

SALESFORCE VIRTUAL INTERNSHIP

SmartInternz

A CRM APPLICATION FOR WHOLESALE RICE MILL

By

Name:Kammari Swarnalatha

Email:swarnakammari2@gmail.com

Project Abstract:

The project aims to develop a Salesforce A CRM APPLICATION FOR WHOLESALE RICE MILL. The primary aim of this project is to develop a comprehensive Customer Relationship Management (CRM) application tailored for the wholesale rice milling industry.

The application will streamline operations, enhance customer engagement, and facilitate efficient management of suppliers, rice mills, consumers, and rice distribution details. The MY RICE CRM application is built on the Salesforce platform, leveraging its robust capabilities to manage custom objects and relationships relevant to the rice wholesale business. This system will allow rice mill operators to efficiently track suppliers, manage customer data, and oversee rice distribution while ensuring seamless communication through automated notifications.

INTRODUCTION:

The wholesale rice milling industry plays a crucial role in the agricultural supply chain, ensuring that rice—a staple food

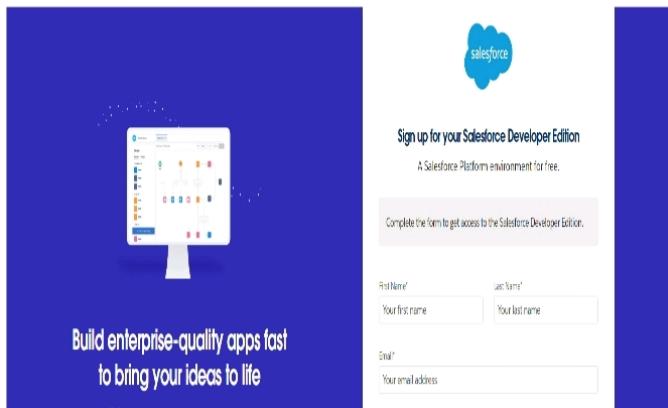
chain, ensuring that rice—a staple food for millions—reaches consumers efficiently and reliably. However, managing the complexities of supplier relationships, consumer interactions, and inventory tracking can be challenging without a robust system in place. To address these challenges, we introduce the MY RICE CRM application, a specialized Customer Relationship Management solution designed for wholesale rice mills. Built on the Salesforce platform, MY RICE integrates various functionalities tailored to the unique needs of this sector, enabling users to manage supplier information, track consumer engagement, and oversee distribution.

Creating a Developer Account

1. Sign Up for Developer Account:

- Visit [Salesforce Developer Signup](#).
- Fill out the form with the following details:
 - **First Name & Last Name**
 - **Email**
 - **Role:** Developer
 - **Company:** College Name
 - **Country:** India
 - **Postal Code:** [Your Pin Code]
 - **Username:** Combination of your name and company (e.g., username@organization.com)

- Click **Sign Me Up.**

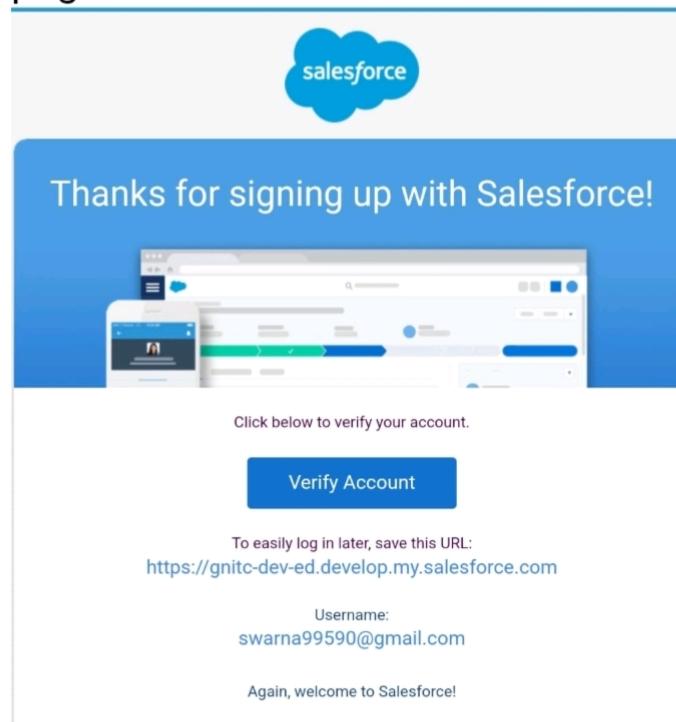




2. Account Activation:

- Check your email inbox for the verification email. It may take 5-10 minutes.
- Click **Verify Account** in the email.
- Set a password and answer a security question, then click **Change Password**.

