Gas Station Management Application using CRM

Prepared & Submitted by:-

Name:- Mareboina Nagamani

Branch:- B.Tech CSE

College:- Malineni Lakshmaiah Women's Engineering College

GitHub Link:- https://github.com/mnagamani10

Project Overview:-

The Gas Filling Store CRM Application is a comprehensive solution designed to streamline and simplify the gas filling process for both customers and store owners. It leverages the power of customer relationship management (CRM) to enhance customer experiences, optimize store operations, and improve overall efficiency in the gas filling industry. This project aims to develop a user-friendly and feature-rich application that addresses the specific needs of gas filling stores.

Project Objectives:-

- To provide a user-friendly CRM-based application for gas filling store management.
- To streamline the gas filling process for both customers and store owners.
- To automate record-keeping of fuel sales, suppliers, and customer details.
- To enhance customer experience by ensuring quick and accurate service.
- To optimize store operations through better data tracking and reporting.
- To reduce manual errors by digitizing transactions and records.
- To support scalability so the system can be used for multiple gas stations.
- To improve decision-making with real-time insights and reports.

Salesforce Development:-

➢ In salesforce development to create this two types:-

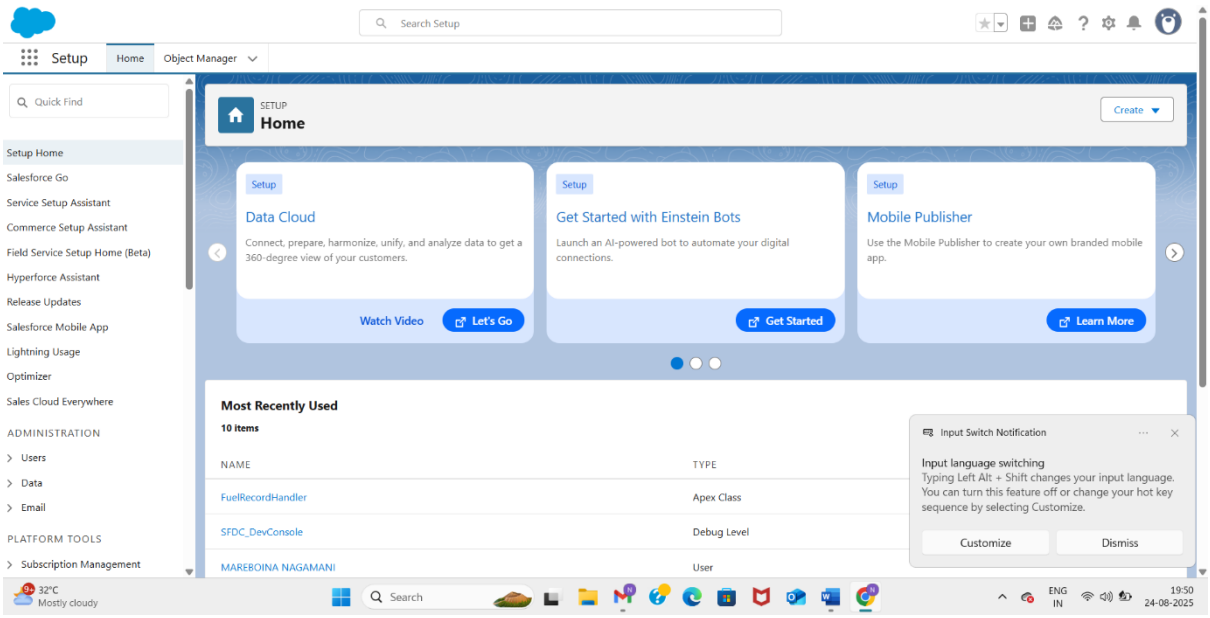
- **Creating Developer Account:-**

  Creating developer org in salesforce following this steps:-

  1. Go to google search https://developer.salesforce.com/signup this link and sign up the form enter the details.
     - ✓ First name and Last name
     - ✓ Email
     - ✓ Role: Developer
     - ✓ Company:College Name
     - ✓ County:India
     - ✓ Postal Code: Pin code

- **Account Activation:-**

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



password. When you will redirect to your salesforce setup page. After following these steps, the Setup will be created as shown in the image below.

Objects:-

- Salesforce objects are database tables that permit you to store data that is specific to an organization.
  - **Customization of objects:-**

    There are 4 types of objects:-

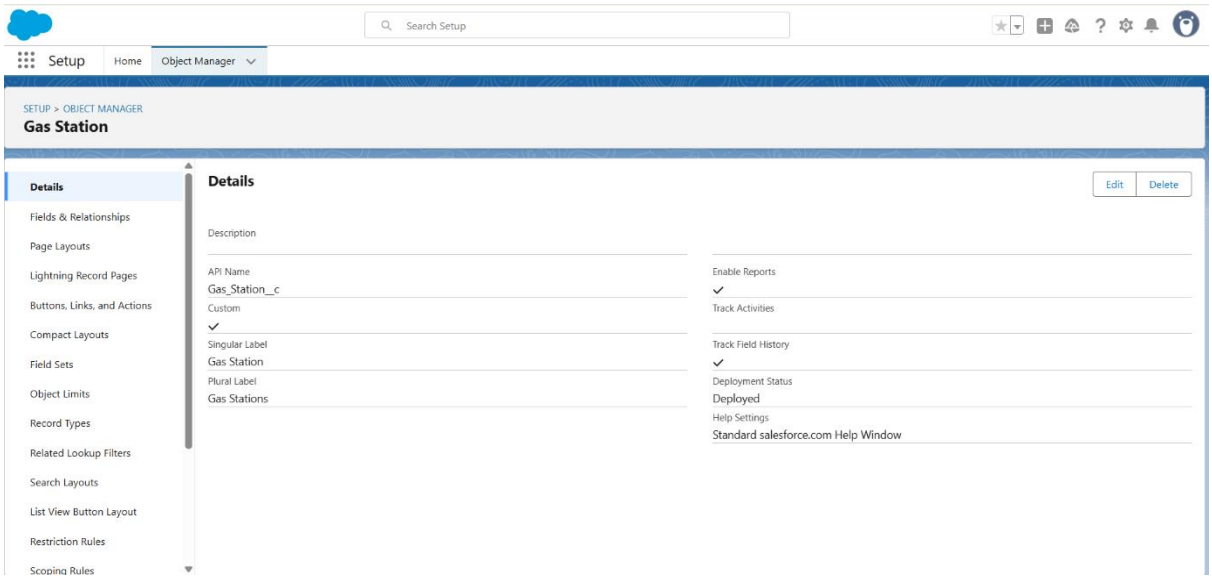
    1.Supplier object

    2.Gas Station object

    3. Buyer object

    4.Fuel details object

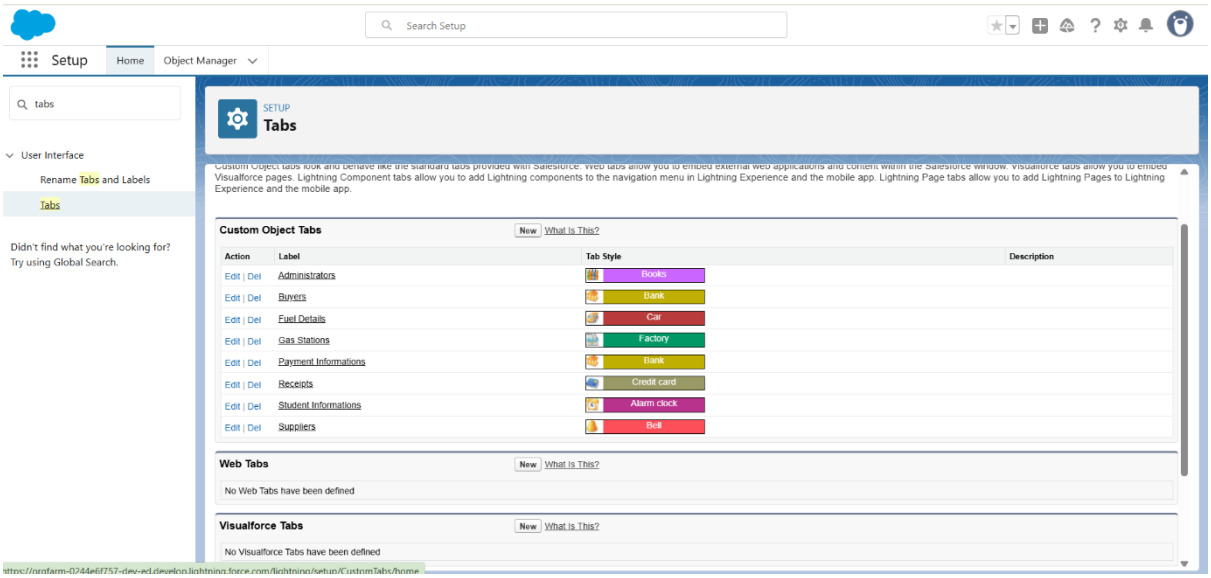
    - ✓ How to create this customization object is following the below steps:-
    - ✓ First we create an object is supplier object :-
    - Go to setup page click on the object manager, click on Create, click on Custom object.
    - After enter the label name Supplier
    - Plural label name Suppliers
    - Enter the Record Name: Supplier Name
    - Enter the Data Type: Name
    - Click on Allow reports and Track Field History
    - Allow search, Save.
- Same above process we can create remaining object are Gas Station, Buyer, Fuel details.
- After following these steps, the customization object will be created as shown in the image below.



