

Garage Management system

ProjectDescription:

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

ProjectFlow:

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

Milestone 1: Creation of developer account

Milestone 2: Object Creation

Milestone 3: Tabs Creation

Milestone 4 : Create App

Milestone 5: Fields & Relationships

Milestone 6 : Page Layout

Milestone 7 : Profile

Milestone 8: Role and role hierarchy

Milestone 9 : Users

Milestone 10: Permission set

Milestone 11: User Adoption

Milestone 12 : Reports

Milestone 13 : Dashboards

Milestone 14 : Flows

Milestone 15 : Apex Classes and Triggers

What you'll learn

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

Submitted by:

CHINNAMANI S (au711221104010)

SURYA PRAKASH H (au711221104054)

GOKUL S (au711221104301)

MANOJ KUMAR S (au711221104035)

JAI SHRI RAM ENGINEERING

TIRUPUR.

9. Email alerts and email templates

Milestone1-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.

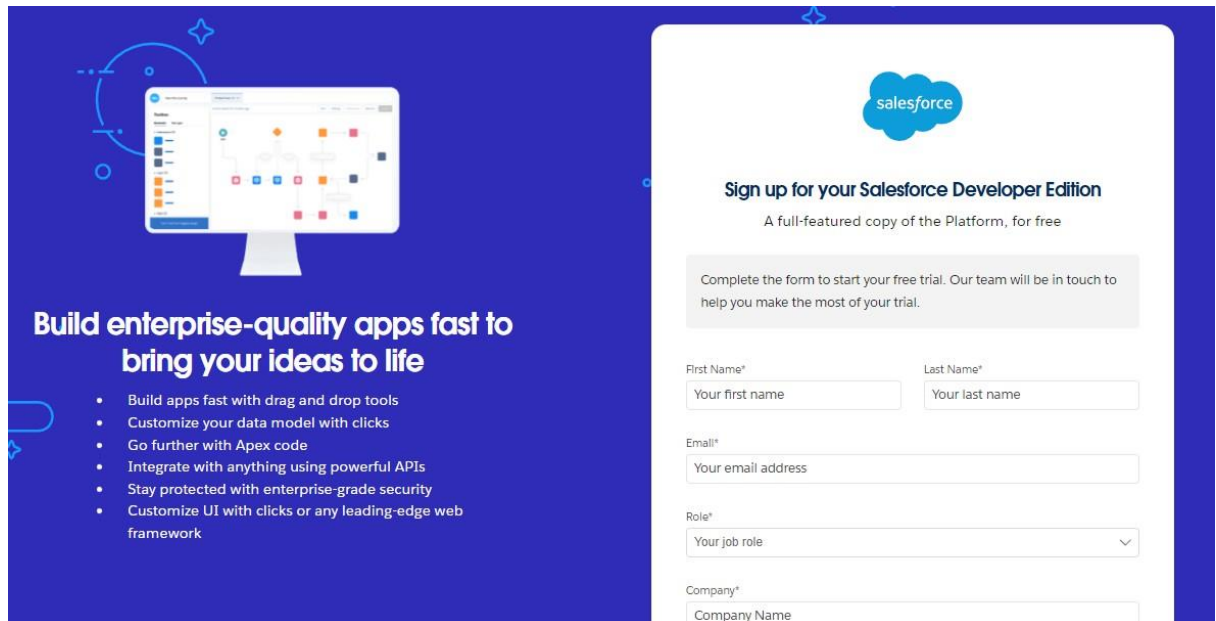
So what does that really mean? Well, before Salesforce, your contacts, emails, follow-up tasks, and prospective deals might have been organized something like this:

<https://youtu.be/r9EX3IGde5k>

Activity1:CreatingDeveloperAccount:

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) Country : India
- 6) Postal Code : pin code
- 7) Username : should be a combination of your name and company

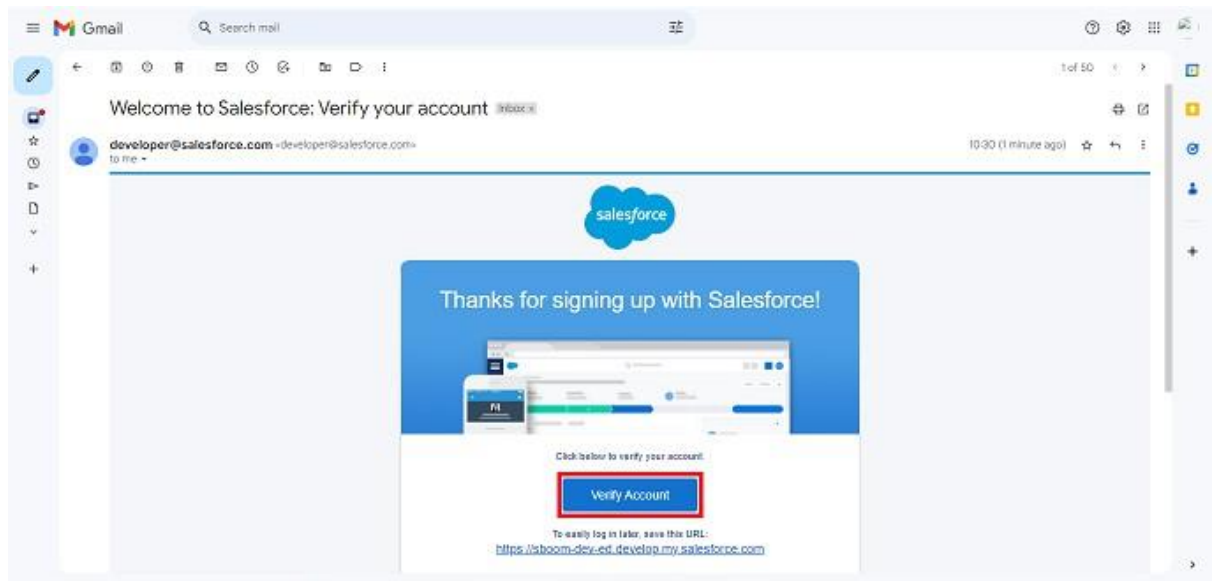
This need not be an actual email id, you can give anything in the format:

username@organization.com

Click on sign me up after filling these.

Activity 2: Account Activation:

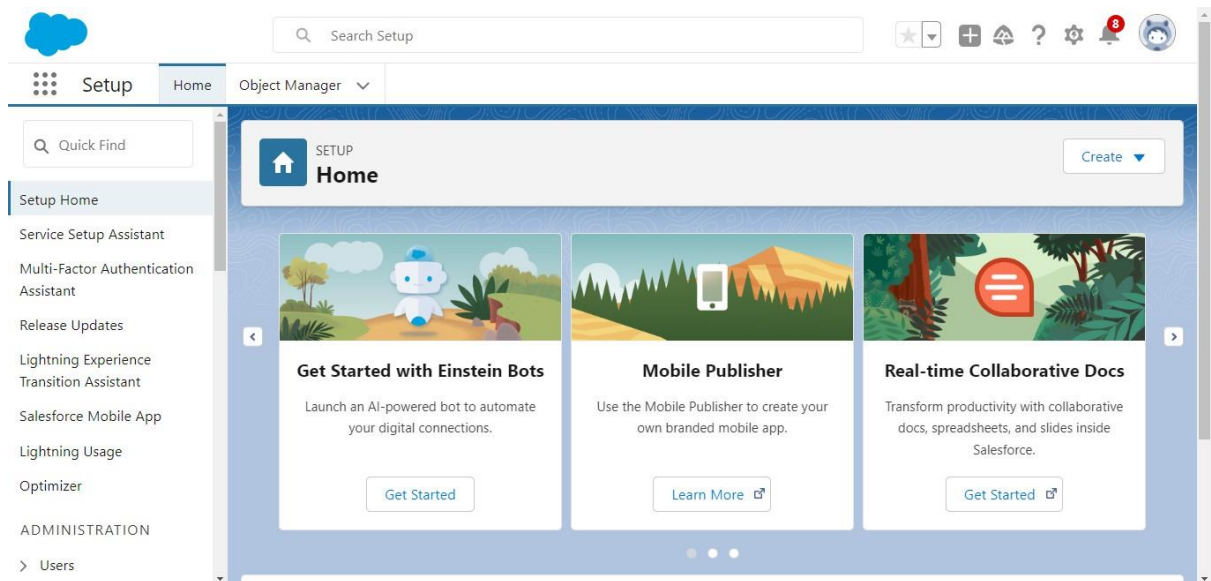
1. Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.



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

A screenshot of the 'Change Your Password' form. The form is titled 'Change Your Password' and asks to 'Enter a new password for lead@sb.com'. It lists requirements: 8 characters, 1 letter, and 1 number. The 'New Password' field shows 'Good' feedback. The 'Confirm New Password' field shows 'Match' feedback. The 'Security Question' dropdown is set to 'In what city were you born?'. The 'Answer' field contains 'asdfghjkl'. A red rectangular box highlights the 'New Password', 'Confirm New Password', 'Security Question', 'Answer', and 'Change Password' button sections.

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



Milestone2-Object

WhatIsanObject?

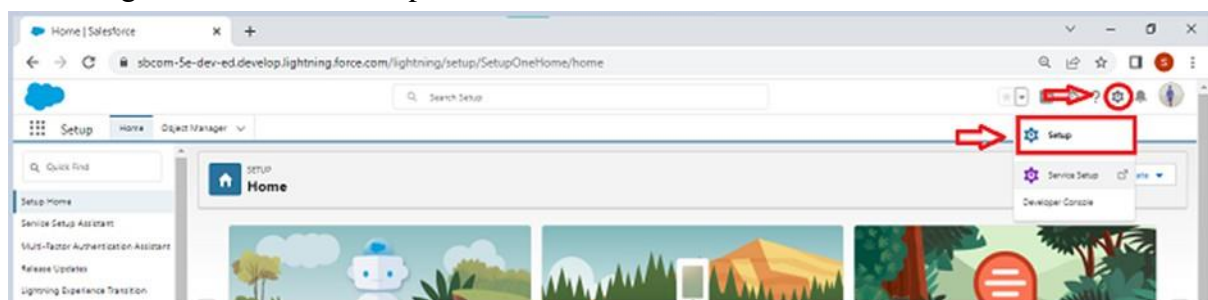
Salesforce objects are database tables that permit you to store data that is specific to an organization. What are the types of Salesforce objects

Salesforceobjectsareoftwotypes:

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 isuniqueandessentialtotheirorganization.Theyaretheheart of any application and provide a structure for sharing data.

ToNavigatetoSetuppage:

Click on gear icon → click setup.

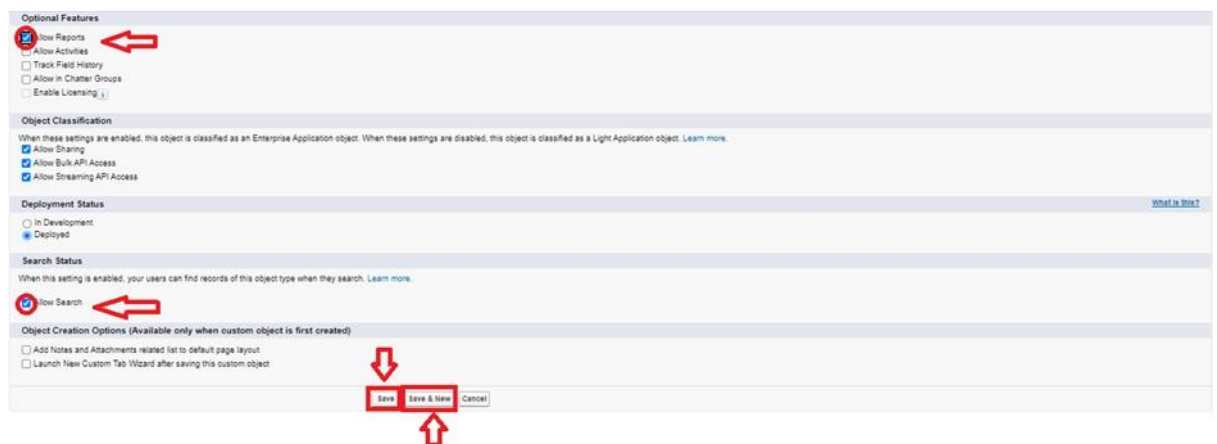
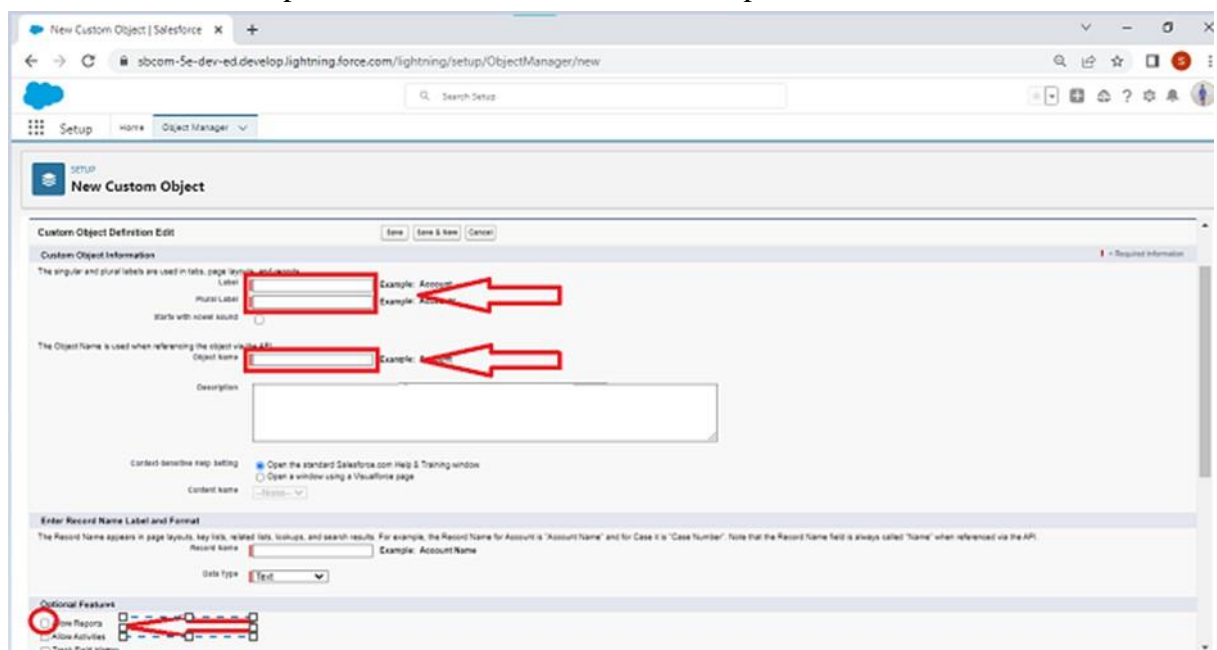


To create an object:

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



2. On Custom object defining page:
3. Enter the label name, plural label name, click on Allow reports, Allow search.



4. Click on Save.

Activity1:CreateCustomerDetailsObject:

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**.