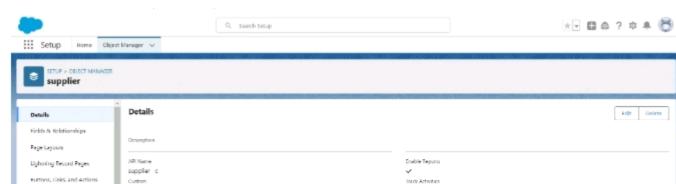
You will be redirected to your Salesforce setup page



Creating Custom Objects

1. To create consumer object:

1. 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
2. Click on Allow reports and Track Field History and allow search
3. Allow search >> Save.



The screenshot shows the 'Object Manager' section of the Salesforce Setup page. A new object is being created with the following details:

- Label Name:** rice mill
- Plural Label Name:** rice mills
- Record Name:** Auto Number
- Display Format:** rice-{000}
- Starting Number:** 1
- Allow Reports:** checked
- Track Field History:** checked
- Allow Search:** checked

2.Create Rice Mill Object:

- From the Setup page, go to **Object Manager**.
- Click **Create > Custom Object**.
- Enter the following details:
 - Label Name:** Rice Mill
 - Plural Label Name:** Rice Mills
 - Record Name:** Auto Number
 - Display Format:** rice-{000}
 - Starting Number:** 1
- Check **Allow Reports, Track Field History, and Allow Search**.

Click Save

The screenshot shows the 'Object Manager' section of the Salesforce Setup page. A new object is being created with the following details:

- Label Name:** rice mill
- Plural Label Name:** rice mills
- Record Name:** Auto Number
- Display Format:** rice-{000}
- Starting Number:** 1
- Allow Reports:** checked
- Track Field History:** checked
- Allow Search:** checked

3.Create Rice Details Object:

- Follow the same steps as above.
- Enter the following details:
 - Label Name:** Rice Details
 - Plural Label Name:** Rice Details
 - Display Format:** rice-{000}
 - Starting Number:** 1

Click Save

The screenshot shows the 'Object Manager' section of the Salesforce Setup page. A new object is being created with the following details:

- Label Name:** rice details
- Plural Label Name:** rice details
- Record Name:** Auto Number
- Display Format:** rice-{000}
- Starting Number:** 1
- Allow Reports:** checked
- Track Field History:** checked
- Allow Search:** checked

4.Create supplier Object:

- From the setup page >> Click on Object Manager>> Click on Create>>Click on Custom Object.
- Enter the label name>>supplier

The screenshot shows the 'Object Manager' section of the Salesforce Setup page. A new object named 'Supplier' is being created. The 'Label Name' is set to 'Supplier', 'Plural Label' to 'Suppliers', and 'Record Type' to 'Supplier'. Under 'Fields & Relationships', 'Job Name' is selected. Under 'Details', 'Allow Reports' and 'Track Activities' are checked. Other options like 'Allow Search' and 'Allow Reports' are also present.

2.Create Rice Mill Object:

- From the Setup page, go to **Object Manager**.
- Click **Create > Custom Object**.
- Enter the following details:
 - **Label Name:** Rice Mill
 - **Plural Label Name:** Rice Mills
 - **Record Name:** Auto Number
 - **Display Format:** rice-{000}
 - **Starting Number:** 1
- Check **Allow Reports, Track Field History, and Allow Search**.

Click Save

The screenshot shows the 'Object Manager' section of the Salesforce Setup page. A new object named 'rice mill' is being created. The 'Label Name' is set to 'rice_mill', 'Plural Label' to 'rice_mills', and 'Record Type' to 'Custom'. Under 'Fields & Relationships', 'rice_mill' is selected. Under 'Details', 'Allow Reports' and 'Track Activities' are checked. Other options like 'Allow Search' and 'Allow Reports' are also present.

