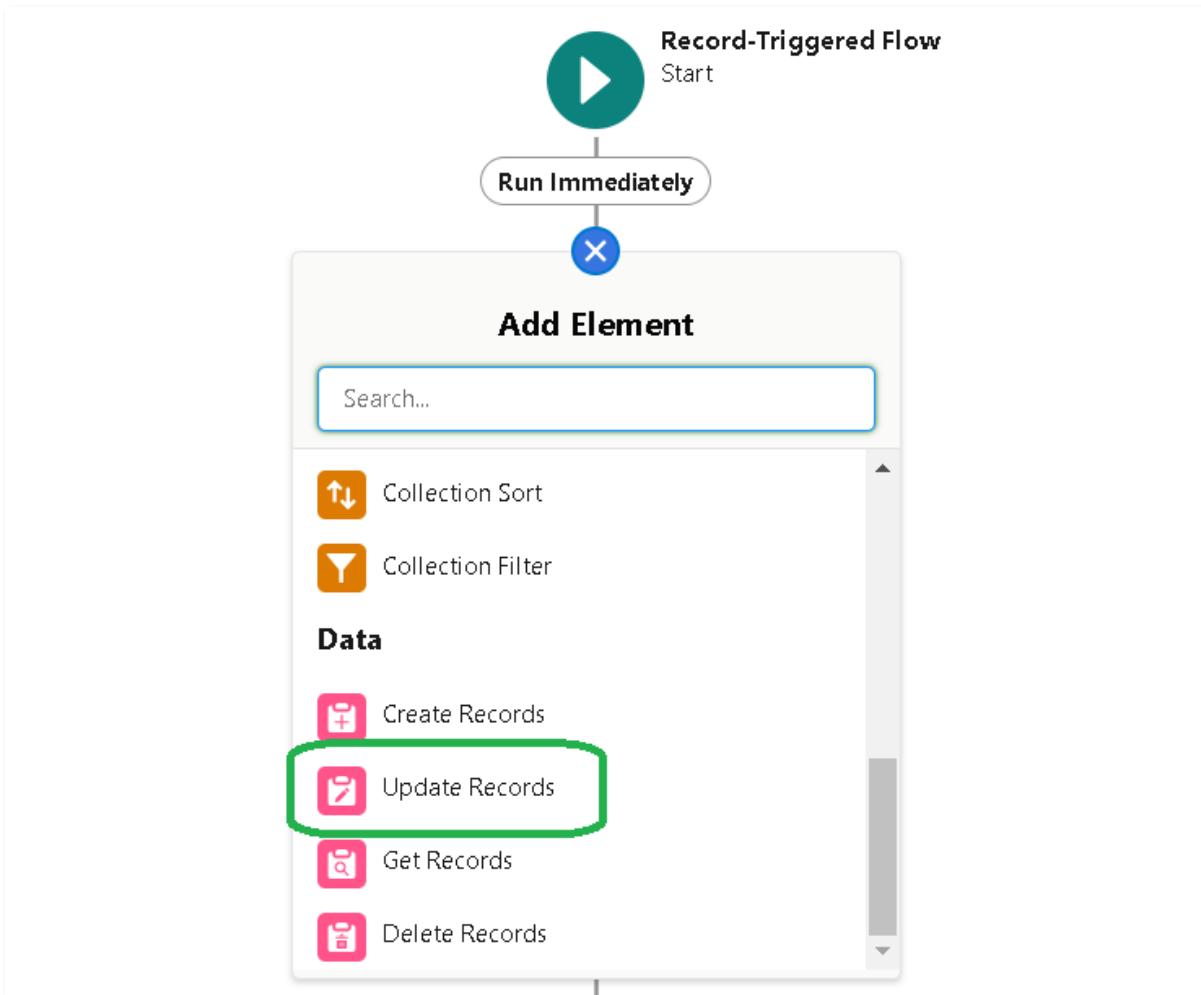
*Optimize the Flow for:

Fast Field Updates	Update fields on the record that triggers the flow to run. This high-performance flow runs <i>before</i> 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 <i>after</i> the record is saved to the database.

3

4 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	
<p>* How to Find Records to Update and Set Their Values</p> <ul style="list-style-type: none"> <input checked="" type="radio"/> Use the billing details and feedback record that triggered the flow <input type="radio"/> Update records related to the billing details and feedback record that triggered the flow <input type="radio"/> Use the IDs and all field values from a record or record collection <input type="radio"/> 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...

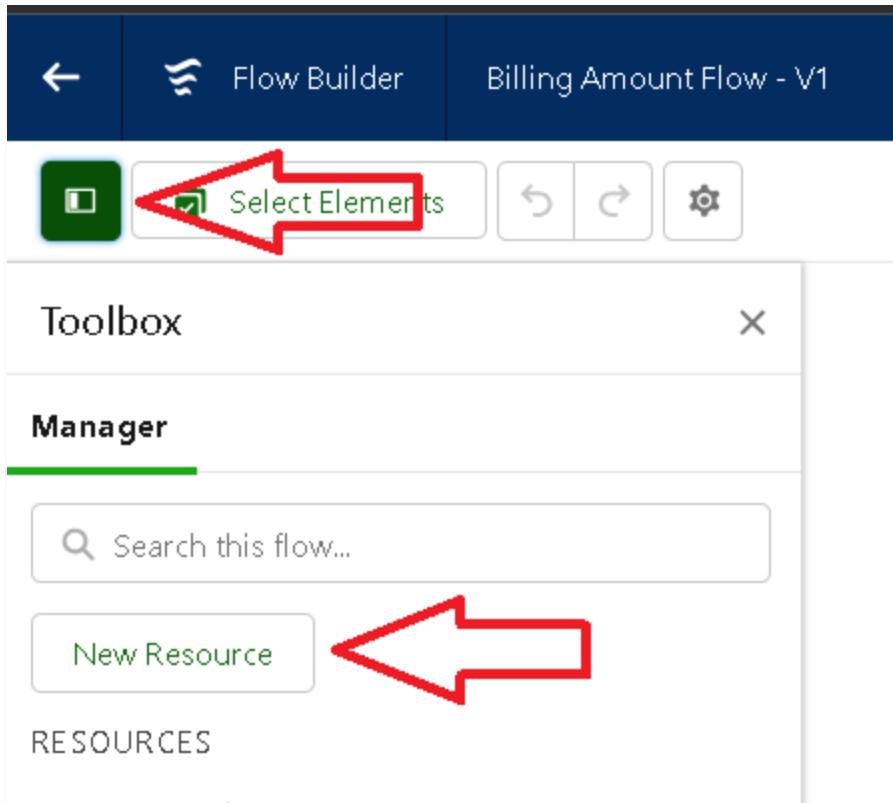
+ Add Field

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.