Activity2:CreateAppointmentObject:

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**.

Activity3:CreateService recordsObject:

To create an object:

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

Activity4:CreateBilling details and feedbackObject:

To create an object:

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

- 8) Plural label name → Billing details and feedback
- 9) Enter Record Name Label and Format
 - Record Name → Billing details and feedback Name
 - DataType → AutoNumber
 - Display Format → bill-{000}
 - Starting number → 1
8. Click on Allow reports and Track Field History,
9. Allow search → **Save**.

Milestone3-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.

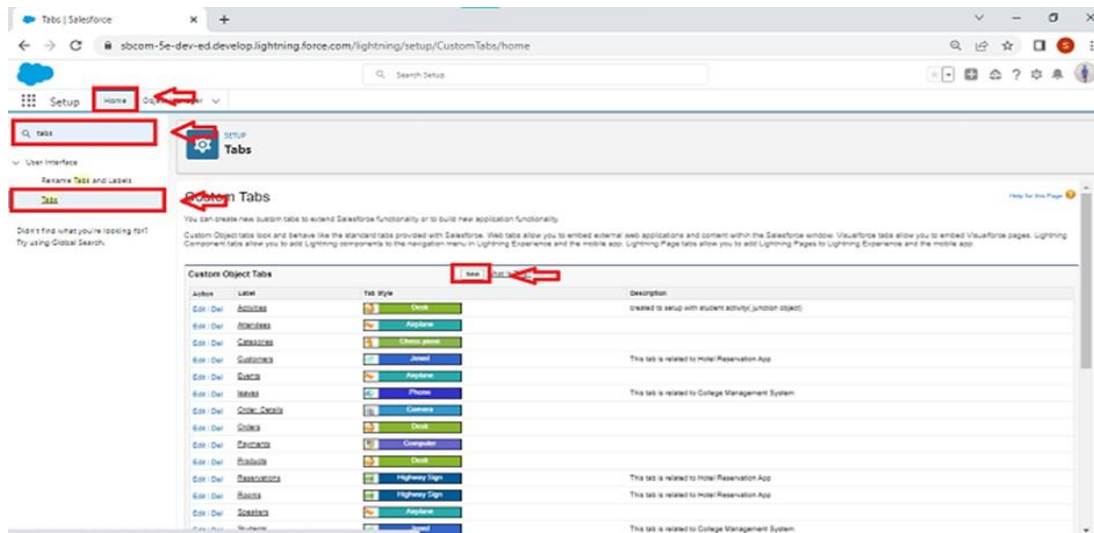
5. Lightning Page Tabs

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.

Activity1: 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

Help for this Page

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:

(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

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

Platform (standard__Platform)

☐

Sales (standard__Sales)

☐

Service (standard__Service)

☐

Marketing (standard__Marketing)

☐

Sample Console (standard__ServiceConsole)

☐

High Volume Customer Portal User

☐

Authenticated Website User

☐

App Launcher (standard__AppLauncher)

☐

Analytics Studio (standard__Insights)

☐

Sales Console (standard__LightningSalesConsole)

☐

Service Console (standard__LightningService)

☐

Sales (standard__LightningSales)

☐

Lightning Usage App (standard__LightningInstrumentation)

☐

Digital Experiences (standard__SalesforceCMS)

☐

Queue Management (standard__QueueManagement)

☐

Bolt Solutions (standard__LightningBolt)

☐

Data Manager (standard__DataManager)

☐

Salesforce Scheduler Setup (standard__LightningScheduler)

☐

☒ Append tab to users' existing personal customizations

Previous

Save

Cancel

Activity2:CreatingRemainingTabs

1. NowcreatetheTabsfortheremainingObjects,theyare“Appointments,Service records,Billing details and feedback”.
2. Follow the same steps as mentioned in Activity -1 .

Milestone4-TheLightningApp:

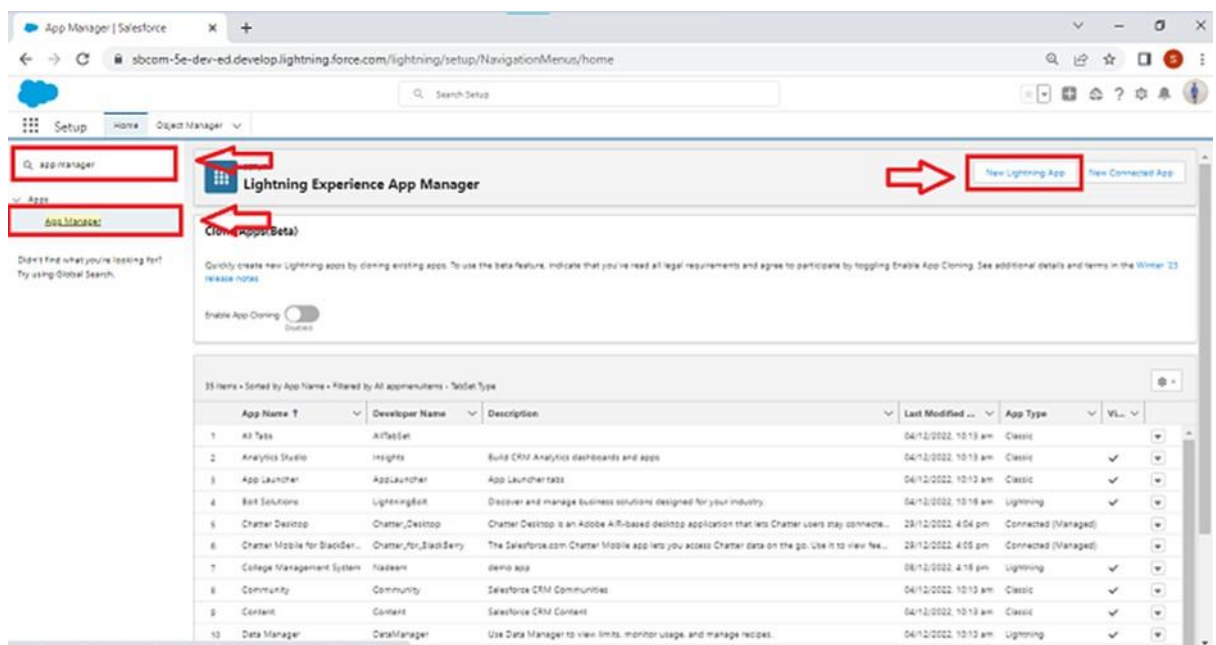
An app is a collection of items that worktogethertoserveaparticularfunction.InLightning Experience, Lightning apps give your users access tosetsofobjects,tabs,andotheritemsall in one convenient bundle in the navigation bar.

Lightning apps let you brand your appswithacustomcolourandlogo.Youcaneveninclude a utility bar and Lightning page tabs in your Lightning app. Members of your org can work more efficiently by easily switching between apps.

Activity1:CreateaLightningApp To

create a lightning app page:

1. Gotosetuppage→search“appmanager”inquickfind→select“appmanager”→ click on New lightning App.



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

3. To Add Navigation Items:

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

4. To Add User Profiles:

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

Milestone 5: Fields

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

Types of Fields

1. Standard Fields
2. Custom Fields

Standard Fields:

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

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

Custom Fields:

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

Activity 1: Creation of fields for the Customer Details object

1. To create fields in an object:

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

Object Manager					
2 items. Sorted by Label					
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

The screenshot shows the Salesforce Object Manager interface for 'Customer1'. The left sidebar contains a navigation menu with 'Fields & Relationships' highlighted. The main content area displays a table of existing fields. At the top right of this section, there are buttons for 'Quick Find', 'New', 'Deleted Fields', 'Field Dependencies', and 'Set History Tracking'. The 'New' button is highlighted with a red box and an arrow.

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	EmailId__c	Email (Unique)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User/Group)		✓
Permanent Address	Permanent_Address__c	Text Area(255)		

3. Select Data Type as a “Phone”

The screenshot shows the 'New Field' dialog in Salesforce Object Manager. The 'Data Type' section is expanded, showing various options. The 'Phone' option is selected and highlighted with a red box and an arrow.

- ☐ 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 spreadsheet.
- ☐ Date: Allows users to enter a date or pick a date from a popup calendar.
- ☐ Date/Time: Allows users to enter a date and time, or pick a date from a popup calendar. When users click a date in the pop-up, that date and the current time are entered into the Date/Time field.
- ☐ Email: Allows users to enter an email address, which is validated to ensure proper format. If this field is specified for a contact or lead, users can choose the address when clicking Send an Email. Note that custom email addresses cannot be used for mass emails.
- ☐ Geolocation: Allows users to define locations. Includes latitude and longitude components, and can be used to calculate distance.
- ☐ Number: Allows users to enter any number. Leading zeros are removed.
- ☐ Percent: Allows users to enter a percentage number, for example, "10" and automatically adds the percent sign to the number.
- ☒ Phone: Allows users to enter any phone number. Automatically formats it as a phone number.
- ☐ Picklist: Allows users to select a value from a list you define.
- ☐ Picklist (Multi-Select): Allows users to select multiple values from a list you define.
- ☐ Text: Allows users to enter any combination of letters and numbers.
- ☐ Text Area: Allows users to enter up to 255 characters on separate lines.
- ☐ Text Area (Long): Allows users to enter up to 131,072 characters on separate lines.
- ☐ Text Area (Rich): Allows users to enter formatted text, add images and links. Up to 131,072 characters on separate lines.
- ☐ Text (Encrypted): Allows users to enter any combination of letters and numbers and store them in encrypted form.
- ☐ Time: Allows users to enter a local time. For example, "2:40 PM", "14:40", and "14:40:50 600" are all valid times for this field.
- ☐ URL: Allows users to enter any valid website address. When users click on the field, the URL will open in a separate browser window.

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.

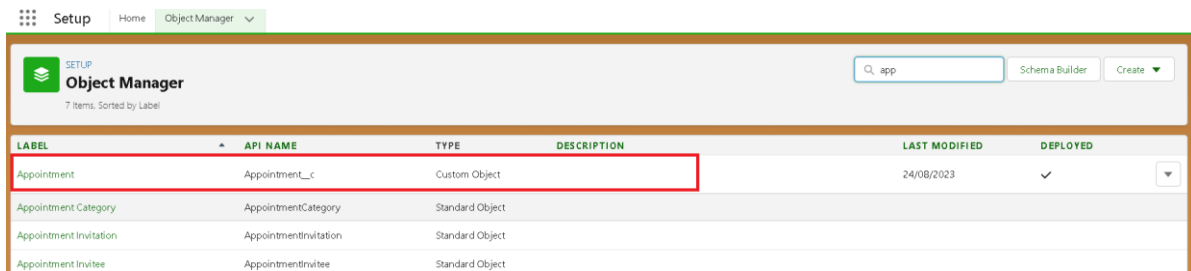
2. To create another field in an object:

1. Go to setup → click on Object Manager → type object name (CustomerDetails) in search bar → click on the object.
2. Now click on “Fields & Relationships” → New
3. Select Data type as a “Email” and Click on Next
4. Fill the Above as following:
 - Field Label : Gmail
 - Field Name : gets auto generated
 - Click on Next → Next → Save and new.

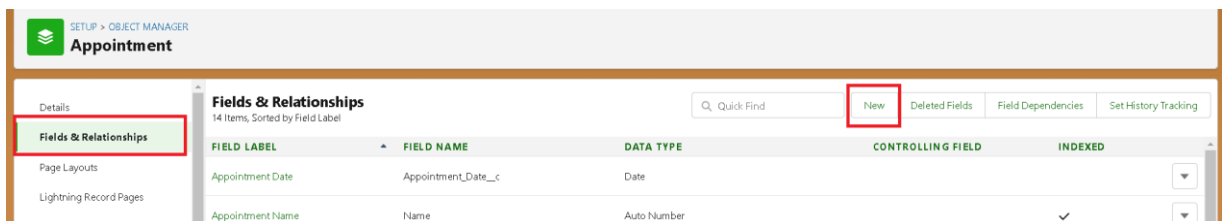
Activity 2: Creation of Lookup Fields:

Creation of Lookup Field on Appointment Object :

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



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



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

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.

☒ **Look-up 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.

Next **Cancel**

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** Child Relationship Name: **Service_records**

Related List Label: **Service records**

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

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.