3.Create Rice Details Object:

- Follow the same steps as above.
- Enter the following details:
 - **Label Name:** Rice Details
 - **Plural Label Name:** Rice Details
 - **Display Format:** rice-{000}
 - **Starting Number:** 1

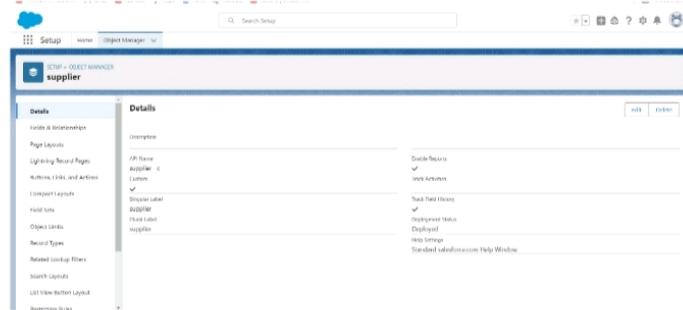
Click Save

The screenshot shows the 'Object Manager' section of the Salesforce Setup page. A new object named 'rice details' is being created. The 'Label Name' is set to 'rice_details', 'Plural Label' to 'rice_details', and 'Record Type' to 'Custom'. Under 'Fields & Relationships', 'rice_details' is selected. Under 'Details', 'Allow Reports' and 'Track Activities' are checked. Other options like 'Allow Search' and 'Allow Reports' are also present.

4.Create supplier Object:

1. From the setup page >> Click on Object Manager>> Click on Create>>Click on Custom Object.
1. Enter the label name>>supplier

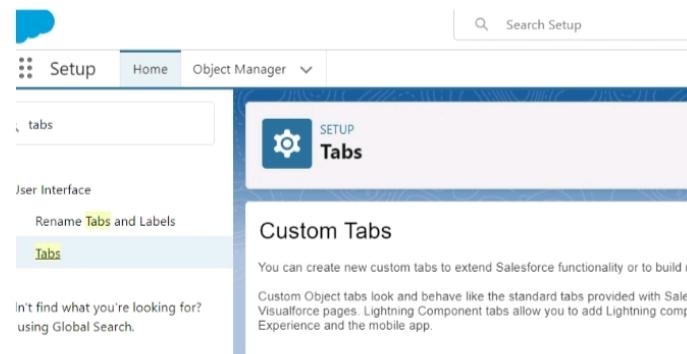
2. Plural label name>>supplier
3. Enter Record Name Label and Format
 - Record Name >> supplier Name
 - Data Type>>Text
2. Click on Allow reports and Track Field History and allow search
3. Allow search >> Save.



Creating Custom Tabs

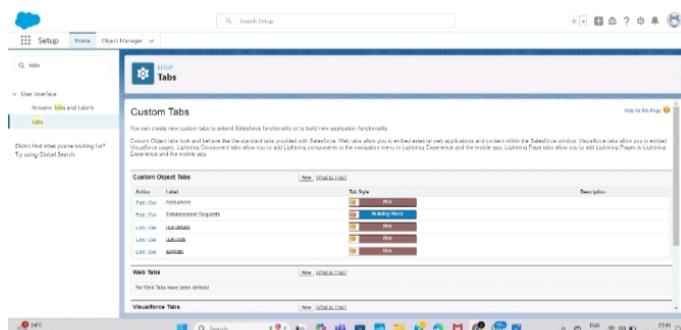
1. Create Tab for Supplier Object:

- Go to the Setup page, type **Tabs** in the Quick Find box, and click **Tabs**.
- Click **New** under **Custom Object Tabs**.
- Select the **Supplier** object and choose a tab style.
- Click **Next**.
- Keep default settings for **Add to Profiles** and **Add to Custom Apps**.
- Click **Save**.
-



2. Create Tabs for Remaining Objects:

- Repeat the above steps for the following objects: **Rice Mill**, **Consumer**, and **Rice Details**.

