KALLAM HARANADHAREDDY INSTITUTE OF TECHNOLOGY

Domain: Salesforce Developer

Branch: Information Technology

Batch:3

Brahma Tejaswini Abburi	218x1a1205@khitguntur.ac.in	tejaswiniabburi7@gmail.com
Peyyela Somaya	218x1a1228@khitguntur.ac.in	somyasudhakar77@gmail.com
Raghukula Mohana Sai Krishna	228x5a1205@khitguntur.ac.in	mohanraghukula@gmail.com
Talari Srihari	218x1a1240@khitguntur.ac.in	srihari.talari64@gmail.com

Project Title: Retail Management Application using Salesforce-DEV

1.Project Overview:

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.

2.Objectives:

Business Goals:

1. **Increase Sales Performance**
   Improve sales performance by 20% within 12 months through enhanced inventory management and sales tracking via Salesforce.
2. **Boost Customer Retention**
   Increase customer retention rate by 15% over the next 6 months through personalized customer experiences driven by Salesforce automation tools.
3. **Reduce Operational Costs**
   Achieve a 10% reduction in operational costs within 12 months by automating retail workflows and processes using Salesforce's cloud platform.
4. **Enhance Customer Service Efficiency**
   Enhance customer service efficiency by reducing response time to customer queries by 25% within 6 months, using Salesforce Service Cloud.

Specific Outcomes:

1. **Inventory Management**

- **Goal**: Connect Salesforce with the inventory system to track stock and sales in real-time.
- **When**: By the end of **Q2**.

2. **Customer Communication**
   - **Goal**: Use Salesforce to send personalized messages to 75% of customers automatically.
   - **When**: By the end of **6 months**.
3. **Sales Reporting**
   - **Goal**: Set up Salesforce dashboards to make sales reports faster and easier.
   - **When**: By the end of the project.
4. **Customer Service**
   - **Goal**: Use Salesforce to handle customer support requests faster.
   - **When**: By the end of **Q1**.
5. **Employee Training**
   - **Goal**: Train 90% of staff to use Salesforce tools effectively.
   - **When**: By the end of **Month 3**.

3. Salesforce Key Features and Concepts Utilized

- Object
- Tab
- App
- Fields and Relationship
- User
- Validation Rules
- User Adoption
- Reports
- Dashboards

