

# **A CRM APPLICATION FOR WHOLESALE RICE MILL**

By

S.Deepika

2111cs030027@mallareddyuniversity.ac.in

## Project

**Abstract Title:** Comprehensive CRM Application for Wholesale Rice Mill.

This project involves the development of a comprehensive Customer Relationship Management (CRM) application tailored for a wholesale rice mill using Salesforce. The primary goal was to create an integrated system that enhances operational efficiency, data management, and customer relations within the rice mill factory. The application features custom objects such as Consumers, Rice Mill, Rice Details, and Suppliers, each with its own dedicated tab for streamlined access. A custom Lightning App was designed to unify these components into a cohesive user experience, integrating various objects with relevant fields and page layouts.

Profiles and role hierarchies were meticulously configured to establish clear access levels and permissions, ensuring that different user roles—from owners to workers—have appropriate visibility and control over the data. User accounts were created and assigned profiles, with permission sets used to extend access rights as needed, ensuring tailored access to specific functionalities.

The application's reporting and dashboard capabilities deliver actionable insights into key metrics such as daily sales volumes, revenue, inventory levels, and customer purchasing patterns. These features facilitate real-time data analysis, aiding in strategic decision-making and operational planning. Custom Apex code was developed to address complex business logic, automate critical workflows, and manage sales transactions and inventory processes efficiently. Validation rules and cross-object formula fields were implemented to ensure data accuracy, with ISBLANK formulas enforcing mandatory field completion. This Salesforce CRM application thus provides a robust and scalable solution for managing the rice mill's operations, improving operational efficiency, and enhancing customer engagement, while also enabling comprehensive data analysis and reporting for strategic business growth.

Additionally, the integration with external systems for data import and export was set up to facilitate seamless synchronization with existing databases and third-party applications. This ensures that the CRM system operates as a central hub for all business processes, providing a comprehensive view of operations and customer interactions. This Salesforce CRM application thus provides a robust and scalable solution for managing the rice mill's operations, improving operational efficiency, and enhancing customer engagement, while also enabling comprehensive data analysis and reporting for strategic business growth.

**Keywords:** Salesforce, CRM Application, Custom Objects, Lightning App, Fields, Page Layouts, Profiles, Role Hierarchy, Permission Sets, Reports, Dashboards, Apex, Validation Rules.

## **INDEX Page**

| <b>Topics</b> |                            | <b>Page no</b> |
|---------------|----------------------------|----------------|
| Topic no 1    | Introduction to Salesforce | 04             |
| Topic no 2    | Object                     | 05             |
| Topic no 3    | Tabs                       | 07             |
| Topic no 4    | The Lightning App          | 08             |
| Topic no 5    | Fields                     | 10             |
| Topic no 6    | Page Layouts               | 16             |
| Topic no 7    | Profiles                   | 18             |
| Topic no 8    | Role & Role Hierarchy      | 20             |
| Topic no 9    | Users                      | 22             |
| Topic no 10   | Permission Sets            | 25             |
| Topic no 11   | Report                     | 25             |
| Topic no 12   | Dashboards                 | 28             |
| Topic no 13   | Apex                       | 30             |

## Task 1: Introduction to Salesforce

### 1.1 Creating Developer Account Creating a developer org in salesforce.

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

First name &

Last name Email

Role :

Developer Company :

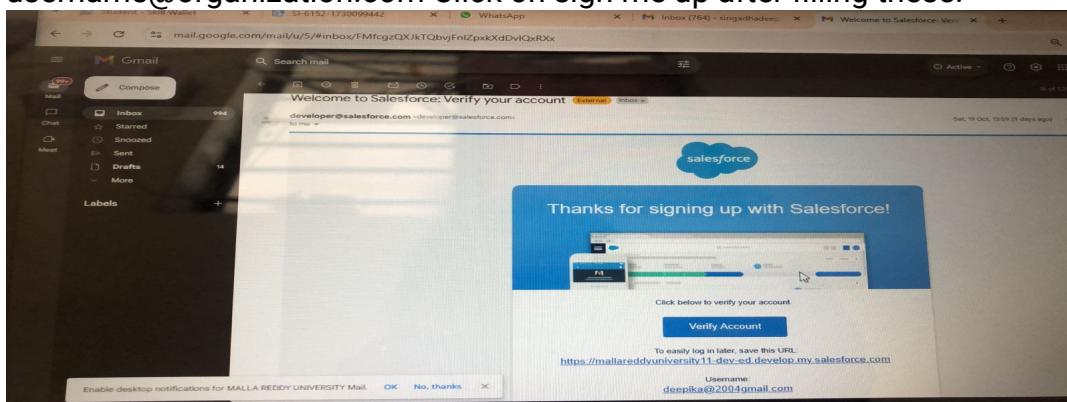
College Name County :

India Postal Code :

pin code Username :should be a combination of your name and company

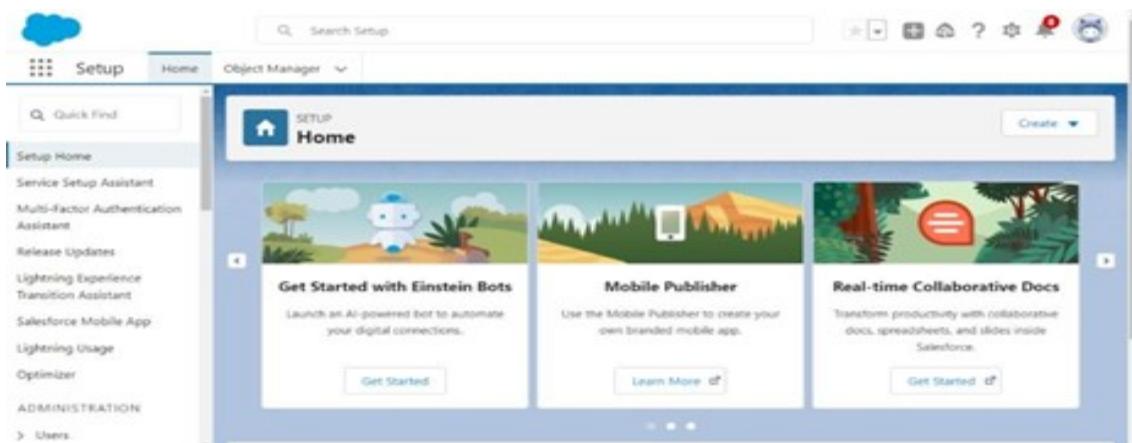
This need not be an actual email id, you can give anything in the format:

username@organization.com Click on sign me up after filling these.



### 1.2. Account Activation

Then you will redirect to your salesforce setup page.



## Task 2: Object

## **Salesforce objects are of two types:**

Standard Objects:

Standard objects are the kind of objects that are provided by salesforce.com such as users, contracts, reports, dashboards, etc.

Custom Objects:Custom objects are those objectsthat are created by users. They supply information that is unique and essential to their organization. They are the heart of any application and provide a structure forsharing data.

To Navigate to Setup page:

Click on gear icon -click setup.

To create an object:

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

2. On Custom object defining page:

3. Enter the label name, plural label name, click on Allow reports, Allow search.

4. Click onSave.

### **2.1. Create Supplier Object To create an object:**

From the setup page >> Click on Object Manager>> Click on Create>>Click on Custom Object.

1. Enter the label name>>supplier

2. Plural label name>>supplier

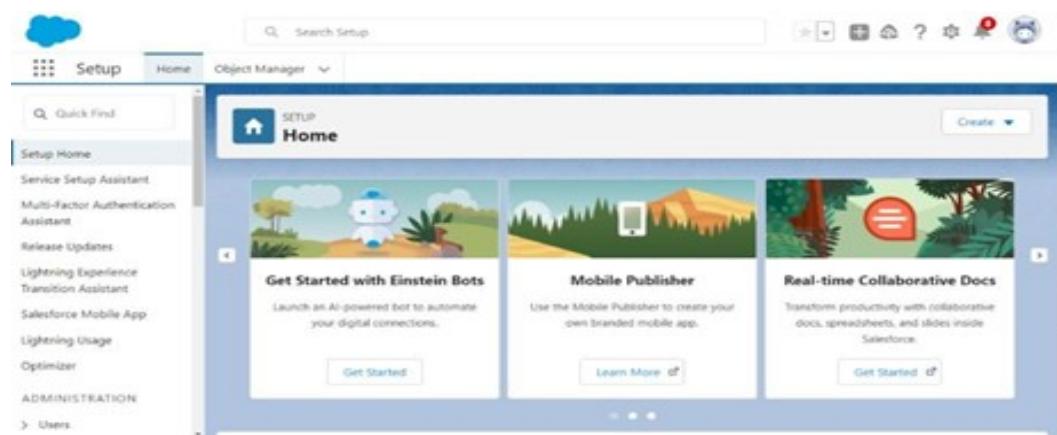
3. Enter Record Name Label and Format

• Record Name >> supplier Name

• Data Type>>Text

4. Click on Allow reports and Track Field History and allow search

5. Allow search >> Save.

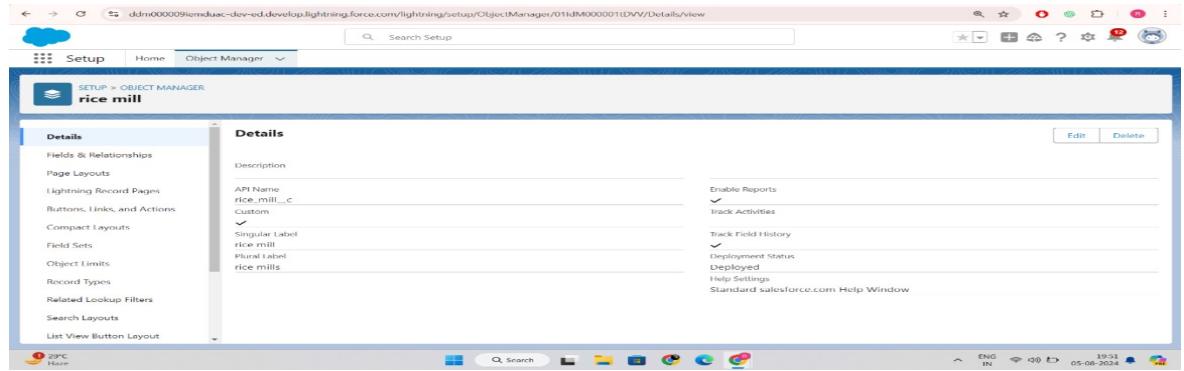


### **2.2Create Rice mill Object**

To create an object:

From the setup page >> Click on Object Manager>>Click on Create >> Click on Custom Object.

