

Project Report on

Retail Management Application Using Salesforce

(DEVELOPER) - (Short-term)

Introduction: Retailing encompasses the business activities involved in selling goods and services to consumers for their personal, family, or household etc. A CRM product owner has requested to create two applications, one is a sales app for sales reps to use this application and store customers data, and the second application is a service app for service reps/agents to provide support to customers in dealing cases. To generate business on top of the customers.

Milestone – 01: Create a Developer Org

Go to developers.salesforce.com/Signup

Click on sign up.

On the sign up form, enter the following details:

1. First name & Last name – Nithya Varadarajulu
2. Email – nithya03v@gmail.com
3. Role : Developer
4. Company : GAYATRI DEGREE COLLEGE - TIRUPATI
5. County : India
6. Postal Code : 517501
7. Username : nithya03v@gdcproject.com

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

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

Sign up for your Salesforce Developer Edition

A full-featured copy of the Platform, for free

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

First Name*
Your first name

Last Name*
Your last name

Email*
Your email address

Role*
Your job role

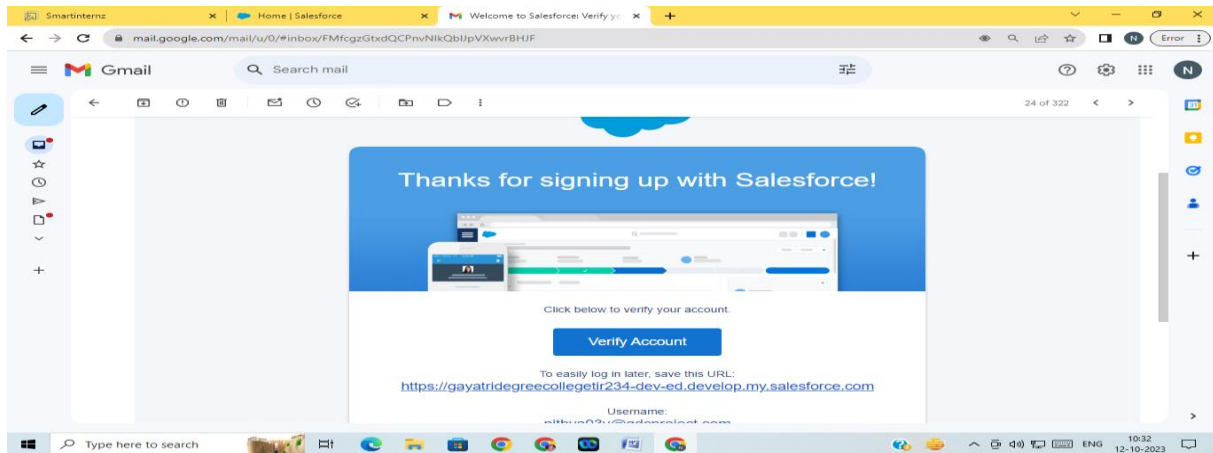
Company*
Company Name

Country/Region*
Country/Region

Postal Code*
Your postal code

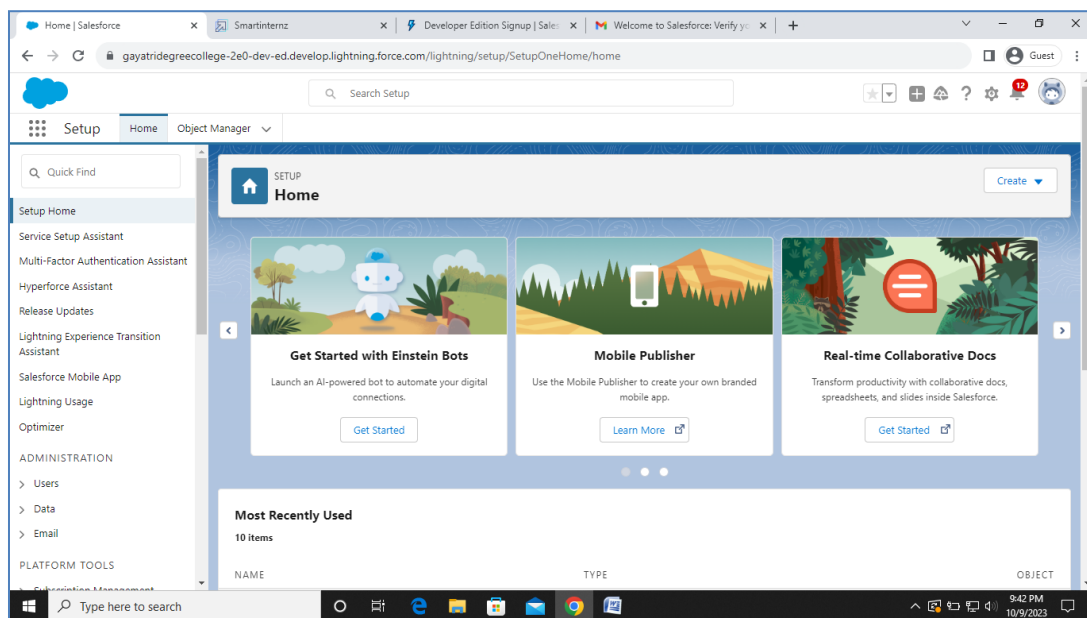
Account Activation

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



Login to your Salesforce Org

1. Go to salesforce.com and click on login.
2. Enter the username and password that you just created.
3. After login this is the home page which you will see.