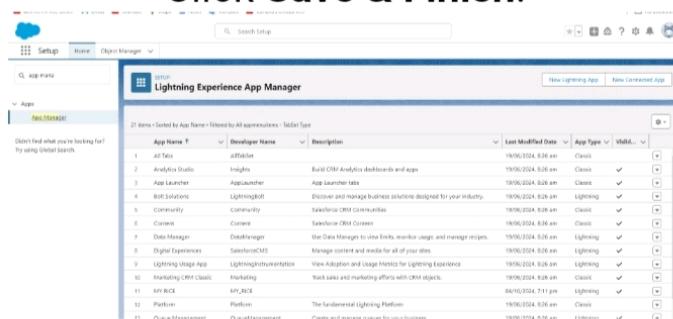


Creating a Lightning App

Creating a Lightning App

1. Create Lightning App:

- Go to the Setup page, search for **App Manager**, and select **App Manager**.
- Click **New Lightning App**.
- Enter **MY RICE** as the app name and click **Next**.
- Keep default settings on the **App Options** page and click **Next**.
- Upload a relevant photo for your app.
- Add Navigation Items:
 - Select **Supplier**, **Rice Mill**, **Consumer**, and **Rice Details**.
 - Click the arrow button to move them to the selected items section.
 - Click **Next**.
- Add User Profiles:
 - Search for **System Administrator** and add it.
 - Click **Save & Finish**.



Creating Fields and Relationships

1. Create a Number Field in Rice Details Object:

- Go to the Setup page, click on **Object Manager**, and edit the **Rice Details** object.
- Click on **Fields & Relationships > New**.
- Select **Number** as the data type and click **Next**.
- Enter:
 - **Field Label:** Rice Distributed
 - **Length:** 5
- Click **Next > Save**.

2. Create Junction Object:

- Edit the **Rice Details** object.
- Click on **Fields & Relationships > New**.
- Select **Master-Detail Relationship** and click **Next**.
- Select **Supplier** as the related object and click **Next**.

- Enter **Field Label:** Supplier Name and click **Next**.
- Click **Save & New**.
- Repeat for **Rice Mill** with **Field Label:** Rice Mill 1 (One).

| FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
|-------------------|--------------------|-------------------------|-------------------|---------|
| Created By | CreatedById | Lookup(Users) | | |
| Last Modified By | LastModifiedById | Lookup(Users) | | |
| rice details Name | Name | Name | | ✓ |
| rice distributed | rice_distributed_c | Auto Number | | |
| rice mill Store | rice_mill_store_c | Master Database mill | | ✓ |
| Supplier Name | supplier_name_c | Master Detail(Supplier) | | ✓ |

3.Create Master-Detail Relationship:

- Edit the **Consumer** object.
- Click on **Fields & Relationships > New**.
- Select **Master-Detail Relationship** and click **Next**.
- Select **Rice Mill** as the related object.
- Enter **Field Label:** Rice Mill Name and click **Next**.
- Click **Save**.

| FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
|------------------|-------------------|-----------------|-------------------|---------|
| Amount Field | Amount (field_c) | Amount (Number) | | |
| Consumer Name | Name | Name | | ✓ |
| Consumer Name | Consumer Name_c | Consumer Name | | |
| Created By | CreatedById | Lookup(Users) | | |
| Email | Email_c | Email | | |
| First Name | First_name_c | Text | | |
| Last Modified By | LastModifiedById | Lookup(Users) | | |
| Last Name | Last_name_c | Text | | |
| Mode of Contact | Mode_of_contact_c | Text | | |

4.Create Roll-up Summary Fields:

- For **Supplier** object:
 - Go to **Object Manager**, select **Supplier**, and click on **Fields & Relationships > New**.
 - Select **Roll-Up Summary** and click **Next**.
 - Enter **Field Label:** Sum of Rice Distributed.
 - Select **Rice Details** as the summarized object and **SUM** as the roll-up type.
 - Select **Rice Distributed** as the field to aggregate.
 - Click **Save**.

| FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
|------------------|------------------|--------------------|-------------------|---------|
| Created By | CreatedById | Lookup(Users) | | |
| Last Modified By | LastModifiedById | Lookup(Users) | | |
| Owner | Owner | Lookup(User Group) | | ✓ |

| Fields & Relationships | | | | | |
|------------------------|------------------|---------------------|-----------------------------------|-------------------|---------|
| | FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
| Created By | CreatedBy | CreatedById | Lookup(Any) | | |
| Last Modified By | LastModifiedBy | LastModifiedById | Lookup(Any) | | |
| Owner | Owner | OwnerId | Lookup(One Great) | | |
| Rice distributed | Rice_distributed | Rice_distributed_id | Roll-up Summary (Sum Rice detail) | | |
| Supplier Name | Supplier Name | Name | Text(50) | | |

- For **Rice Mill** object, follow similar steps with **Field Label**: Rice Distributed to Shops.

| Fields & Relationships | | | | | |
|---------------------------|---------------------------|------------------------------|-----------------------------------|-------------------|---------|
| | FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
| Created By | CreatedBy | CreatedById | Lookup(Any) | | |
| Last Modified By | LastModifiedBy | LastModifiedById | Lookup(Any) | | |
| Rice distributed to shops | Rice_distributed_to_shops | Rice_distributed_to_shops_id | Roll-up Summary (Sum Rice detail) | | |
| Rice mill Name | Rice mill Name | Name | Auto Number | | |
| Rice price | Rice price | Rice_price_id | Number(5, 0) | | |
| Rice taken | Rice taken | Rice_taken_id | Roll-up Summary (Sum consumer) | | |
| Rice taken in shops | Rice taken in shops | Rice_taken_in_shops_id | Number(5, 0) | | |
| Supplier Name | Supplier Name | Name | Master Detail(supplier) | | |

5.Creating Fields in Objects:

Creating the number field in rice details object

- Go to the **setup page** >> click on **object manager** >> From drop down click edit for rice details object.
- Click on fields & relationship >> click on **New**.
- Select Data type as “**master detail**” and click **Next**.
- Given the Field Label as “ **supplier name** ” and length as “ **5** ”
- Field Name will be auto populated, and click on **Next>> Next >>Save**.

| Fields & Relationships | | | | | |
|------------------------|------------------|---------------------|--------------------------|-------------------|---------|
| | FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
| Created By | CreatedBy | CreatedById | Lookup(Any) | | |
| Last Modified By | LastModifiedBy | LastModifiedById | Lookup(Any) | | |
| Rice distributed | Rice distributed | Rice_distributed_id | Number(5, 0) | | |
| Rice mill Name | Rice mill Name | Rice_mill_name_id | Master Detail(Rice mill) | | |
| Supplier Name | Supplier Name | Name | Master Detail(supplier) | | |

6.Creating Fields in rice mill Objects:

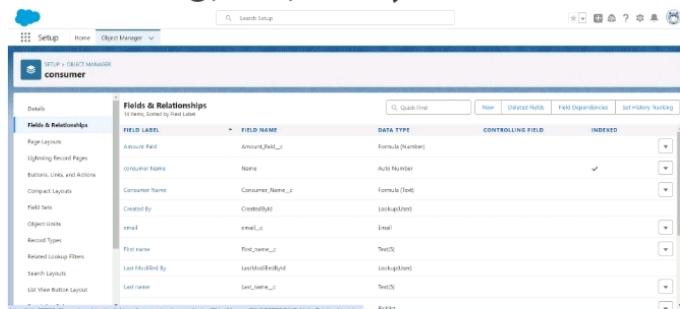
- Select Data type as “**Number**” and click **Next**.
- Given the Field Label as “ **rice price/kg** ” and length as “ **5** ”

7.Creating Fields in consumer Objects:

Object name:**consumer**

- Field:**First name**; Data type:**Text**
- Field:**Last name**; Data type:**Text**
- Field:**Phone number**; Data type:**phone**

- Field:**email**; Data type:**email**
- Field:**Rice taken by shops**; Data type:**Number (length=5)**
- Field:**Rice type**; Data type:(Picklist values:**basmati,normal rice**)
- Field: **Mode of payment**; Data type: (Picklist values :**Credit card,Debit card,Net banking,UPI,Cash**)