☒ 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: AppointmentDate → Operator: less than → select field → Appointment: Created Date

9. Filter type should be Required.

10. ErrorMessage: Valuedoesnotmatchthecriteria.

11. Enable the filter by click on Active.

12. Next → Next → Save.

CreationofLookupFieldonBillingdetailsandfeedbackObject:

1. Gotosetup→clickonObjectManager→typeobjectname(**Billingdetailsand 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. Selectthe related object “ **Servicerecords**” andclick next.
5. Next → Next → Save&new.

Activity3:CreationofCheckboxFields:

CreationofCheckbox Fieldon AppointmentObject :

1. Gotosetup→clickonObjectManager→typeobjectname(**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

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

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.
- When a user deletes the master record, all detail records are deleted.
- You can create rollup summary fields on the master record to summarize the detail records.

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.

The relationship field 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

Next Cancel

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

Default Value ☐ Checked ☒ Unchecked

Field Name

Description

Help Text

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

Previous Next Cancel

7. Click on next → next → save.

Creation of Another Checkbox Field on Appointment Object :

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

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

Creation of Checkbox Field on Service records Object :

1. Gotosetup→clickonObjectManager→typeobjectname(Servicerecords)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

Activity4:Creationofdate Fields:

CreationofDateField onAppointmentObject :

1. Gotosetup→clickonObjectManager→typeobjectname(Appointment)inthe search bar
→ click on the object.
2. Now click on “Fields & Relationships” → New.
3. Select “Date” as data type and click Next.
4. Give the Field Label : Appointment Date
5. Field Nme : is auto populated
6. Make it as a Required field by clicking on the Required option.
7. Click on next → next → save.

Appointment
New Custom Field

Step 2 of 4: Enter the details

Field Label: Appointment Date

Field Name: Appointment_Date

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

Activity5:CreationofCurrencyFields:

CreationofCurrencyFieldonAppointmentObject:

1. Gotosetup→clickonObjectManager→typeobjectname(Appointment)inthe 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

Field Label: Service Amount

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: 18

Decimal Places: 0

Field Name: Service_Amount

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

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"/>

8. Click on next → save.

Creation of Currency Field on Billing details and feedback Object:

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

Activity 6: Creation of Text Fields:

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

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

Step 2. Enter the details Step 2 of 4

Previous Next Cancel

Field Label

Please enter the maximum length for a text field below.

Length

Field Name

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 .
8. Click on next → next → save

Activity 7: Creation of Picklist Fields:

Creation of Picklist Fields in Service records object:

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

New Custom Field

Step 2. Enter the details Step 2 of 4

Previous Next Cancel

Field Label

Values

☐ Use global picklist value set

☒ Enter values, with each value separated by a new line

☐ 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

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.

Activity 8: Creating Formula Field in Service records Object

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

Step 2. Choose output type Step 2 of 5

Previous Next Cancel

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

☐ Currency

☒ Date

☐ Date/Time

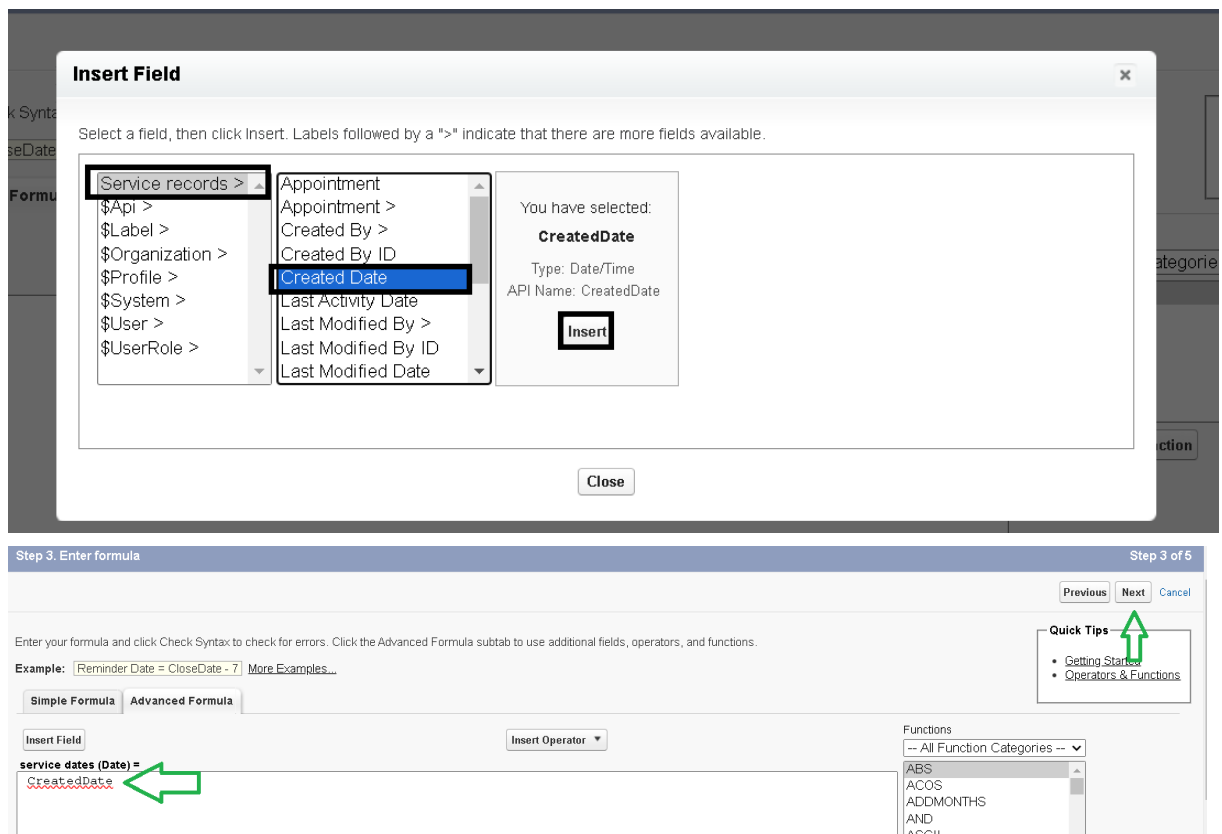
Calculate a boolean value
Example: `TODAY() > CloseDate`

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

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

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

5. Insert field formula should be : CreatedDate



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

Milestone6:Validationrule

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.

Activity1:TocreateavalidationruletoanAppointmentObject

1. Gotothesetuppage→clickonobjectmanager→Fromdropdownclickeditfor **Appointment** object.
2. Click on the validation rule → click New.

SETUP > OBJECT MANAGER

Appointment

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

Flow Triggers

Validation Rules

Validation Rules
1 Items, Sorted by Rule Name

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

New

3. Enter the Rule name as "Vehicle".

4. Insert the Error Condition Formula as :-

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

Validation Rule Edit Save Save & New Cancel

Rule Name:

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

Insert Field Insert Operator

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

Check Syntax

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](#)

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:

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

Save Save & New Cancel

Activity2: To create a validation rule to an Billing details and feedback Object

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



Validation Rule Edit

Save Save & New Cancel

Rule Name **rating_should_be_less_than_5**

Active ☒

Description

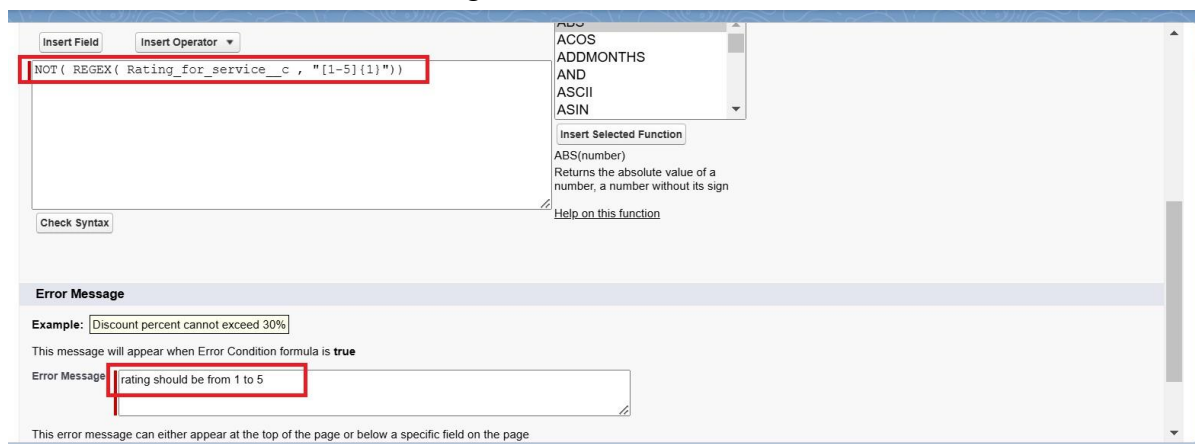
Error Condition Formula

Quick Tips

- Operators & Functions

! = Required Information

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



Insert Field Insert Operator

NOT(REGEX(Rating_for_service_c , "[1-5]{1}"))

Check Syntax

Insert Selected Function

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

Error Message

Example: Discount percent cannot exceed 30%

This message will appear when Error Condition formula is true

Error Message **rating should be from 1 to 5**

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

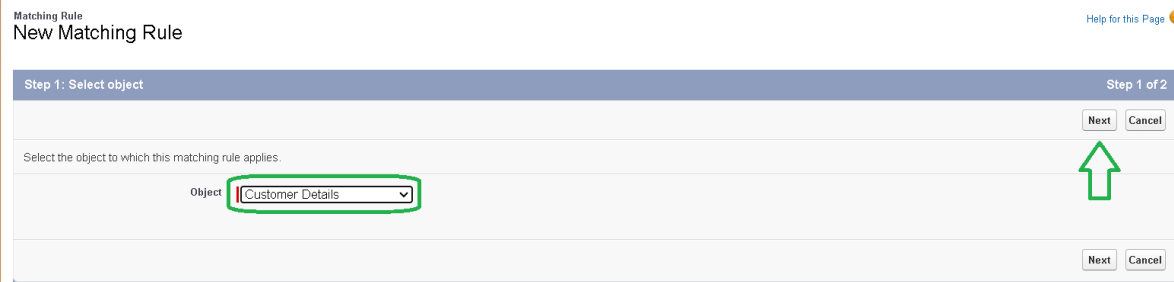
Milestone7: Duplicate rule

Activity1: 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

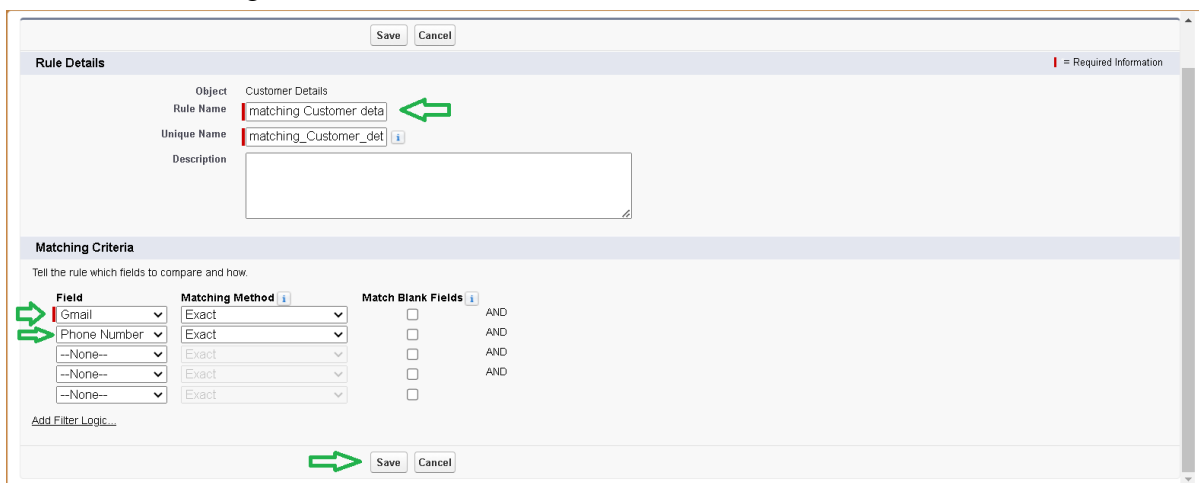
Exact

2. Phone Number

Exact

8. Click save.

9. After Saving Click on Activate.



Activity2: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 Salesforce Duplicate Rules setup interface. On the left, the navigation menu includes 'Data', 'Duplicate Management', 'Duplicate Error Logs', 'Duplicate Rules' (highlighted), and 'Matching Rules'. The main area is titled 'All Duplicate Rules' and includes a 'What Are Duplicate Rules?' section. Below this is a table of existing rules. A 'New Rule' dropdown menu is open, showing a list of objects: Account, Appointment, Billing details and feedback, Contact, Customer Details (highlighted), Environment, Individual, Laptop, and Lead. The table of existing rules has columns: Rule Name, Description, Matching Rule, Active, Last Modified By, and Last Modified Date.

Rule Name	Description	Matching Rule	Active	Last Modified By	Last Modified Date
Customer Detail duplicate		Matching Customer details	<input type="checkbox"/>	p2	10/10/2023
Standard Account Duplicate Rule	Identify accounts that duplicate other accounts.	Standard Account Matching Rule	<input checked="" type="checkbox"/>	p2	24/08/2023
Standard Contact Duplicate Rule	Identify contacts that duplicate other contacts and leads.	Standard Lead Matching Rule	<input checked="" type="checkbox"/>	p2	24/08/2023
Standard Lead Duplicate Rule	Identify leads that duplicate other leads and contacts.	Standard Contact Matching Rule	<input checked="" type="checkbox"/>	p2	24/08/2023

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' page for 'Customer Detail duplicate'. The 'Rule Details' section includes fields for 'Rule Name' (Customer Detail duplicate), 'Description', 'Object' (Customer Details), and 'Record-Level Security' (Enforce sharing rules). The 'Actions' section includes 'Action On Create' (Allow, Alert, Report) and 'Action On Edit' (Allow, Alert, Report). The 'Alert Text' field is also visible.