- i. Enter the label name>>rice mill
- ii. Plural label name>> rice mills
- iii. Enter Record Name Label and Format
  1. Record Name >>
  2. Data Type >> Auto Number
  3. Display Format >>rice-{000}
  4. Starting number >>1

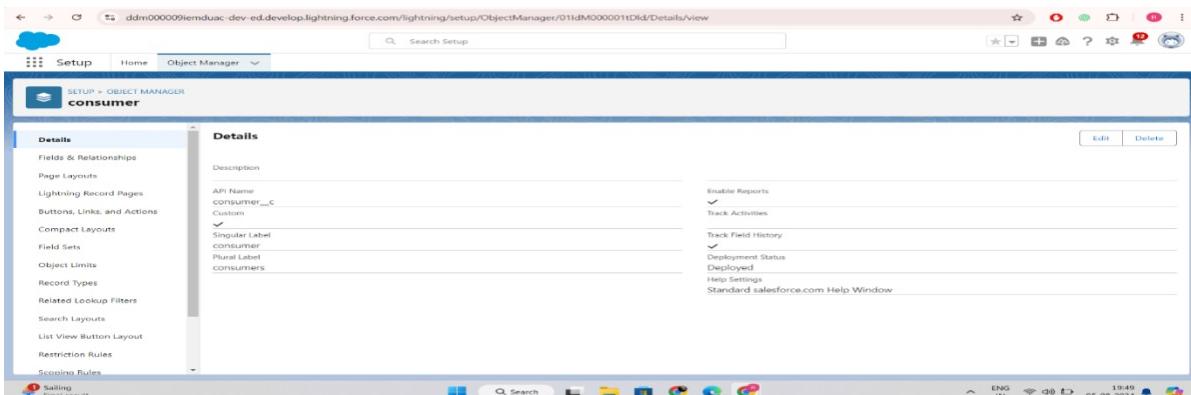


4 Click on Allow reports and Track Field History, Allow Search and Save.

## 2.3 Create consumer Objects

Note: Follow the same steps as mentioned in Activity 2 for the and Receipt objects.

- i. Use these display format for the consumer
  - label name >> consumer
  - Plural label name >> consumers
  - Display Format >> consumers-{000}



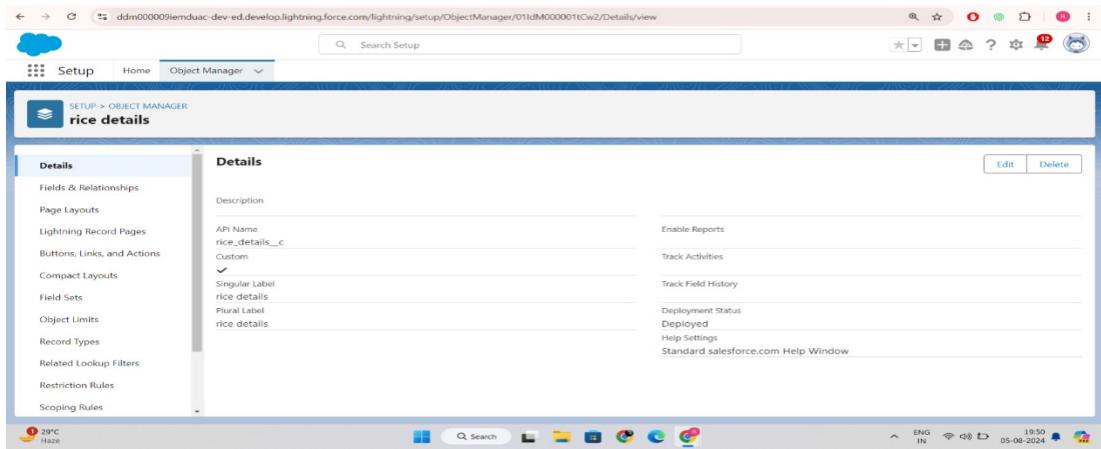
- Starting number >> 1

## 2.4 Create rice details Objects

1. Use these display format for the rice details

- a. label name >> rice details

- b. Plural label name >> rice details
- c. Display Format >> rice-{000}
- d. Starting Number >>1



## Task 3 : Tabs

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:

#### 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.

#### 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.

#### 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.

#### Lightning Component Tabs

Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app.

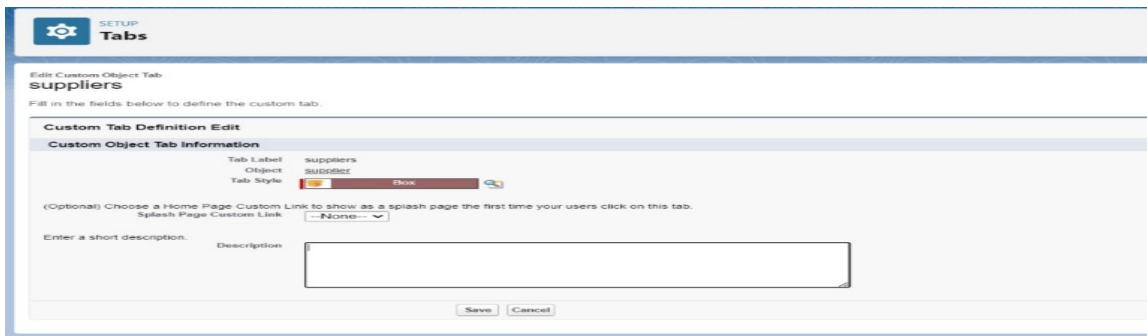
#### Lightning Page Tabs

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 customize the tabs for your apps.

### 3.1 Creating a Custom Tab

- To create a Tab:(supplier)
- Go to setup page >> type Tabs in Quick Find bar >> click on tabs >> New (under custom object tab)
- Select Object(supplier) >> Select the tab style >> Next (Add to profiles page) keep it as default >> Next(Add to Custom App) uncheck the include tab .
- Make sure that the Append tab to users' existing personal customizations is checked.

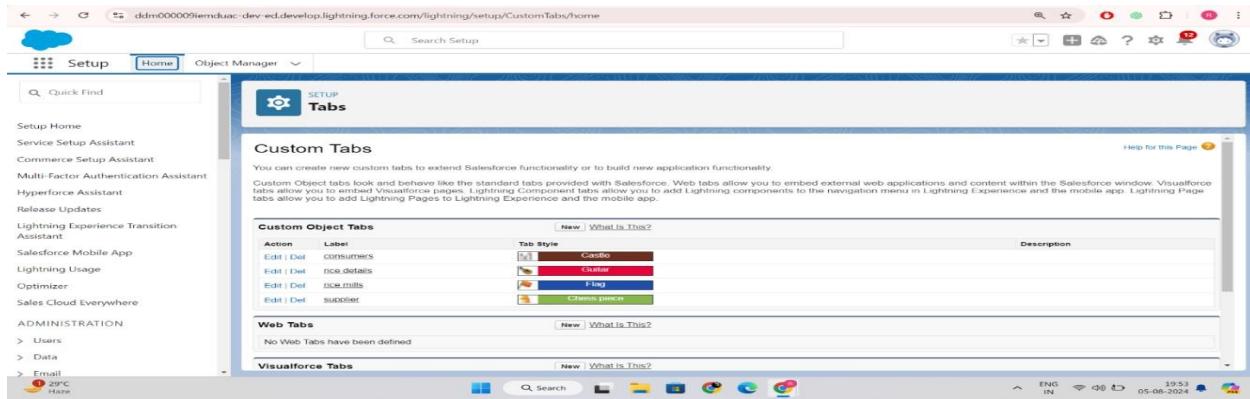


Click save.

### 3.2Creating Remaining Tabs

Now create the Tabsfor the remaining Objects,they are “ rice mill,consumer, rice details” .

Follow the same stepsas mentioned in Activity -1 .