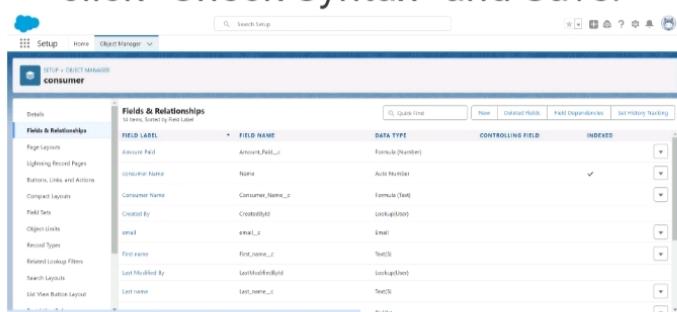


8. Creating Cross Object Formula Field in consumer Object:

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.

- Go to **setup** >> click on **Object Manager** >> type **object name(consumer)** 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:
 - `rice_taken_by_shops_c * rice_mill_name_r.rice_price_kg_c`
- Under Advanced Formula write down the formula and click "Check Syntax" and Save.
- Creating the Formula field in consumer Object
- Note : check whether that the fields that mentioned in the formula field are created are not , if not go to activity 9 and create that fields mentioned in consumer object
- Go to **setup** >> click on **Object Manager** >>

- Go to setup >> click on Object Manager >> type object name(consumer) 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 “Consumer 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.



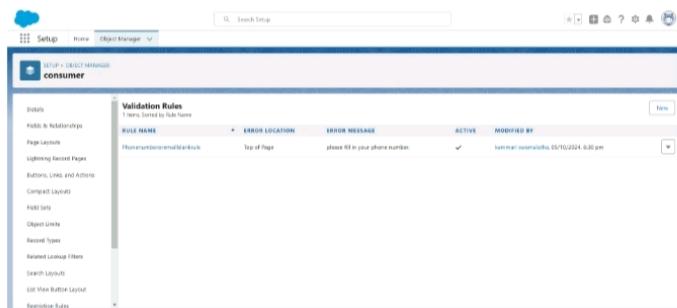
9.Creating the validation rule:

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 consumer object

- Go to the setup page >>click on object manager >> From drop down click edit for consumer object.
- Click on the validation rule >> click New.
- Enter the Rule name as “Phonenumberoremailblankrule”.
- Enter the description as “phone number and email number should not be blank”

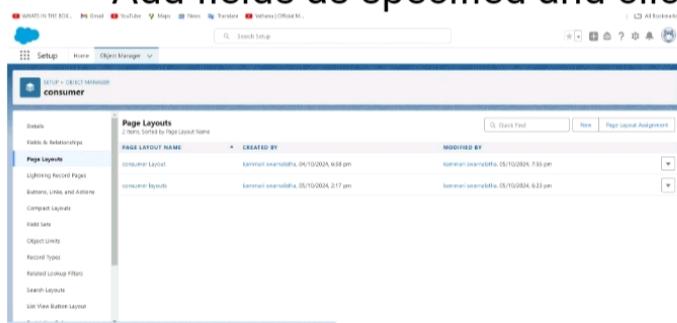
- Enter the formula as “OR(ISBLANK(phone_number_c), ISBLANK(email_c))” and check the syntax.
- Under the error message write as “please fill in your phone number.”
- Select error location “top of page”.
- Save the validation rule.



Creating Page Layouts:

Create Page Layout for Consumer Object:

- Go to **Setup > Object Manager > Consumer**.
- Click on **Page Layouts > New**.
- Select an existing layout, name it **Consumer Layout**, and click **Save**.
- Drag and drop fields into sections named **Personal Details**, **Rice Details**, and **Receipt Details**.
- Add fields as specified and click **Save**.



Creating Profiles and Roles

1. Create Owner Profile:

- Go to **Setup > Profiles**.
- Clone **Standard User**, name it **Owner**, and click **Save**.
- Set permissions as shown in the diagram and save.



1. Create Employer Profile:

- Clone Standard Platform User, name it **Employer**, and click **Save**.

1. Set default app settings and permisssions

Worker Profile:

- Clone Standard Platform User, name it **Worker**, and click **Save**.