The screenshot shows the 'Matching Rules' section. It includes a 'Compare Customer Details With' dropdown set to 'Customer Details' and a 'Matching Rule' dropdown set to 'matching Customer details'. The 'Matching Criteria' section shows the criteria for matching records: (Customer Details: Email EXACT MatchBlank = FALSE) AND (Customer Details: Phone_Number EXACT MatchBlank = FALSE). The 'Field Mapping' section shows 'Mapping Selected'.

Milestone8: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:

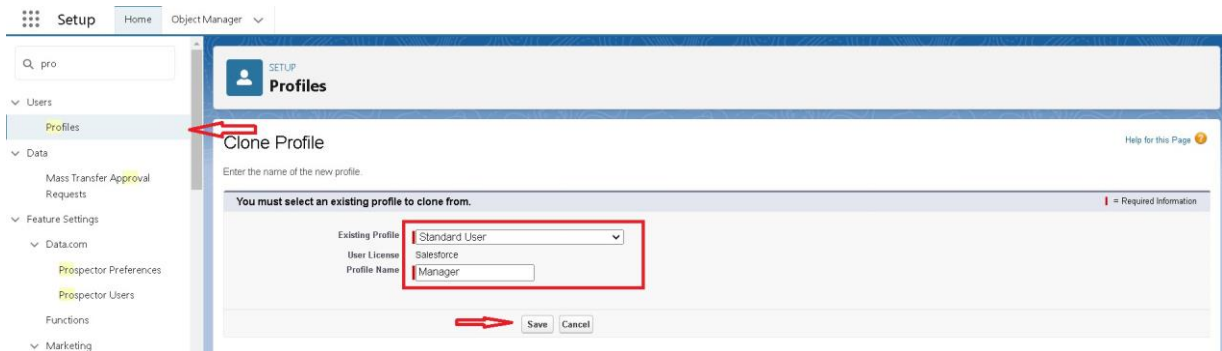
Custom ones defined by us.

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

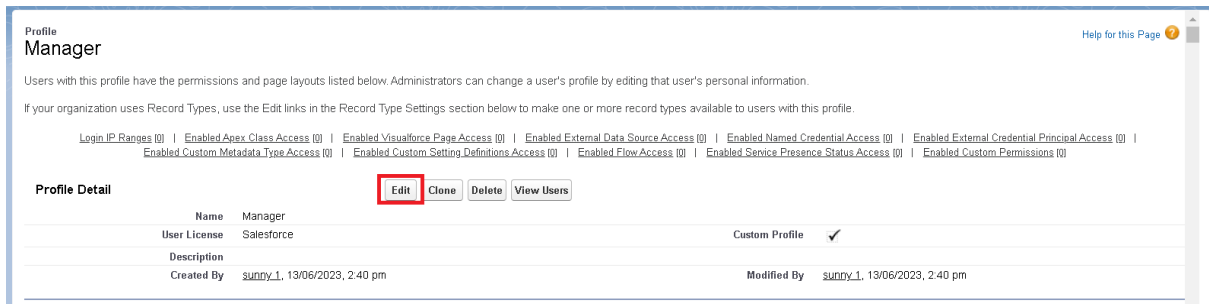
Activity1:ManagerProfile

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.

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"/>

	Basic Access				Data Administration	
	Read	Create	Edit	Delete	View All	Modify All
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 “ 8 ”, and click save.

Activity2:salespersonProfile

1. Gotosetup→typeprofilesinquickfindbox→clickonprofiles→clonethedesired 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,Billingdetailsandfeedback,servicerecordsandcustomerdetailsobjects 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"/>

	Basic Access				Data Administration	
	Read	Create	Edit	Delete	View All	Modify All
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.

Milestone9:Role&RoleHierarchy

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.

Activity1:CreatingManagerRole

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, the 'Roles' link is highlighted. The main content area is titled 'Understanding Roles' and contains a diagram of a sample role hierarchy. The hierarchy starts with 'Executive Staff' (CEO, President, CFO, VP, Sales) at the top. Below this are three branches: 'Western Sales Director', 'Eastern Sales Director', and 'International Sales Director'. Each branch further divides into regional sales reps (e.g., CA Sales Rep, NY Sales Rep, etc.). A red box highlights the 'Set Up Roles' button at the bottom right of the main content area.

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. At the top, there are 'Collapse All' and 'Expand All' buttons. Below this is a list of roles: 'Nick Enterprises', 'CEO', 'HR', 'Manager', 'On Site Emp', and 'Remote Emp'. Each role has an expand/collapse icon and an 'Add Role' button. A red box highlights the 'Expand All' button at the top left. Another red box highlights the 'Add Role' button next to the 'Manager' role.

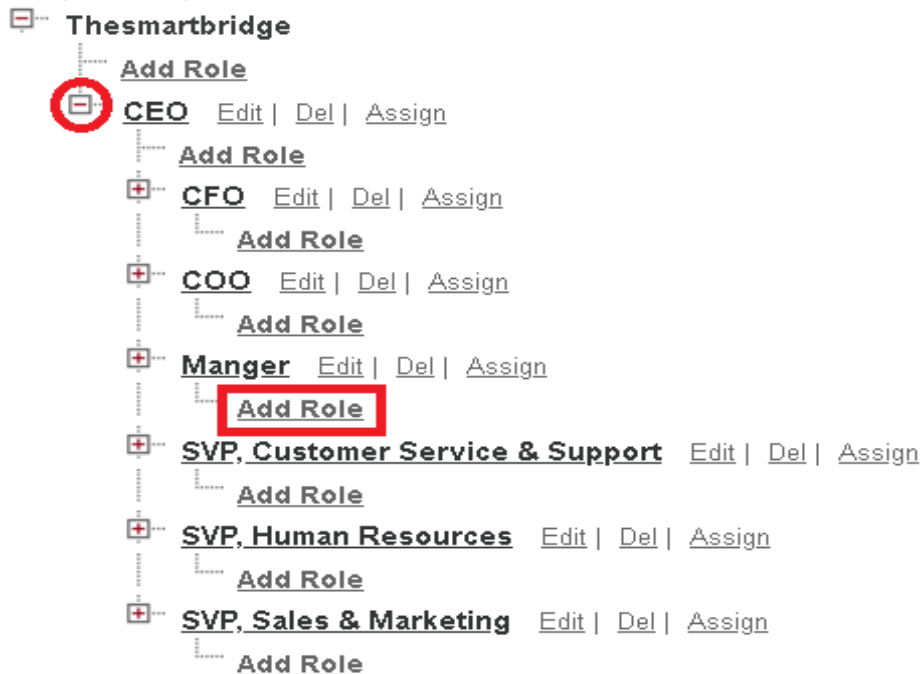
3. Give Label as “Manager” and Role name gets auto populated. Then click on Save.

Activity2:Creatinganotherroles

Creating another two roles under manager

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

[Collapse All](#) [Expand All](#)



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

Milestone10: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.

Activity1:CreateUser

1. Goto 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. Userlicence: Salesforce
 9. Profiles : Manager

New User

Help for this Page

User Edit

Save Save & New Cancel

General Information

First Name Niklaus

Last Name Mikaelson

Alias nmika

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.

Activity2:creatinganotherusers

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

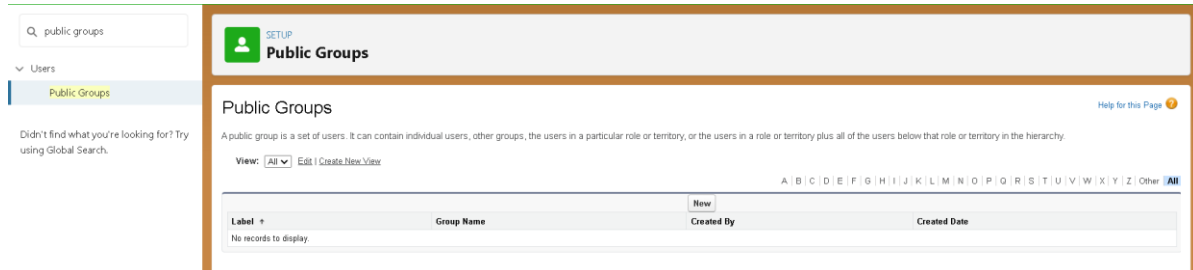
Note:create atleast3users withthese permissions.

Milestone11:Publicgroups

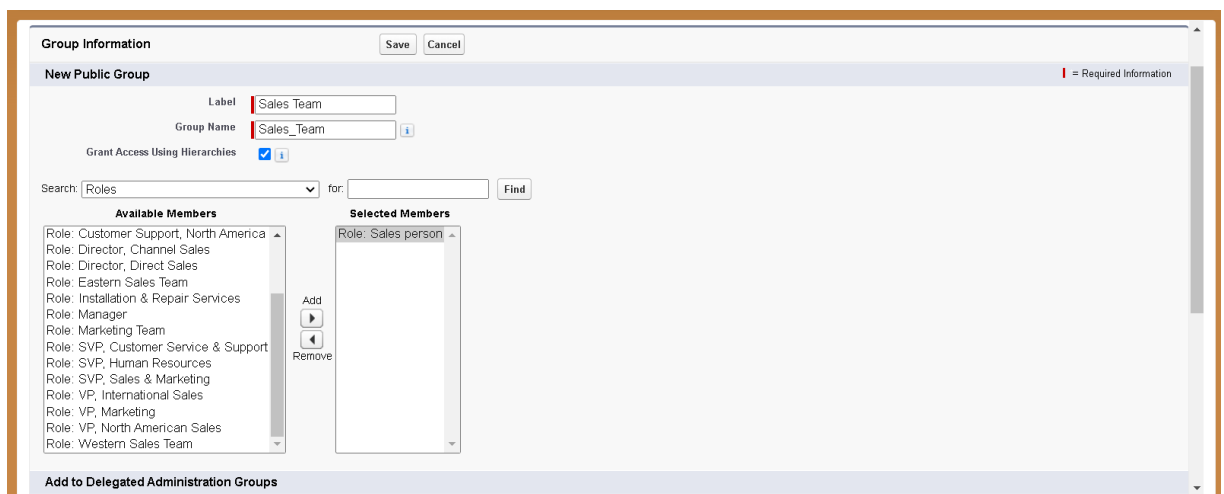
Public groups are a valuable tool for Salesforce administrators and developers to streamline user management, data access, and security settings. By creating and using public groups effectively,youcanmaintainasecureandorganizedSalesforceenvironmentwhileensuring that users have appropriate access to the resources they need.

Activity1:CreatingNewPublicGroup

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.



Milestone12:SharingSetting

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

Organization-Wide Default (OWD) Settings:

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

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

Role Hierarchy:

Salesforce uses a role hierarchy to determine record access.

Users at higher levels in the hierarchy have greater access to records owned by or shared with users lower in the hierarchy.

The role hierarchy is often used in combination with OWD settings to grant different levels of access.

Profiles and Permission Sets:

Profiles and permission sets allow administrators to specify object-level and field-level permissions for users.

Profiles are typically used to grant general object and field access, while permission sets can be used to extend those permissions to specific users.

Sharing Rules:

Sharing rules are used to extend access to records for users who meet specific criteria.

They can be used to grant read-only or read-write access to records owned by other users. Manual Sharing:

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

Activity 1: Creating Sharing settings

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

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 cases ☒ Secure guest user record access ☒ Require permission to view record names in lookup fields ☐

Save **Cancel**

3. Clickonsaveandrefresh.
4. Scrolldownabit,ClicknewonServicerecordssharingRules. 5.

Service records Sharing Rules **New** **Recalculate**

No sharing rules specified.

6. GivetheLabelnameas“Sharingsetting”
7. Rulenameisautopopulated.
8. Instep3:Selectwhichrecordstobeshared,memberof“Roles”>>“Sales person”
9. Instep4:sharewith,select“Roles”>>“Manager”
10. Instep5:Changetheaccesslevelto“Read/write”.
11. Clickonsave.

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
Description:

Step 2: Select your rule type

Rule Type: ☒ Based on record owner ☐ Based on criteria

Step 3: Select which records to be shared

Service records: owned by members of Roles Sales person

Step 4: Select the users to share with

Share with: Roles Manager

Step 5: Select the level of access for the users

Access Level: Read/Write

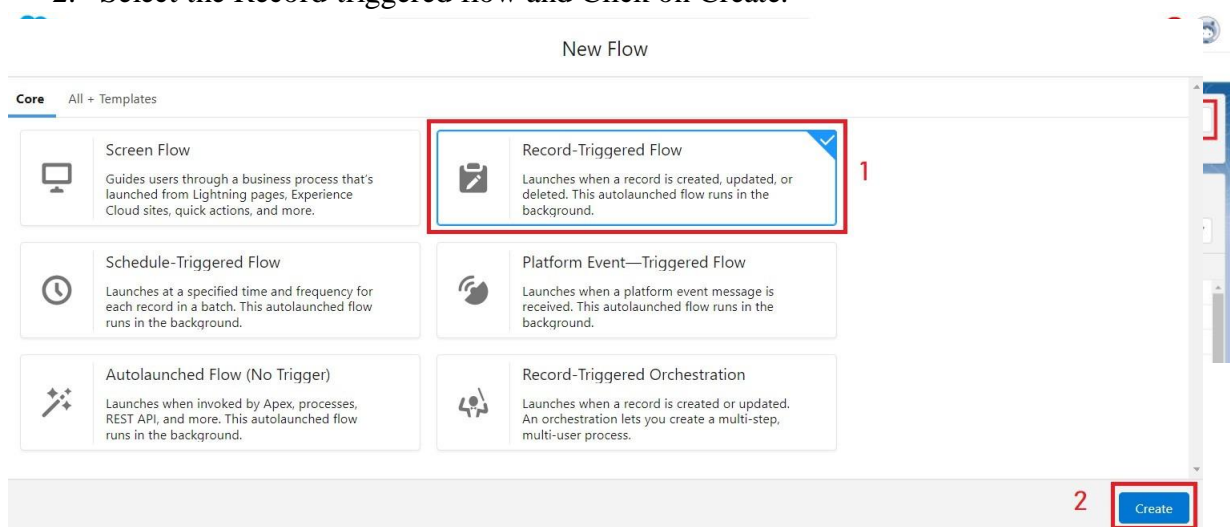
Save **Cancel**

Milestone13: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.