Tabs:-

- A tab is like a user interface that is used to build records for objects and to view the records in the objects.
- In this we can create a Custom Tabs
    - **Creating a Custom Tab:-**

      To create a tab for Supplier tab following steps:-
        - ✓ Go to setup page, type tabs in Quick Find bar, click on tabs, click on New.
        - ✓ Select object:- Supplier
        - ✓ Select tab style.
        - ✓ Cleck on Next (add to profiles page) keep it is default.
        - ✓ Next (add to Custom App) uncheck the include tab.
        - ✓ Make sure that append tab to users existing personal customization is checked.
        - ✓ Click save.
- Same above steps to follow to create the tabs for the remaining objects, they are Gas Station, Buyer, Fuel details.
- After following these steps, the Tabs will be created as shown in the image below.



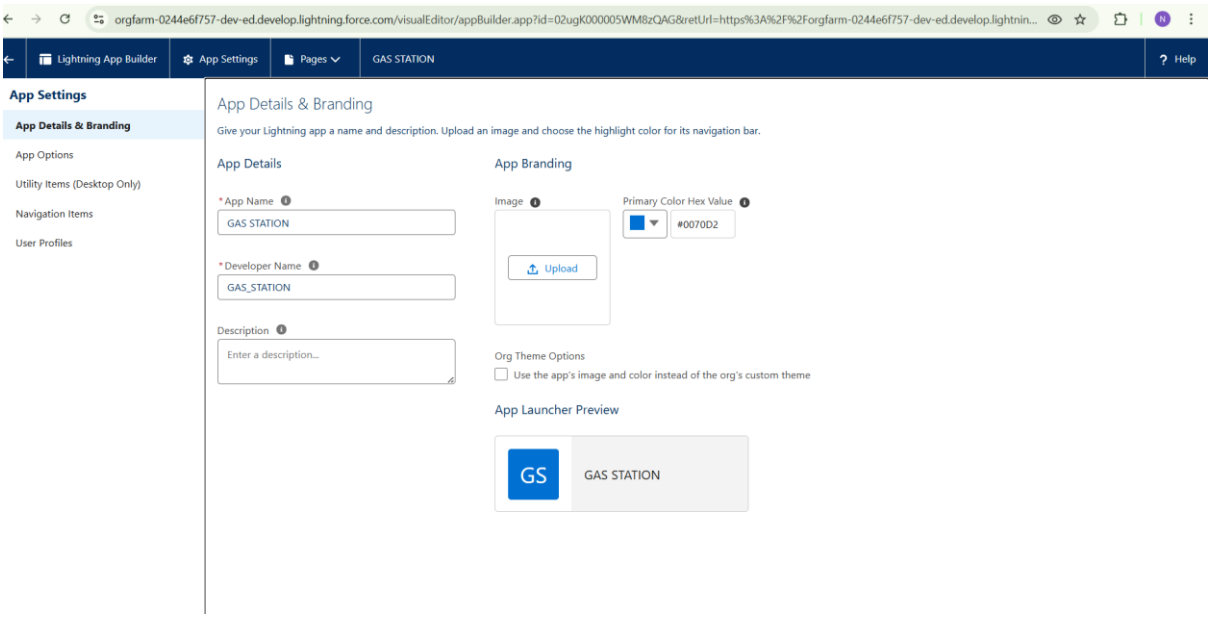
The Lightning App:-

- 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.
    - **Creating A Lightning App:-**

To create a lightning app page to following the below steps:-

- ✓ Go to setup page, search app manager in quick find, select app manager, click New lightning App.
- ✓ Fill the app name in app details as GAS STATION, click on Next(app option page) keep it as default, click on Next,(utility items) keep it as default, click on Next.
- ✓ To add Navigation items:-
- ▪ Select the items(Supplier, Gas Station, Buyer, Receipt) from the search bar and move it using the arrow button, click on Next.
- ✓ To add user profiles:-
- ▪ Search profiles (System administrator) in the search bar, click on the arrow button, save & finish.

➢ After following these steps, the Lightning App will be created as shown in the image below.



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

- **Creating Junction Object:-**

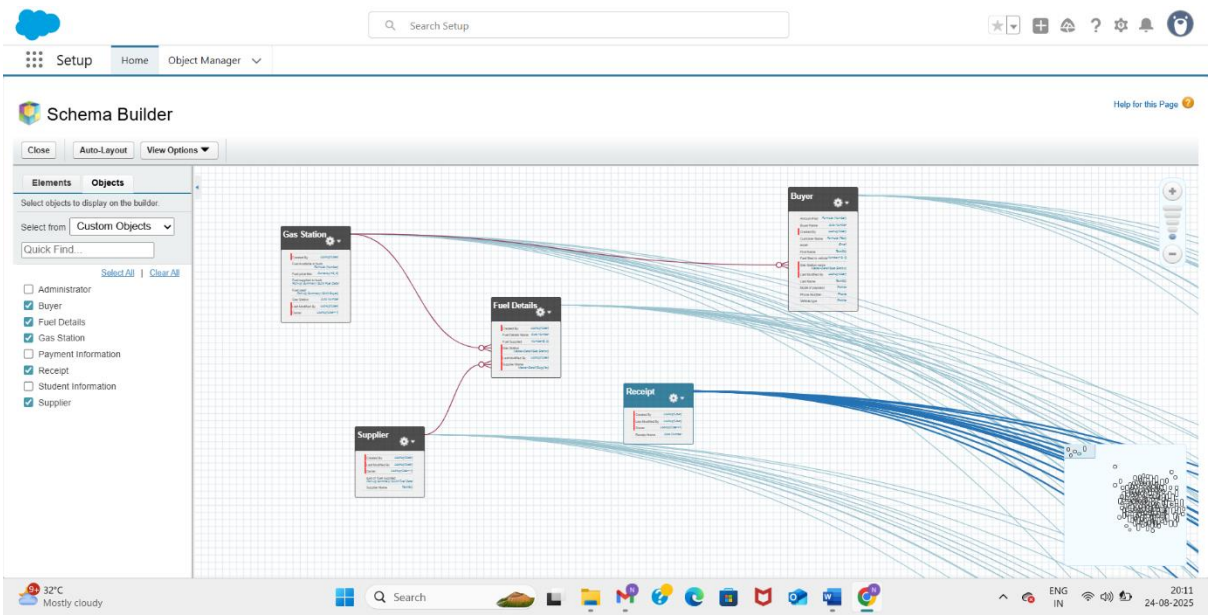
Junction object is a custom object that serves as a bridge between two relate objects in a many-to-many relationship. It allows you to create a relationship between records of two different objects by creating a many-to-many relationship model.

Creating junction object as Fuel details with Supplier & Gas station To create junction object.

Now following the below steps to create junction object:-

- ✓ Go to the setup page, click on object manager, From drop down click edit for Fuel details object.
- ✓ Select Click on fields & relationship, click on New.
- ✓ Master-Detail relationship as data type and click Next.
- ✓ Give Field Label as Supplier Name and click Next.
- ✓ Click on Next and Next Save & New.
- ✓ Follow the same steps form to select the related object Gas Station and click Next.
- ✓ Give Field Label as Gas Station and click on Next.
- ✓ Click on Next and Next , Save.

➢ After following these steps, the Junction Object will be created as shown in the image below.



- **Creating A Master-Detail Relationship:-**

Master-detail relationship is a type of relationship between two objects where the master object controls certain behaviors and settings of the detail object. Here are a few use cases that demonstrate the use of master-detail relationships.

- ✓ Creating Master-Detail Relationship between Buyer & Gas Station Object

- ✓ To Create a Master-Detail relationship follow the steps:-
- ✓ Go to the setup page, click on object manager, From drop down click edit for Buyer object.
- ✓ Click on fields & relationship, click on New.
- ✓ Select Master-Detail relationship as data type and click Next.
- ✓ Same steps to create to Select the related object Gas station. Give Field Label as Gas Station name and click Next. Click on Next and Next, Save.

- **Creating the number field in Fuel details object:-**

Creating the number field in Fuel details object follow the steps:-

- ✓ Repeat steps 1 and 2 mentioned in activity of Junction object.
- ✓ Select Data type as Number and click Next.
- ✓ Given the Field Label as Fuel Supplied and length as 5.
- ✓ Field Name will be auto populated, and click on Next and Next click on Save.

- **Creating The Roll-Up Summary:-**

A rollup summary field is a field that summarizes data from a child object to a parent object that share a master-detail relationship. Rollup summary fields can use the COUNT, SUM, MIN, and MAX functions. For example, you could use a rollup summary field to display the total value (amount of fuel supplied) from Fuel details on a related Supplier.