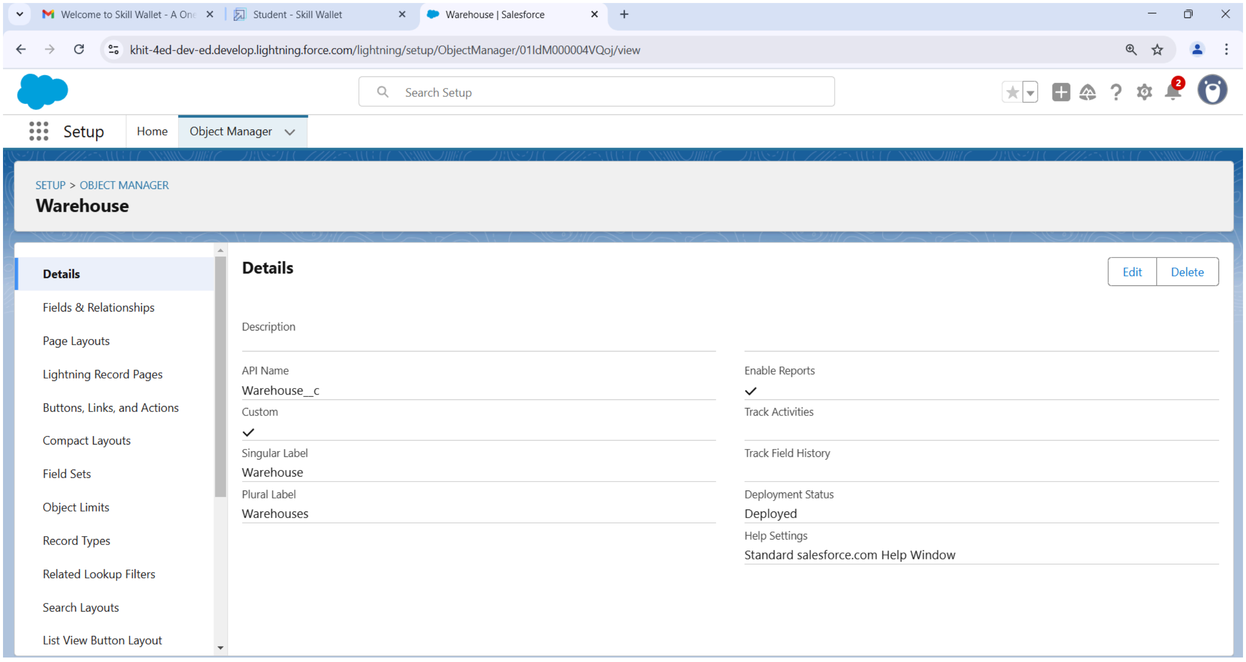
- Flows
- Triggers

4.Detailed steps to solution Design

Object

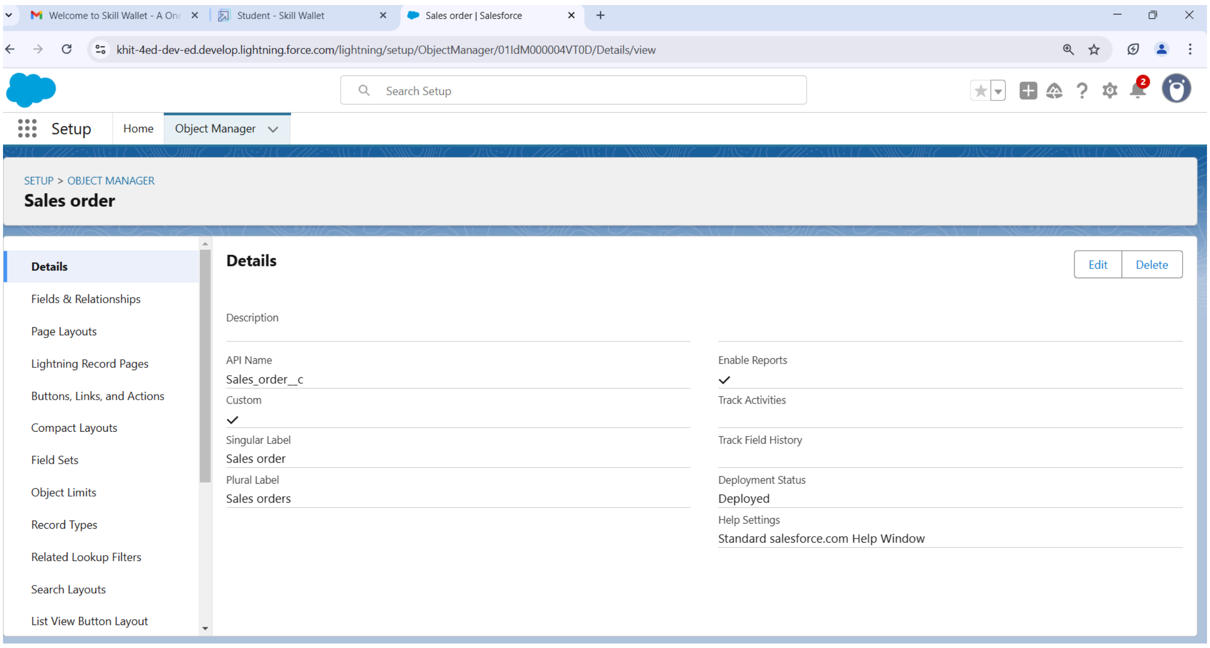
Creation of Warehouse Object

- Click on the gear icon and then select Setup.
- Click on the object manager tab just beside the home tab.
- After the above steps, have a look on the extreme right you will find a Create Dropdown click on that and select Custom Object.
- On the Custom Object Definition page, create the object as follows:
- Label: Warehouse
- Plural Label: Warehouses
- Record Name: Warehouse Name
- Check the Allow Reports checkbox
- Check the Allow Search checkbox
- Click Save.

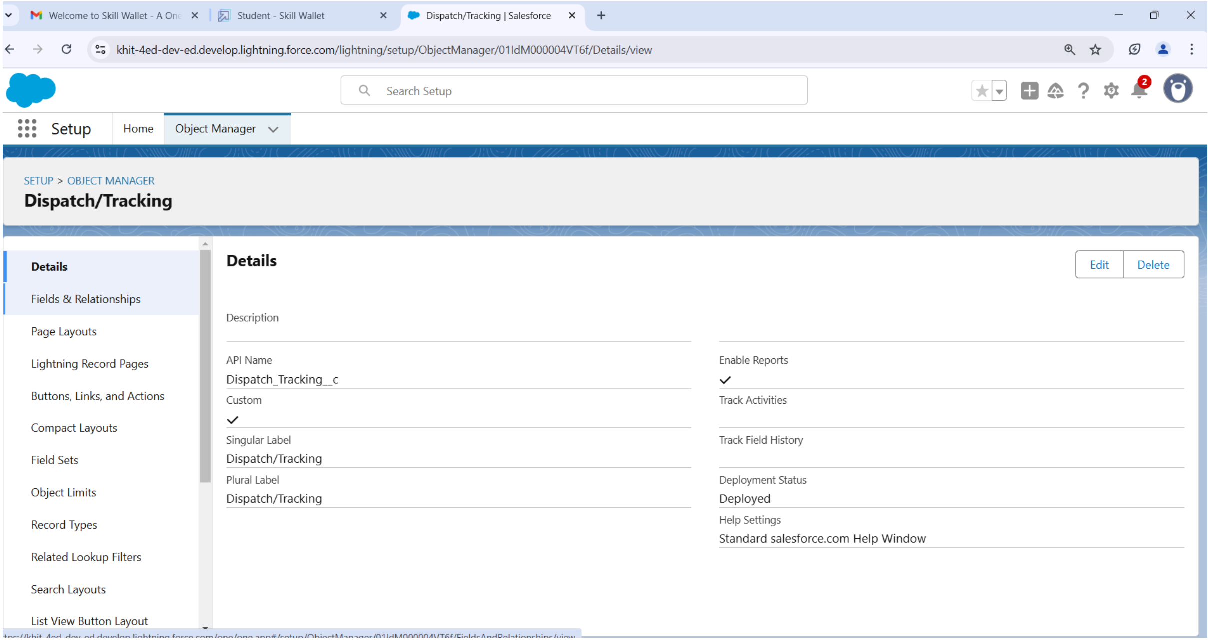


Similarly do the same process for other objects.