Activity1:CreateaFlow

1. Go to setup → type Flow in quick find box → Click on the Flow and Select the New Flow.
2. Select the Record-triggered flow and Click on Create.



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

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

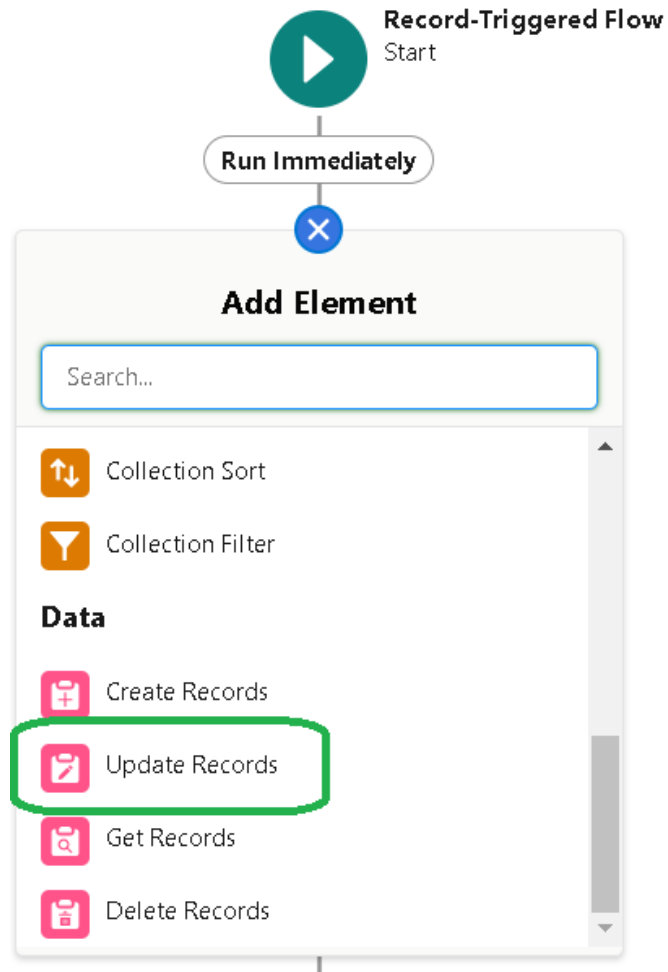
Update any record and perform actions, like send an email. This more flexible flow runs *after* the record is saved to the database.

☐ 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 DropDownList 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**

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	<div style="border: 1px solid #ccc; width: 20px; height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;"> <div style="width: 10px; height: 10px; background-color: #000;"></div> </div>

+ Add Condition

Set Field Values for the Billing details and feedback Record

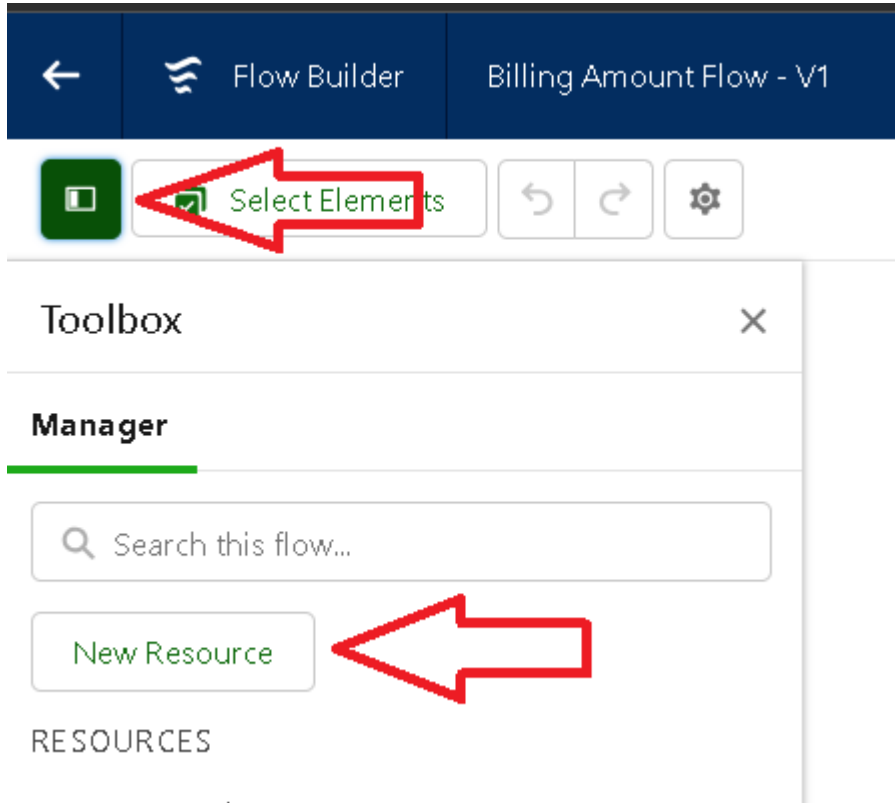
Field		Value	
Payment_Paid__c	←	<div style="display: flex; align-items: center;"> <div style="width: 20px; height: 20px; background-color: #000; margin-right: 5px;"></div> \$Record > Service records > Appointment > Service A... </div>	<div style="border: 1px solid #ccc; width: 20px; height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;"> <div style="width: 10px; height: 10px; background-color: #000;"></div> </div>

+ Add Field

Cancel
Done

9. Set a filter condition : All Conditions are met(AND)
10. Field : Payment_Statusc
11. Operator : Equals
12. Value:Completed
13. AndSetFieldValuesfortheBillingdetailsandfeedbackRecord
14. Field: Payment_Paidc
15. Value:{ !\$Record.Service_recordsr.Appointmentr.Service_Amountc }
16. Click On Done.

17. Before creating another element, create a new resource from the toolbox for the 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 RichText → View to Plain Text.

22. In body field paste the syntax that is given below. Dear
{!\$Record.Service_recordsr.Appointmenttr.Customer_Namer.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 your 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_Paidc}
Thank you for Coming .

23. Click done.

Edit Text Template

*API Name
alert

Description

*Body ⓘ

Insert a resource... View as Plain Text ▼

Dear {\$Record.Service_records_r.Appointment_r.Customer_Name_r.Name},

Cancel 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_recordsr.Appointmentr.Customer_Namer.Gmailc }
32. Includesubjectas“ThankYouforYourPayment-GarageManagement”.
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

A_a Body ⓘ

{!alert}

✓

Include

A_a Email Template ID

Don't Include

🔗

 Log Email on Send

Don't Include

Edit Action

A_a Recipient Address List ⓘ

{!\$Record.Service_records__r.Appointment__r.Cus

✓

Include

A_a Recipient ID

Don't Include

A_a Related Record ID

Don't Include

🔗

 Rich-Text-Formatted Body

Don't Include

A_a Sender Email Address

Don't Include

A_a Sender Type

Don't Include

A_a Subject ⓘ

Thank You for Your Payment - Garage Manageme

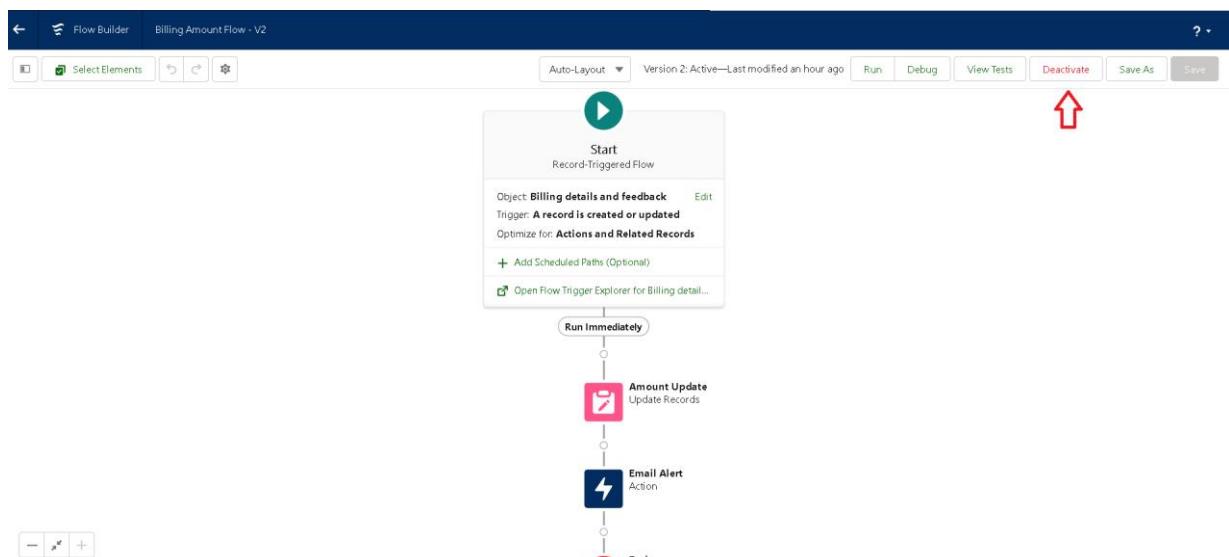
✓

Include

Cancel

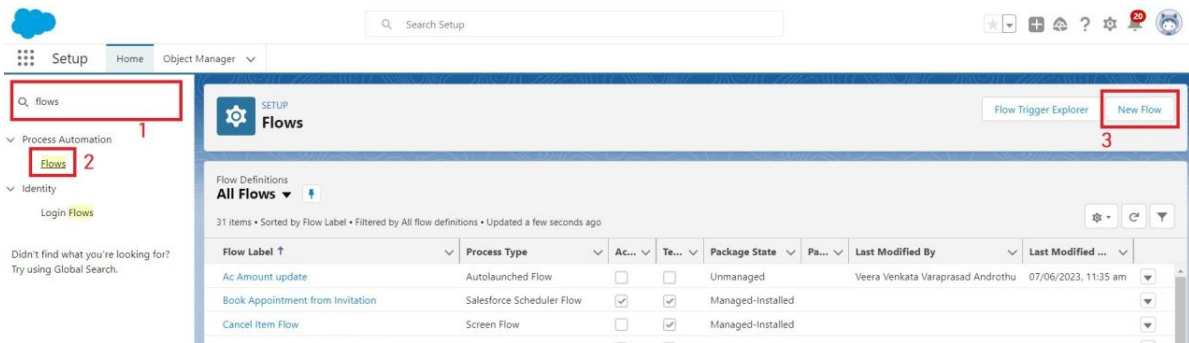
Done

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

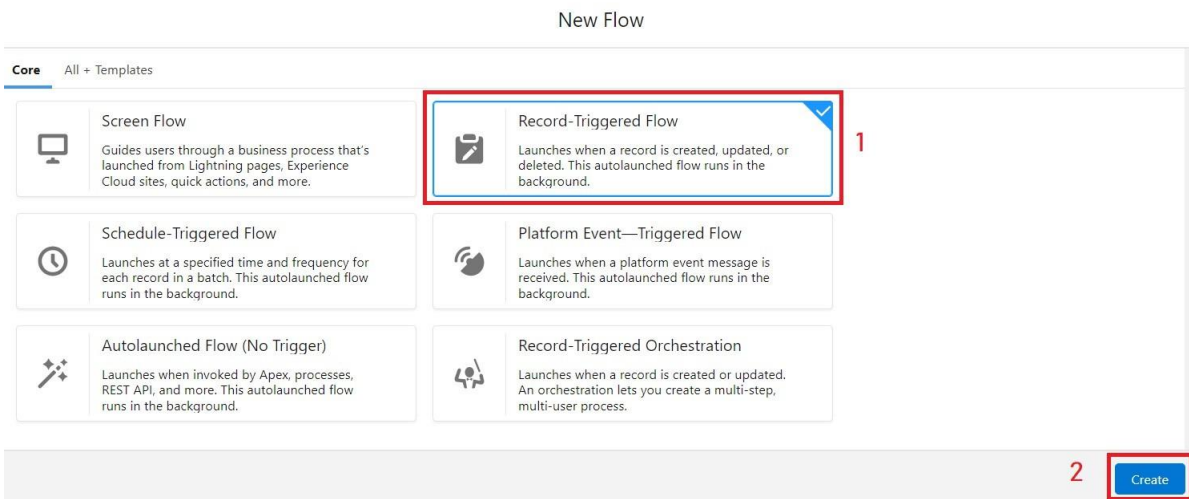


Activity2:CreateanotherFlow

36. Go to setup → type Flow in quick find box →ClickontheFlowandSelecttheNew Flow.



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



38. Select the Object as “Service records” in the Drop down list.

39. Select the Trigger Flow when: “A record is created or updated”.

40. Select the Optimise the flow for: “Actions and Related Records” and Click on Done.

41. Under the Record-triggered Flow Click on “+” Symbol and In the Dropdown List select the “Update records Element”.

42. Set a filter condition : All Conditions are met(AND)

43. Field : Quality_Check_Statusc

44. Operator : Equals

45. Value: True

46. And Set Field Values for the Billing details and feedback Record

47. Field: Service_Statusc

48. Value: Completed

Set Filter Conditions

Condition Requirements to Update Record

All Conditions Are Met (AND) ▼

Field	Operator	Value
Quality_Check_Status__c	Equals ▼	True X

+ Add Condition

Set Field Values for the Service record Record

Field	Value
Service_Status__c	Completed

+ Add Field

49. Click On Done.

50. Click on save

51. Given the Flow label as **UpdateServiceStatus**, Flow Api name will be auto populated.

52. And click save, and click on activate.

Milestone14:ApexTrigger

Apex can be invoked by using triggers. Apex triggers enable you to perform custom actions before or after changes to Salesforce records, such as insertions, updates, or deletions.