Creating the Roll-up summary field on Supplier & Gas Station Objects following steps:-

- ✓ Go to setup, click on Object Manager, type object name(Supplier) in search bar, click on the object.
- ✓ Now click on Fields & Relationships, click on New
- ✓ Select the data type as Rollup summary and click Next.
- ✓ Give the Field label as sum of Fuel supplied, Field Name will be Auto generated, and click Next.
- ✓ Select the summarized object as Fuel details.
- ✓ Select the Rollup type as sum.
- ✓ Select the field to aggregate as Fuel supplied, and click Next and Next and Save.
- ✓ Follow the same steps for the Gas Station object from 1 to 3.
- ✓ Give the Field label as Fuel supplied to bunk, Field Name will be Auto generated, and click Next.
- ✓ Select the summarized object as Fuel details.
- ✓ Select the Rollup type as sum.
- ✓ Select the field to aggregate as Fuel supplied, and click Next and Next and Save.

Note : create the field as Fuel filled in vehicle using number datatype in Buyer object.

- ✓ Follow the same steps for the Gas station Object from 1 to 3.

- ✓ Give the Field label as Fuel used, Field Name will be Auto generated, and click Next.
- ✓ Select the summarized object as Buyer.
- ✓ Select the Rollup type as sum.
- ✓ Select the field to aggregate as Select the field to aggregate as, and click Next and click Next and Save.

- **Creating Formula Field In Gas Station Object:-**

A formula field is a custom field that can be used to calculate or display data on a Salesforce record.
Formula fields can be used to perform a variety of tasks, such as:-

- Calculating totals or averages
- Creating custom fields that display data from other fields
- Validating data entry
- Automating processes

To create Formula field in Gas Station object follow the steps:-

- ✓ Go to setup, click on Object Manager, type object name(Gas station) in search bar, click on the object.
- ✓ Click on fields & relationship, click on New.
- ✓ Select Data type as Formula and click Next.
- ✓ Give Field Label and Field Name as Fuel Available in bunk and select formula return type as Number and click next.
- ✓ Under Advanced Formula write down the formula and click Check Syntax and Save.

Insert field formula should be :-

- Fuel_supplied_to_bunk__c – Fuel_Used__c

Creating the Formula field in Buyer Object following same steps:-

- ✓ Go to setup, click on Object Manager, type object name(Buyer) in search bar, click on the object.
- ✓ Click on fields & relationship, click on New.
- ✓ Select Data type as Formula and click Next.
- ✓ Give Field Label and Field Name as Customer Name and select formula return type as TEXT and click next.

Insert field formula should be :-

- ✓ First_Name__c +''+ Last_Name__c
- ✓ Click Check Syntax and Save.

- **Creating Cross Object Formula Field In Buyer Object:-**

A cross-object formula field is a formula field that references fields from another object in Salesforce. This type of formula allows users to calculate and display data from multiple objects on a single record.
To create the cross-object formula field in Buyer object follow the steps:-

- ✓ Go to setup, click on Object Manager, type object name(Buyer) in search bar click on the object.
- ✓ Click on fields & relationship, click on New.

- ✓ Select Data type as Formula and click Next.
- ✓ Give Field Label and Field Name as Amount Paid and select formula return type as Number and click next.

Insert fields formula should be:

- ✓ Fuel_filled_in_vehicle__c * Gas_Station_name__r.Fuel_price_liter__c
- ✓ Under Advanced Formula write down the formula and click Check Syntax and Save.

- **Creating Picklist Field In Buyer Object:-**

To follow steps to create Buyer Object:-

- ✓ Go to setup, click on Object Manager, type object name(Buyer) in search bar, click on the object.
- ✓ Click on fields & relationship, click on New.
- ✓ Select Data type as Picklist and click Next.
- ✓ Enter Field Label as Vehicle type, under values select Enter values, with each value separated by a new line and enter values as shown below.

The values are: -

- two wheeler
- three wheeler
- four wheeler
- six wheeler
- eight wheeler and Others.

- ✓ Click Next, and Next And Click on Next and Save & New.
- ✓ Repeat the process 1 and 2 steps.
- ✓ Enter Field Label as Mode of payment, under values select Enter values, with each value separated by a new line" and enter values as shown below.

The values are :-

- credit card
- debit card
- net banking
- upi
- cash.

- ✓ Click Next, and Next And Click on Next and Save & New.

- **Creating The Validation Rule:-**

Creating the validation rule for phone number field in Buyer object follow the steps:-

- ✓ Go to the setup page, click on object manager, From drop down click edit for Buyer object.
- ✓ Click on the validation rule, click New.
- ✓ Enter the Rule name as Phone.

Insert the Error Condition Formula as:-

- NOT(REGEX(Phone_Number__c , "[6-9]{1}[0-9]{9}")).

- ✓ Enter the Error Message as incorrect data, select the Error location as Field and select the field as phone number, and click Save.

Page Layouts:-

- ➢ Page Layout in Salesforce allows us to customize the design and organize detail and edit pages of records in Salesforce. Page layouts can be used to control the appearance of fields, related lists, and custom links on standard and custom objects' detail and edit pages.

To create a page layout following steps:-

- ✓ Go to Setup, Click on Object Manager, earch for the object (Buyer) From drop down select the object and click on it.
- ✓ Click on Page layout, Click on New.
- ✓ Select the existing page layout, and give the page layout name as customer layout, and click save.
- ✓ Drag and drop the section field to Buyer details and create the section.
- ✓ Enter the section name as Personal details, click Ok.

Now drag the fields to this section that mentioned, they are :-

- First name
- last name
- customer name
- phone number
- email
- Gas station name.

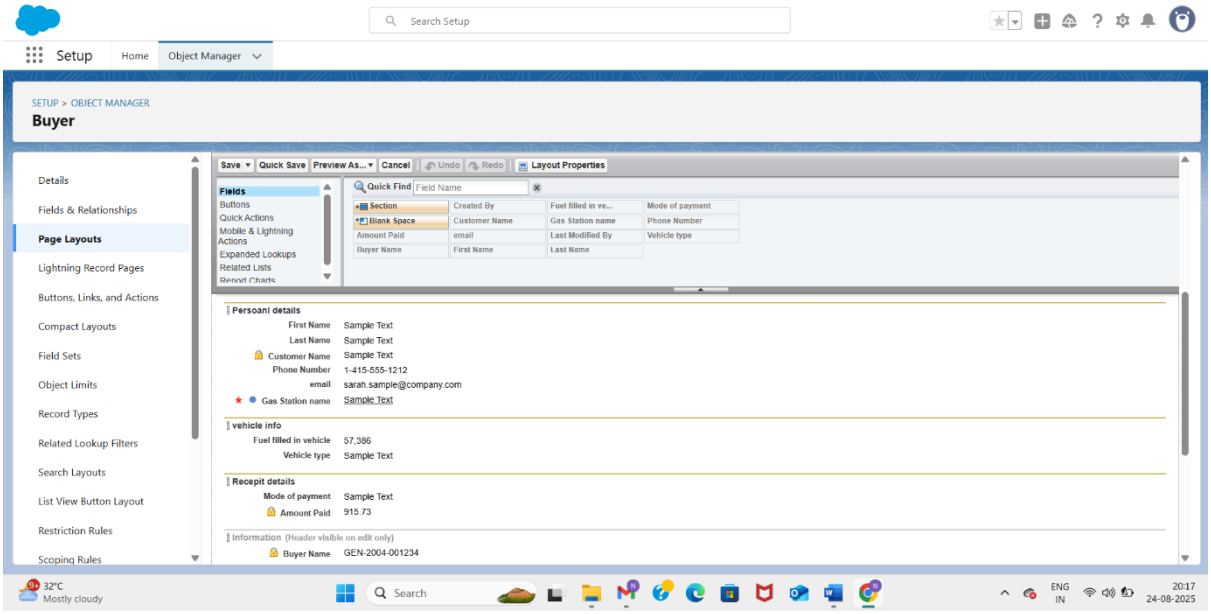
Follow the same process for another two sections as shown above, they are One section is vehicle info, drag the fields that are:-

- Fuel filled in vehicle
- vehicle type.

Another section is Receipt details, and drag the fields that are :-

- Mode of payment
- Amount paid.

- ✓ Then, Click Save.
- ➢ After following these steps, the Page Layouts will be created as shown in the image below.



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:-

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.

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 follow the steps:-

  - ✓ Go to setup, type profiles in quick find box, click on profiles, clone the desired profile (Standard User), enter profile name (Manager), Save.
  - ✓ While still on the profile page, then click Edit.
  - ✓ Select the Custom App settings as default for the Gas station.
  - ✓ Scroll down to Custom Object Permissions and Give access permissions for Buyers, Fuel details, gas station and suppliers objects.
  - ✓ Change the session times out after should be 8 hours of inactivity.
  - ✓ Change the password policies as mentioned, User passwords expire in should be never expires. Minimum password length should be 8, and click save.