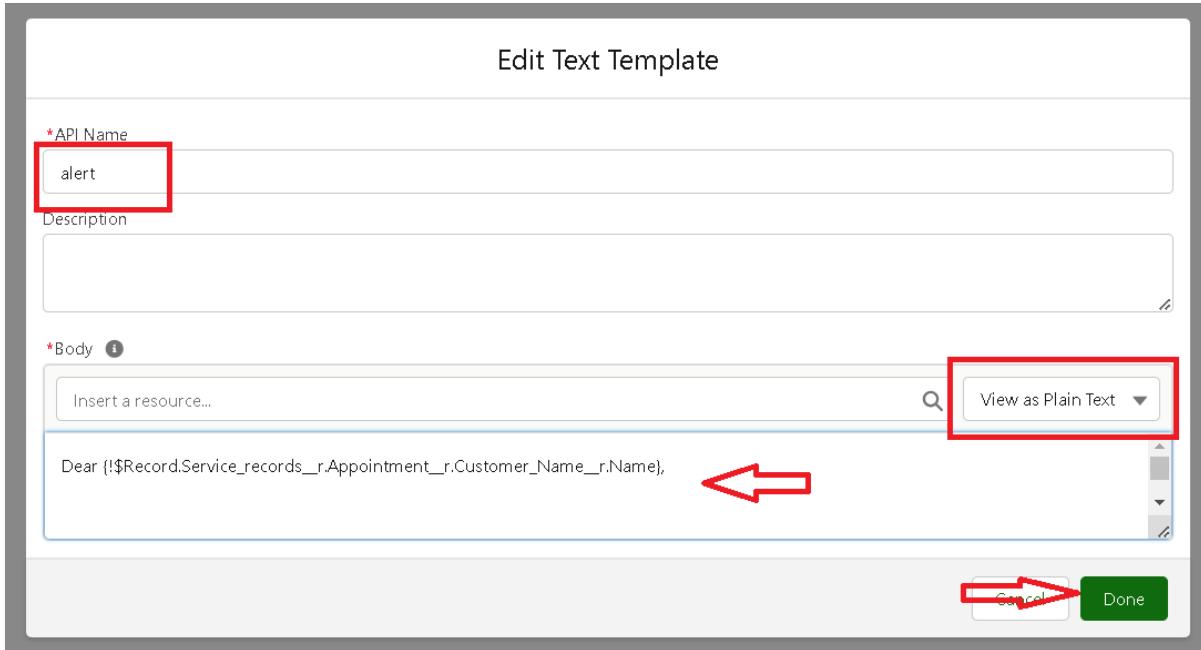
Dear {!\$Record.Service_records__r.Appointment__r.Customer_Name__r.Name},