Milestone – 02: Creation of Objects

Object – Warehouse

1. Click on the gear icon and then select Setup.
2. Click on the object manager tab just beside the home tab.

3. After the above steps, have a look on the extreme right you will find a Create Dropdown click on that and select Custom Object.
4. On the Custom Object Definition page, create the object as follows:
5. Label: **Warehouse**
6. Plural Label: Warehouses
7. Record Name: Warehouse Name
8. Check the Allow Reports checkbox
9. Check the Allow Search checkbox
10. Click Save.

Object – Sales order

1. Click on the gear icon and then select Setup.
2. Click on the object manager tab just beside the home tab.
3. After the above steps, have a look on the extreme right you will find a Create Dropdown click on that and select Custom Object.
4. On the Custom Object Definition page, create the object as follows:
5. Label: **Sales order**
6. Plural Label: Sales orders
7. Record Name: Sales order Name
8. Check the Allow Reports checkbox
9. Check the Allow Search checkbox
10. Click Save.

Object – Dispatch/Tracking

1. Click on the gear icon and then select Setup.
2. Click on the object manager tab just beside the home tab.
3. After the above steps, have a look on the extreme right you will find a Create Dropdown click on that and select Custom Object.
4. On the Custom Object Definition page, create the object as follows:
5. Label: **Dispatch/Tracking**
6. Plural Label: Dispatch/Trackings
7. Record Name: Dispatch/Tracking Name
8. Check the Allow Reports checkbox
9. Check the Allow Search checkbox
10. Click Save.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Dispatch/Tracking	Dispatch_Tracking__c	Custom Object		08/10/2023	✓
Sales order	Sales_order__c	Custom Object		08/10/2023	✓
Warehouse	Warehouse__c	Custom Object		08/10/2023	✓

Milestone – 03: Create tabs

Tab- Warehouse

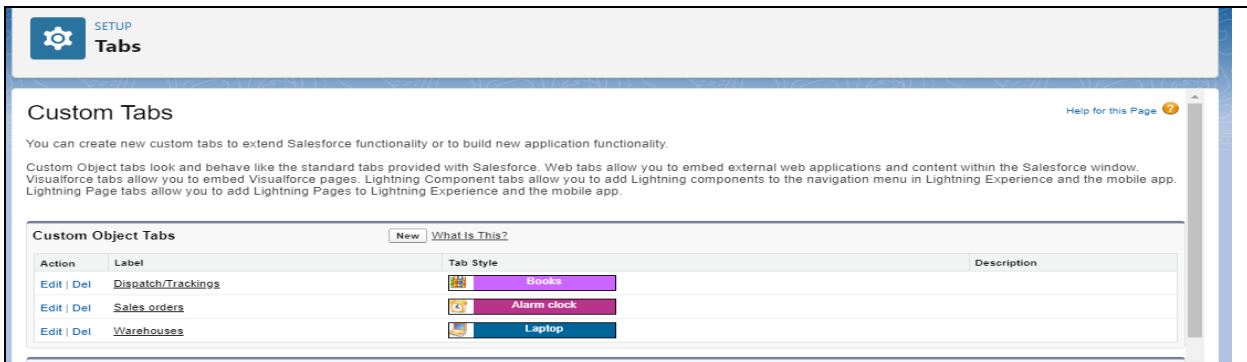
1. Enter Tabs in Quick Find and select Tabs.
2. Under Custom Object Tabs, click New.
3. For Object, select Warehouse.
4. For Tab Style, select any icon.
5. Leave all defaults as is. Click Next, Next, and Save

Tab- Sales order

1. Enter Tabs in Quick Find and select Tabs.
2. Under Custom Object Tabs, click New.
3. For Object, select Warehouse.
4. For Tab Style, select any icon.
5. Leave all defaults as is. Click Next, Next, and Save

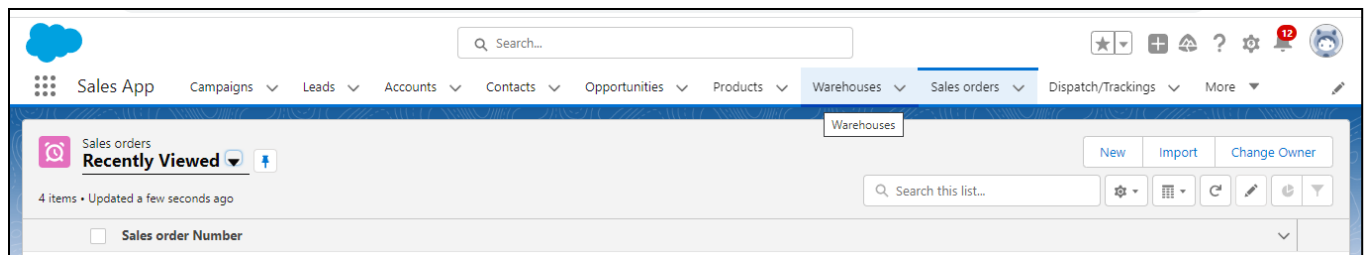
Tab- Dispatch /Tracking

1. Enter Tabs in Quick Find and select Tabs.
2. Under Custom Object Tabs, click New.
3. For Object, select Warehouse.
4. For Tab Style, select any icon.
5. Leave all defaults as is. Click Next, Next, and Save