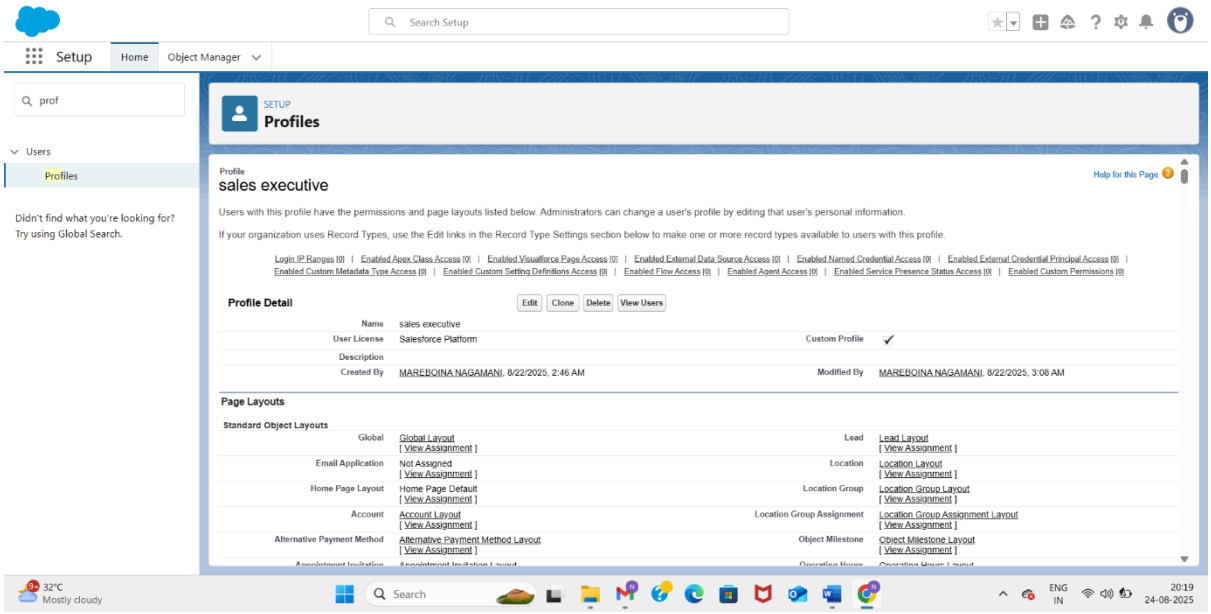
- ## Sales Executive Profile:-

  - Go to setup, type profiles in quick find box, click on profiles clone the desired profile (Salesforce Platform User), enter profile name (sales executive), Save.
  - While still on the profile page, then click Edit.
  - Select the Custom App settings as default for the Gas station.
  - Scroll down to **Custom** Object Permissions and Give access permissions for Buyers, Fuel details , gas station and suppliers objects, Click Save.

- ## Sales Person Profile:-

  - ✓ 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.
  - ✓ While still on the profile page, then click Edit.
  - ✓ Select the Custom App settings as default for the Gas station.
  - ✓ Scroll down to Custom Object Permissions and Give access permissions for Buyers, Fuel details , gas station and suppliers objects. And click on Save.

➢ After following these steps, the Profiles will be created as shown in the image below.



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 follow the steps:-

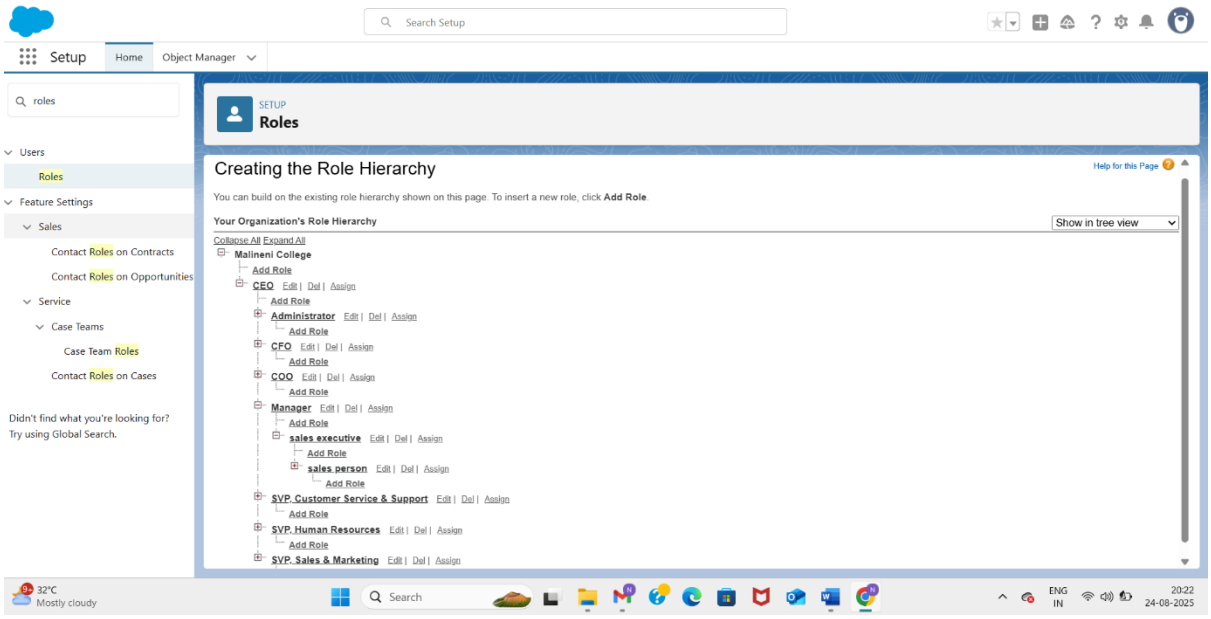
  - ✓ Go to quick find, Search for Roles, click on set up roles.
  - ✓ Click on Expand All and click on add role under whom this role works.
  - ✓ Give Label as Manager and role name get auto populated. Then clicked save.

- **Creating Another Roles:-**

  Creating another two roles under manager follow the steps:-

  - ✓ Go to quick find, Search for Roles, click on set up roles.
  - ✓ Click plus on CEO role, and click add role under manager.
  - ✓ Give Label as sales executive, and Role name gets auto populated. Then click on Save.
  - ✓ Repeat the same steps, another role.
  - ✓ Click plus on CEO role, and click plus on manager, and click add role under sales executive .
  - ✓ give Label as sales person, and Role name gets auto populated. Then click on Save.

➢ After following these steps, the Role & Role Hierarchy will be created as shown in the image below.



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:-

  Go to setup, type users in quick find box, select users, click New user.

  - Fill in the fields:-
  - First Name : Niklaus
  - Last Name : Mikaelson
  - Alias : Give a Alias Name
  - Email id : Give your Personal Email id
  - Username : Username should be in this form: text@text.text
  - Nick Name : Give a Nickname
  - Role : Manager
  - User licence : Salesforce
  - Profiles : Manager

  ✓ Save

- ## Creating Another Users:-

  Follow the same steps from above activity and create another user using:-

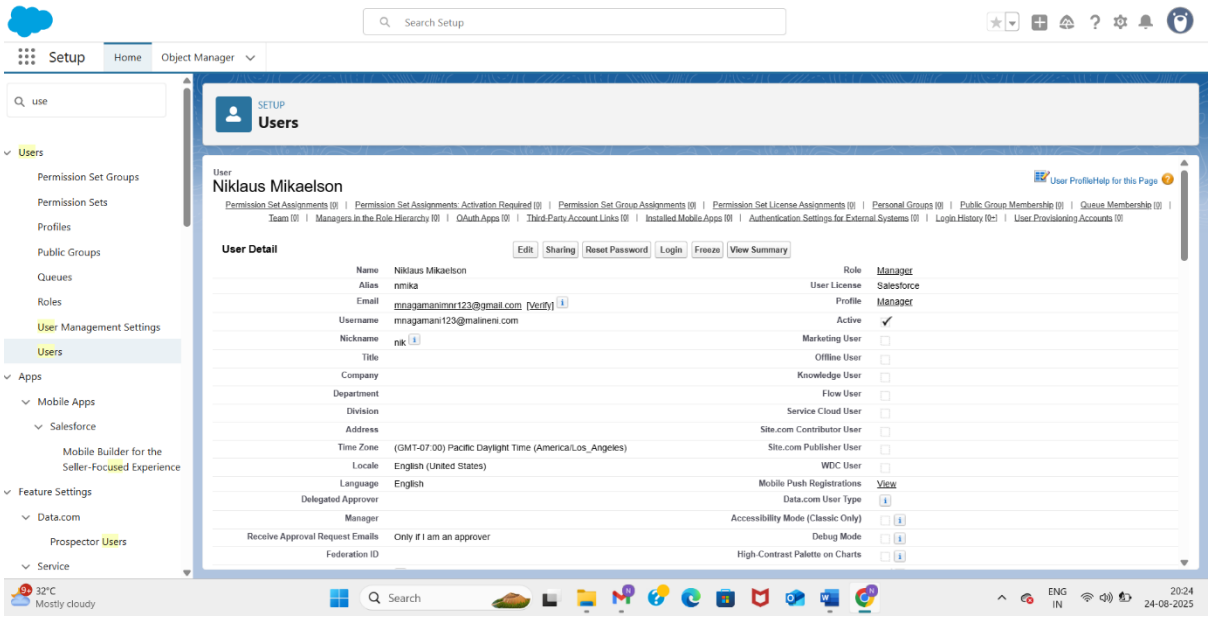
  1) Role : sales executive
  2) User licence : Salesforce Platform

3) Profile : sales executive

Repeat the steps and create another user using:-

1) Role : sales person
2) User licence : Salesforce Platform
3) Profile : sales person

➢ After following these steps, the Users will be created as shown in the image below.