## Task 4 : The Lightning App

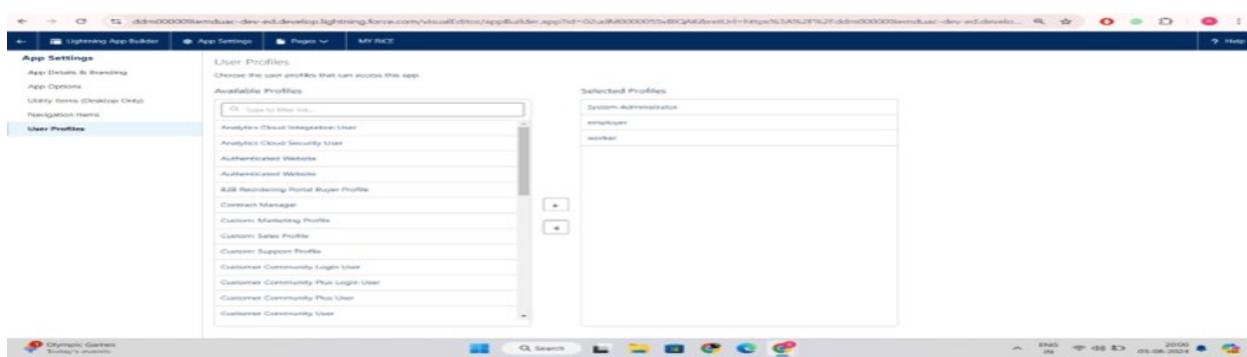
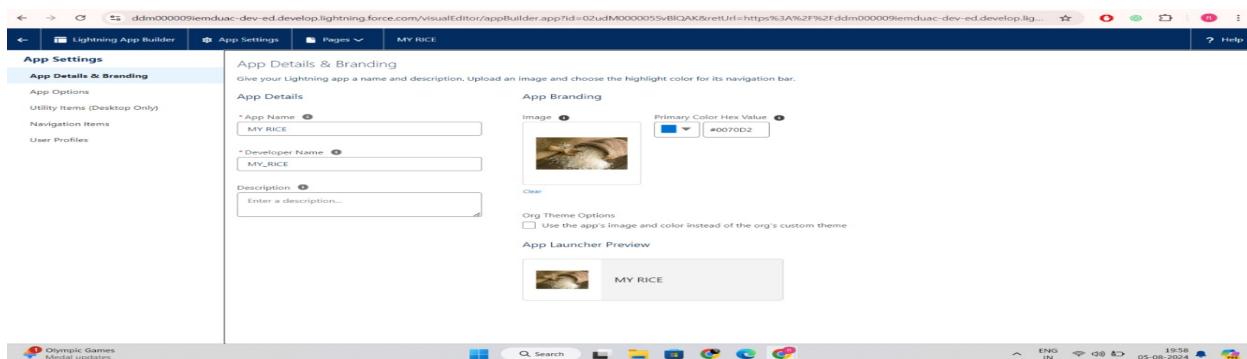
An app is a collection of items that work together to serve a particular function. In Lightning Experience, Lightningapps give your users access to sets of objects, tabs, and other items all in one

convenient bundle in the navigationbar.

Lightning apps let you brand your apps with a customcolor 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 switchingbetweenapps.

## 4.1Create a LightningApp

- To create a lightning app page:
- Go to setup page >> search “app manager” in quick find >> select “app manager”>> click on Newlightning App.
- Fill the app name in app details as MY RICE >>Next >> (App option page) keep it as default>>Next >>(Utility Items) keep it as default >> Next.
- Upload a photo that is related to your app.
- To add Navigation Item:
- Select the items (supplier, rice mill, consumer,Rice details ) from the search bar and move it usingthe arrow button >> Next.
- 
- To Add User Profiles:Search profiles (System administrator) in the searchbar >> click on the arrow button>> save & finish.



## Task 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

#### StandardFields:

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,

1. Created By
2. Owner
3. Last Modified
4. 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 organization 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.

### 5.1 Creating the number field in rice details object

Creating the number field in rice details object

- i. Go to the setup page >> click on object manager >> From drop down click edit for rice details object
- ii. Click on fields & relationship >> click on New.
- iii. Select Data type as "Number" and click Next.
- iv. Given the Field Label as "rice distributed" and length as "5".

| FIELD LABEL       | FIELD NAME         | DATA TYPE                | CONTROLLING FIELD | INDEXED |
|-------------------|--------------------|--------------------------|-------------------|---------|
| Created By        | CreatedBy          | Lookup(P>User)           |                   |         |
| Last Modified By  | LastModifiedBy     | Lookup(P>User)           |                   |         |
| rice details Name | Name               | Auto Number              |                   | ✓       |
| rice distributed  | rice_distributed_c | Number(0, 0)             |                   |         |
| rice mill 1(one)  | rice_mill_1_one_c  | Master-Detail(rice mill) |                   | ✓       |
| supplier Name     | supplier_Name_c    | Master-Detail(supplier)  |                   | ✓       |

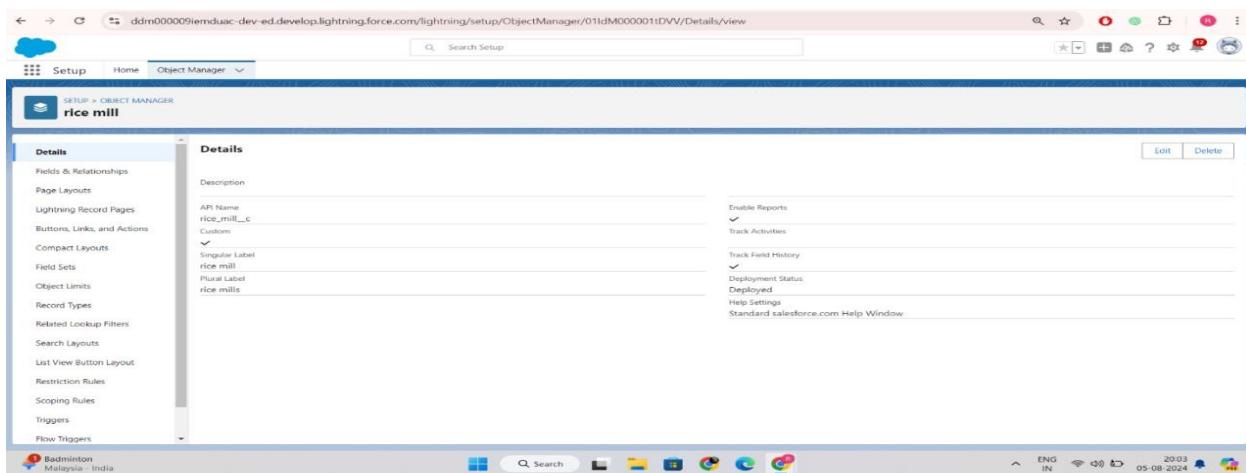
V. Field Name will be auto populated, and click on Next- Next >> Save.

## 5.2 Creating JunctionObject

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

Creating junction object as rice details with supplier& rice millTo create junctionobject

1. Go to the setup page >> click on object manager >> From drop down click edit for rice details
- objectClick on fields & relationship - click on New.
- 2.
3. Select “Master-Detail relationship” as data type and click Next.
4. Select the related object “ supplier ” and click next.
5. Give Field Label as “supplier Name” and click Next.
6. Next >> Next >>Save & New.
- 7.
8. Follow the same steps from 1 to 3.
9. Select the related object “ rice mill ” and click Next.
10. Give Field Label as “rice mill 1(one)” and click Next.
11. Next >> Next >> Save.



## 5.3 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 consumer & rice mill Object To Create a Master-Detail relationship

1. Go to the setup page >> click on objectmanager >> From drop down click edit for consumerobject.
2. Click on fields & relationship >> click on New.
3. Select “Master-Detail relationship” as data type and click Next.
4. Select the related object “rice mill”.
5. Give Field Label as “ricemill name” and click Next.
6. Next >> Next >> Save.

## 5.4 Creating the Roll-up summaryfield on supplier & rice mill Objects.

1. Go to setup >> click on Object Manager >> type object name(supplier) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select the data type as “Rollup summary”, and click Next.
4. Give the Fieldlabel as “sum of rice distributed”, Field Name will be Auto generated, and click Next.
5. Select the summarized object as “rice details”.

6. Select the Rolluptype as “sum”.
7. Select the field to aggregate as “rice distributed”, and click Next >>Next >>Save.
8. Follow the same steps for the rice mill Object from 1 to 3
9. Give the Fieldlabel as “rice distributed to shops”, Field Name will be Auto generated, and clickNext.
10. Select the summarized object as “rice details”.
11. Select the Rolluptype as “sum”.
12. Select the field to aggregate as “rice distributed”, and click Next >> Next >> Save.
13. Note :create the field as “rice taken by shops in kgs” using number datatype in consumer object
14. Follow the same steps for the rice mill Object from 1 to 3
15. Give the Field label as “rice taken”, Field Name will be Auto generated, and click Next.
16. Select the summarized object as “consumer”.
17. Select the Rolluptype as “sum”.
18. Select the field to aggregate as “rice taken in shops”, and click Next >> Next >>Save.

| FIELD LABEL                | FIELD NAME                             | DATA TYPE                | CONTROLLING FIELD | INDEXED |
|----------------------------|--|--------------------------|-------------------|---------|
| Amount Paid                | Amount_Paid__c                         | Formula (Number)         |                   |         |
| consumer Name              | Name                                   | Auto Number              |                   |         |
| Consumer Name              | Consumer_Name__c                       | Formula (Text)           |                   |         |
| Created By                 | CreatedById                            | Lookup(User)             |                   |         |
| email                      | email__c                               | Email                    |                   |         |
| First name                 | First_name__c                          | Text(20)                 |                   |         |
| Last Modified By           | LastModifiedById                       | Lookup(User)             |                   |         |
| Last name                  | Last_name__c                           | Text(20)                 |                   |         |
| Mode of payment            | Rice_type_Picklist_values_1_basmati__c | Picklist                 |                   |         |
| Phone number               | Phone_number__c                        | Phone                    |                   |         |
| rice mill name             | rice_mill_name__c                      | Master-Detail(rice mill) |                   |         |
| Rice taken by shops        | Rice_taken_by_shops__c                 | Number<0, 0>             |                   |         |
| rice taken by shops in kgs | rice_taken_by_shops_in_kgs__c          | Number(18, 0)            |                   |         |
| Rice type                  | Rice_type__c                           | Picklist                 |                   |         |

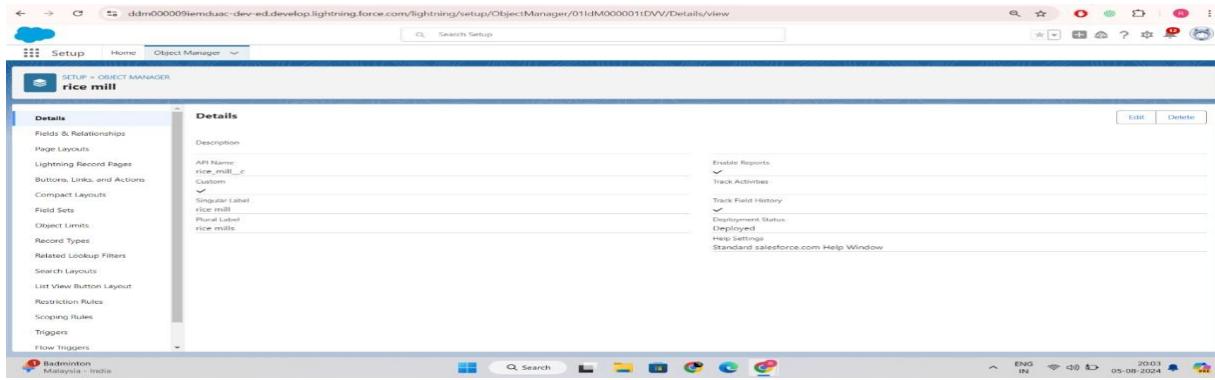
## 5.5 Creating the number field in rice details object

1. Go to the setup page >> click on object manager >> From drop down click edit for rice details object.
2. Click on fields & relationship >> click on New.