Set default app settings and permissions as shown in the diagram and saveions as shown in the diagram and save

1. Create Roles:

- **Owner Role:**
 - Go to **Setup > Roles > Set Up Roles**.
 - Click **Add Role** under **CEO** role.
 - Enter **Label:** Owner and save.
- **Employer Role:**
 - Add role under **Owner** with **Label:** Employer and save.
- **Worker Role:**
 - Add role under **Employer** with **Label:** Worker and save.



Creating Users

1. Create User (Owner):

- Go to **Setup > Users > New User**.
- Enter details:
 - **First Name:** Vicky
 - **Last Name:** Y
 - **Alias:** [Alias Name]
 - **Email:** [Your Email]
 - **Username:** [Username@domain.com]
 - **Role:** Owner
 - **User License:** Salesforce
 - **Profile:** Owner
- Click **Save**.
- Click **Save**.

1. Create User (Employer):

- Enter details:
 - **First Name:** Ram
 - **Last Name:** Ram
 - **Alias:** [Alias Name]
 - **Email:** [Your Email]
 - **Username:** [Username@domain.com]
 - **Role:** Employer
 - **User License:** Salesforce Platform
 - **Profile:** Standard Platform User
- Click **Save**.

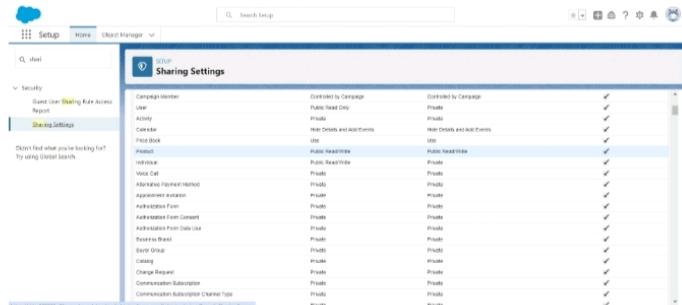
2. Create User (Worker):

- Enter details:
 - **First Name:** Ragu
 - **Last Name:** Raj
 - **Alias:** [Alias Name]
 - **Email:** [Your Email]
 - **Username:** [Username@domain.com]
 - **Role:** Worker
 - **User License:** Salesforce Platform
 - **Profile:** Standard Platform User
- Click **Save**.

Creating Sharing Settings

- **Set Organization-Wide Defaults:**
 - Go to **Setup > Sharing Settings**

- Go to **Setup > Sharing Settings**
 - Scroll down, change the default internal access to “public read-only” for rice mill and supplier object.
 - Click save.
 - **Set Sharing Rules:**
 - Create sharing rules as needed to grant access to different user roles or profiles.

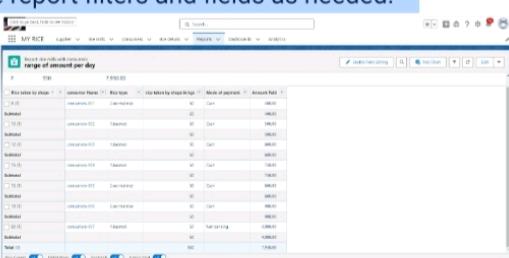


Creating Reports and Dashboards

1.Create Reports:

- Go to **Reports** and click **New Report**.
 - Select the report type for your custom objects (e.g., **Supplier**, **Rice Mill**, **Consumer**, **Rice Details**).
 - Customize report filters and fields as needed.

- CUSTOMIZE report filters and fields as needed**



2.Create Dashboards:

- Go to **Dashboards** and click **New Dashboard**.
 - Add components (charts, graphs) based on the reports you created.



Creating Apex Classes and Triggers

1.Create Apex Class:

- Go to **Setup > Apex Classes > New**.
 - Enter class name and code as needed.