Permission Sets:-

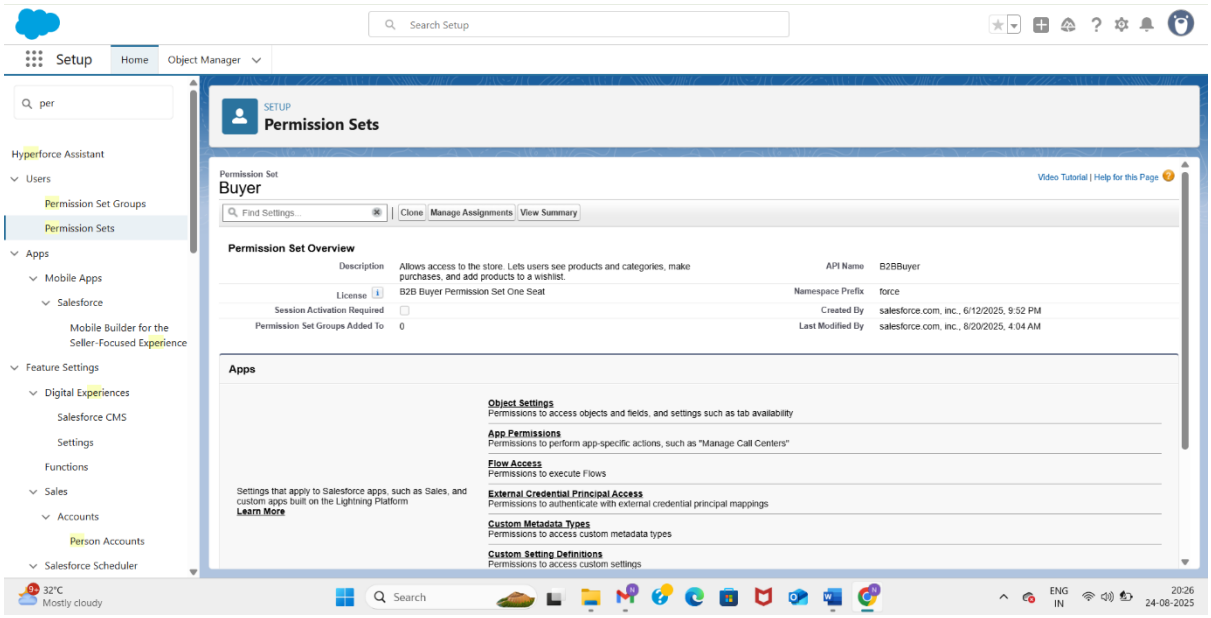
A standard permission set consists of a group of common permissions for a particular feature associated with a permission set license. Using a standard permission set saves you time and facilitates administration because you don't need to create the custom permission set.

- **Creating Permission Set:-**

Creating permission set to follow the steps:-

✓ Go to setup, type permission sets in quick search ? select permission sets and click New.
✓ Enter the label name as P1, API will be auto populated and Save.
✓ Under Apps Select object settings.
✓ Click on Fuel details object, click on Edit under object permission check for read and create.
✓ Click on Save.
✓ After saving the permission click on the Manage assignment.
✓ Now click on the Add Assignment.
✓ Now select the users which you have created in user milestone, using sales executive profile and click on Next, Assign, Done.

➢ After following these steps, the permission set will be created as shown in the image below.



Setup for QWD:-

Organization-Wide Defaults, or OWDs, are the pattern security rules that you can follow for your Salesforce instance. Organization Wide Defaults are utilized to confine who can access what information in your CRM. You can award access through different methods that we will discuss later (sharing principles, Role Hierarchy, Sales Teams, and Account groups, manual sharing, and so forth).

Primarily, there are four levels of access that can be set in Salesforce OWD and they are:-

- Public Read/Write/Transfer (only available of Enquiry and Cases)
- Public Read/Write
- Public Read/Only
- Private

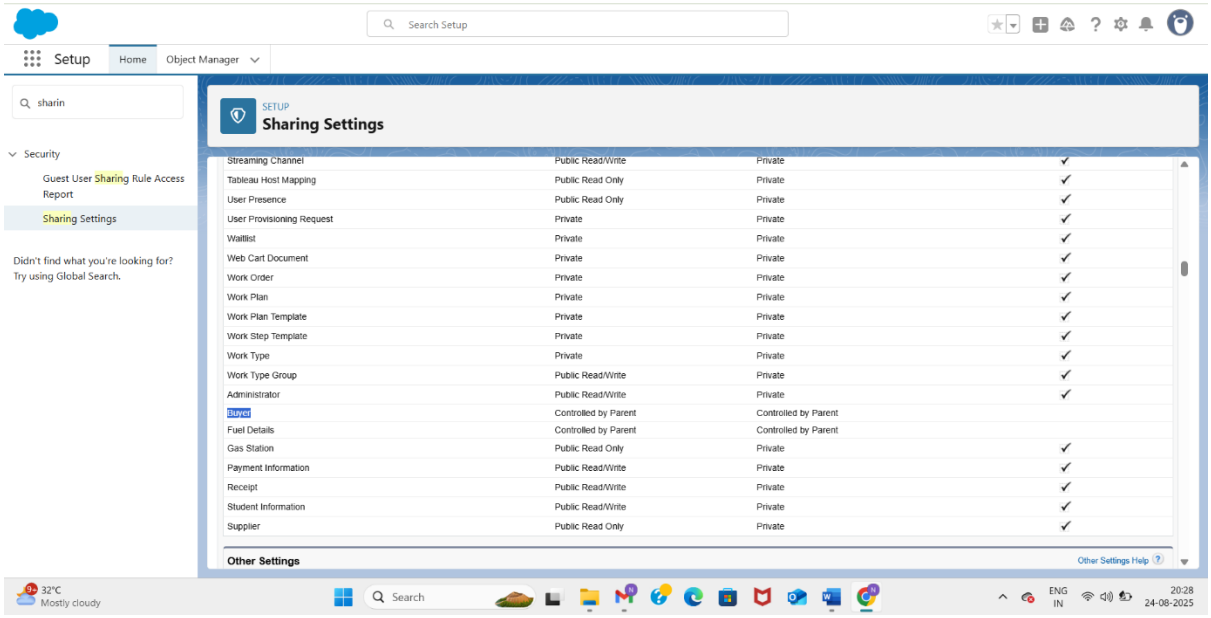
- ## Creating OWD Setting:-

  To creating OWD Setting follow this steps:-

  - ✓ Go to setup, type sharing settings in quick search, Click edit.
  - ✓ Scroll down, change the default internal access to public read-only for Gas station and Supplier object.
  - ✓ Click Save.
  - ✓ Extra information, By these every profile has their own access, according to their profile.

- ✓ But in our case we created a roles and given the roles in such a way that manager can see sales executive and sales person records , sales executive can see the sales person records.

➢ After following these steps, the setup OWD Setting will be created as shown in the image below.