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 ⓘ

{!alert}

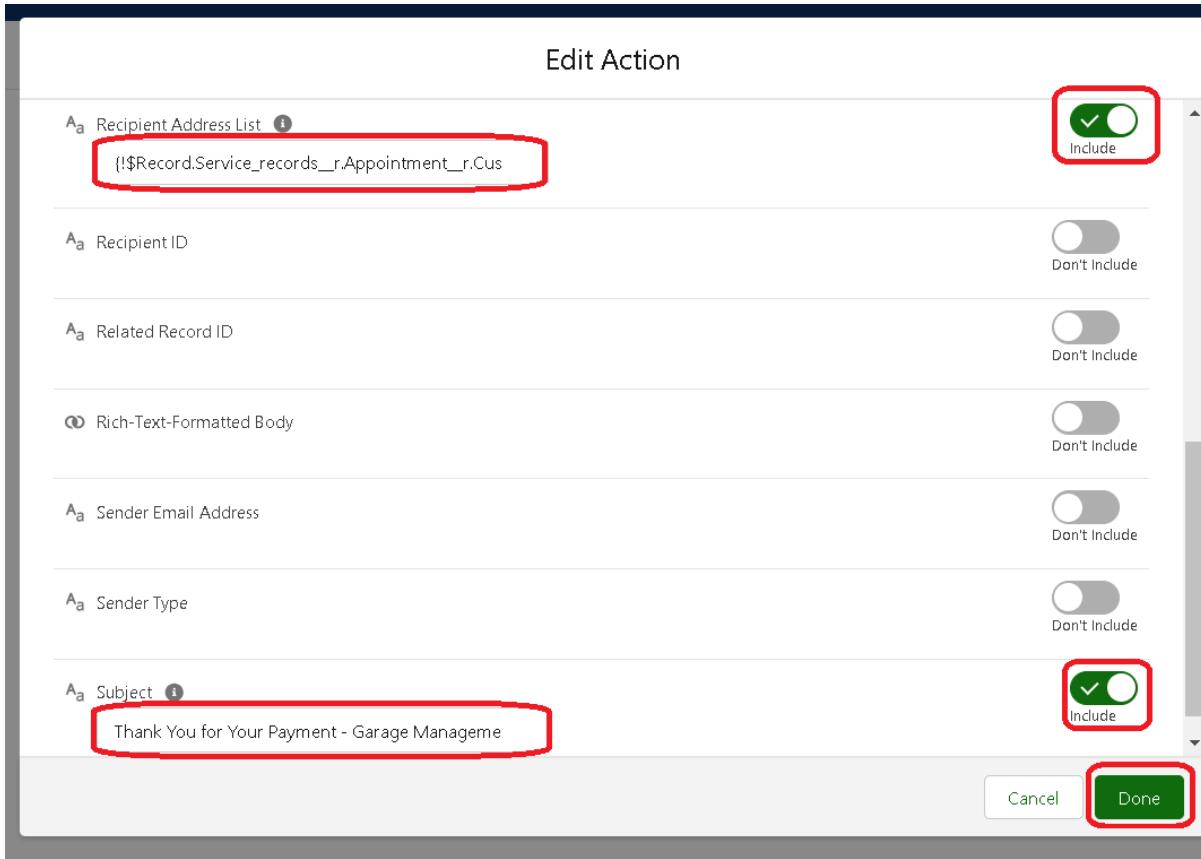


Aa Email Template ID



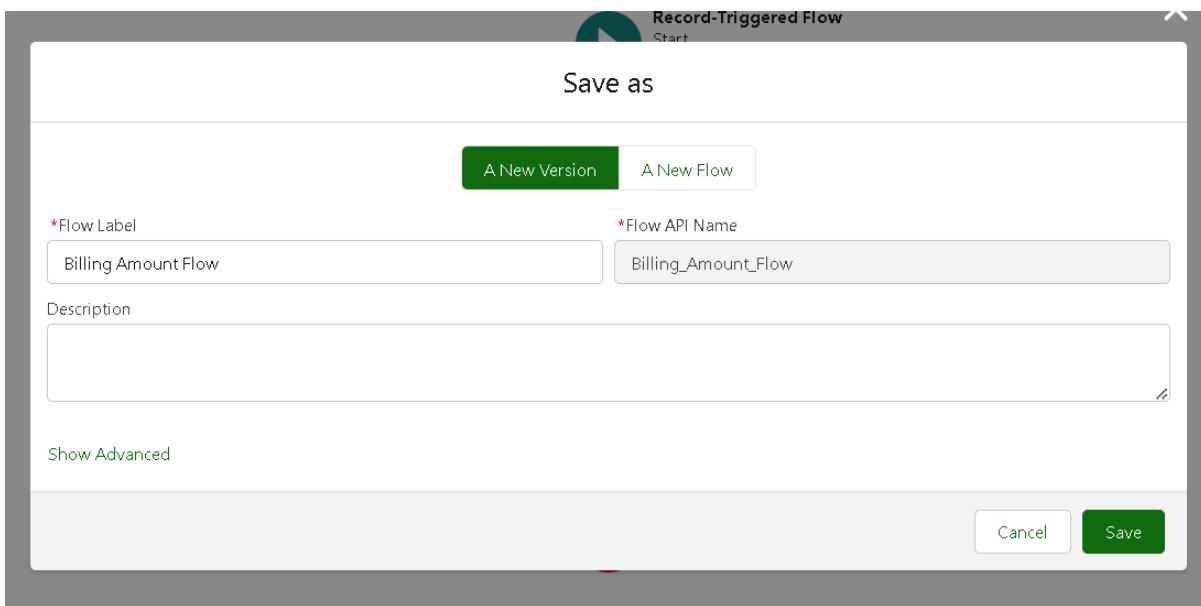
@@ Log Email on Send

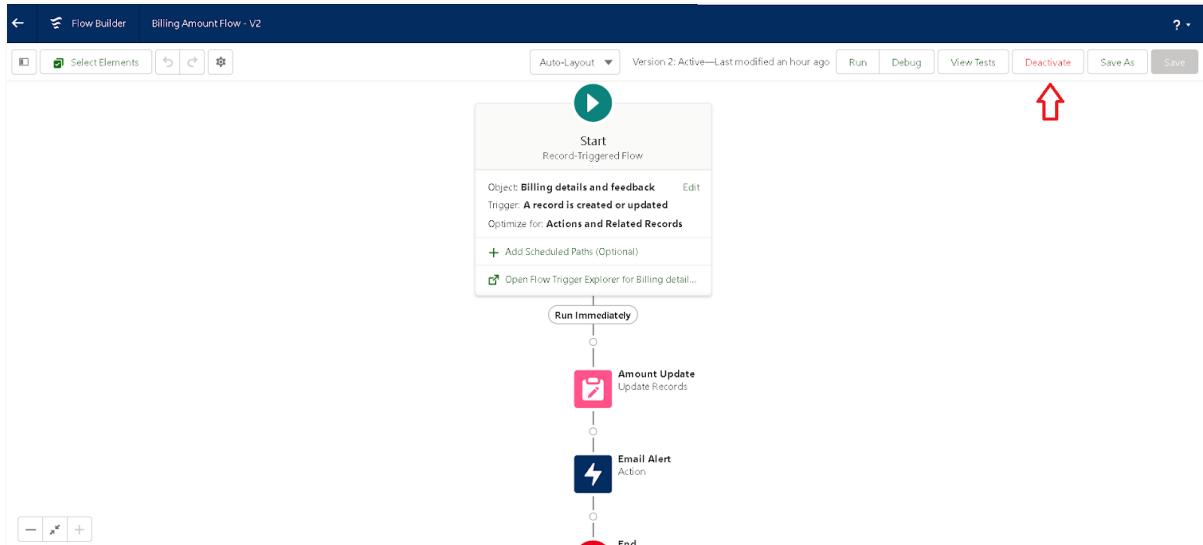




34. Click on save. Give the Flow label , Flow Api name will be autopopulated.

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:

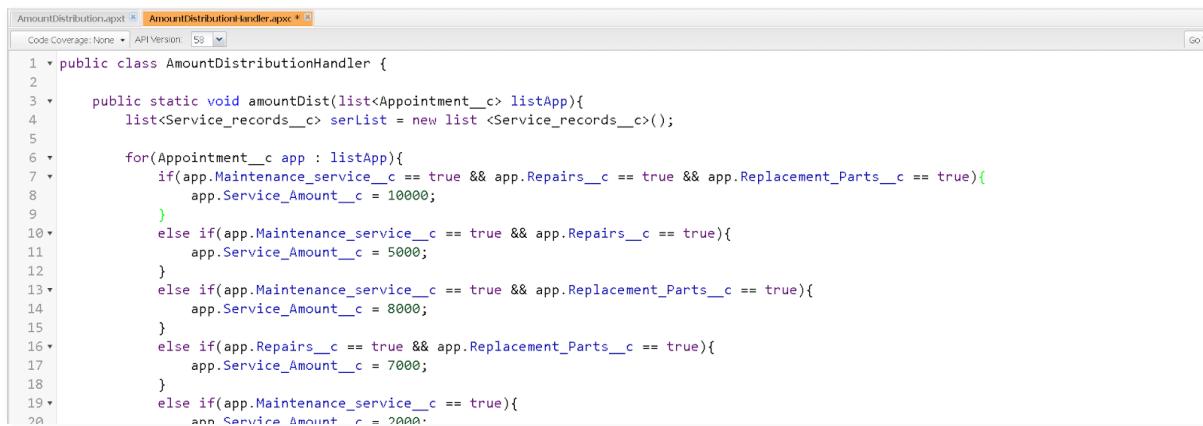
Before Trigger: This type of trigger in Salesforce is used either to update or validate the values of a record before they can be saved into the database. So, basically, the before trigger validates the record first and then saves it. Some criteria or code can be set to check data before it gets ready 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.apxtx AmountDistributionHandler.apxc * 
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.Repairs__c == true &&
14             app.Replacement_Parts__c == true){
15         app.Service_Amount__c = 8000;
16     }
17     else if(app.Repairs__c == true && app.Replacement_Parts__c == true){
18         app.Service_Amount__c = 7000;
19     }
20     else if(app.Maintenance_service__c == true){
21         app.Service_Amount__c = 2000;
22     }
23     else if(app.Repairs__c == true){
24         app.Service_Amount__c = 3000;
25     }
26     else if(app.Replacement_Parts__c == true){
27         app.Service_Amount__c = 5000;
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;
}

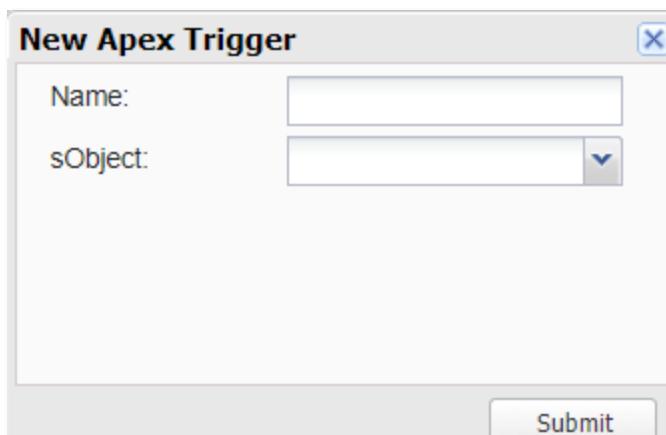
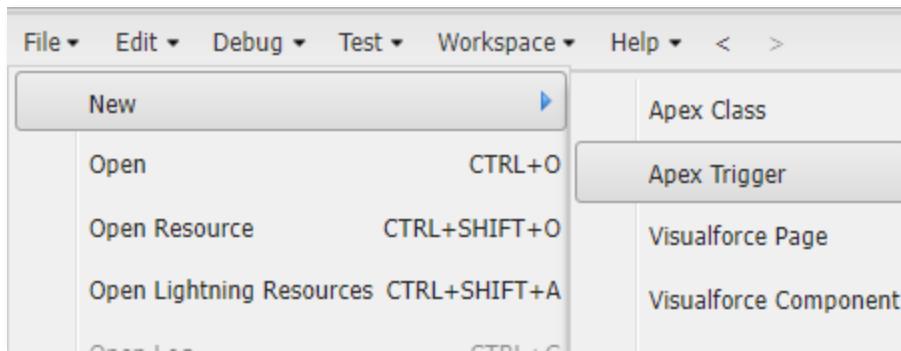
}
}
}

```

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

The screenshot shows a Salesforce code editor window. The title bar includes 'File', 'Edit', 'Debug', 'Test', 'Workspace', 'Help', and navigation buttons. The active tab is 'AmountDistribution.apxt'. Below the tabs, there's a dropdown for 'Code Coverage: None' and an API Version selector set to '58'. The main area contains the following Apex code:

```
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

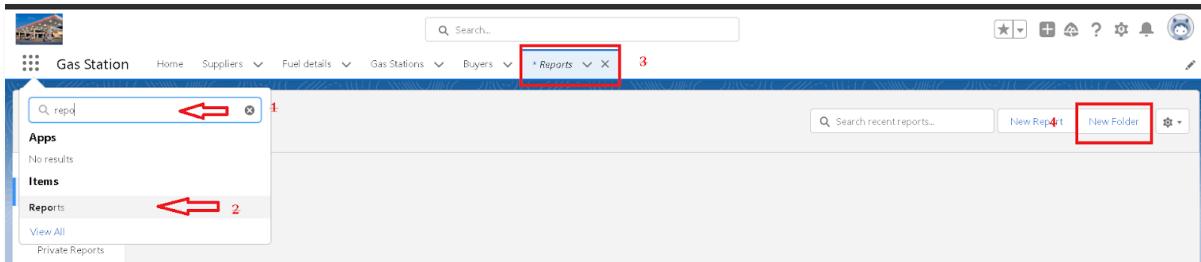
Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics.