Creation of Sales order object



Creation of Planning/Tracking Object



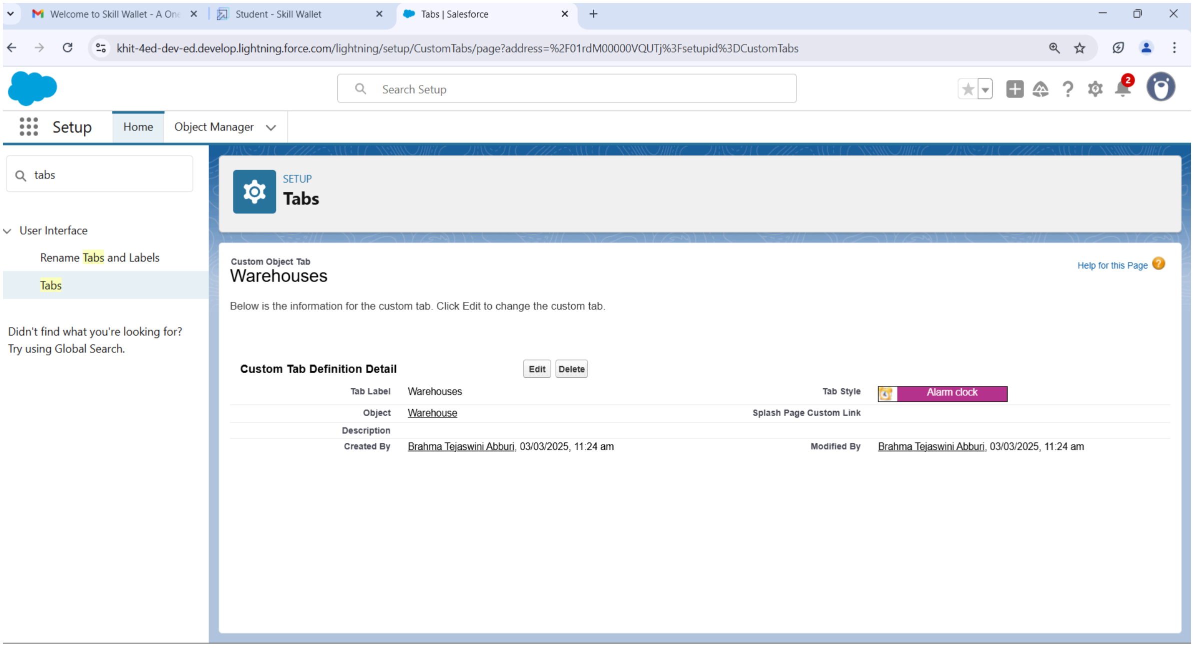
Tab

Tabs in Salesforce help users view the information at a glance. It displays the data of objects and other web content in the application.

Create a tab for Warehouse object

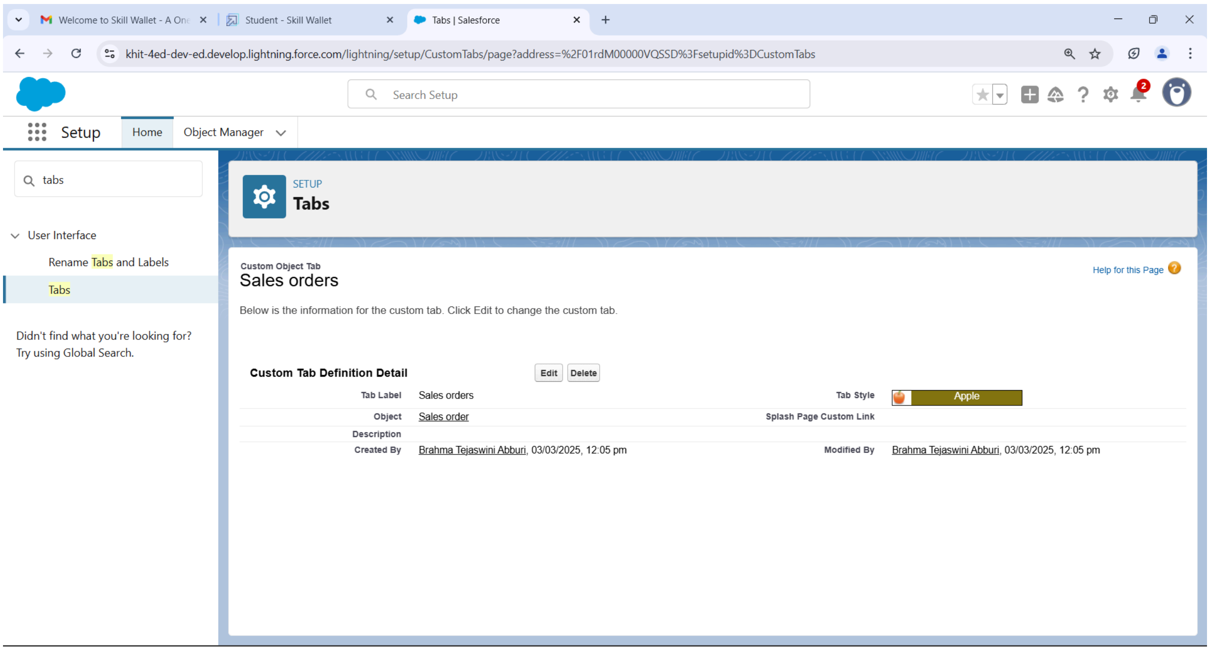
Now create a custom tab. Click the Home tab.