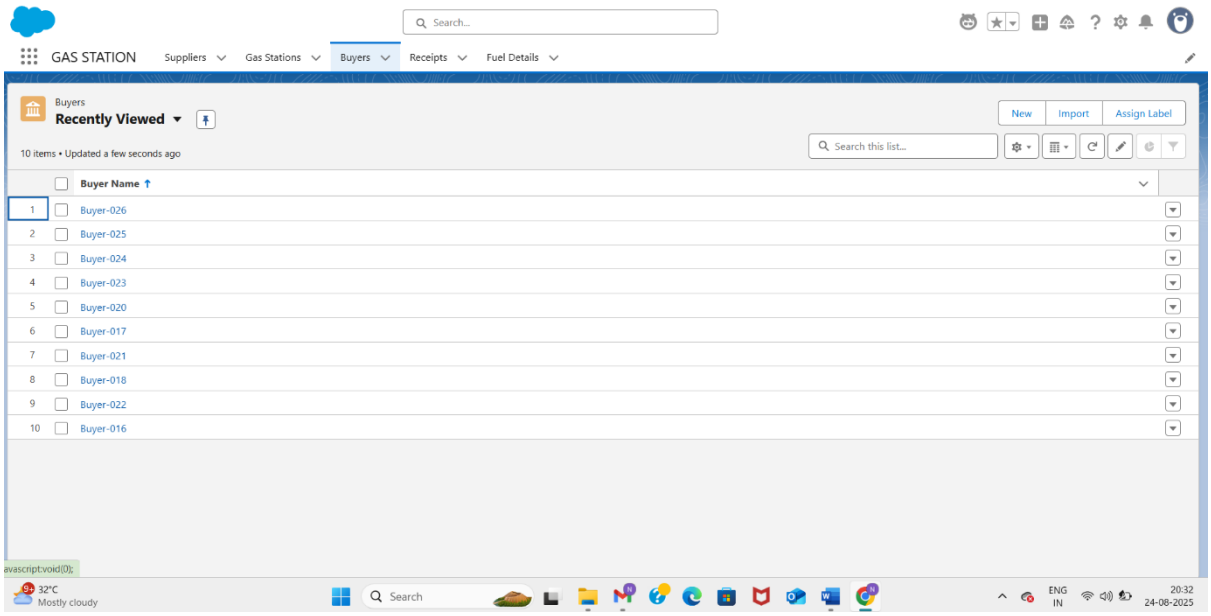
User Adoption:-

- **Create A Record:-**

To create a record in junction object follow these steps**:-**

- ✓ Click on the app launcher locate at left side of the screen.
- ✓ Search for Gas Station and click on it.
- ✓ Click on fuel detail tab.
- ✓ Click on new and fill the details as shown below figs, and click save.
- ✓ Creating the supplier record in fuel detail record, by clicking the new supplier.
- ✓ Fill the details in supplier record and click on save.
- ✓ Creating the Gas station record in fuel details record, by clicking on new gas station.
- ✓ Fill the details in gas station record, Click save.
- ✓ Fill the remaining details in fuel detail record , and click save.

➢ Followed by these create 10 more records in Buyer object.
➢ After following these steps, the 10 Buyer objects will be created as shown in the image below.

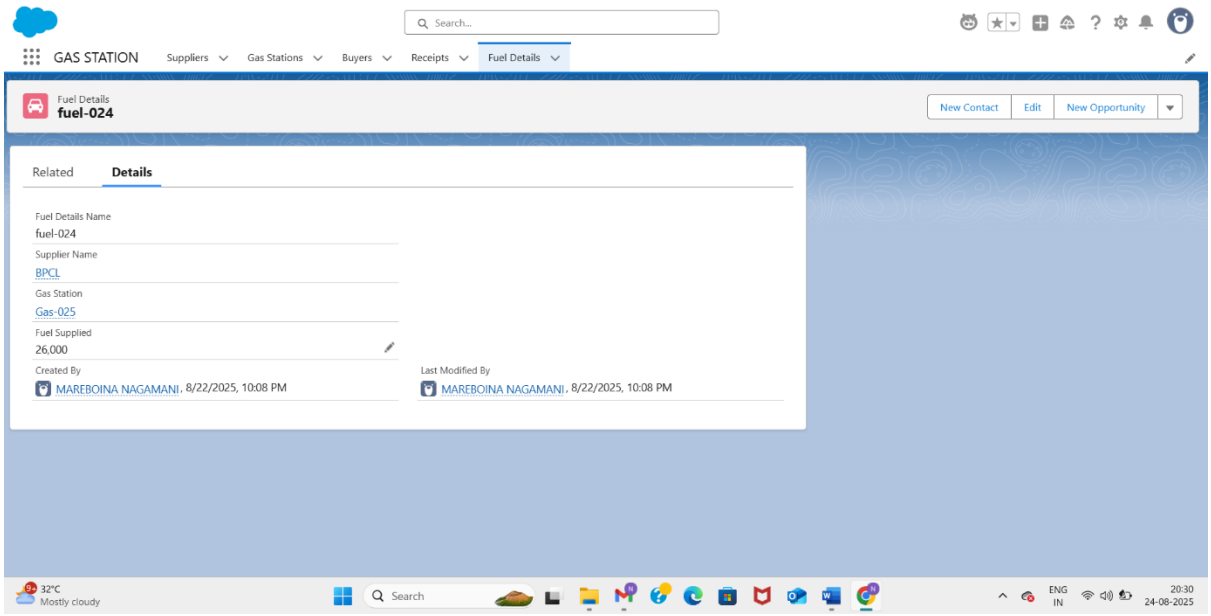


- **View A Record:-**

  To create a record in junction object follow these steps:-

  - ✓ Click on the app launcher locate at left side of the screen.
  - ✓ Search for Gas Station and click on it.
  - ✓ Click on fuel detail tab.
  - ✓ Click on the records that are already created.

➢ After following these steps, the View the Record will be created as shown in the image below.



- ## Delete A Record:-

  To create a record in junction object follow these steps:-

  - ✓ Click on the app launcher locate at left side of the screen.
  - ✓ Search for Gas Station and click on it.
  - ✓ Click on fuel detail tab and click on it.
  - ✓ Click on Arrow at right hand side on that Particular record.
  - ✓ Click delete and delete again.

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

- ## Creating A Report Folder:-

  - ✓ Click on the app launcher and search for reports.
  - ✓ Double click on the report, report tab will be in navigation bar.
  - ✓ Click on the report tab, click on new folder.
  - ✓ Give the Folder label as Fuel Estimation, Folder unique name will be auto populated.
  - ✓ Click on Save.

- ## Sharing A Report Folder:-

  - ✓ Go to the app, click on the reports tab.
  - ✓ Click on the All folder , click on the arrow for Fuel estimation folder, and Click on share.
  - ✓ Select the share with as roles, in name field search for manager give view access for that role.
  - ✓ Then click share, and click on Done.

- ## Create A Report:-

  Create a report follow the steps:-

  - ✓ Go to the app, click on the reports tab.
  - ✓ Click New Report.

- ✓ select for report type, search for Gas station with buyers click on it. And click on start report.

Their outline pane is opened alredy, select the fields that mentioned below in column section.

- Fuel filled in vehicle
- Amount paid

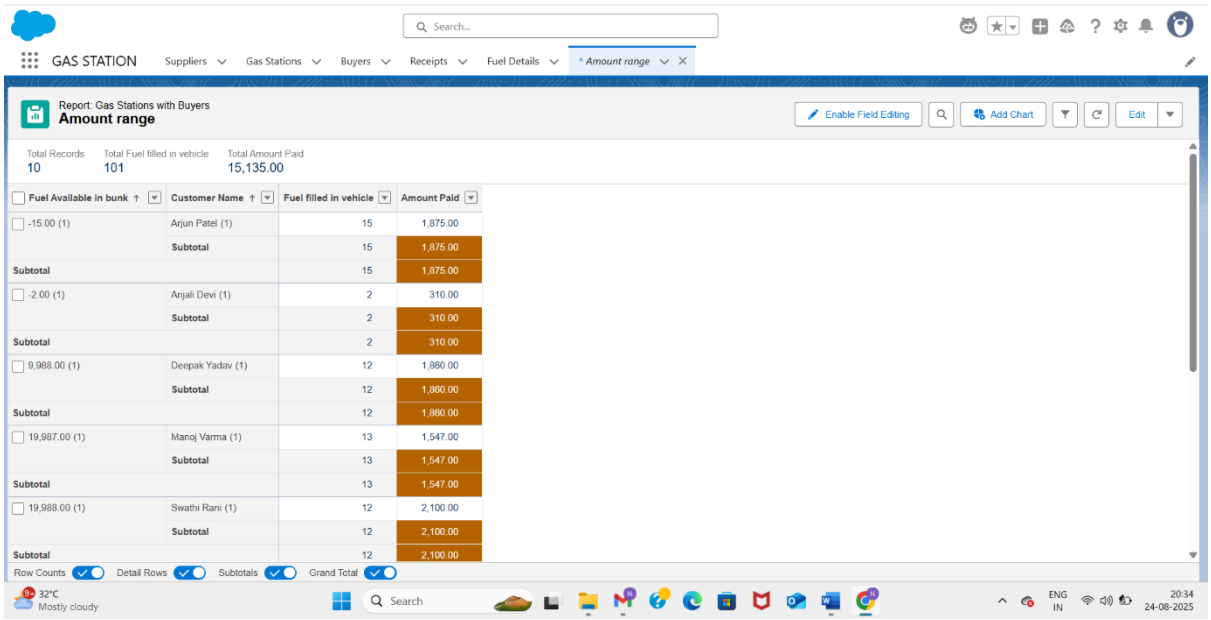
- ✓ Remove the unnecessary fields.

Select the fields that mentioned below in GROUP ROWS section.

- Fuel Available in bunk
- Customer name

- ✓ Change the apply conditional formatting to sum of Amount paid.
- ✓ Mention the range form 1000 to 5000 .
- ✓ Don't change the colours, and click on Done.
- ✓ Click apply.
- ✓ Click save, give the report name as Amount range, report unique name will be auto populated.
- ✓ Click on select folder, select Fuel estimation click select folder
- ✓ Click save & run , then the preview will be shown below.

- ➢ After following these steps, the create a report will be created as shown in the image below.



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.