A trigger is Apex code that executes before or after the following types of operations:

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

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

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

There are primarily two types of Apex Triggers:

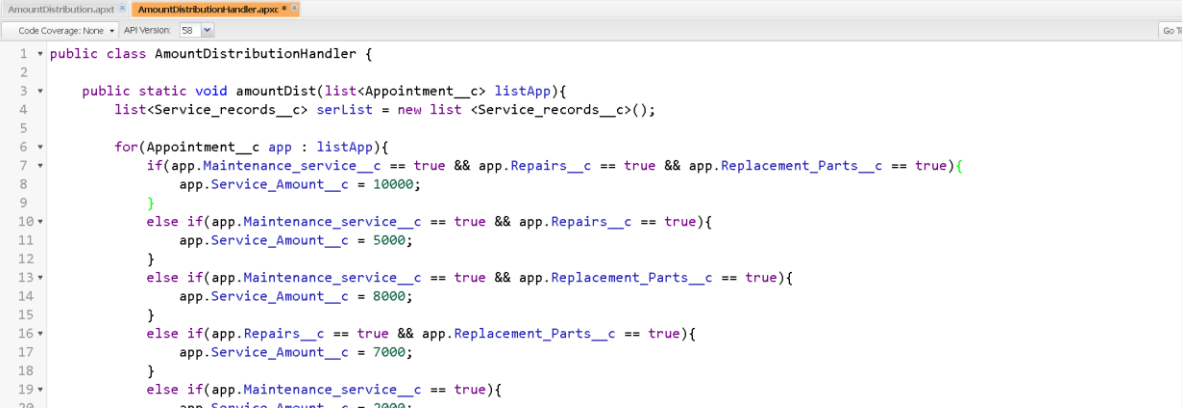
Before Trigger: This type of trigger in Salesforce is used either to update or validate the values of a record before they can be saved into the database. So, basically, the before trigger validates the record first and then saves it. Some criteria or code can be set to check data before it gets ready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

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

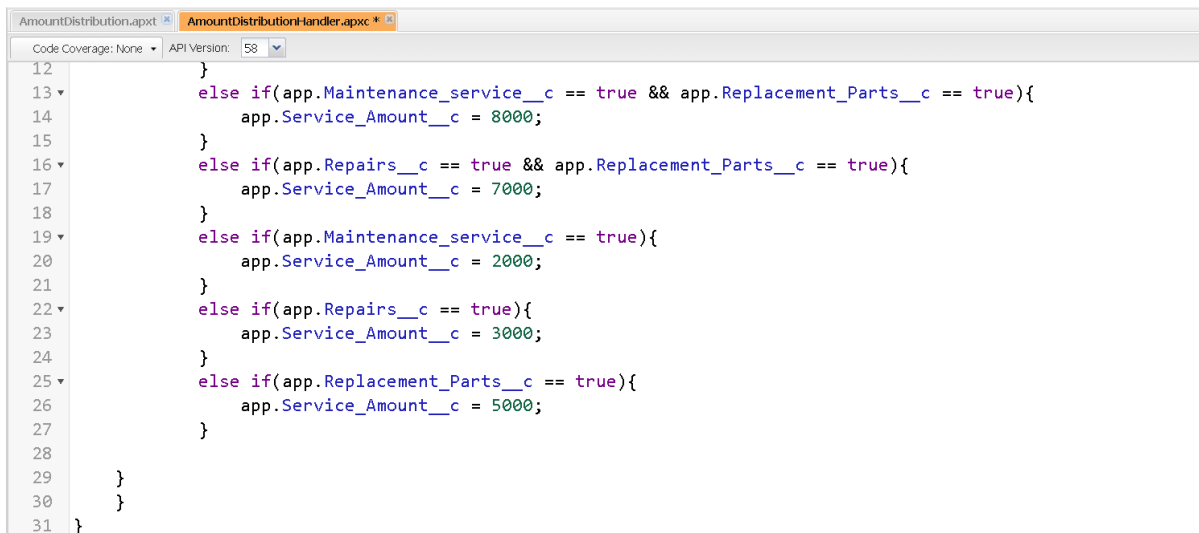
Activity-1: Apex handler

Use Case: This use case works for Amount Distribution for each Service the customer selected for their Vehicle.

- 1) Log in 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".



```
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;
            }
            elseif(app.Maintenance_service__c == true && app.Repairs__c ==
true) {
                app.Service_Amount__c = 5000;
            }
            elseif(app.Maintenance_service__c == true &&
app.Replacement_Parts__c == true) {
                app.Service_Amount__c = 8000;
            }
            elseif(app.Repairs__c == true && app.Replacement_Parts__c == true) {
```

```

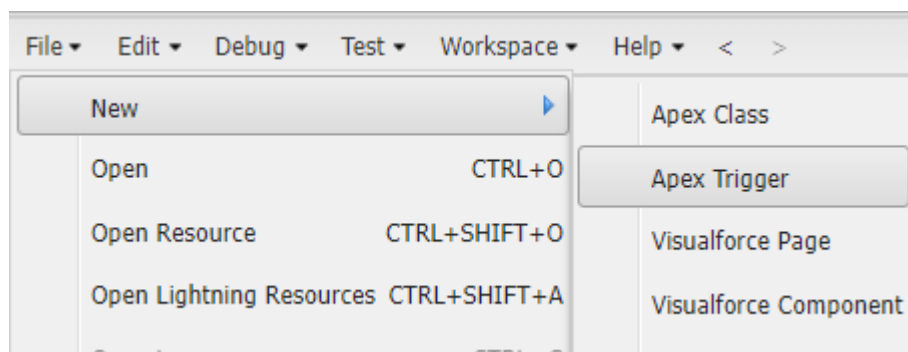
        app.Service_Amount__c=7000;
    }
    else if(app.Maintenance_service__c==true){
        app.Service_Amount__c = 2000;
    }
    else if(app.Repairs__c == true){
        app.Service_Amount__c=3000;
    }
    else if(app.Replacement_Parts__c==true){
        app.Service_Amount__c = 5000;
    }
}
}
}

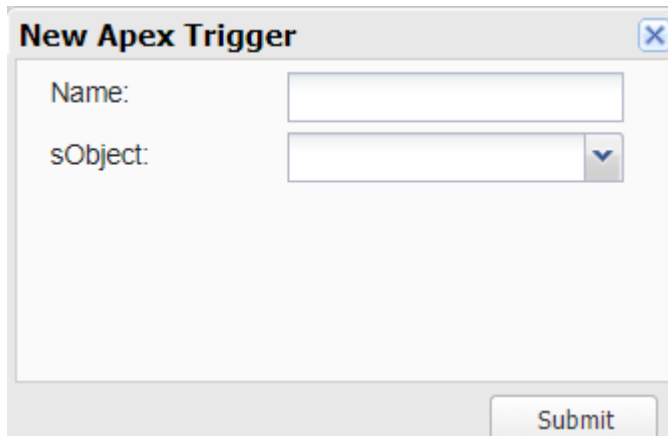
```

TriggerHandler:

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 toolbar, and click on new → Trigger.
- 4) Enter the trigger name and the object to be triggered.
- 5) Name: AmountDistribution
- 6) sObject : Appointmentc





The image shows a 'New Apex Trigger' dialog box. It has a title bar with a close button. Inside, there are two fields: 'Name:' with a text input box, and 'sObject:' with a dropdown menu. At the bottom right, there is a 'Submit' button.

Syntax For creating trigger:

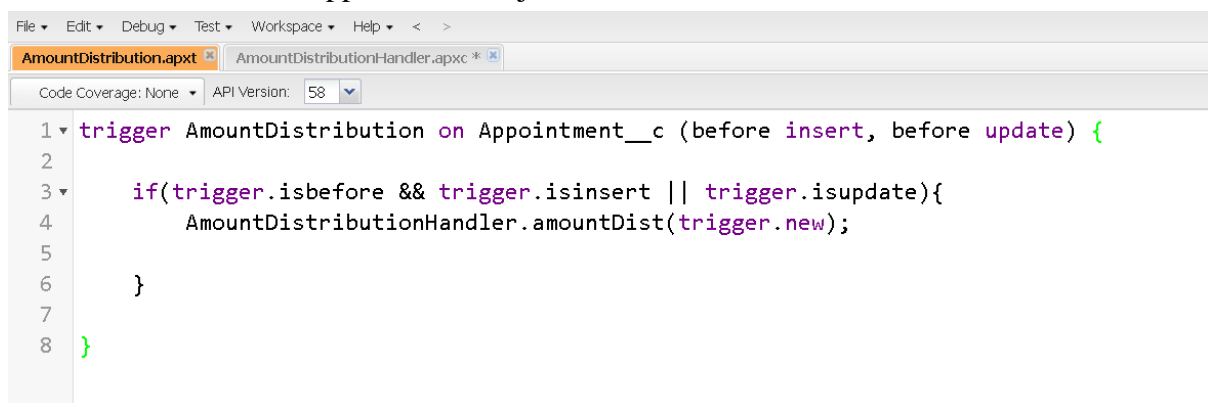
The syntax for creating trigger is :

```
Trigger[triggername]on[objectname](Before/Afterevent)
{

}
```

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

1. Handler for the Appointment Object



The image is a screenshot of an IDE window. The title bar shows 'File', 'Edit', 'Debug', 'Test', 'Workspace', and 'Help'. Below the title bar, there are two tabs: 'AmountDistribution.apxt' (active) and 'AmountDistributionHandler.apxc *'. Below the tabs, there is a status bar showing 'Code Coverage: None' and 'API Version: 58'. The main area displays 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 Appointmentc (before insert, before update) {