Milestone – 04: Create the sales app

1. From Setup, enter App Manager in the Quick Find and select App Manager.
2. Click New Lightning App.
3. Enter Sales App as the App Name, then click Next
4. Under App Options, leave the default selections and click Next.
5. Under Utility Items, leave as is and click Next.
6. From Available Items, select Campaign, Leads, Accounts, Contacts, Opportunities, Products, Warehouse, Sales order, Dispatch/Tracking, Reports, and Dashboards and move them to Selected Items. Click Next.
7. From Available Profiles, select System Administrator and move it to Selected Profiles. Click Save & Finish.



Milestone – 05: Creation of fields and relationships

Creation of fields for the dispatch/tracking object:

1. Click the gear icon and select Setup. This launches Setup in a new tab.
2. Click the Object Manager tab next to Home.
3. Select Dispatch/Tracking
4. Select Fields & Relationships from the left navigation
5. Click New
6. Select the Text as the Data Type, click Next.
7. For Field Label, enter Tracking ID & length = 40.
8. Click Next, Next, then Save & New.

Create a master-detail-relationship on dispatch/delivery object:

1. Select Master-Detail Relationship as the Data Type and click Next.
2. For Related to, enter Sales order.
3. Click Next.
4. For Field Label, enter Sales order.
5. Click Next, Next, Next and Save.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)	Data Type	
Dispatch/Tracking Name	Name	Text(80)		✓
Dispatched	Dispatched__c	Text(40)		
Expected date of delivery	Expected_date_of_delivery__c	Text(40)		
Last Modified By	LastModifiedById	Lookup(User)		
Sales order	Sales_order__c	Master-Detail(Sales order)		✓
Tracking ID	Tracking_ID__c	Text(40)		

Create a pick-list field on sales order:

1. From Setup, click Object Manager and select Sales order.
2. Click Fields & Relationships, then New.
3. Select Picklist as the Data Type and click Next.
4. For Field Label enter Status
5. Select Enter values, with each value separated by a new line and enter these values:

6. Open
7. Hold
8. Shipped
9. Returned
10. Click Next, Next, then Save & New.

Create a look relationship on sales order object with account object:

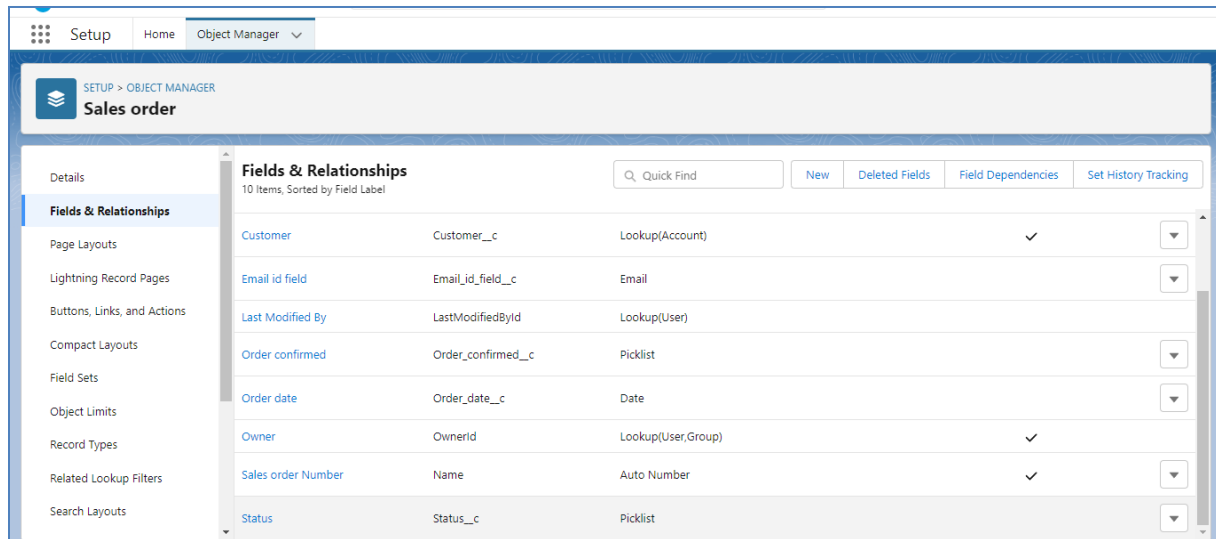
1. Follow steps 1 to 5 of field creation then follow below steps.
2. Select look up Relationship as the Data Type and click Next.
3. For Related to, enter Account.
4. Click Next.
5. For Field Label, enter Customer.
6. Click Next, Next, Next and Save.

Create a look relationship on sales order object with contact object with use of look filter:

1. Select look up Relationship as the Data Type and click Next.
2. For Related to, enter Contact.
3. Click Next.
4. For Field Label, enter Contact.
5. Click lookup filter.
6. Provide filter as given below & also refer picture
7. Contact: Account ID equals Sales Order: Customer
8. Click Next, Next, Next and Save.