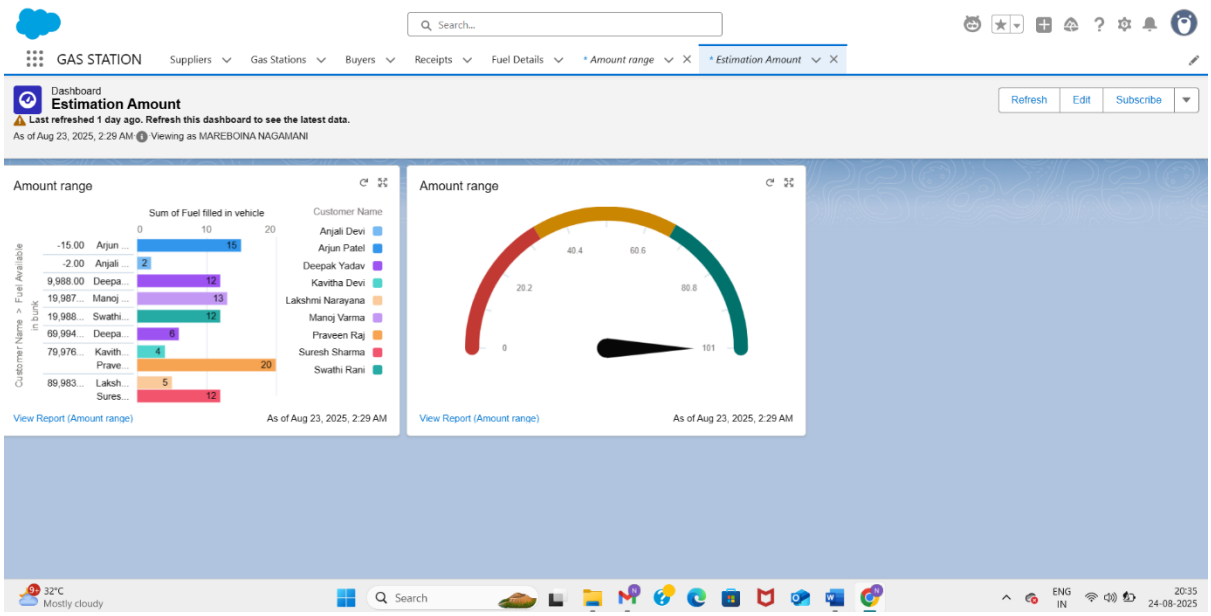
- **Create Dashboard Folder:-**

  - ✓ Click on the app launcher and search for dashboard.
  - ✓ Click on dashboard tab.
  - ✓ Click new folder, give the folder label as Amount estimation dashboard.
  - ✓ Folder unique name will be auto populated.
  - ✓ Click on Save.
  - ✓ Follow the same steps, form milestone 12, and activity 2, and provide the sharing settings for the folder that just created.

- **Create Dashboard:-**

✓ Go to the app, click on the Dashboards tabs.
✓ Give a Name and select the folder that created, and click on create.
✓ Select add component.
✓ Select a Report and click on select.
✓ Click Add then click on Save and then click on Done.
✓ Preview is shown below.

➢ After following these steps, the create Dashboard will be created as shown in the image below.



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:-

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

  Select the Record-triggered flow and Click on Create.

  Select the Object as a buyer in the Drop down list.

  Select the Trigger Flow when: A record is Created or Updated.

  Select the Optimize the flow for: Actions and Related Records and Click on Done.

  Now change the mode form Auto-layout to free-form.

  Now select the manger option in toolbox, click New resource.

  Select the resource type as text template.

  Enter the API name as emailbody.

  In body field paste the syntax that given below.

  Hello {!$Record.Customer_name__c},

  Thank you for coming , we are glad and considering that we provided the best survise.

  RECEPIT DETAILS :

  Customer name : {!$Record.Customer_name__c}

  Amount paid by Customer : {!$Record.Amount_Paid__c}

  Vehicle type : {!$Record.Vehicle_type__c}

  Fuel intake in vehicle : {!$Record.Fuel_filled_in_vehicle__c}

  Change the view as Rich Text, View to Plain Text.

  Click done.

  Now click on elements, and drag the action element into the preview pane.

  Their action bar will be opened in that search for send email and click on it.

  Give the label name as notice.

API name will be auto populated.

Enable the body in set input values for the selected action.

Select the text template that created.

Include recipient address list select the email form the record.

Include subject as welcome to gas station.

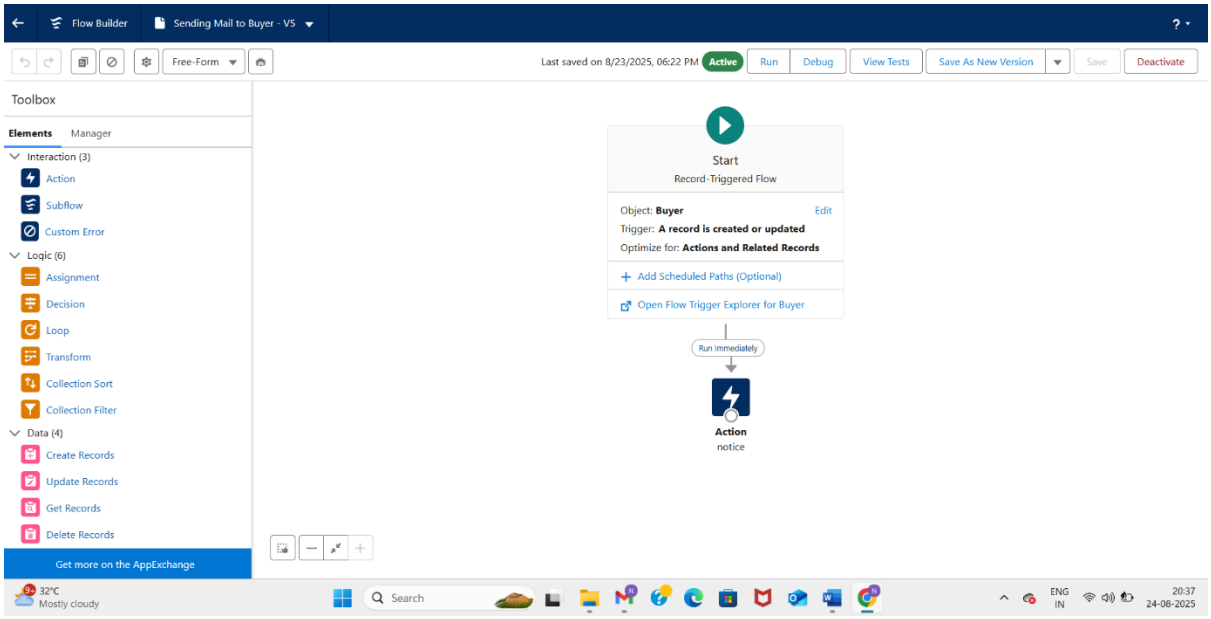
Click done.

Now drag the path form the start to action element.

Click on save. Give the Flow label , Flow Api name will be auto populated.

And click save, and click on activate.

- After following these steps, the Flow will be created as shown in the image below.



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.

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:-**

Use Case : This use case works for the Fuel details Object and Gas Station Object were The Fuel details data is important for us. So, Before deleting the records we need to the text. And Another scenario is like Fuel price should be greater than 50 rupees In Gas Station Object.

- ✓ Login to the respective trailhead account and navigate to the gear icon in the top right corner.
- ✓ Click on the Developer console. Now you will see a new console window.
- ✓ In the toolbar, you can see FILE. Click on it and navigate to new and create New apex class.

- Name the class as FuelRecordHandler.

Code:-

```
public class FuelRecordHandler {

   public static void beforeDeleteInfo(list<Fuel_details__c> fuelList){

      //fuelList = [select Id from Fuel_details__c];

      for(Fuel_details__c ful : fuelList){

         if(ful.Fuel_supplied__c > 500){

            ful.addError('you cannot delete the fuel details record because it is
associated with supplier and Gas station records');

         }

      }

   }
```

```
    public static void beforeDeleteGas(list<Gas_Station__c> gasList){

        //fuelList = [select Id from Fuel_details__c];

        for(Gas_Station__c gas : gasList){

            if(gas.Fuel_price_liter__c <= 50){

                gas.addError('enter the fuel price before saving the record, Minimum price
should be 50');

            }

        }

    }

}
```

Trigger Handler :

How to create a new trigger :

While still in the trailhead account, navigate to the gear icon in the top right corner.

Click on developer console and you will be navigated to a new console window.

Click on File menu in the tool bar, and click on new, Trigger.

Enter the trigger name and the object to be triggered.