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

    }

}

```

Milestone-15:UserAdoption Activity

1 : creating record

To create a record in the following objects follow these steps

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

The screenshot shows a mobile application interface for creating a new customer record. The form is titled "New Customer Detail". It includes a legend indicating that an asterisk (*) denotes required information. The form is divided into two main sections: "Information" and "Owner". The "Information" section contains three input fields: "Customer Name" (with the value "Mac"), "Phone number" (with the value "5678765567"), and "Gmail" (with the value "mac@gmail.com"). The "Owner" section displays a user icon and the name "Annapurna SmartBridge". At the bottom of the form, there are three buttons: "Cancel", "Save & New", and "Save".

Now, Create the Appointment Record

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

Garage Manageme... Customer Details ▾ Appointments ▾ Service records ▾ Billing details and feedback ▾ Reports ▾ Dashboards ▾

Appointment
app-016

Appointment Name
app-016

Owner
Annapurna SmartBridge

Customer Details

Mac

* Appointment Date
13/11/2024

Maintenance service
☒

Repairs
☒

Replacement Parts
☐

Service Amount

* Vehicle number plate
TS30EU0443

Created By
Annapurna SmartBridge, 18/11/2024, 3:28 pm

Cancel Save

Last Modified By
Annapurna SmartBridge, 18/11/2024, 3:28 pm

Now, Create a service record

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

New Service record

* = Required Information

Information

Service Record Name

Owner
Annapurna SmartBridge

* Appointment
app-016


Quality Check Status
☐

Service Status
Started


Cancel Save & New Save

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

Service Record Name
ser-010

Owner
 Annapurna SmartBridge

* Appointment


 app-016

Quality Check Status
☒


Service Status

Started

service date
18/11/2024
This field is calculated upon save

Created By
 Annapurna SmartBridge, 18/11/2024, 4:32 pm

Cancel Save

Created By
 Annapurna SmartBridge, 18/11/2024, 4:34 pm

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

Related Details


Service Record Name
ser-010


Appointment
[app-016](#)


Quality Check Status
☒

Service Status
Completed

service date
18/11/2024

Created By
 Annapurna SmartBridge, 18/11/2024, 4:32 pm

Owner
 Annapurna SmartBridge

Last Modified By
 Annapurna SmartBridge, 18/11/2024, 4:34 pm

Milestone16: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.

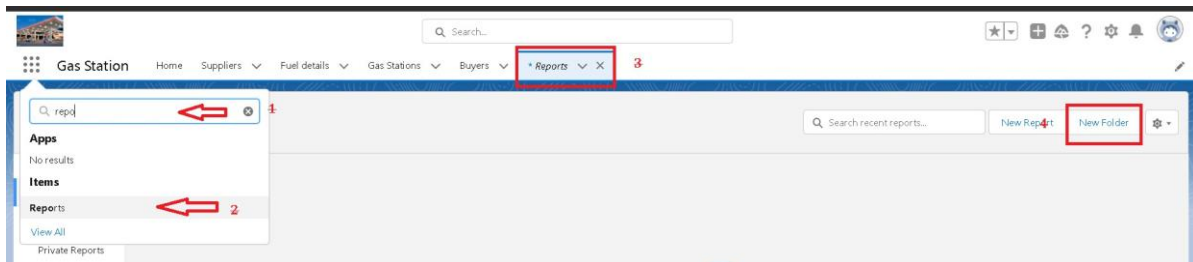
Types of Reports in Salesforce

1. Tabular
2. Summary
3. Matrix

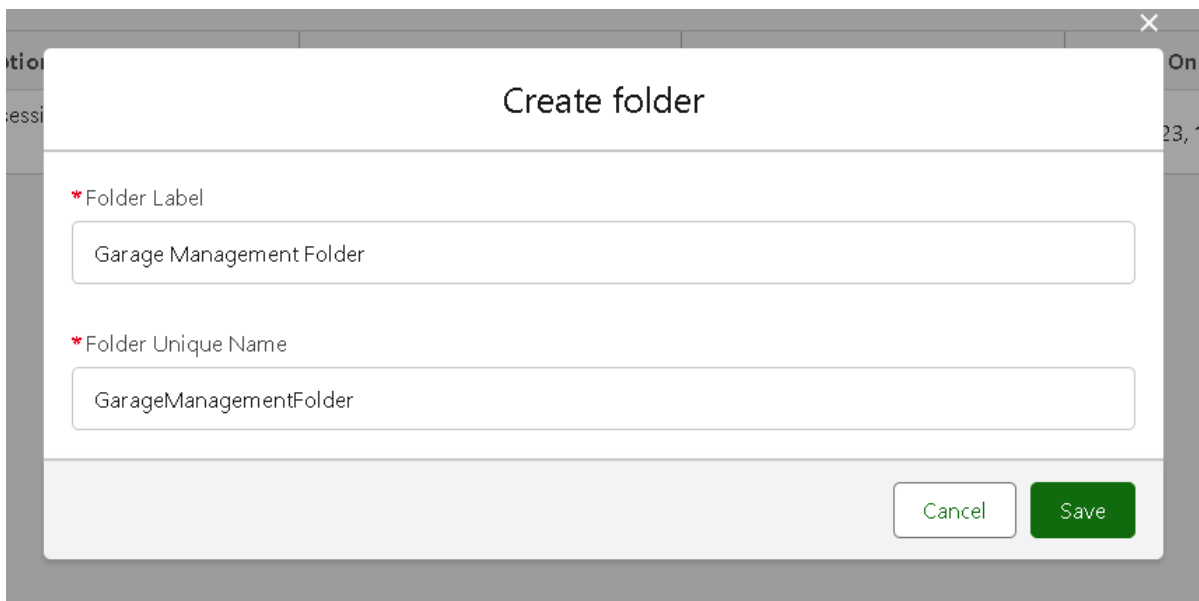
4. Joined Reports

Activity1:createareportfolder

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

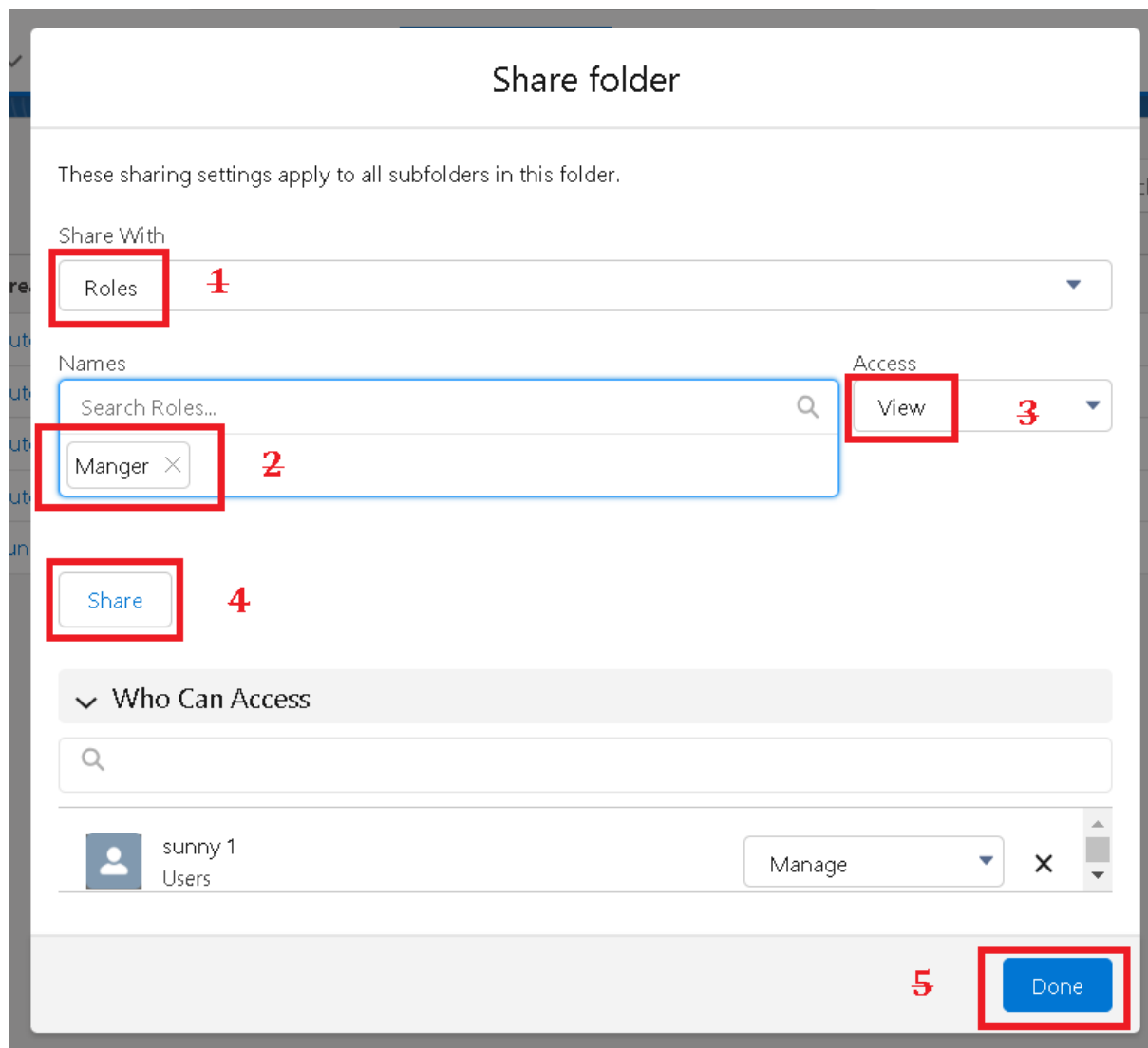


3. Give the Folder label as “Garage Management Folder”, Folder unique name will be auto populated.
4. Click save.



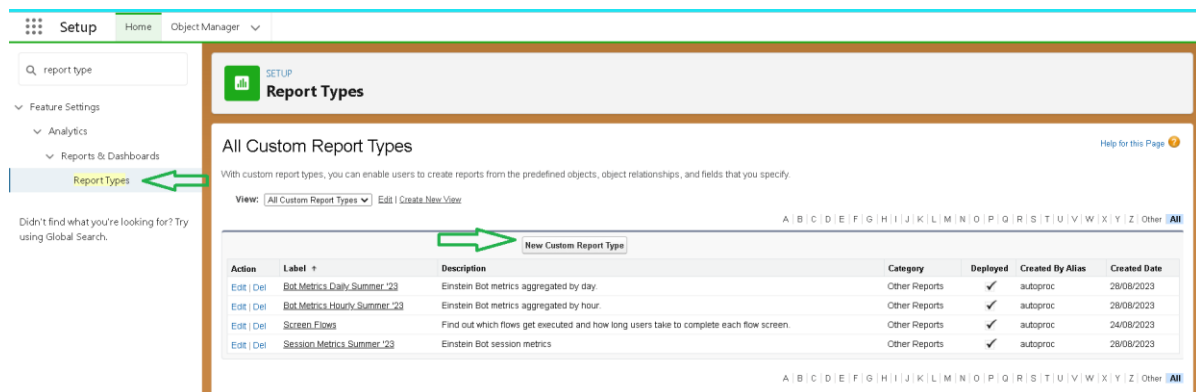
Activity2:Sharingareportfolder

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.



Activity3:CreateReportType

1. Gotosetup→typeusersinquickfindbox→selectReportType→clickonContinue.
2. Click on new custom report type.



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

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

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 **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

A Customer Details

Primary Object