Types of Reports in Salesforce

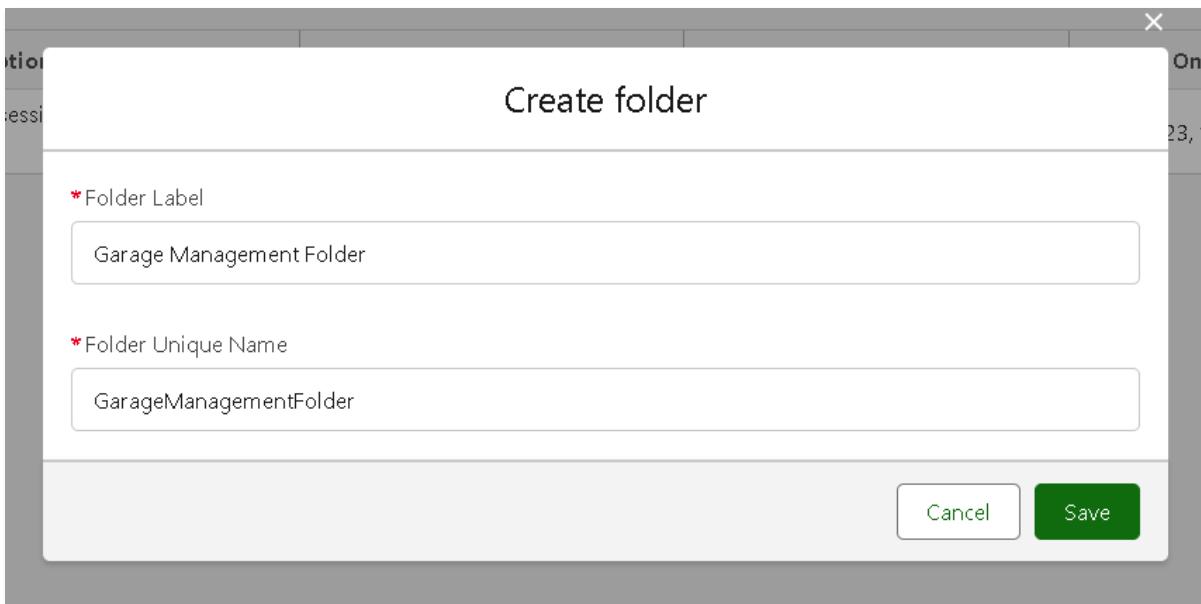
1. Tabular
2. Summary
3. Matrix
4. Joined Reports

create a report folder

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

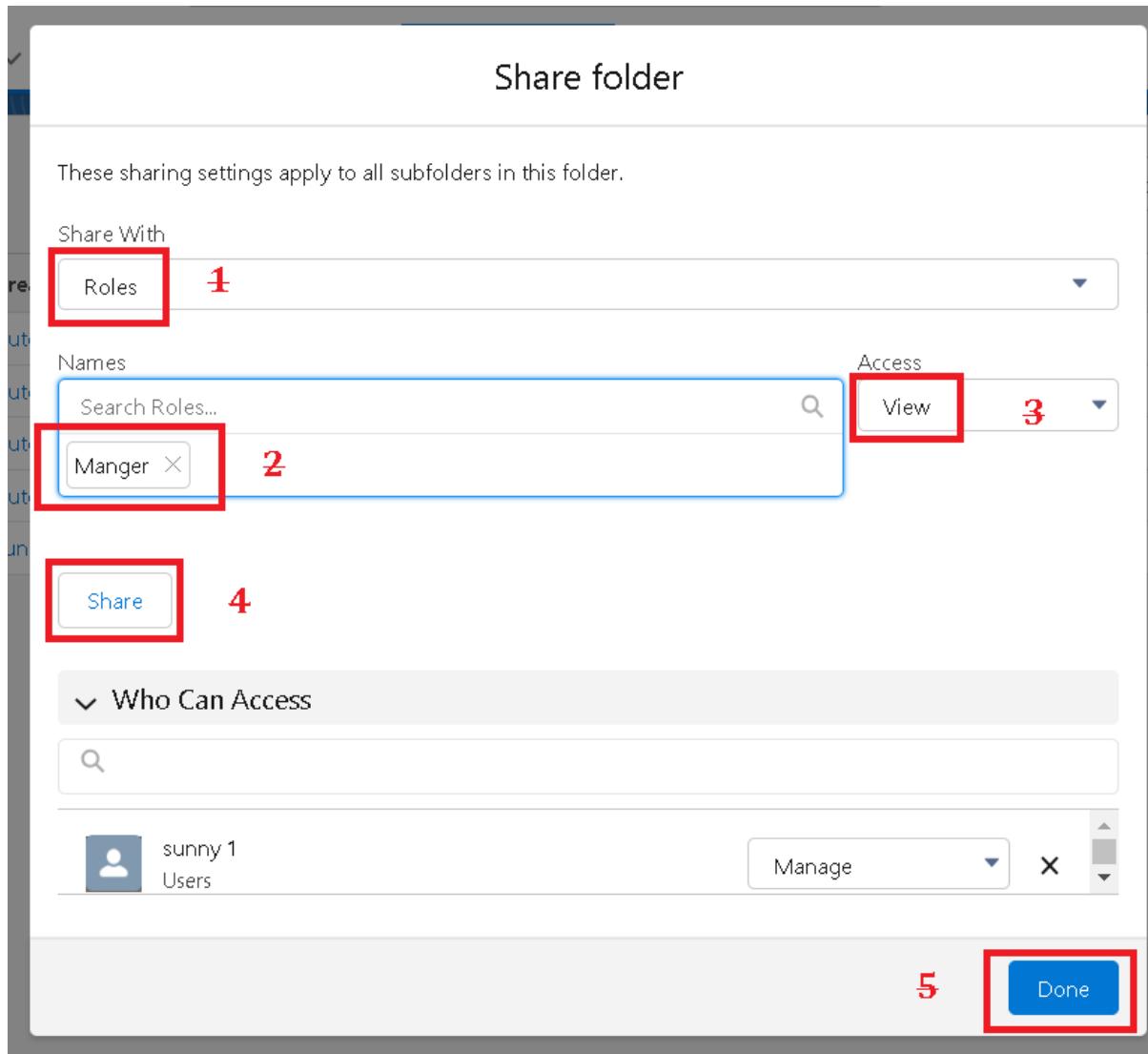


3. Give the Folder label as "Garage Management Folder", Folder unique name will be auto populated.
4. Click save.



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.