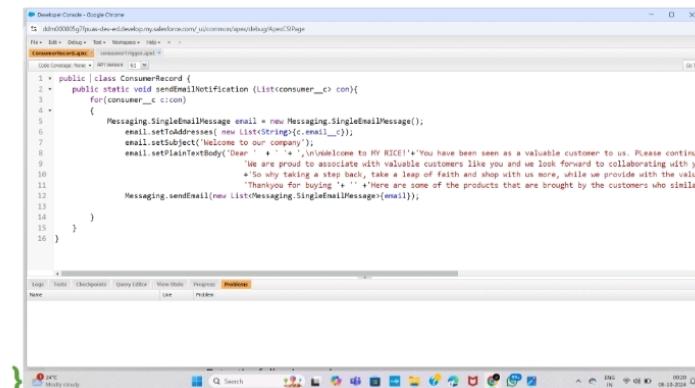
- Enter class name and code as needed for your functionality.
 - Enter the following code
- ```
class ConsumerRecord {
 public static void sendEmailNotification
(List<consumer__c> con){
 for(consumer__c c:con {
 Messaging.SingleEmailMessage email = new
 Messaging.SingleEmailMessage();
 email.setToAddresses(new 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});
}
```



The screenshot shows the Developer Console window in the Salesforce IDE. The code editor contains the following Apex class:

```
1 * Enter class name and code as needed
2 * for your functionality.
3 *
4 * Enter the following code
5
6 class ConsumerRecord {
7 public static void sendEmailNotification (List<consumer__c> con){
8 for(consumer__c c:con {
9 Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
10 email.setToAddresses(new List<String>
11 {c.email__c});
12 email.setSubject('Welcome to our company');
13 email.setPlainTextBody('Dear ' + ' +
14 '\n\nWelcome to MY RICE!+'You have been seen as a valuable customer to us. Please continue
15 your journey with us, while we try to provide you with good quality
16 resources.'+'\n'+
```

The code continues with a multi-line string containing a welcome message and a thank you note, followed by a closing brace for the class definition.

## 2.Create Apex Trigger:

- Go to **Setup > Apex Triggers > New**.
- Enter trigger name and code to define the behavior.
- Enter the following code

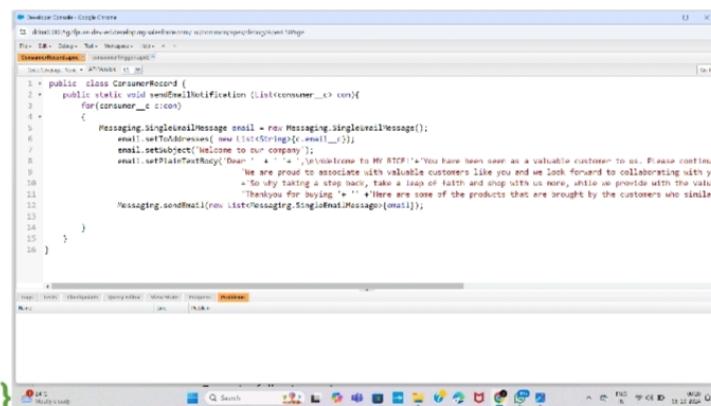
trigger consumerTrigger on  
consumer\_\_c (After insert)

"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});
}
```



```
1+ public class ConsumerRecord {
2+ public static void sendEmailNotification (List<consumer_c> con){
3+ for(consumer_c c :con){
4+ Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
5+ email.setToAddresses(new List<String>{c.email__c});
6+ email.setSubject('Welcome to our company');
7+ email.setPlainText('Dear ' + c.name__c + ',
8+ 'We are proud to associate with valuable customers like you and we look forward to collaborating with you.
9+ '+ 'So why taking a step back, take a leap of faith and shop with us more, while we provide with the valuable
10+ 'Thankyou for buying ' + " + Here are some of the products that are brought by the customers who similar"
11+ Messaging.sendEmail(new List<Messaging.SingleEmailMessage>{email});
12+ }
13+ }
14+ }
15+ }
16+ }
```

## 2.Create Apex Trigger:

- Go to **Setup > Apex Triggers > New.**
- Enter trigger name and code to define the behavior.
- Enter the following code

```
trigger consumerTrigger on
consumer_c (After insert)
{
```

```
 if(trigger.isAfter &&
trigger.isInsert) {
```

```
 ConsumerRecord.sendEmailNotification(trigger.new
 }
}
```

- Click Save



```
1+ trigger consumerTrigger on consumer_c (After insert) {
2+ if(trigger.isAfter && trigger.isInsert) {
3+ ConsumerRecord.sendEmailNotification(trigger.new);
4+ }
5+ }
```