3. Select Data type as “masterdetail” and click Next.
4. Given the Field Label as “ supplier name ” and length as “ 5
5. Field Name will be auto populated, and click on Next>> Next >>Save.

## 5.6 Creating Fields in rice mill Objects

1. Select Data type as “Number” and click Next.
2. Given the Field Label as “ rice price/kg” and length as “ 5 ”



## 5.7 Creating Fields in consumer Objects

| Fields & Relationships    |                              |                          |                   |         |  |
|---------------------------|------------------------------|--------------------------|-------------------|---------|--|
| FIELD LABEL               | FIELD NAME                   | DATA TYPE                | CONTROLLING FIELD | INDEXED |  |
| Amount Paid               | Amount_Paid__c               | Formula (Number)         |                   |         |  |
| Consumer Name             | Consumer_Name__c             | Auto Number              |                   |         |  |
| Created By                | CreatedById                  | Formula (Text)           |                   |         |  |
| Email                     | email__c                     | Lookup(Email)            |                   |         |  |
| Last Name                 | last_name__c                 | Text(20)                 |                   |         |  |
| Last Modified By          | LastModifiedById             | Text(20)                 |                   |         |  |
| Last Name                 | LastName__c                  | Text(20)                 |                   |         |  |
| Mode of payment           | Mode_of_payment__c           | Text                     |                   |         |  |
| Phone number              | Phone_number__c              | Phone                    |                   |         |  |
| rice mill name            | rice_mill_name__c            | Master-Detail(rice mill) |                   |         |  |
| Rice taken by shop        | Rice_taken_by_shop__c        | Number(5, 0)             |                   |         |  |
| Rice taken by shops in kg | Rice_taken_by_shops_in_kg__c | Number(15, 0)            |                   |         |  |
| rice type                 | rice_type__c                 | Text                     |                   |         |  |

## 5.8 Creating Cross Object Formula Field in consumerObject

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.

Note :check whether the fields mentioned in the formula field are created or not , if not go to activity 9 and create those fields mentioned in consumerobject.

1. Go to setup >> click on Object Manager >> type object name(consumer) 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 “Amount Paid ” and select formula return type as “Number” and click next.
5. Insert fields formula should be :
6. `rice_taken_by_shops_c * rice_mill_name__r.rice_price_kg_c`
7. Under Advanced Formula write down the formula and click “Check Syntax” and Save.



## 5.9 Creating the validation rule

Improve the quality of your data using validation rules. Validation rules verify that the data a user enters in a record meets the standards you specify before the user can save the record. A validation rule can contain a formula or expression that evaluates the data in one or more fields and returns a value of “True” or “False”. Validation rules also include an error message to display to the user when the rule returns a value of “True” due to an invalid value.

Creating the validation rule for phone number field in consumerobject

Note :check whether the fields mentioned in the formula field are created or not , if not go to activity 9 and create those fields mentioned in consumerobject.

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

The screenshot shows the Salesforce Object Manager interface for the 'consumer' object. On the left, a sidebar lists various configuration options: 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, and Restriction Rules. The main content area is titled 'Validation Rules' and displays one item: 'phonenumeroremailblankrule'. The table columns are RULE NAME, ERROR LOCATION, ERROR MESSAGE, ACTIVE, and MODIFIED BY. The rule name is 'phonenumeroremailblankrule', error location is 'Top of Page', error message is 'please fill phone number', it is active, and it was modified by 'udeyashri.yelagandula' on '05/07/2023, 12:57 pm'.

- i. Enter the Rule name as "Phonenumberoremailblankrule"
- ii. Enter the description as "phone number and email number should not be blank"
- iii. Enter the formula as "OR( ISBLANK( phone\_number\_c ) , ISBLANK( email\_c ) )" and check the syntax.
- iv. Under the error message write as "please fill in your phone number."
- v. Select error location "top of page".
- vi. Save the validation rule.

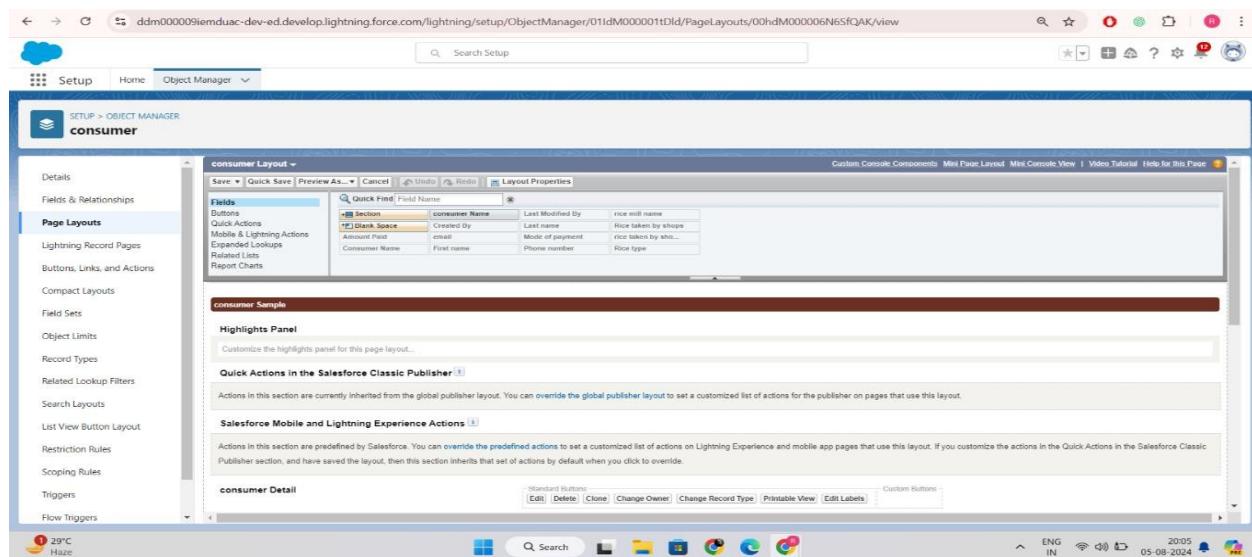
## Task 6 : Page Layouts

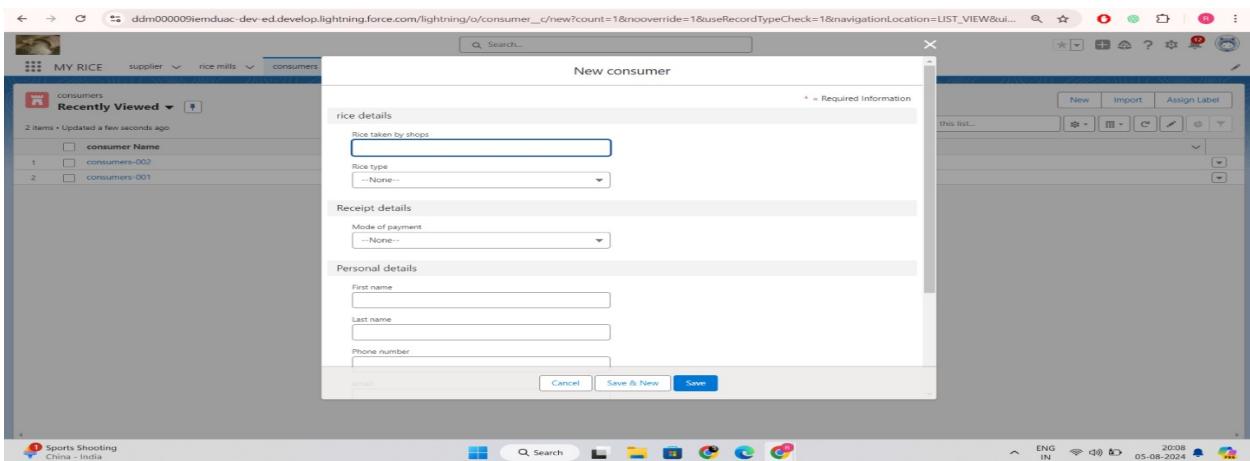
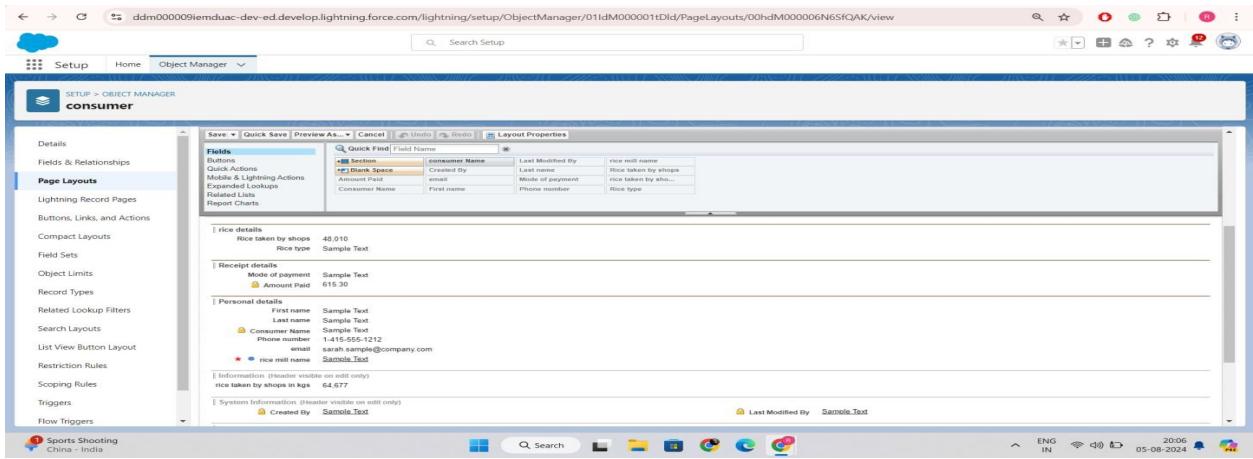
Page Layout in Salesforce allows us to customize the design and organize detail and edit pages of records in Salesforce. Page layouts can be used to control the appearance of fields, related lists, and custom links on standard and custom objects' detail and edit pages.