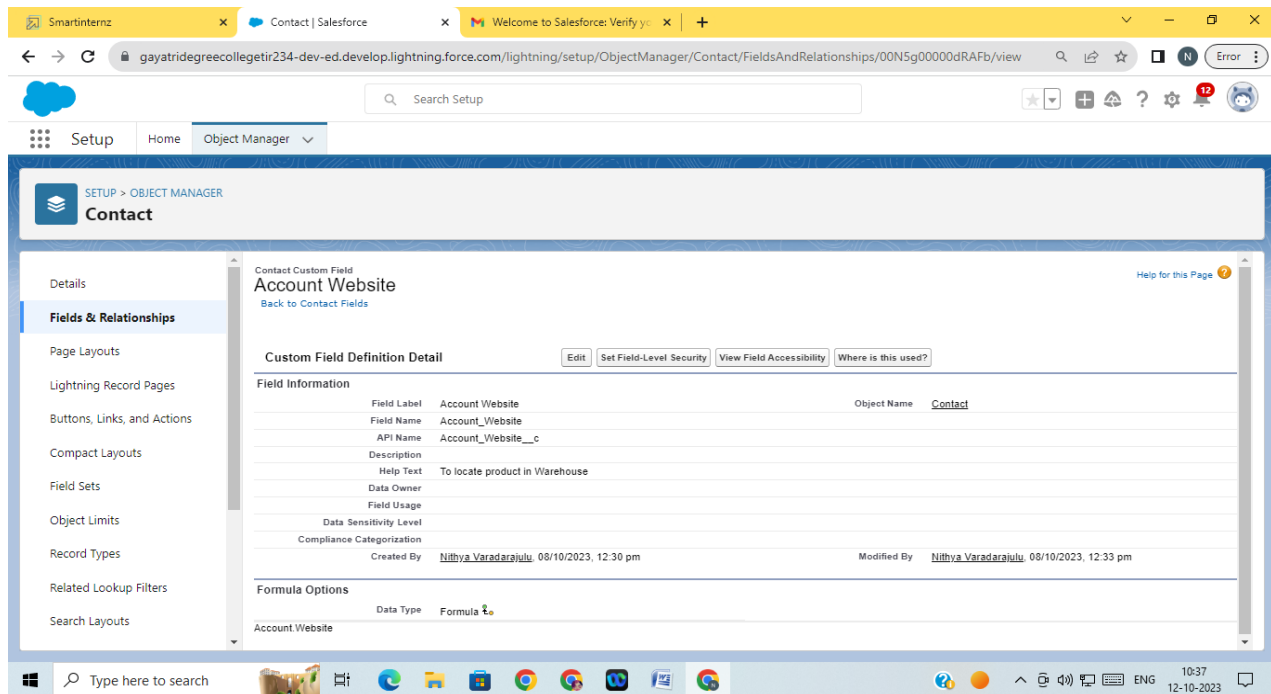
Create order date field on sales order:

1. Click the gear icon and select Setup. This launches Setup in a new tab.
2. Click the Object Manager tab next to Home.
3. Select Sales Order
4. Select Fields & Relationships from the left navigation
5. Click New
6. Select the Date as the Data Type, click Next.
7. For Field Label, enter Order date.
8. Click Next, Next, then Save & New.



Cross-object formula field:

1. Select your object from object selection has Contact.
2. And select the option fields and relationships.
3. At the top right side you can find a new select that option.
4. Now you have to select data type as formula.
5. And you will navigate to enter the details page where you give the field label.
6. And give the label name has Account Website
7. Select formula return type Text
8. In the formula field enter this formula Account. Website.
9. Click next you will navigate to field level security click on visible checkbox so that it is visible to all profiles.
10. Select the next option, select the page layout and save it.



Milestone – 06: Create a user

1. From Setup, in the Quick Find box, enter Users.
2. Select Users.
3. Click New User.
4. Enter the First Name and Last Name manager and (Your) email address and a unique username in the form of an email address. By default, the username is the same as the email address.
5. Select a User License as Salesforce
6. Select Standard User profile.
7. Check Generate new password and notify the user immediately to have the user's login name and a temporary password emailed to your email.

The screenshot shows the Salesforce Setup interface with the 'Users' section selected. The user profile for Nithya Varadarajulu is displayed, including details like Name, Alias, Email, Username, Nickname, Title, Company, Department, Division, Address, Time Zone, Locale, and Language. The 'User Detail' section shows the user's role as 'Salesforce System Administrator' and various permissions.

Milestone – 07: Creation of validation rule

1. Navigate to object manager and select Account object.
2. In details section scroll down and find validation rule in it.
3. Click new, give the label name and in edit error conditional formula give the formula
 - `LEFT(Phone, 1) <> "+"` .
4. And in error message give the description has Phone number must begin with + (country code).
5. In error location select field.
6. Save

The screenshot shows the Salesforce Setup interface with the 'Object Manager' section selected. The 'Account' object is selected, and the 'Validation Rules' section is visible. A validation rule named 'phone_number_has_international_format' is listed with the error message 'give the description has Phone number must begin with + (country code)'.