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

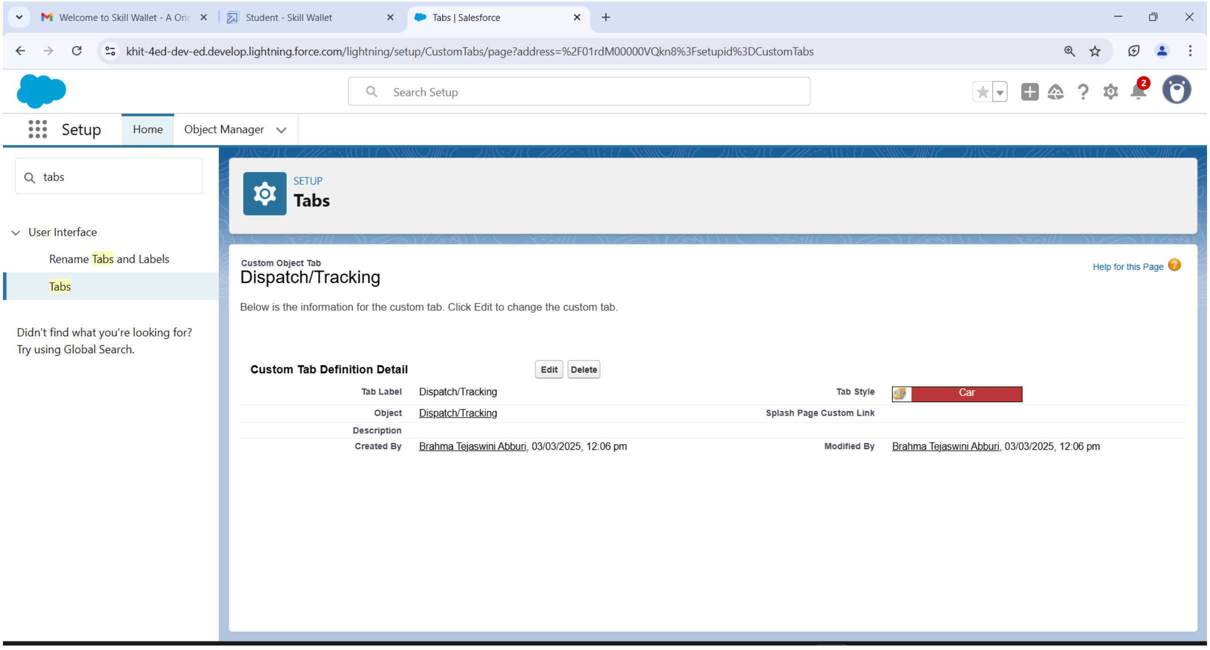


Similarly do the same process for creation of tabs for other objects

Create a tab for sales order object



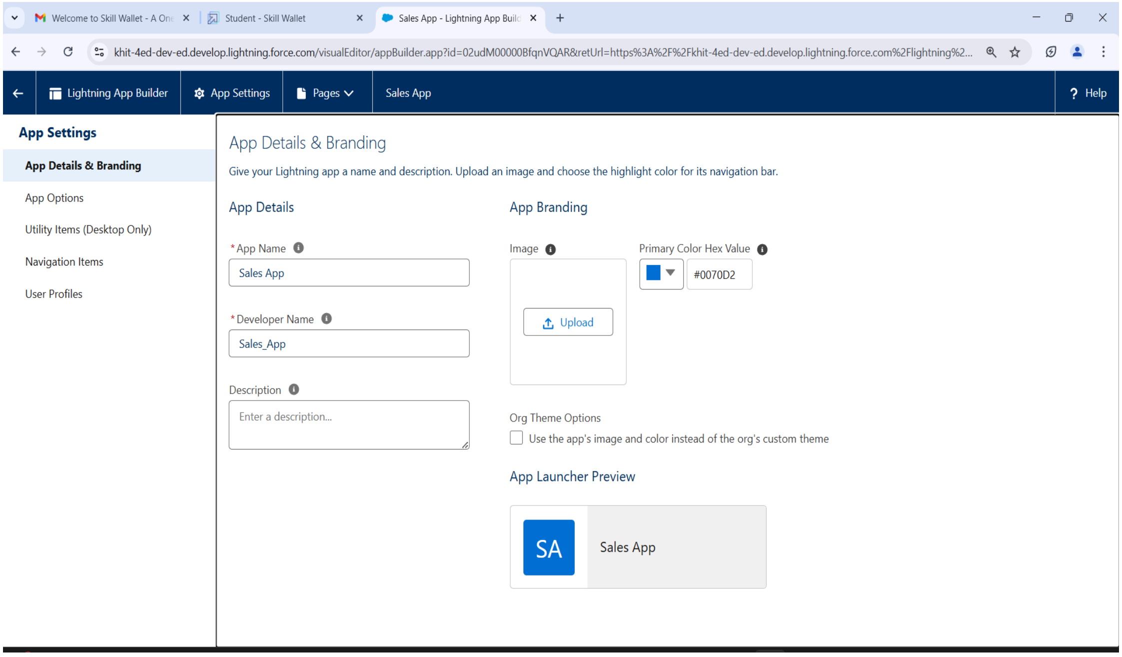
Creation of tab for Dispatch/tracking object



App

Create the sales App

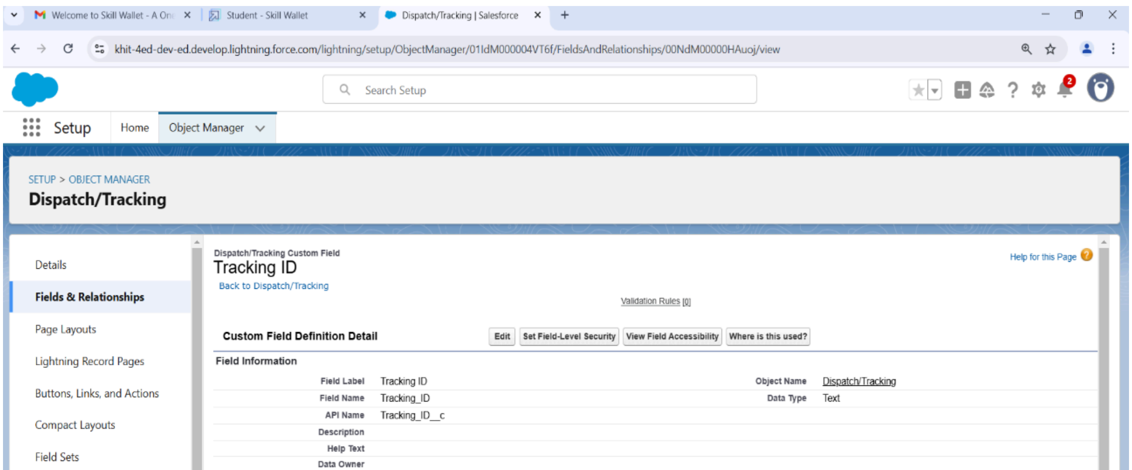
- From Setup, enter App Manager in the Quick Find and select App Manager.
- Click New Lightning App.
- Enter Sales App as the App Name, then click Next
- Under App Options, leave the default selections and click Next.
- Under Utility Items, leave as is and click Next.
- From Available Items, select Campaign,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