### 6.1 creating the page layout

- i. To Create a Page layout:
- ii. Go to Setup >> Click on Object Manager >> Search for the object (consumer) >> From drop down select the object and click on it.
- iii. Click on Page layout >> Click on New.
- iv. Select the existing page layout, and give the page layout name as "consumer layout", and click save.

- v. Drag and drop the sectionfield to consumer detailsand create the section.
- vi. Enter the section name as “Personal details”, - click Ok.
- vii. Now drag the fields to this section that mentioned , they are
- viii. First name , last name , consumer name , phone number, email, rice mill name.
- ix. Follow the same process for another two sections as shown above , they are
- x. One section is “ rice details ” , drag the fields that are
- xi. Rice taken by shop, rice type.
- xii. Another section is “Receipt details ”, and drag the fields that are Mode of payment , Amount paid. Then , Click save.





## Task 7 : Profiles

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 SystemAdministrator, 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 systemAdministrator. We cannot deleted standard ones

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

## 2. Custom Profiles:

Custom ones defined by us.

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

### 7.1 owner Profile

To create a new profile:

Go to setup >>type profiles in quick find box >> click on profiles >> clone the desired profile (StandardUser)>>enter profile name (owner) >>Save.

Scrolldown to Custom Object Permissions and Give access permissions for consumers, rice details , ricemill and suppliers objects as mentionedin the below diagram.

| Custom Object Permissions |                                     |                                     |                                     |                                     |                                     |                                     |                                     |                                     |  |
|---------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--|
|                           | Basic Access                        |                                     |                                     |                                     | Data Administration                 |                                     |                                     |                                     |  |
|                           | Read                                | Create                              | Edit                                | Delete                              | View All                            | Modify All                          |                                     |                                     |  |
| Assets                    | <input type="checkbox"/>            |  |
| Asset Services            | <input type="checkbox"/>            |  |
| books                     | <input type="checkbox"/>            |  |
| books                     | <input type="checkbox"/>            |  |
| Brokers                   | <input type="checkbox"/>            |  |
| consumers                 | <input checked="" type="checkbox"/> |  |
| Employees                 | <input type="checkbox"/>            |  |
| energy audits             | <input type="checkbox"/>            |  |
| item details              | <input type="checkbox"/>            |  |
| nick names                | <input type="checkbox"/>            |  |
| positions                 | <input type="checkbox"/>            |  |
| Projects                  | <input type="checkbox"/>            |  |
| ProjectTasks              | <input type="checkbox"/>            |  |
| Properties                | <input type="checkbox"/>            |  |
| purchasers                | <input type="checkbox"/>            |  |
| reviews                   | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |  |
| rice details              | <input checked="" type="checkbox"/> |  |
| rice mills                | <input checked="" type="checkbox"/> |  |
| SolarBots                 | <input type="checkbox"/>            |  |
| SolarBot Status           | <input type="checkbox"/>            |  |
| studis                    | <input type="checkbox"/>            |  |
| students                  | <input type="checkbox"/>            |  |
| super marts               | <input type="checkbox"/>            |  |
| suppliers                 | <input checked="" type="checkbox"/> |  |
| teachers                  | <input type="checkbox"/>            |  |
| tickets                   | <input type="checkbox"/>            |  |
| vendors                   | <input type="checkbox"/>            |  |

Give access and save it.

### 7.2 employer Profile

1. Go to setup >> type profiles in quick find box >>click on profiles >> clone the desired profile(Standard Platform User) >> enter profile name (employer) >> Save.
2. While still on the profile page, then click Edit.
3. Select the Custom App settings as default for the rice mill
4. Scroll down to Custom Object Permissions and Give access permissions for consumer,rice details ,rice mill and suppliersobjects as mentioned in the below diagram.
5. And click save.

The screenshot shows the 'Profiles' section in the Salesforce setup. It displays two separate tables of permissions for different objects:

| Object         | Basic Access                        |                          |                          |                          |                          |                            | Data Administration |  |
|----------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|----------------------------|---------------------|--|
|                | Read                                | Create                   | Edit                     | Delete                   | <a href="#">View All</a> | <a href="#">Modify All</a> |                     |  |
| Assets         | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| Asset Services | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| books          | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| books          | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| Brokers        | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| consumers      | <input checked="" type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| Employees      | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| energy audits  | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| item details   | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| nick names     | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| positions      | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| Projects       | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| ProjectTasks   | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| Properties     | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |

| Object          | Basic Access                        |                          |                                     |                          |                          |                            | Data Administration |  |
|-----------------|-------------------------------------|--------------------------|-------------------------------------|--------------------------|--------------------------|----------------------------|---------------------|--|
|                 | Read                                | Create                   | Edit                                | Delete                   | <a href="#">View All</a> | <a href="#">Modify All</a> |                     |  |
| purchasers      | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| reviews         | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| rice details    | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| rice mills      | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| SolarBots       | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| SolarBot Status | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| studs           | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| students        | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| super marts     | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| suppliers       | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| teachers        | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| tickets         | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| vendors         | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |

### 7.3 worker Profile

1. Go to setup >>type profiles in quick find box >>click on profiles>> clone the desired profile(StandardPlatform User) >> enter profile name (worker) >> Save.
2. While still on the profile page, then click Edit.
3. Select the Custom App settings as default for the rice mill.
4. Scroll down to CustomObject Permissions and Giveaccess permissions for consumer, rice details , ricemill and suppliers objects as mentioned in the below diagram.

The screenshot shows the 'Profiles' section in the Salesforce setup. It displays two separate tables of permissions for different objects, similar to the previous screenshot but with different checkbox states:

| Object         | Basic Access                        |                                     |                          |                          |                          |                            | Data Administration |  |
|----------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|----------------------------|---------------------|--|
|                | Read                                | Create                              | Edit                     | Delete                   | <a href="#">View All</a> | <a href="#">Modify All</a> |                     |  |
| Assets         | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| Asset Services | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| books          | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| books          | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| Brokers        | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| consumers      | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| Employees      | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| energy audits  | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| item details   | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| nick names     | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| positions      | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| Projects       | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |

| Object          | Basic Access                        |                                     |                          |                          |                          |                            | Data Administration |  |
|-----------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|----------------------------|---------------------|--|
|                 | Read                                | Create                              | Edit                     | Delete                   | <a href="#">View All</a> | <a href="#">Modify All</a> |                     |  |
| purchasers      | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| reviews         | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| rice details    | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| rice mills      | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| SolarBots       | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| SolarBot Status | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| studs           | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| students        | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| super marts     | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| suppliers       | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| teachers        | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |
| tickets         | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/>   |                     |  |

5. And click save.

## Task 8 : Role &Role Hierarchy

A role in Salesforce defines a user's visibility access at the record level. Roles may be used to specify the types of access that people in your Salesforce organization can have to data. Simply put, it describes what a user could see within the Salesforce organization.

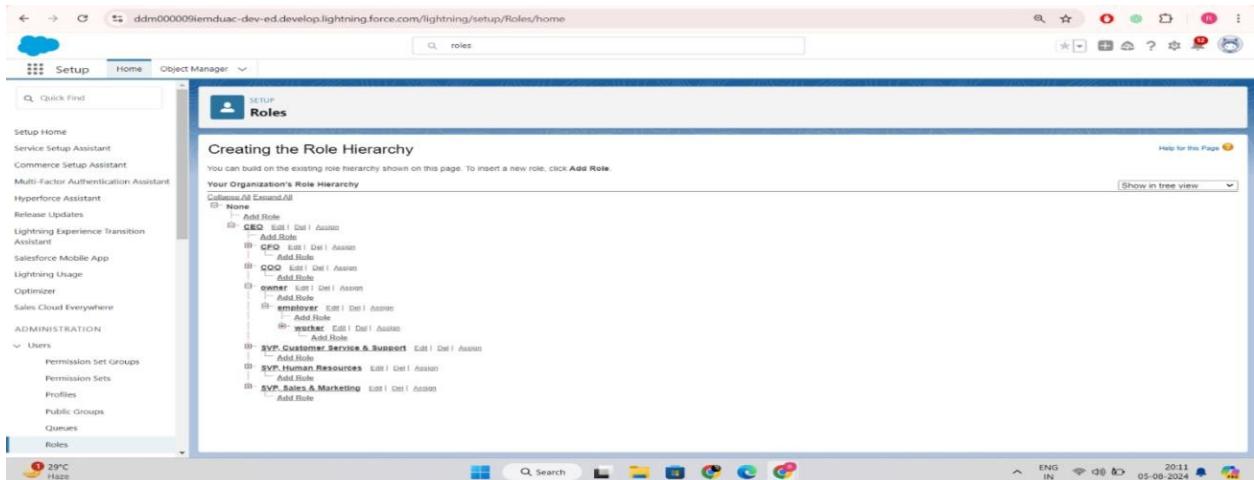
## **8.1 Creating owner Role**

1. Creating owner Role:
2. Go to quick find >> Search for Roles >> click on set up roles.
3. Go to quick find >> Search for Roles >> click on set up roles.
4. Click on Expand All and click on add role under whom this role works.
5. Give Label as “owner” and Role name gets auto populated. Then click on Save.