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
phone_number_has_international_format	Account.TextName	give the description has Phone number must begin with + (country code)	✓	Nithya Varadarajulu, 08/10/2023, 1:11 pm

Milestone – 08: User adoption

Create a record (sales order):

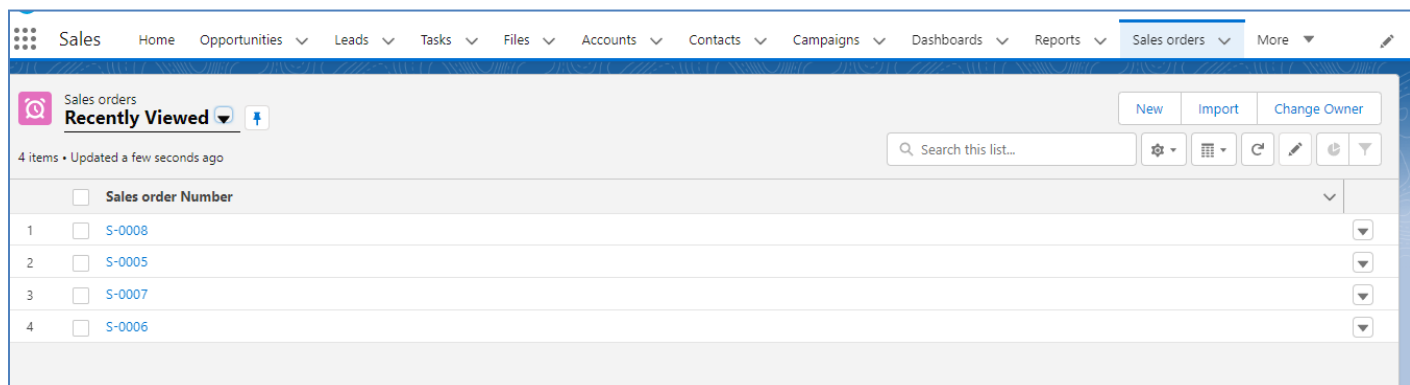
1. Click on App Launcher on left side of screen.
2. Search Sales App & click on it.
3. Click on Sales Order tab.
4. Click new button
5. Fill all Sales Order record details.
6. Click on Save Button

View record (sales order):

1. Click on App Launcher on left side of screen.
2. Search Sales App & click on it.
3. Click on Sales Order Tab.
4. Click on any record name. you can see the details of the Sales Order

Delete records (sales order):

1. Click on Sales Order Tab.
2. Click on Arrow at right hand side on that Particular record.
3. Click on Sales Order Tab.
4. Click on Arrow at right hand side on that Particular record.
5. Click delete and delete again.



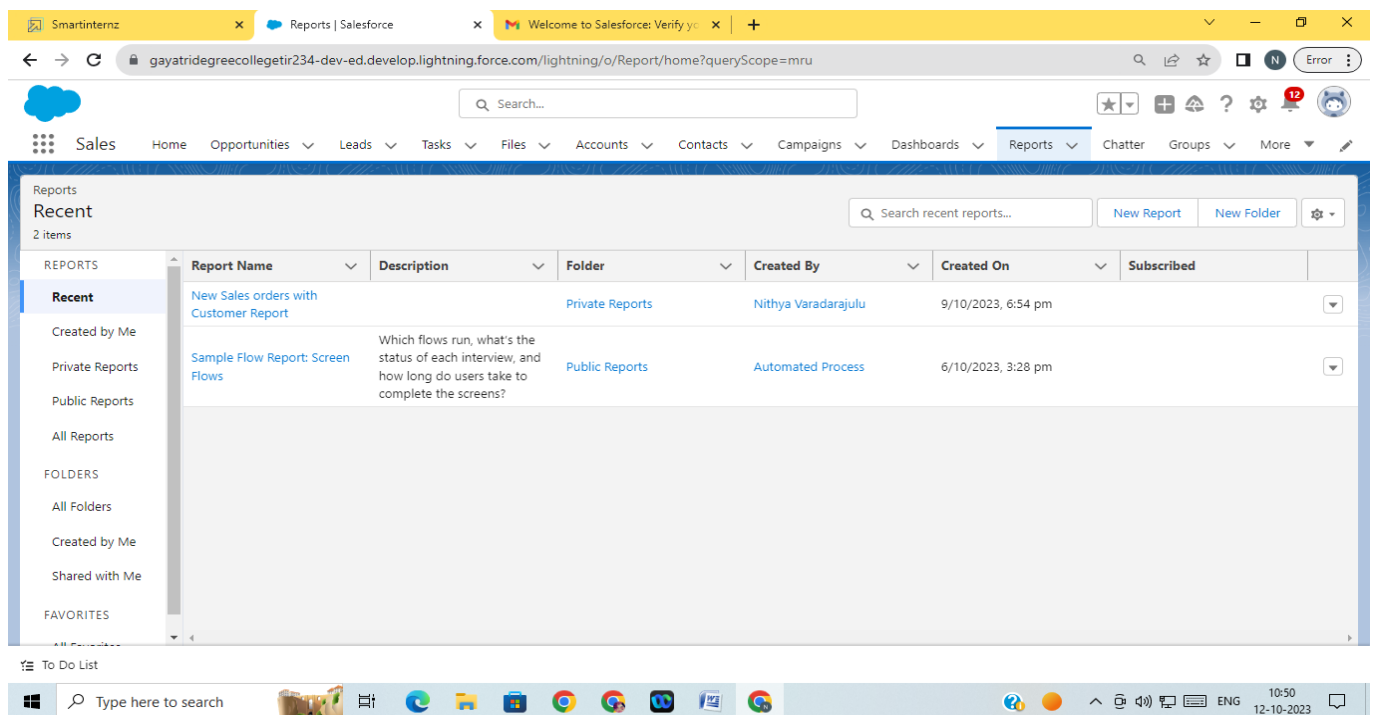
Milestone-09: Report

1. Create report:

2. Click App Launcher and
3. Select Sales App
4. Click reports tab
5. Click New Report.
6. Click the report type as Sales order with customer Click Start report.