Fields and Relationship

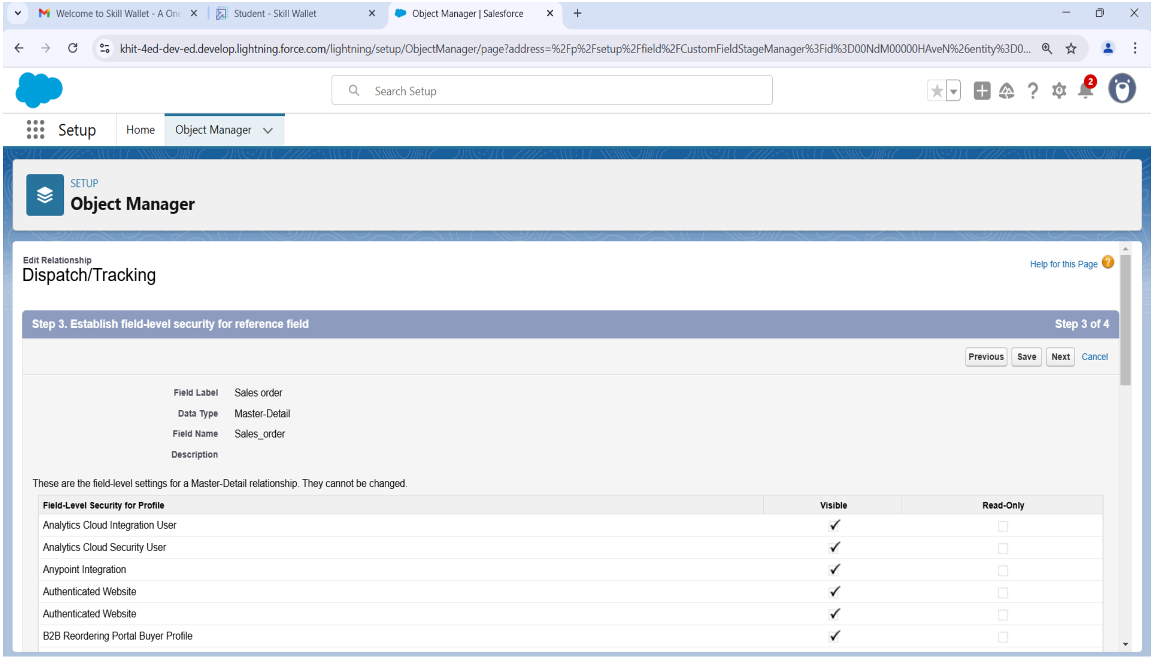
Creation of fields for the Dispatch/Tracking object

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



Create a master-detail relationship on Dispatch/Delivery object

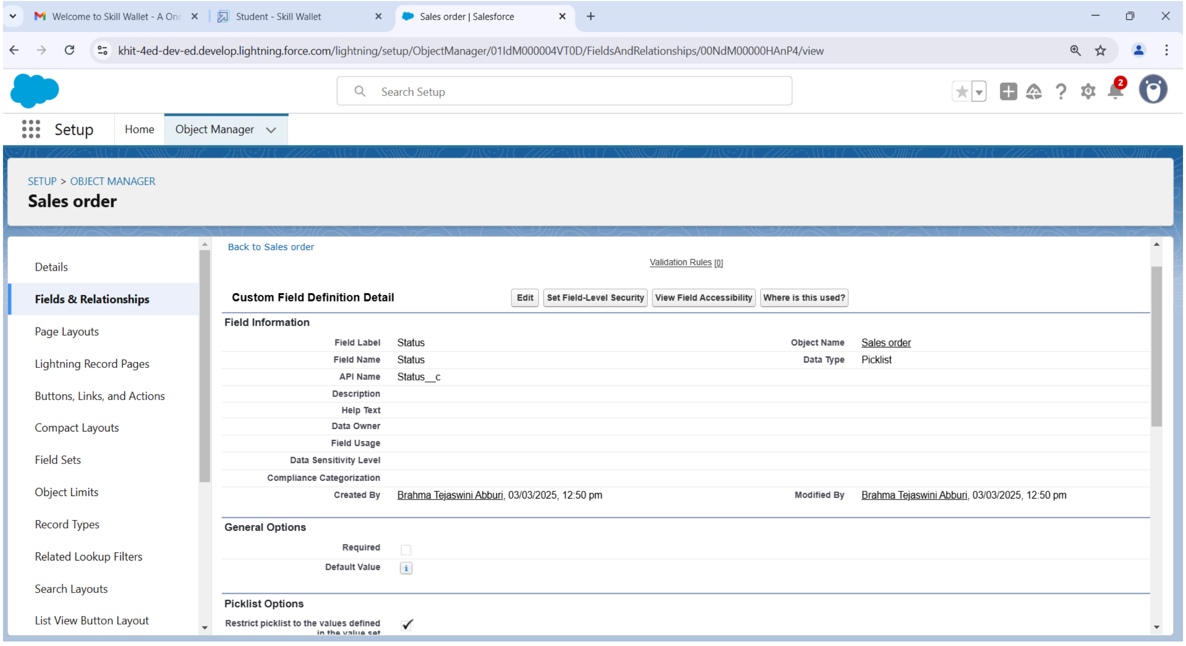
- o Select Master-Detail Relationship as the Data Type and click Next.
- o For Related to, enter Sales order.
- o Click Next.
- o For Field Label, enter Sales order.
- o Click Next, Next, Next and Save



Create a Pick-List field on Sales order

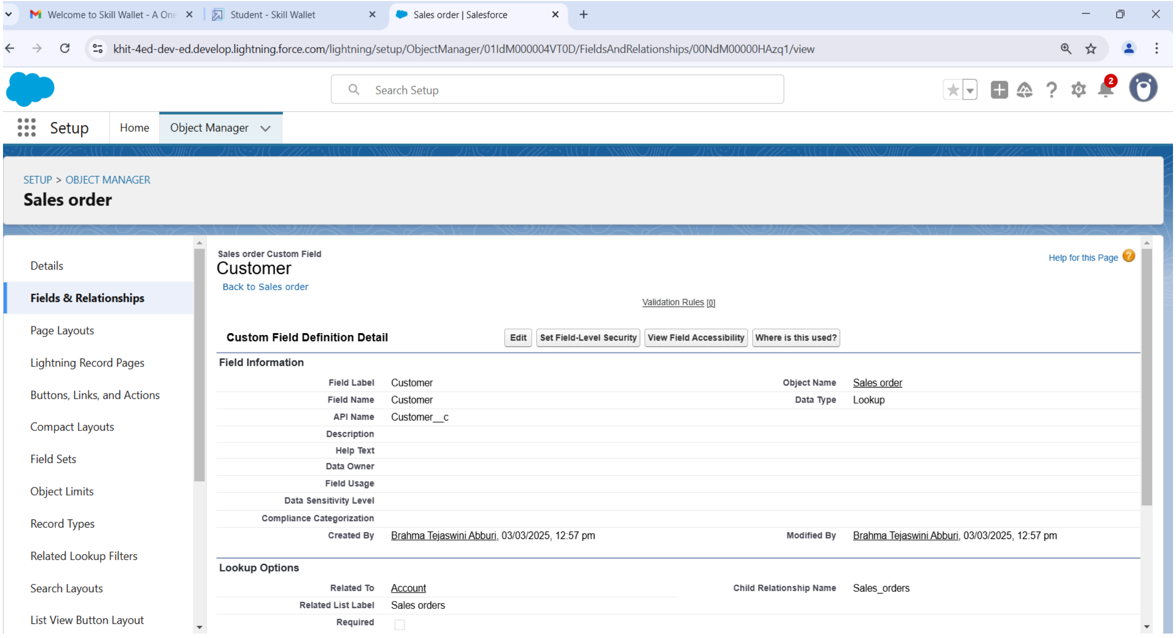
- o From Setup, click Object Manager and select Sales order.
- o Click Fields & Relationships, then New.
- o Select Picklist as the Data Type and click Next.
- o For Field Label enter Status
- o Select Enter values, with each value separated by a new line and enter these values:
- o Open
- o Hold