The screenshot shows a 'Role Edit' page titled 'New Role'. It has fields for 'Label' (containing 'owner'), 'Role Name' (containing 'owner'), and 'This role reports to' (containing 'CEO'). At the bottom are 'Save', 'Save & New', and 'Cancel' buttons.

## **8.2 Creating employer roles**

- a. Creating another two roles under manager
- b. Go to quick find >>Search for Roles >>click on set up roles.
- c. Click plus on CEO role, and click add role under owner.
- d. Give Label as “employer” and Role name gets auto populated. Then click on Save.
- e. Repeat the same steps, for another role.
- f. Click plus on CEO role, and click plus on owner, and click add role under employer.
- g. Give Label as “worker” and Role name gets auto populated. Then click on Save.



## Task 9 : 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.

### 9.1 Create User

1. Go to setup >> type users in quick find box >> select users >> click New user.
2. Fill in the fields
3. First Name: vicky
4. Last Name: y
5. Alias: Give a Alias Name
6. Email id: Give your Personal Email id
7. Username: Username should be in this form: [text@text.text](mailto:text@text.text)
8. Nick Name: Give a Nickname
9. Role: owner
10. User license: Salesforce
11. Profiles: owner.
12. Save it.

The screenshot shows the 'User Edit' page for a user named 'vicky'. The page is titled 'User Edit vicky' and includes a 'General Information' section and a 'Role' section.

**General Information:**

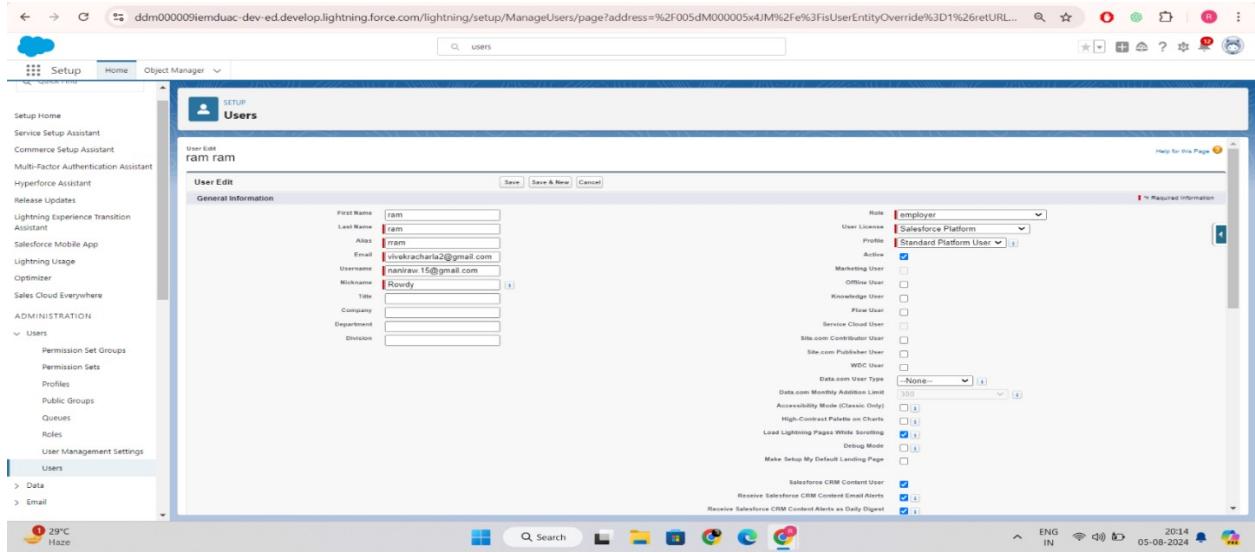
- First Name: vicky
- Last Name: y
- Alias: vy
- Email: ramesh0820@gmail.com
- Username: ramesh0820@754123gmail
- Nickname: vicky
- Title: (empty)
- Company: (empty)
- Department: (empty)
- Division: (empty)

**Role:**

- Role: owner
- User License: Salesforce
- Profile: owner
- Active: checked
- Marketing User: (checkbox)
- Offline User: (checkbox)
- Knowledge User: (checkbox)
- Flow User: (checkbox)
- Service Cloud User: (checkbox)
- Site.com Contributor User: (checkbox)
- Site.com Publisher User: (checkbox)
- WDC User: (checkbox)
- Data.com User Type: --None--
- Data.com Monthly Addition Limit: 300

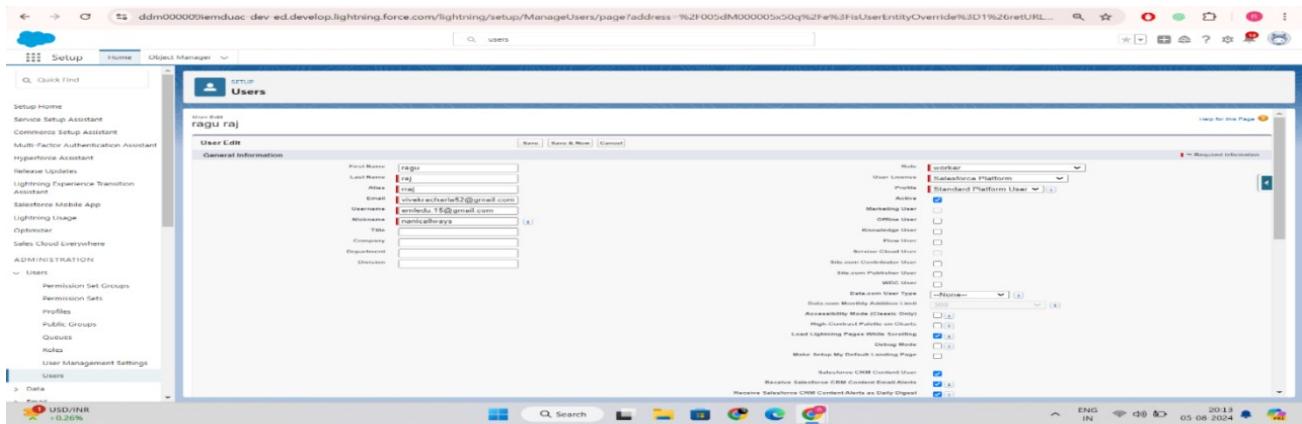
## 9.2 creating another users

1. Go to setup ? type users in quick find box ? selectusers ? click New user.
2. Fill in the fields
3. First Name: ram
4. Last Name: ram
5. Alias: Give a Alias Name
6. Email id: Give your Personal Email id
7. Username: Username should be in this form: [text@text.text](mailto:text@text.text)
8. Nick Name: Give a Nickname
9. Role: employer
10. User license: Salesforce platform
11. Profiles: standardplatform user.



### 9.3 Create Another User

1. Go to setup ? type users in quick find box ? select users ? click New user.
2. Fill in the fields
3. First Name : ragu
4. Last Name : raj
5. Alias : Give a AliasName
6. Email id : Give your Personal Email id
7. Username : Username should be in this form: [text@text.text](mailto:text@text.text)
8. Nick Name : Give a Nickname
9. Role : worker
10. User license : Salesforce platform

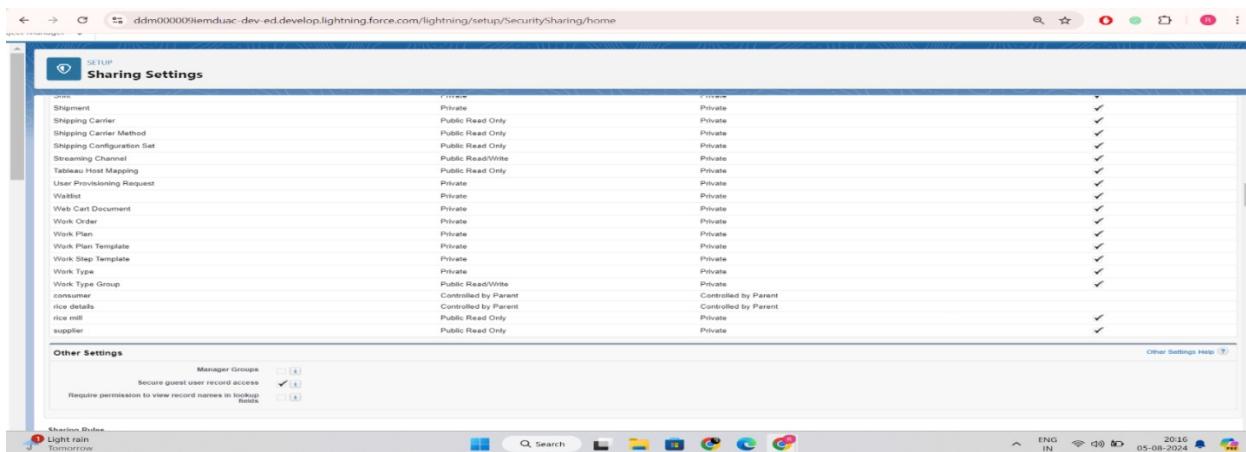


## Task 10 : Permission sets

A permission set is a collection of settings and permissions that give users access to various tools and functions. Permission sets extend users' functional access without changing their profiles and are the recommended way to manage your users' permissions.

### 10.1 Creating OWD setting.

1. Go to setup >> type "sharing settings " in quick search >> Click edit.
2. Scroll down, change the default internal access to " public read-only" for rice mill and supplier object.
3. Click save.
4. Extra information, By these every profile has their own access, according to their profile.
5. But in our case we created roles and given the roles in such a way that the owner can see employer and worker records, and the employer can see the worker records.