The screenshot shows the Salesforce Setup interface with the 'Report Types' page selected. The sidebar on the left has a 'Report Types' link highlighted with a green arrow. The main area displays a list of 'All Custom Report Types' with a 'New Custom Report Type' button at the top. Another green arrow points to this button.

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.

The screenshot shows the 'Report Type Focus' configuration page. The 'Primary Object' field is set to 'Customer Details'. The 'Report Type Label' and 'Report Type Name' fields are both set to 'Service information'. The 'Description' field contains 'Service information'. The 'Store in Category' field is set to 'Other Reports'. In the 'Deployment' section, the 'Deployed' radio button is selected. A green arrow points to the 'Next' button at the bottom right.

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

Step 2. Define Report Records Set Step 2 of 2

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--

Activities
Appointments
Duplicate Record Items

At least one related "B" record.
related "B" records.

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

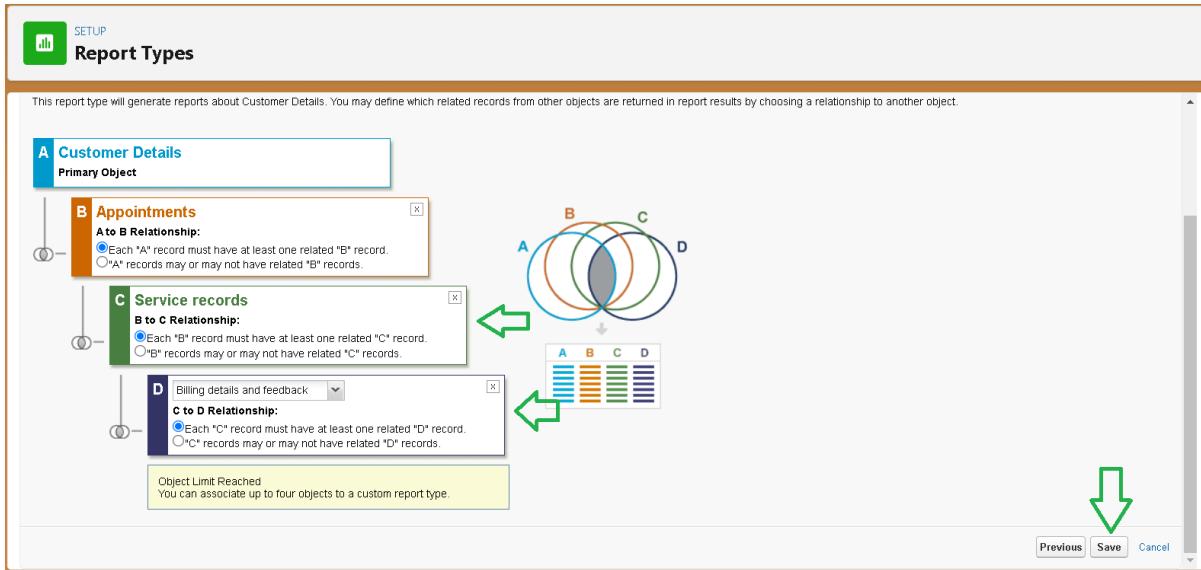
B Appointments --Select Object--

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.

Click to relate another object

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.

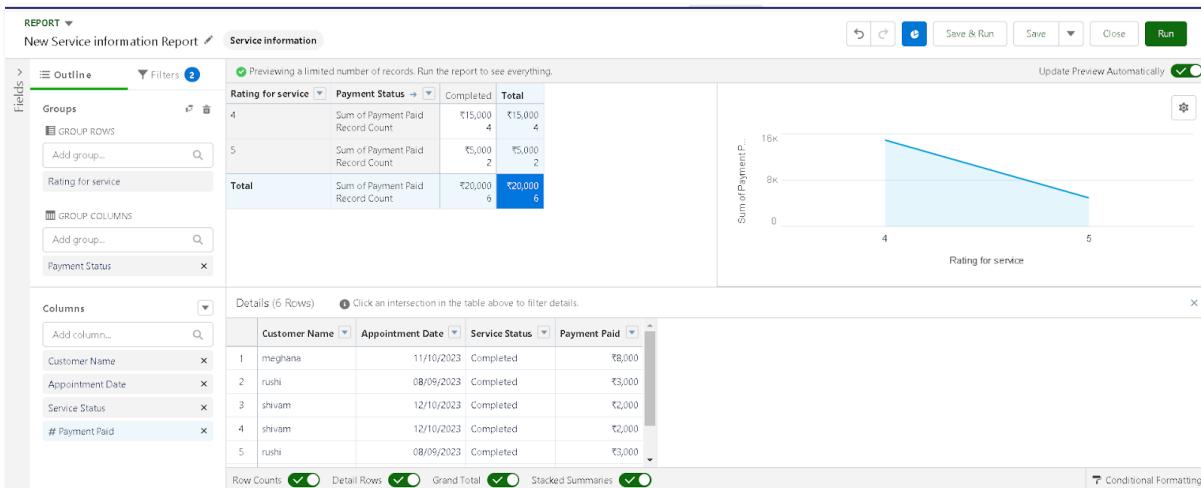
REPORTS	Report Name	Description	Folder	Created By	Created On	Subscribed
Recent	Employee's working on projects report	Private Reports	Employee Project	5/6/2023, 9:33 am		
	Assets assigned to Employees	Private Reports	Employee Project	5/6/2023, 9:36 am		

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 section for 'Other Reports'. The main area is titled 'Select a Report Type' with a search bar. It lists several report types: Service records (Standard), Service records with Appointment (Standard), Service records History (Standard), Billing details and feedback with Service records (Standard), and Service information (Custom). Two green arrows point to the 'Service information' row: one from the right towards the 'Category' dropdown, and another from the bottom towards the 'Start Report' button. To the right of the list is a 'Details' panel for the 'Service information' report, which includes sections for 'Service information' (Custom Report Type), 'Start Report' (button), 'Details' (link), 'Fields (49)' (link), 'Description' (Service information), 'Created By You' (No Reports Yet), and 'Created By Others' (No Reports Yet).