- Shipped
- Returned
- Click Next, Next, then Save & New.



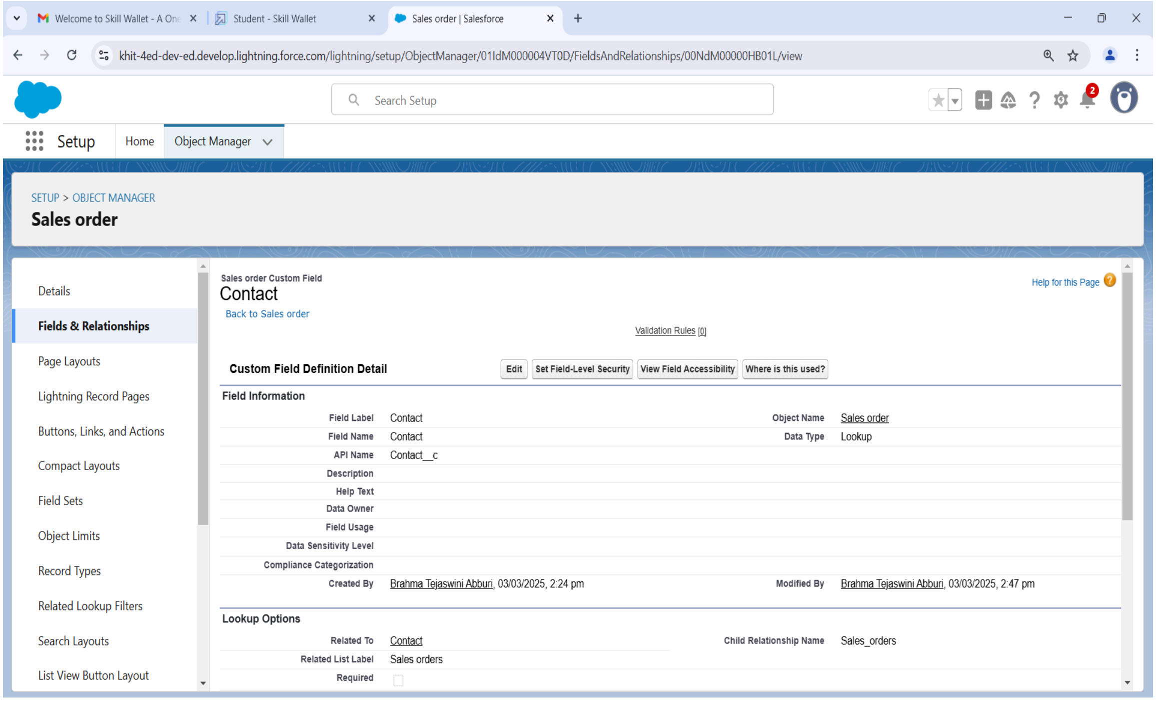
Create a Lookup relationship on Sales Order object with Account object

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



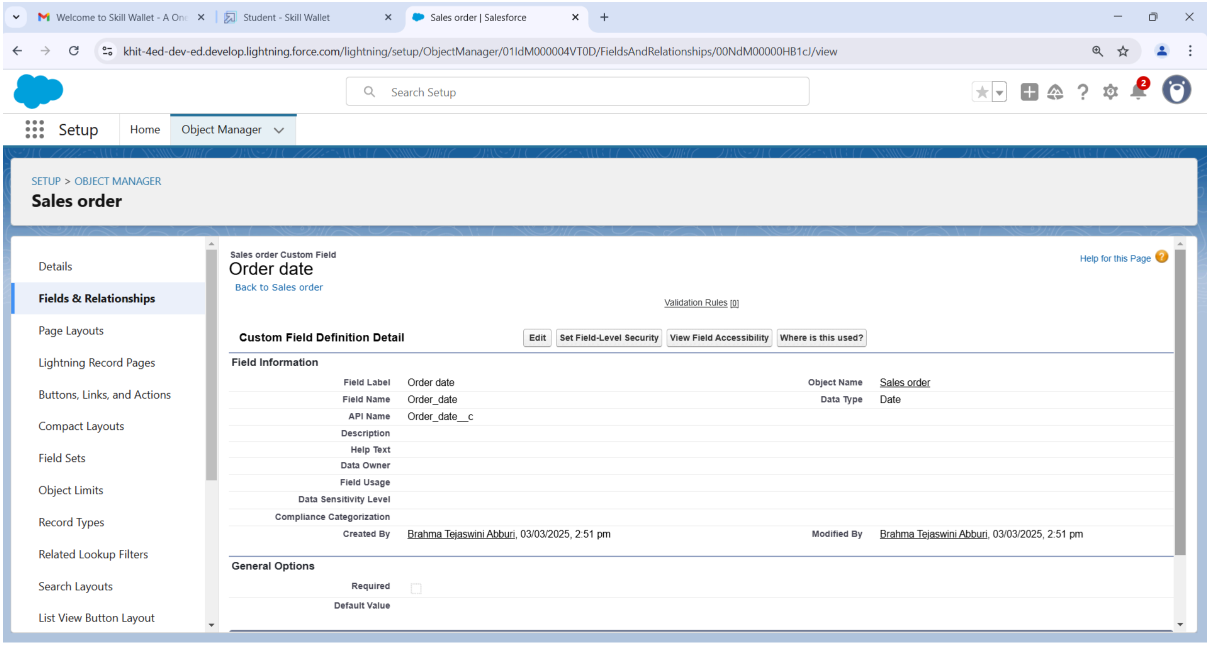
Create a Lookup relationship on Sales Order object with Contact object with use of Lookup Filter

- o Select look up Relationship as the Data Type and click Next.
- o For Related to, enter Contact.
- o Click Next.
- o For Field Label, enter Contact.
- o Click lookup filter.
- o Provide filter as given below & also refer picture (Screenshot of Step 6)
  Contact: Account ID equals Sales Order: Customer
- **o** Click Next, Next, Next and Save.