View report:

1. Click on App Launcher on left side of screen.
2. Search “Sales App” & click on it.
3. Click on Reports Tab.
4. Click on New Sales orders with Customer Report and see records.



Milestone-10: Dashboard

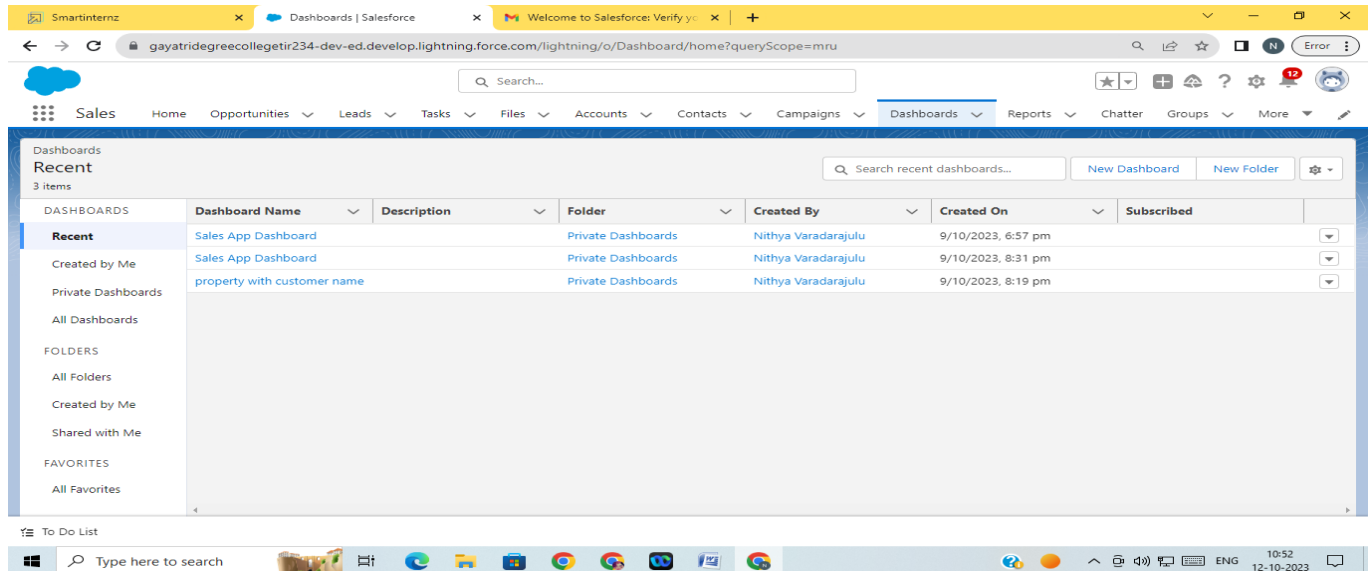
Create dashboard:

1. Click on Dashboards tab from the “Sales App” application,
2. Click on new dashboard
3. Give name- Sales App Dashboard
4. Click create
5. Give your dashboard a name and click on +component
6. Select the New Sales orders with Customer Report which you created.

7. For the data visualization select any of the chart, table etc as your wish.
8. Click add
9. Click save

View dashboard:

1. Click on App Launcher on left side of screen.
2. Search Sales App & click on it.
3. Click on Dashboard Tab.
4. Click on sales app dashboard and see graph view of records.



Milestone-11: Flows

1. Create flow:
2. Click on setup gear
3. In quick find search for Flows
4. Choose new flow option at right side of the page
5. Now select screen flow as a new flow
6. Left side corner of the page you can find a toggle click on that and select a new resource. and select resource type has variable
7. Give api name as Recordid
8. and select data type as Text
9. At bottom for Availability outside the flow check the box as Available for Input
10. Click on done
11. Now below the start button click on add element and choose Get Records

12. Now give the label name as Get Account Record
13. For Get record of object choose object as - Account
14. For Filter account records condition requirements are - All conditions are met
15. Field- Account id Operator- equals Value- Recordid (variable which we had created)
16. For how many records to share - Only the first record
17. How to store record data- Automatically stores all fields.
18. Click on done.
19. Now again add the element below the Get account record and select Screen as your element
20. Give the label name as Get Account & Contact
21. Left side in the component section search for Name and drag it to the screen
22. Give the api name as Name
23. Now drag Email from component section and move it to the screen
24. Give the Api name- Email
25. Required - `{!$GlobalConstant.True}`
26. Now drag the Phone from component to screen below the email
27. Give the Api name as - Phone
28. Required- `{!$GlobalConstant.True}`
29. Now Drag Address from component section to screen
30. Give the Api name as - Address

- a. City Value- `{!Get_Account_Record.ShippingCity}`
- b. Country Value- `{!Get_Account_Record.BillingCountry}`
- c. Postal code- `{!Get_Account_Record.ShippingPostalCode}`
- d. State/province value- `{!Get_Account_Record.ShippingState}`
- e. Street Value- `{!Get_Account_Record.ShippingStreet}`

31. Click on done and save it. Give the label name as Get Account & Contact.

To send an email alert to the customer once order is confirmed.