4. Their outline pane is opened already, select the fields that mentioned below in column section.
 1. Customer name
 2. Appointment Date
 3. Service Status
 4. Payment paid
5. Remove the unnecessary fields.
6. Select the fields that mentioned below in GROUP ROWS section.
 1. Rating for Service
7. Select the fields that mentioned below in GROUP ROWS section.
 1. 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.



Save Report

*Report Name
New Service information Report 

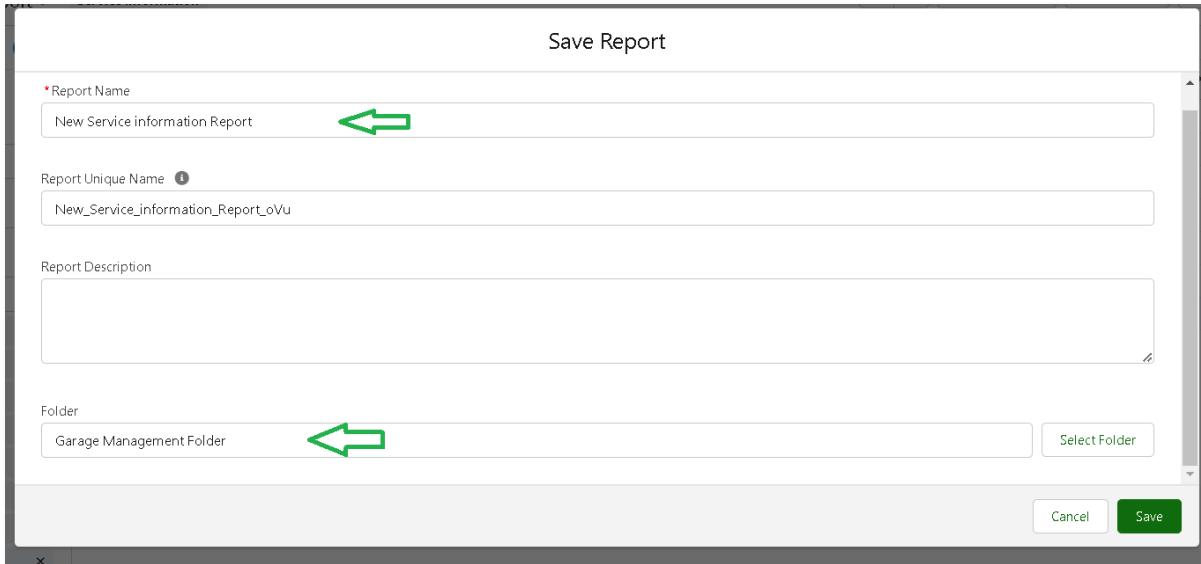
Report Unique Name ⓘ
New_Service_information_Report_oVu

Report Description

Folder
Garage Management Folder 

Select Folder

Cancel Save

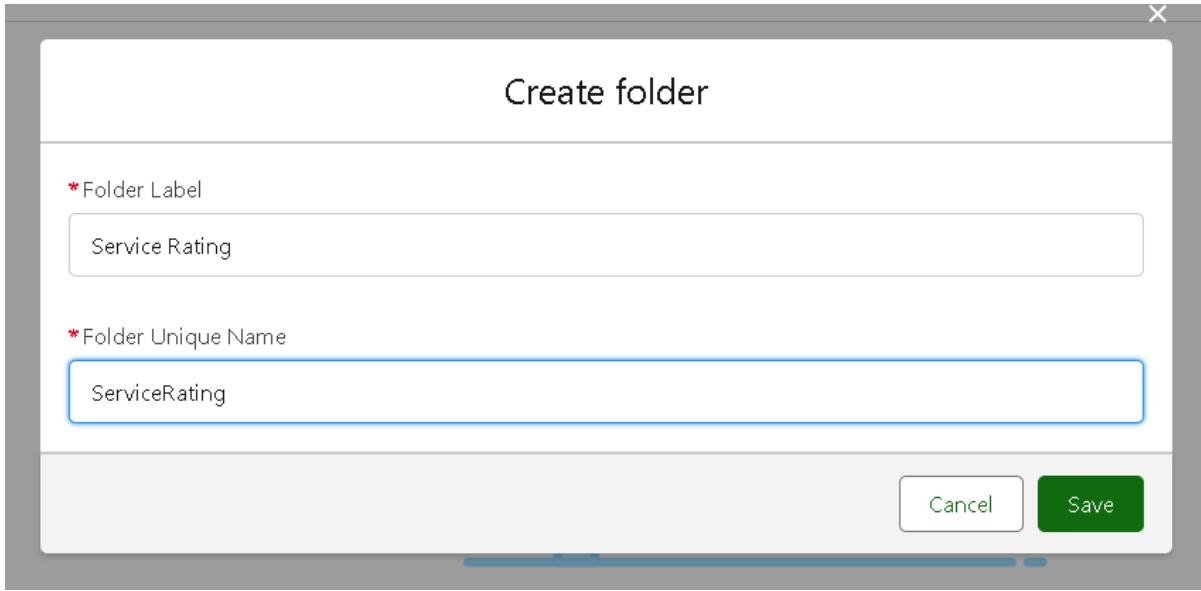


Dashboards

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.



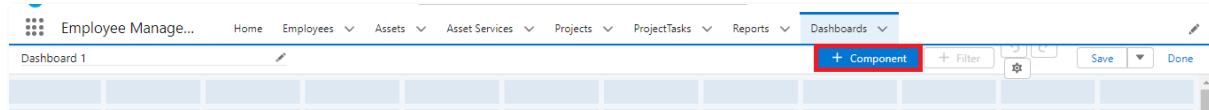
6. Follow the same steps, from milestone 15, and activity 2, and provide the sharing settings for the folder that just created.