Create Order date field on Sales Order

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



Cross-object formula field

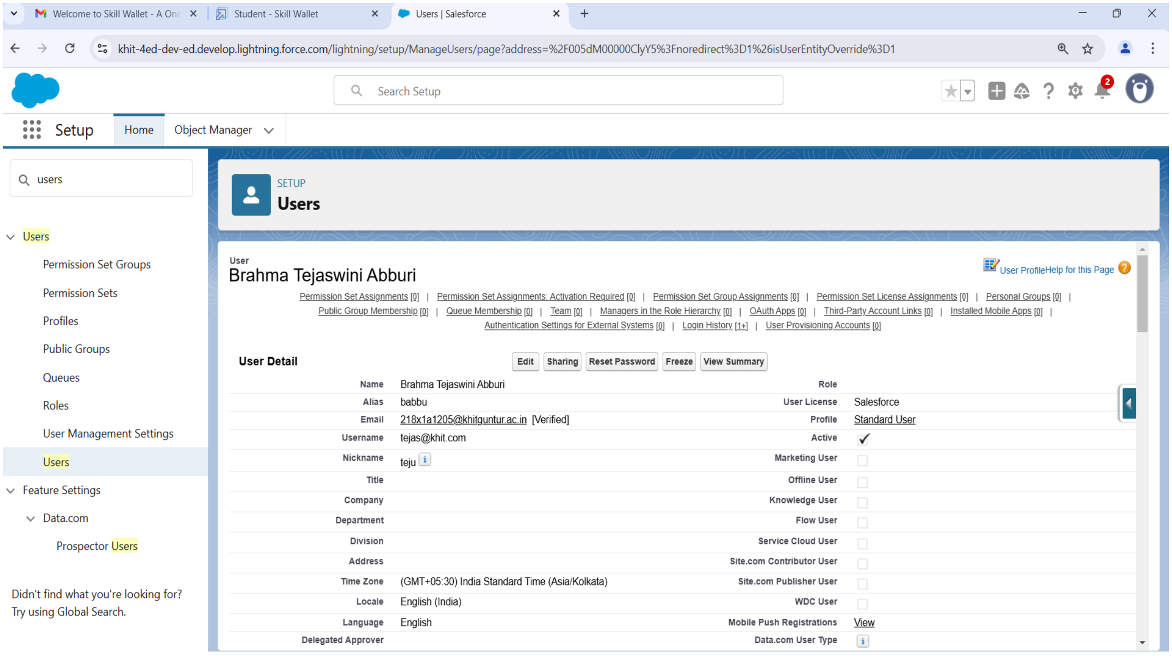
- Select your object from object selection has Contact.
- And select the option fields and relationships.
- At the top right side you can find a new select that option.
- Now you have to select data type as formula.
- And you will navigate to enter the details page where you give the field label.
- And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

User

Creating a User

- From Setup, in the Quick Find box, enter Users.
- Select Users.
- Click New User

- 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.
- Select a User License as Salesforce
- Select Standard User profile
- Check Generate new password and notify the user immediately to have the user's login name and a temporary password emailed to your email

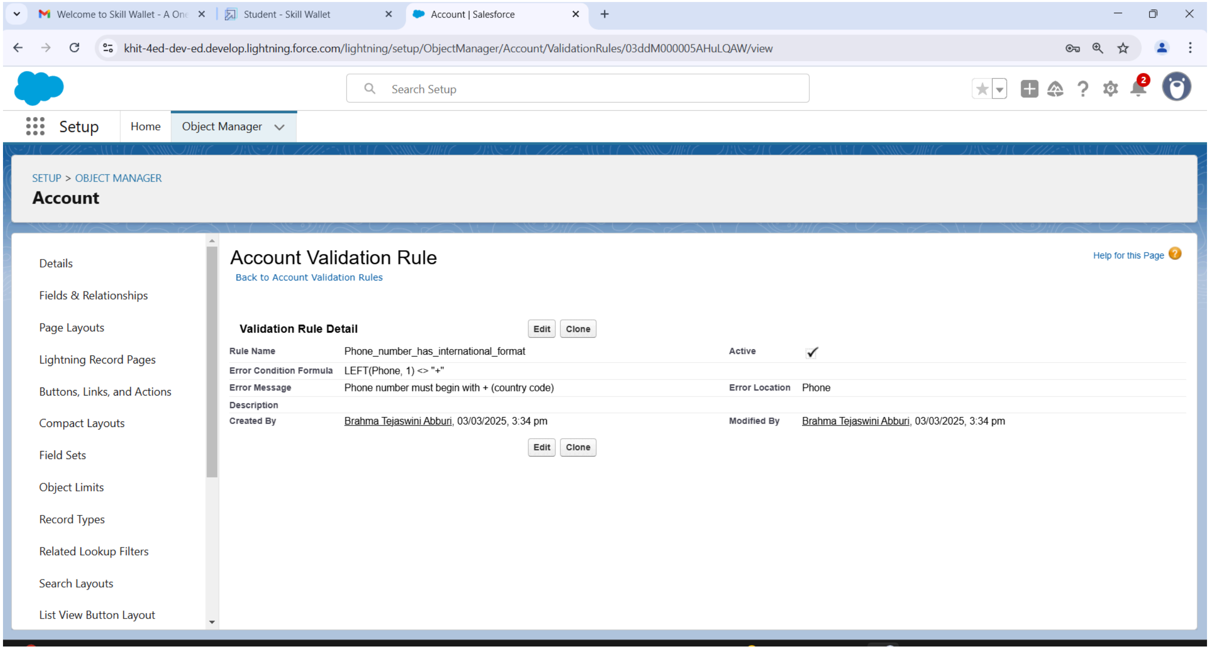


Validation Rules

Creation of validation rule

- Navigate to object manager and select Account object
- In details section scroll down and find validation rule in it.
- Click new, give the label name and in edit error conditional formula give the formula - LEFT (Phone, 1) <> "+".