Syntax For creating trigger:

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

{

}

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

- **Handler for the Fuel details Object:-**

Code:

```
trigger beforeDelete on Fuel_details__c  (before Delete) {
```

```
    if(trigger.isbefore && trigger.isDelete){
        FuelRecordHandler.beforeDeleteInfo(trigger.old);
    }
}
```

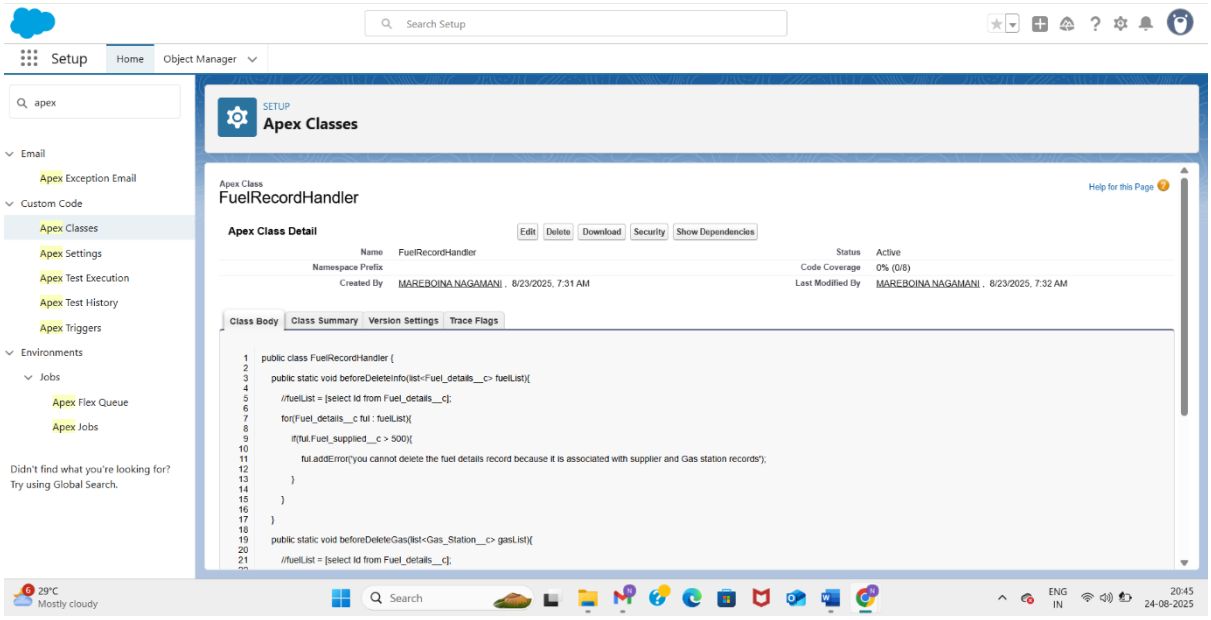
- **Handler for the Gas Station Object:-**

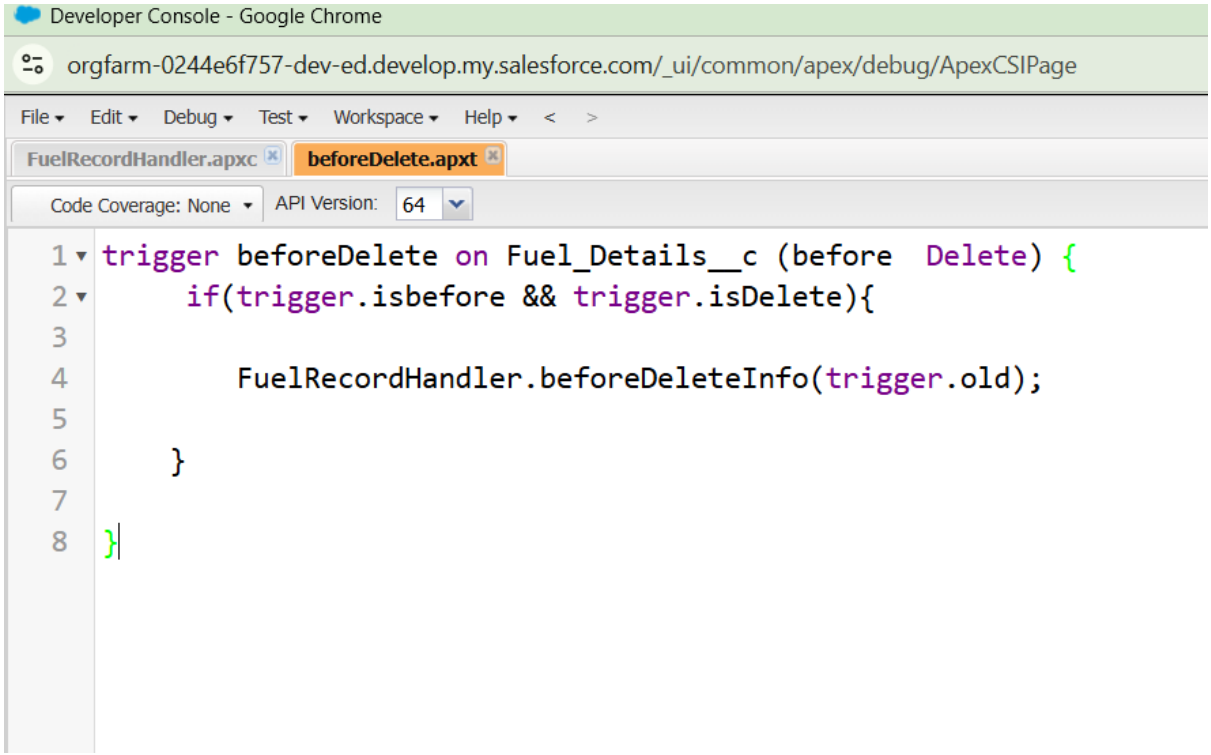
Code:-

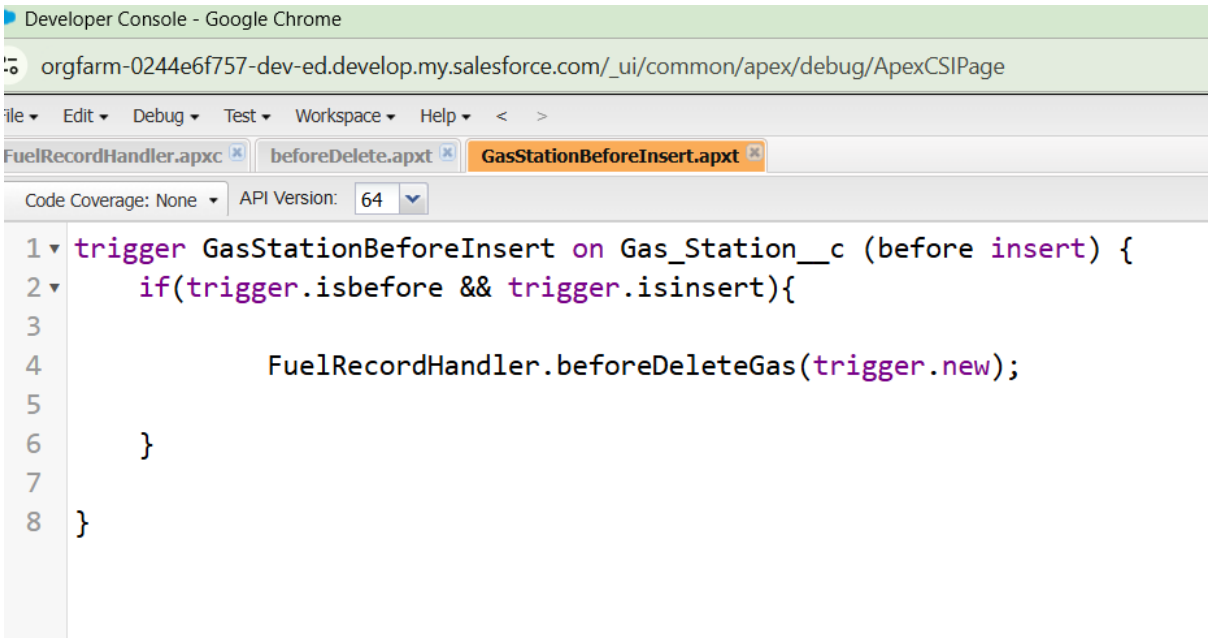
```
trigger beforeInsert on Gas_Station__c (before insert ) {
    if(trigger.isbefore && trigger.isinsert){
        FuelRecordHandler.beforeDeleteGas(trigger.new);
    }

}
```

➢ After following these steps, the Apex Triggers will be created as shown in the image below.




Developer Console - Google Chrome
orgfarm-0244e6f757-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help < >
FuelRecordHandler.apxc
beforeDelete.apxt
Code Coverage: None
API Version: 64
1 trigger beforeDelete on Fuel_Details__c (before Delete) {
2 if(trigger.isbefore && trigger.isDelete){
3
4 FuelRecordHandler.beforeDeleteInfo(trigger.old);
5
6 }
7
8 }


Developer Console - Google Chrome
orgfarm-0244e6f757-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help < >
FuelRecordHandler.apxc
beforeDelete.apxt
GasStationBeforeInsert.apxt
Code Coverage: None
API Version: 64
1 trigger GasStationBeforeInsert on Gas_Station__c (before insert) {
2 if(trigger.isbefore && trigger.isinsert){
3
4 FuelRecordHandler.beforeDeleteGas(trigger.new);
5
6 }
7
8 }


thank
you