## Task 11 : Reports

Note : Before creating a report, create the latest "10" records in consumer objects. 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.

select for report type, search for “rice mill with consumers” click on it. And click on start report. Their outlinepane is opened already, select the fields that are mentioned below in the column section.

- a. consumer name
- b. rice type
- c. rice price/kg
- d. mode of payments
- e. amount paid
- f. Remove the unnecessary fields.
- g. Select the fields that are mentioned below in the GROUP ROWS section.
- h. Rice taken by shops
- i. Click save and run and save the report as “range of amount per day”. and save it.

The screenshot shows the Salesforce Lightning interface for the 'MY RICE' account. The top navigation bar includes links for 'supplier', 'rice mills', 'consumers', 'rice details', and a search bar. The main content area is titled 'Reports' and shows a 'Recent' section with one item: 'range of amount per day'. This report was created by Vivek Racharla on April 8, 2024, at 11:22 PM. The report details are listed in a table:

| Report Name             | Description            | Folder         | Created By         | Created On | Subscribed |
|-------------------------|------------------------|----------------|--------------------|------------|------------|
| range of amount per day | estimated rice per day | Vivek Racharla | 4/8/2024, 11:22 pm | ✓          |            |

The sidebar on the left provides navigation links for 'Reports', 'Recent', 'Folders', and 'Favorites'. The bottom of the screen shows the Windows taskbar with various application icons and system status indicators.

### 11.1 Sharing report to owner

Click edit drop down and select subscribe option

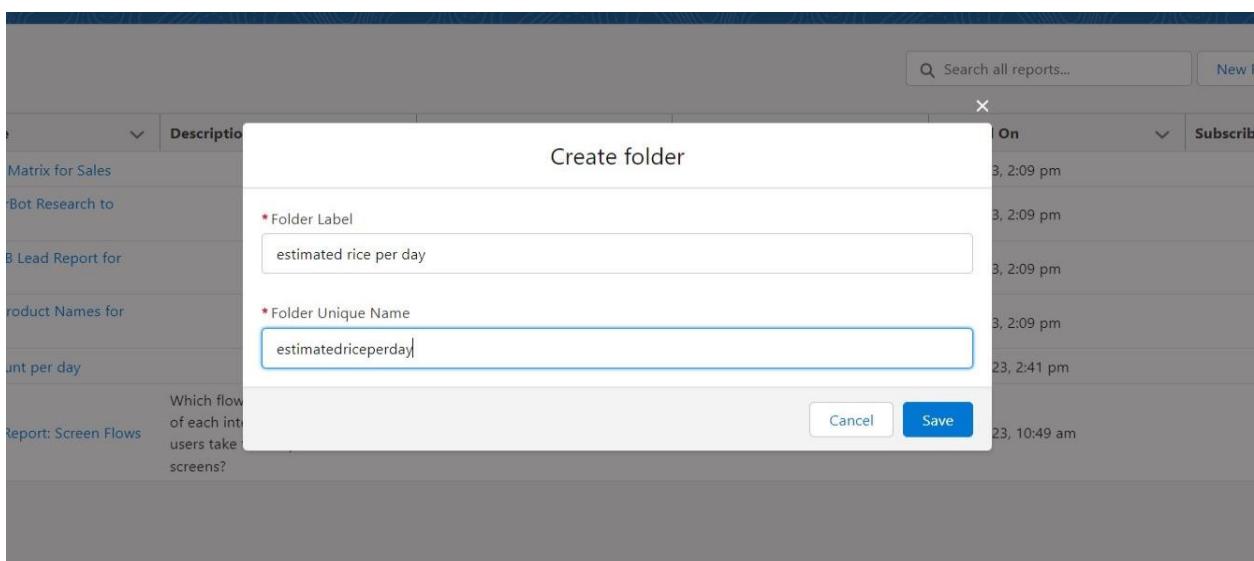
- i. Follow as per below image.



## 11.2 create a report folder

- 1.navigate to app launcher and click reports on that.
- 2.click all reports.
3. Select the rangeof amount per day drop downin that click move.
5. Select estimated rice per day folderand select folder.

Note: if you want to see the report which you have created then go to reports - all folders - estimated riceper day - your report will appear in this way.



The screenshot shows the Salesforce Reports page. On the left, there's a sidebar with categories like REPORTS, FOLDERS, and FAVORITES. The main area displays a table of reports with columns for Report Name, Description, Folder, Created By, Created On, and Subscribed. One specific report, "range of amount per day", is selected and highlighted in blue. A context menu is open over this report, listing options: Run, Edit, Subscribe, Export, Delete, Add to Dashboard, Favorite, and Move.

This screenshot shows a modal dialog titled "Move range of amount per day". It displays a hierarchical tree view of folders under "All Folders". The tree structure includes "All Folders", "Created by Me", "Shared with Me", "Private Reports", "Public Reports", and several sub-folders for "Acquisition Reports", "Einstein Bot Reports", and "Report Recycle Bin". At the bottom of the dialog are "New Folder" and "Select" buttons.

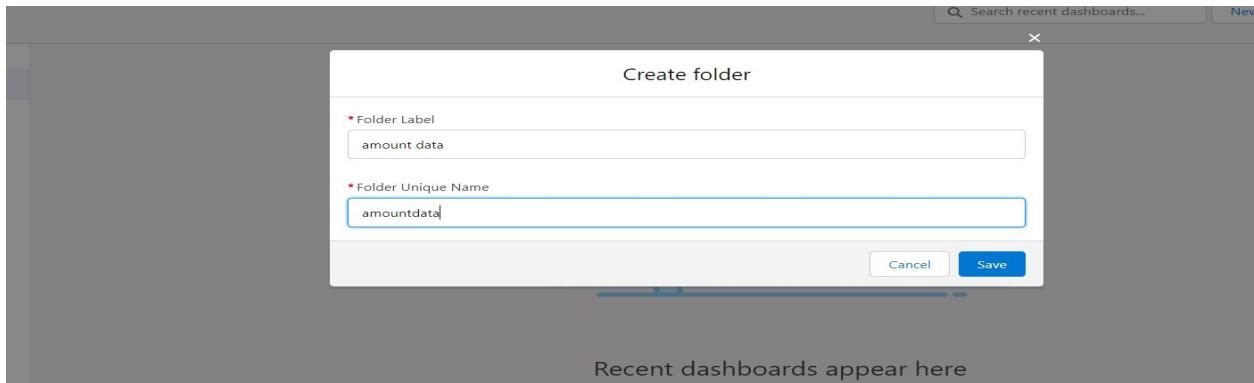
## Task 12 : Dashboards

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.

### 12.1 Create Dashboard Folder

1. Click on the app launcher and search for the dashboard.
2. Click on the dashboard tab.
3. Click the new folder, give the folder label as “amount data dashboard”.

4. Folder unique names will be auto populated.
5. Click save.



## 12.2 Create Dashboard

1. Go to the app >> click on the Dashboards tabs.
2. Give a Name and select the folder that was created, and click on create.
3. Select add component
4. Select a Report and click on select.

preivous is shown below. Display as>> vertical bar chart X-axis >> rice taken by shops Y-axis >> sum of amount Y-axis range >> automatic Sort by >> rice taken by shops

1. Component theme >> dark.
2. Add the component
3. Again select add component with above same steps
4. .display as donut chart
5. sort by >> sum of amount
6. .title>>range of amount per day
7. component theme dark
8. click add.
9. Click save and done.

The screenshot shows a Salesforce Lightning interface. The top navigation bar includes links for 'supplier', 'rice mills', 'consumers', 'rice details', 'range of amount per day', 'Dashboards', and a search bar. On the left, a sidebar titled 'MY RICE' lists 'Recent' dashboards, 'Dashboards', 'Folders', and 'Favorites'. The main content area displays a table for the 'estimated data' dashboard, with columns for 'Dashboard Name', 'Description', 'Folder', 'Created By', 'Created On', and 'Subscribed'. The table entry for 'estimated data' shows it was created by Vivek Rachana on 5/8/2024 at 5:45 pm. At the bottom, there's a status bar with system information like battery level, signal strength, and date.

This screenshot shows the same Salesforce Lightning interface as above, but the main content area now displays two charts from the 'estimated data' dashboard. The top chart is a bar chart titled 'range of amount per day' showing 'Rice taken by shops' with values 125 and 150. The bottom chart is a donut chart titled 'range of amount per day' showing 'Sum of Amount Paid' with a value of 15k. Both charts have a 'View Report' link below them. The status bar at the bottom remains the same.

## Task 13 : Apex

Apex is a stronglytyped, object-oriented programming language that allows developers to execute flow and transaction control statements on the Lightning platform server in conjunction with calls to the Lightning Platform? API. Using syntax that looks like Java and acts like database stored procedures, Apex enables developers to add business logic to most system events, including button clicks, related record updates, and Visualforce pages. Apex code can be initiated by Web service requests and from triggers on objects

It is as similaras java i.e, it also supports OOP( Object oriented programming) like Classes, objects,methods.

### **Creating Classes :**

Apex classesare modeled on their counterparts in Java. You'll define, instantiate, and extend classes, and you'llwork with interfaces, Apex class versions,properties, and other related

class concepts.

- **Class:**

As in Java, you can create classes in Apex. A class is a template or blueprint from which objects are created. An object is an instance of a class.

- **Object**

Object is an instance of a class, where it can access all the properties that are present in a class i.e, variables and methods.

### **13.1 Creating an Apex Class(ConsumerRecord)**

1. Login to the Salesforce account and navigate to the gear account in the top right corner.
2. Then we can see the Developer console. Click on the developer console and you will navigate to a new console window.
3. Then you can see many tools in the Toolbar of the new console window. Click on File, New and ApexClass.
4. Enter the name of the class(ConsumerRecord) to create a new class file.