Create Dashboard

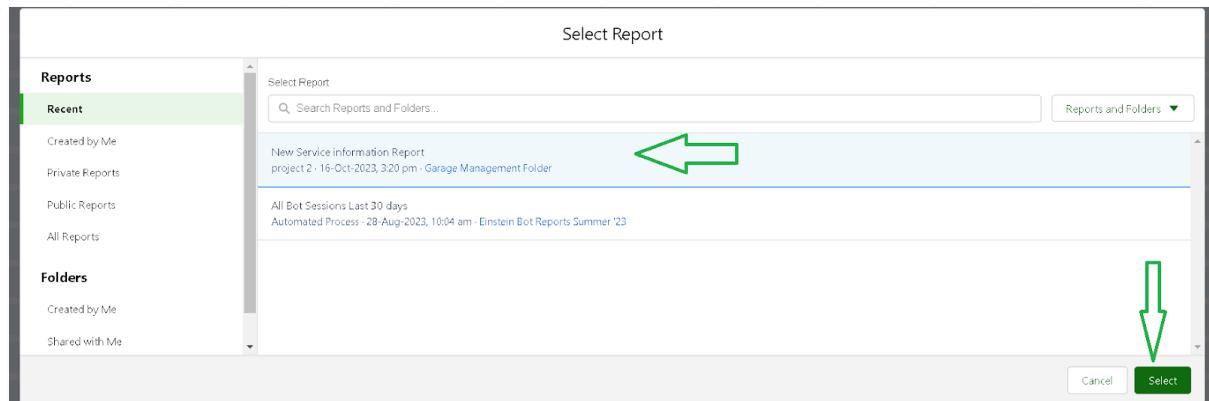
1. Go to the app >> click on the Dashboards tabs.
2. Give a Name and select the folder that created, and click on create.

The screenshot shows a 'New Dashboard' dialog box. It has three main input fields: 'Name' (containing 'Customer review'), 'Description' (empty), and 'Folder' (containing 'Service Rating'). A 'Select Folder' button is located to the right of the 'Folder' field. At the bottom right are 'Cancel' and 'Create' buttons, with 'Create' being highlighted with a green border.

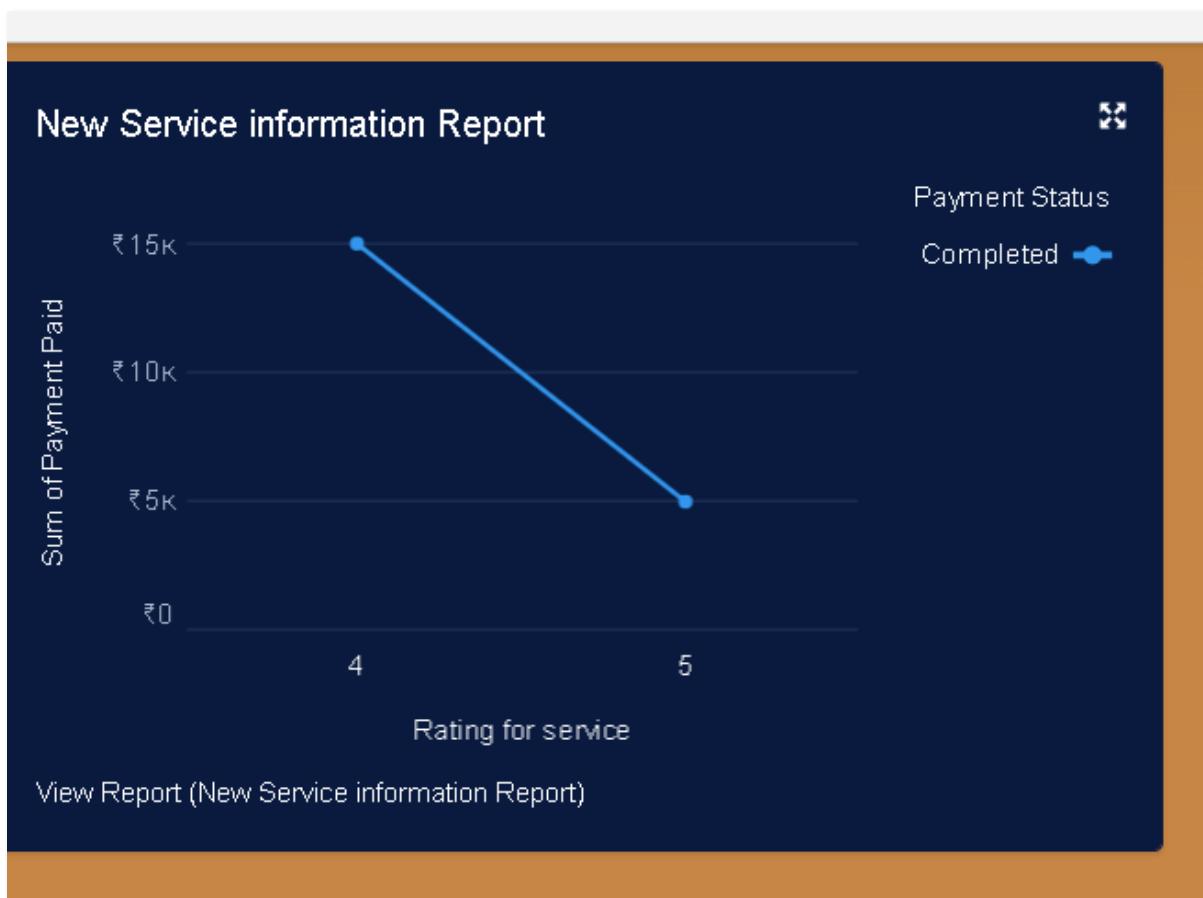
3. Select add component.



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.

Edit Subscription

Schedule dashboard refreshes and subscribe to receive results.

Settings

Frequency

Daily Weekly Monthly

Days

Sun Mon Tue Wed Thu Fri Sat

Time

3:00 pm

Recipients

Receive new results by email when dashboard is refreshed. (i)

Send email to

Me

[Edit Recipients](#)

[Cancel](#) [Save](#)