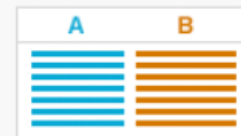
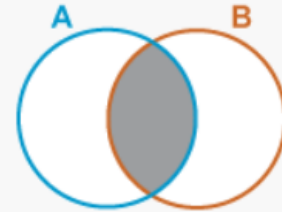
B

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.

(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.



Report Types

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.

C Service records

B to C Relationship:

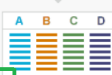
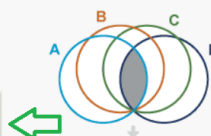
- ☒ Each "B" record must have at least one related "C" record.
☐ "B" records may or may not have related "C" records.

D Billing details and feedback

C to D Relationship:

- ☒ Each "C" record must have at least one related "D" record.
☐ "C" records may or may not have related "D" records.

Object Limit Reached
You can associate up to four objects to a custom report type.

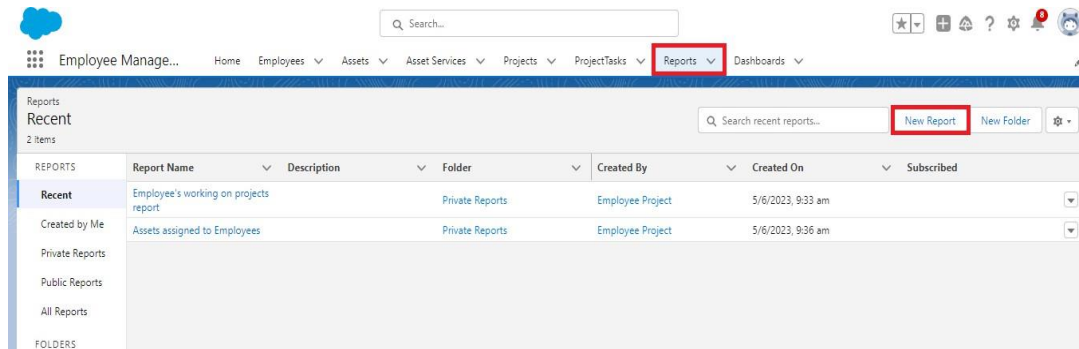


Previous Save Cancel

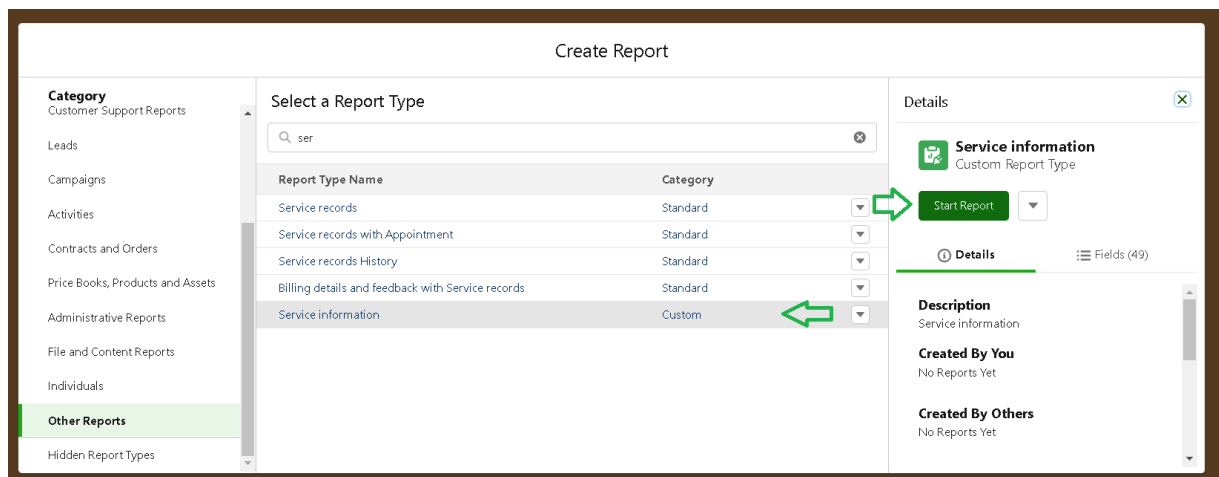
Activity4: Create Report

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

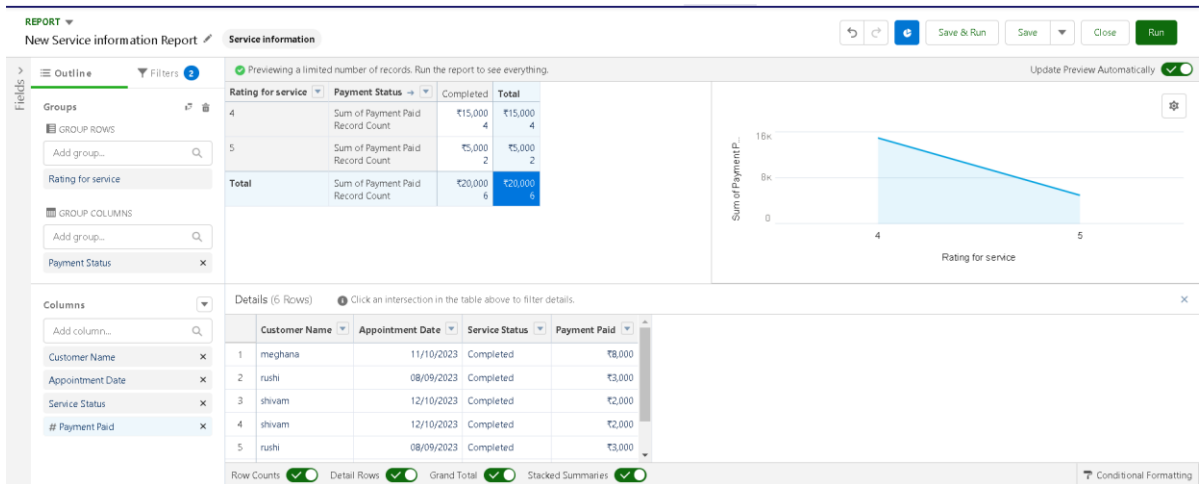
1. Go to the app → click on the reports tab
2. Click New Report.



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



4. The outline pane is opened already, select the fields that are mentioned below in the column section.
 - a. Customer name
 - b. Appointment Date
 - c. Service Status
 - d. Payment paid
5. Remove the unnecessary fields.
6. Select the fields that mentioned below in GROUP ROWS section.
 - a. Rating for Service
7. Select the fields that mentioned below in GROUP ROWS section.
 - a. Payment Status
8. Click on Add Chart , Select the Line Chart.
9. Click on save, Give the report Name : New Service information Report
10. Report unique Name is auto populated.
11. Select the folder the created and Click on save.



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

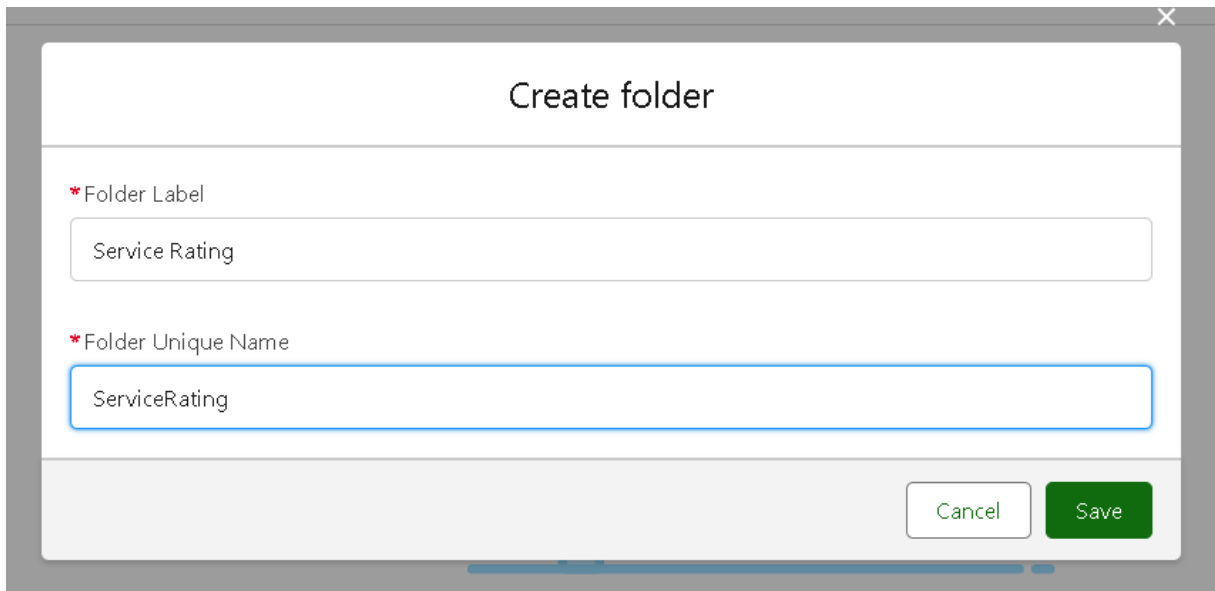
Milestone17:Dashboards

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.

Activity1:CreateDashboardFolder

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

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

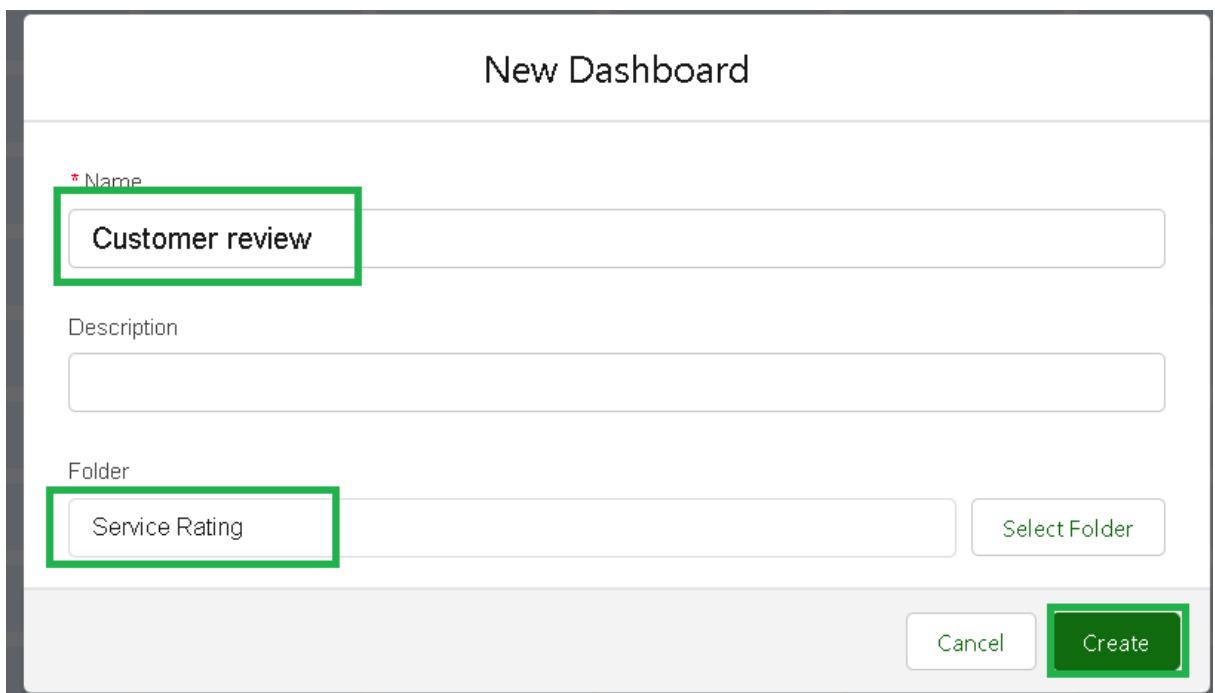


A screenshot of a 'Create folder' dialog box. The title bar says 'Create folder'. There are two input fields: the first is labeled '* Folder Label' and contains the text 'Service Rating'; the second is labeled '* Folder Unique Name' and contains the text 'ServiceRating'. At the bottom right, there are two buttons: 'Cancel' and 'Save'.

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

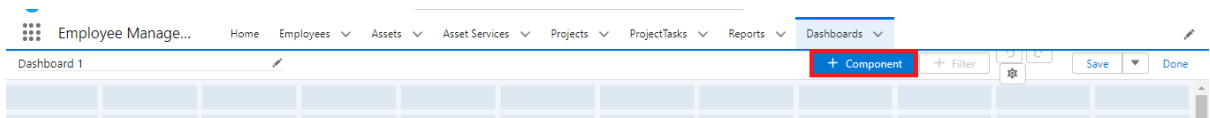
Activity 2: Create Dashboard

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

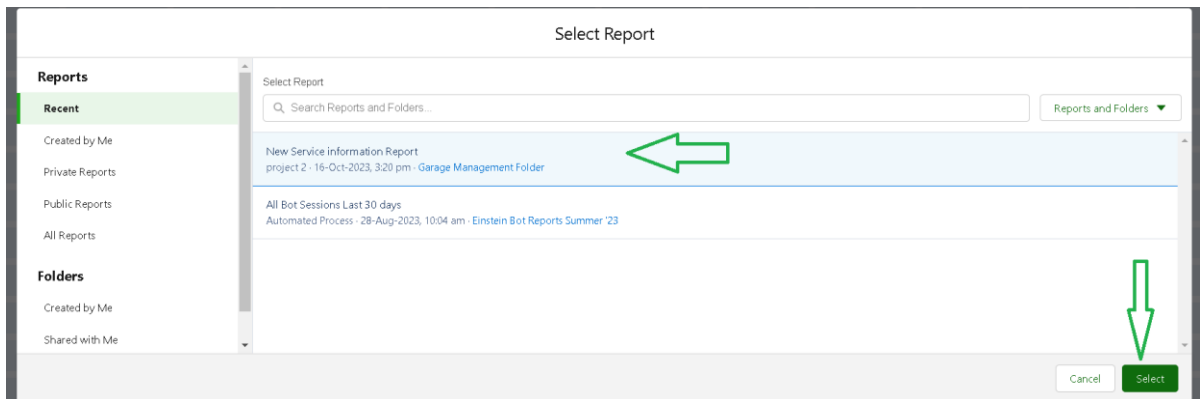


A screenshot of a 'New Dashboard' dialog box. The title bar says 'New Dashboard'. There are three input fields: the first is labeled '* Name' and contains the text 'Customer review'; the second is labeled 'Description' and is empty; the third is labeled 'Folder' and contains the text 'Service Rating'. To the right of the 'Folder' field is a button labeled 'Select Folder'. At the bottom right, there are two buttons: 'Cancel' and 'Create'.

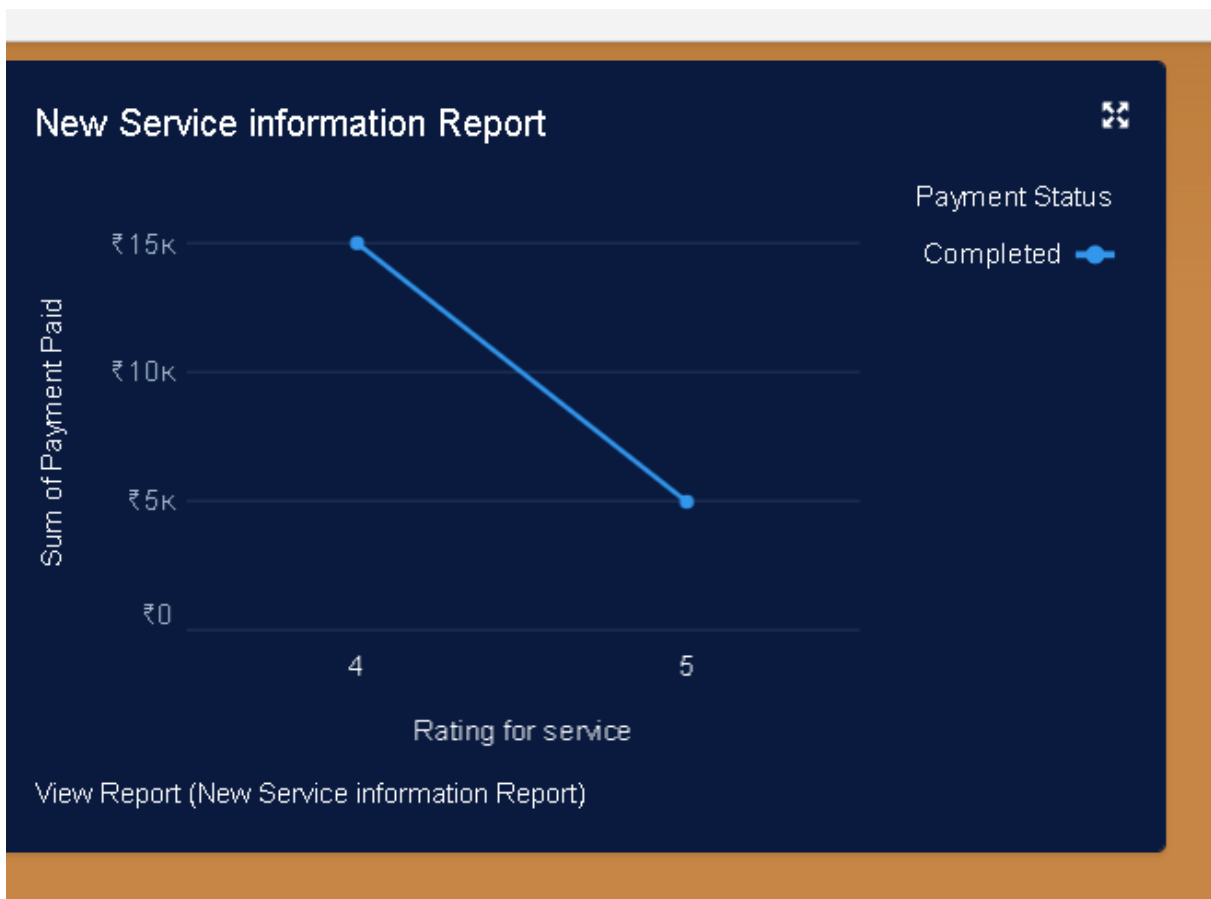
3. Select add component.



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.

Edit Subscription

Schedule dashboard refreshes and subscribe to receive results.

Settings

Frequency

DailyWeeklyMonthly



Days

SunMonTueWedThuFriSat



Time

3:00 pm

Recipients

☒ Receive new results by email when dashboard is refreshed. ⓘ

Send email to

Me

Edit Recipients



Cancel

Save