#### **Code Snippet :**

```
class ConsumerRecord {  
    public static void sendEmailNotification  
        (List<consumer_c> con){for(consumer_c c:con)  
    {  
        Messaging.SingleEmailMessage email = new  
            Messaging.SingleEmailMessage();  
        email.setToAddresses( new List<String>{c.email_  
            c});email.setSubject('Welcome to our company');  
        email.setPlainTextBody('Dear ' + ' ' + ',\n\nWelcome to MY  
RICE!'+ 'You have been seen as a valuable customer to us. Please continue your  
journey with us, while we try to provide you with good quality resources.'+'\n'+  
            "We are proud to associate with valuable customers  
like you and we look forward to collaborating with you by providing more and  
more exciting discounts or even product offers too.' + '\n'  
            +'So why taking a step back, take a leap of faith  
and shop with us more, while we provide with the valuable products and  
offers'+'\n'+ '\n'+ '\n'+
```

"Thankyou for buying'+'Here are some of the  
products that are brought by the customers who similarly bought products like  
this'+'\n\n');

```
        Messaging.sendEmail(new  
List<Messaging.SingleEmailMessage>{email});  
  
    }  
}  
}
```



```
ConsumerRecord.apc
1  public class ConsumerRecord {
2      public static void sendEmailNotification(List<consumer__c> con) {
3          List<Messaging.SingleEmailMessage> emails = new List<Messaging.SingleEmailMessage>();
4
5          for (consumer__c c : con) {
6              Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
7              email.setTo(c.Email__c);
8              email.setSubject('Welcome to our company');
9              email.setPlainTextBody('Dear ' + c.Name + ',\n\n' +
10                'We are proud to associate with valuable customers like you and we look forward to collaborating with you by providing more and more exciting discounts or even product offers too.\n\n' +
11                'So why taking a step back, take a leap of faith and shop with us more, while we provide with valuable products and offers.\n\n' +
12                'Thank you for buying ' + 'Here are some of the products that are bought by customers who similarly bought products like this.\n\n' +
13                'Haze' + '\n\n' +
14                'Haze' + '\n\n' +
15                'Haze' + '\n\n' +
16                'Haze' + '\n\n' +
17                'Haze' + '\n\n' +
18                'Haze' + '\n\n' +
19                'Haze' + '\n\n' +
20                'Haze' + '\n\n' +
21                'Haze' + '\n\n' +
22            );
23
24            emails.add(email);
25        }
26
27        if (!emails.isEmpty()) {
28            Messaging.sendEmail(emails);
29        }
30    }
31
32 }
```



The screenshot shows the Salesforce Apex code editor with the following details:

- File Path:** ddm000005iemduac-dev-ed.develop.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage
- File Name:** ConsumerRecord.apcpx
- Class Name:** consumerTrigger.apcpx
- Apex Class:** RiceDetails
- Code Coverage:** None - API Version: 61
- Code Content:**

```
1 public class RiceDetails {
2     // Method to process rice details records
3     public static void processRiceDetails(List<rice_details__c> riceDetailsList) {
4         // Check if the list is not empty
5         if (riceDetailsList == null || riceDetailsList.isEmpty()) {
6             return;
7         }
8
9         // Example logic: Debugging output (useful for testing)
10        System.debug('Processing rice details: ' + riceDetailsList);
11
12        // Implement your specific logic here
13        for (rice_details__c riceDetail : riceDetailsList) {
14            // Example: Log each rice detail record's information
15            System.debug('Rice Detail Record: ' + riceDetail);
16
17            // Implement specific actions for each record
18            // For instance, you might send notifications or update related records
19        }
20    }
21 }
```

## 13.2 Creating an Apex Trigger

## 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 the Filemenu in the toolbar, and click on new? Trigger.Enter the

`triggername` and the object to be triggered.

### Syntax For creating trigger :

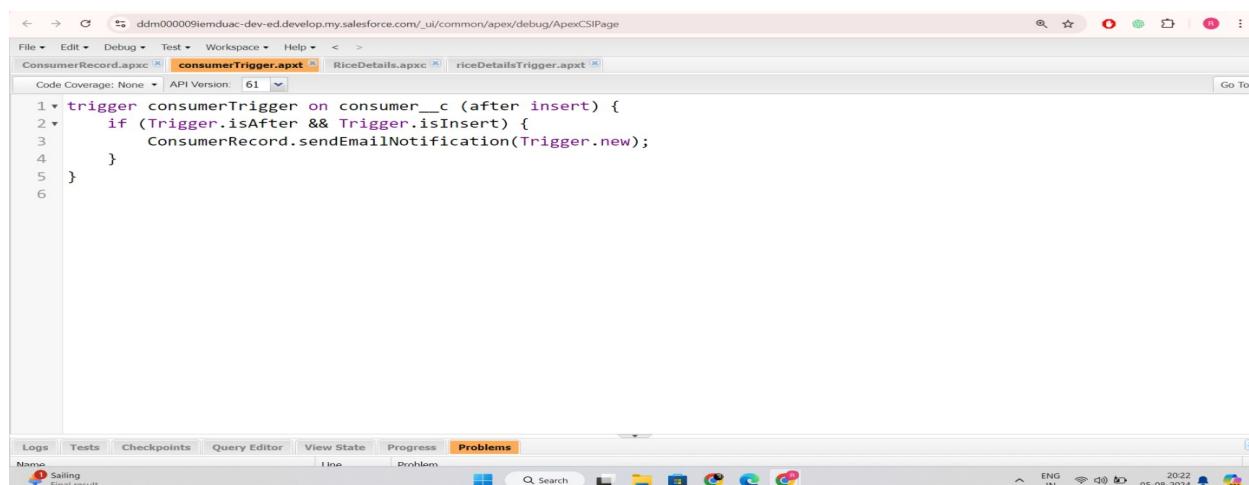
The syntax for creating trigger is :

```
Trigger [triggername] on [objectname]( Before/After event) {  
    //Trigger Logic  
}
```

### Trigger code:

```
trigger consumerTrigger on consumer__c (After insert)
```

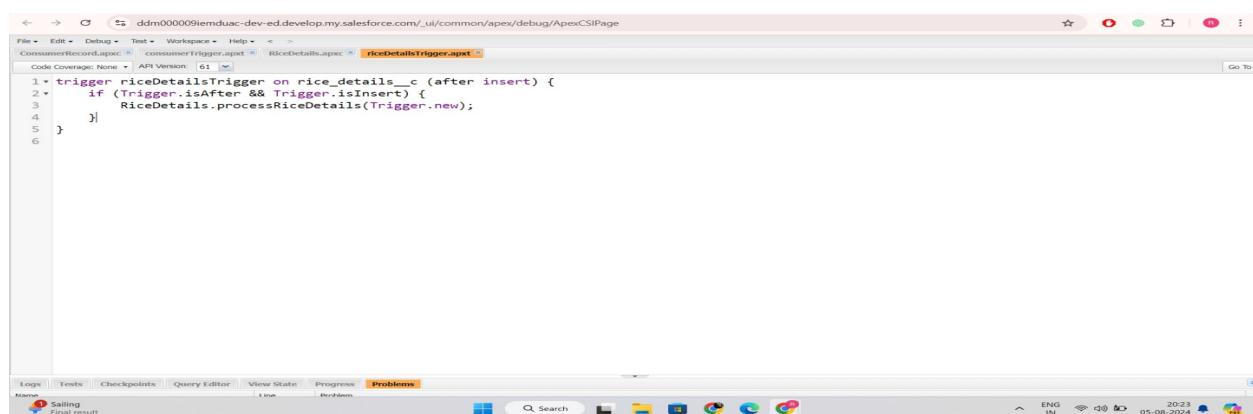
```
{if(trigger.isAfter && trigger.isInsert) {  
    ConsumerRecord.sendEmailNotification(trigger.new);  
}  
}
```



A screenshot of the Salesforce Apex Editor interface. The title bar shows the URL: ddm000009iemduac-dev-ed.develop.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage. The tabs at the top include 'File', 'Edit', 'Debug', 'Test', 'Workspace', 'Help', 'ConsumerRecord.apxc', 'consumerTrigger.apxt' (which is the active tab), 'RiceDetails.apxc', and 'riceDetailsTrigger.apxt'. Below the tabs, the code for 'consumerTrigger.apxt' is displayed:

```
1 trigger consumerTrigger on consumer__c (after insert) {  
2     if (Trigger.isAfter && Trigger.isInsert) {  
3         ConsumerRecord.sendEmailNotification(Trigger.new);  
4     }  
5 }
```

The editor has a toolbar with icons for Save, Undo, Redo, Cut, Copy, Paste, Find, Replace, and others. Below the toolbar is a menu bar with 'Logs', 'Tests', 'Checkpoints', 'Query Editor', 'View State', 'Progress', and 'Problems'. The 'Problems' tab is selected, showing one error: 'Sailing' (Final result). The bottom status bar shows the language as 'ENG IN', the date as '05-08-2024', and the time as '20:22'.



A screenshot of the Salesforce Apex Editor interface, identical to the previous one but with a different file open. The title bar shows the URL: ddm000009iemduac-dev-ed.develop.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage. The tabs at the top include 'File', 'Edit', 'Debug', 'Test', 'Workspace', 'Help', 'ConsumerRecord.apxc', 'consumerTrigger.apxt', 'RiceDetails.apxc', and 'riceDetailsTrigger.apxt' (which is the active tab). Below the tabs, the code for 'riceDetailsTrigger.apxt' is displayed:

```
1 trigger riceDetailsTrigger on rice_details__c (after insert) {  
2     if (Trigger.isAfter && Trigger.isInsert) {  
3         RiceDetails.processRiceDetails(Trigger.new);  
4     }  
5 }
```

The interface and status bar are identical to the first screenshot.

# **THANK YOU**