1. Navigate to setup click on object manager

2. Select sales order as object
3. click on fields and relationships
4. Select Email id field and click on edit
5. Than in general options select Required Field
6. Save it.

Now Create a new field Order confirmed

1. On the same object sales order create a Picklist field.
2. Give the label name as Order confirmed.
3. And in Values give 1) Yes
 - i. 2) No
4. Make it as Required field.

To create record triggered flow

1. Click on setup and search for flows.
2. Than click on new flow
3. Select Record Trigger flow as your flow
4. In Object search for Sales order
5. In Configure Trigger select A record is created
6. In set entry conditions All conditions are Met

7. In fields search for Order confirmed

8. Operator - Equals

9. In values select - Yes

10. In optimize the flow for Select Action and related Records

Now click on add element and in Logic section select Decision element

1. Give the element name as Is Order confirmed

2. In Outcome Details Label as Order Status

3. In condition requirement All Conditions are Met

4. In resource select \$Record than field as Order confirmed

5. Operator - Equals

6. Value as Yes

7. Click on done

8. Give the label as Send an email to customer

9. Now in body select \$Record and than select Id

10. Move your cursor to front before \$record and give the text as Hi Your order with than comes \$record is confirmed

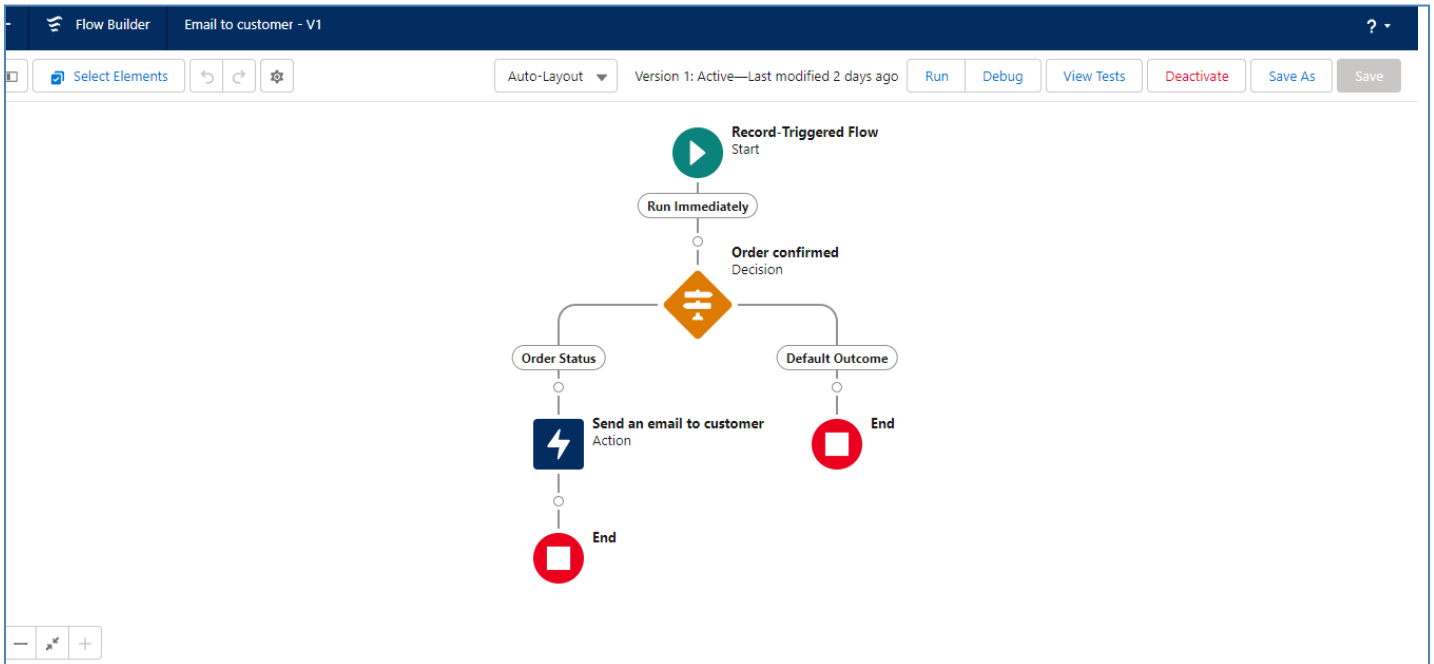
11. In recipients email address list - select Record and than Email id as your field

12. In subject give it as - Order confirmed.

13. Click on done.

14. Save the flow as Email to customer

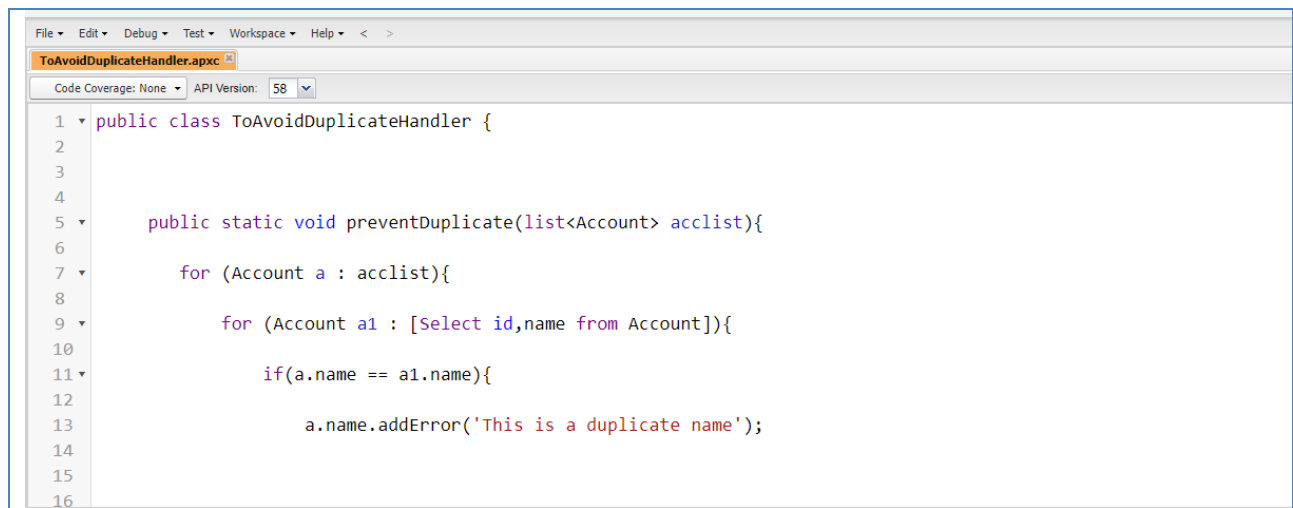
15. And activate the flow.



Milestone-12: Triggers

Trigger on Account to prevent Duplicate Name.

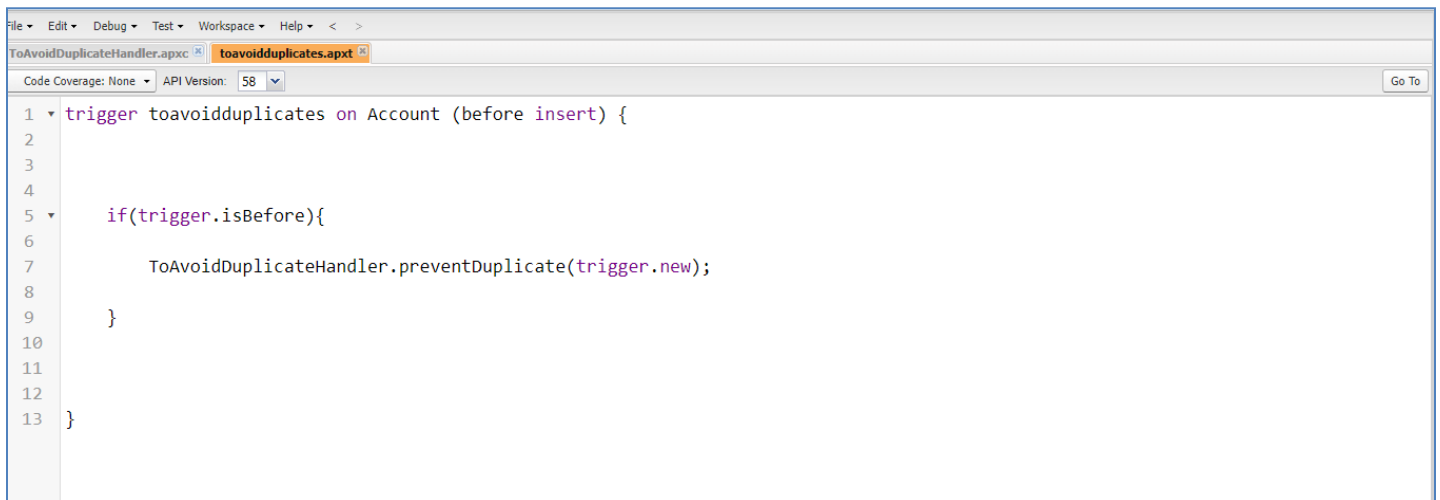
1. Click on Setup and select developer console
2. Click on file and than New
3. Select Apex Class give the name



```
1 public class ToAvoidDuplicateHandler {
2
3
4
5     public static void preventDuplicate(list<Account> acclist){
6
7         for (Account a : acclist){
8
9             for (Account a1 : [Select id,name from Account]){
10
11                 if(a.name == a1.name){
12
13                     a.name.addError('This is a duplicate name');
14
15
16
```

Trigger:

1. Click on Setup and select developer console
2. Click on file and then New
3. Select Apex trigger give the name as Toavoidduplicates
4. Subject as Account.



```
1 trigger toavoidduplicates on Account (before insert) {
2
3
4
5     if(trigger.isBefore){
6
7         ToAvoidDuplicateHandler.preventDuplicate(trigger.new);
8
9     }
10
11
12
13 }
```

The End