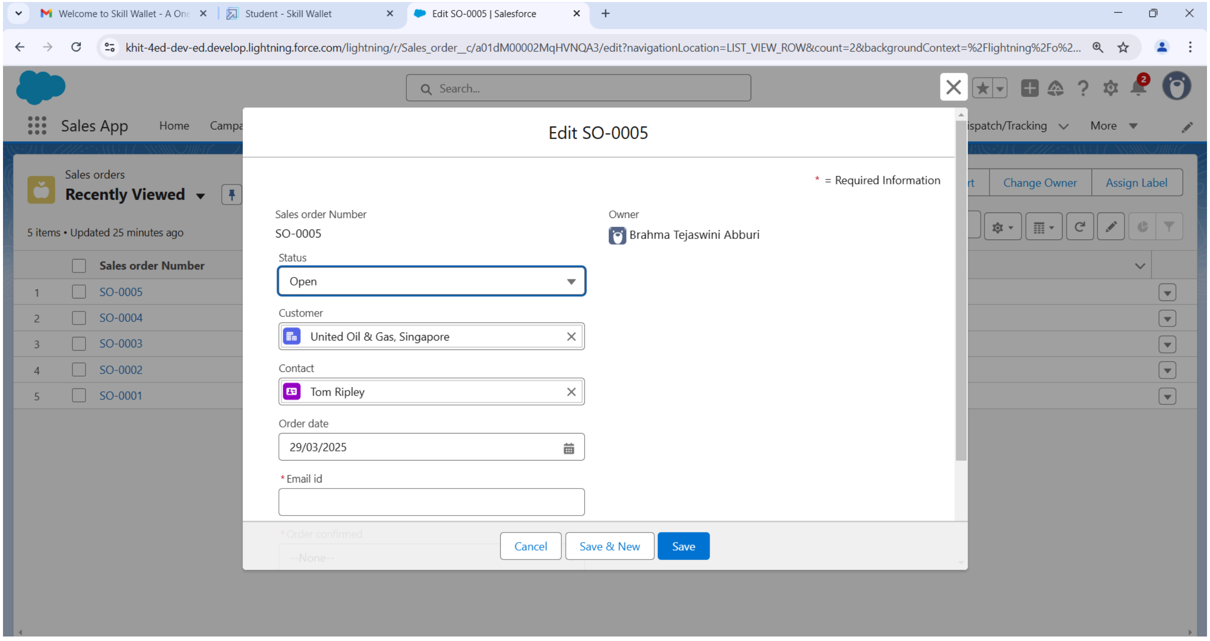
- And in error message give the description has Phone number must begin with + (country code).
- In error location select field.
- Save



User Adoption

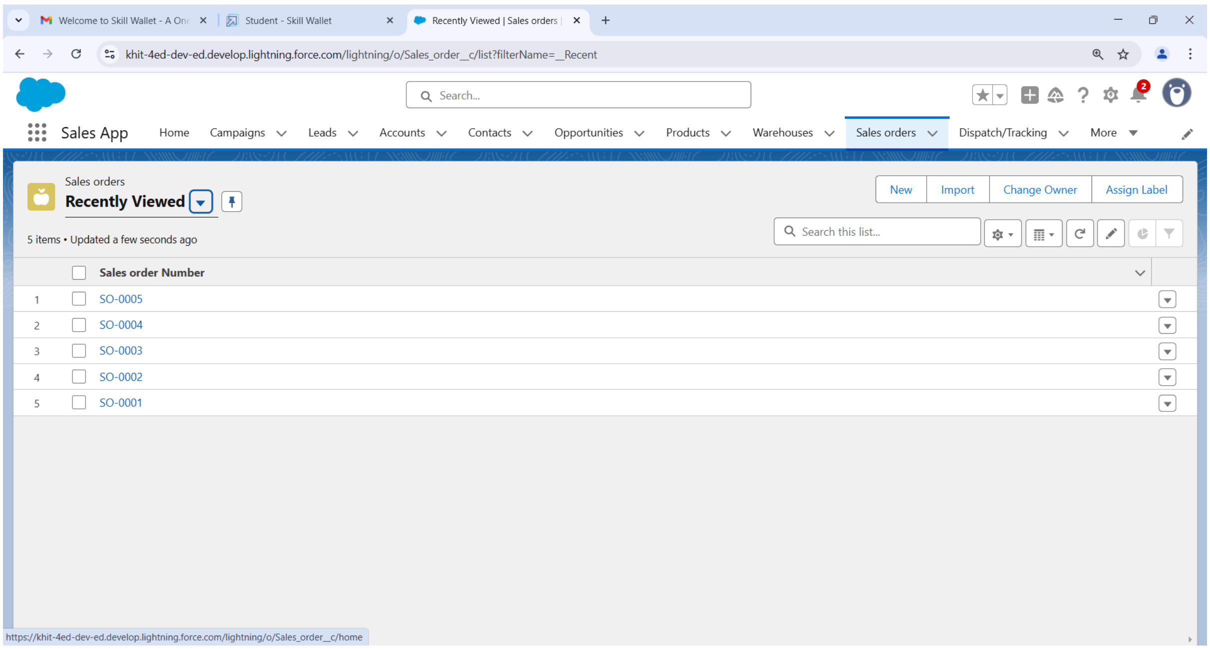
Create Record (Sales Order)

- Click on App Launcher on left side of screen.
- Search Sales App & click on it.
- Click on Sales Order tab.
- Click new button
- Fill all Sales Order record details.
- Click on Save Button



View Record (Sales Order)

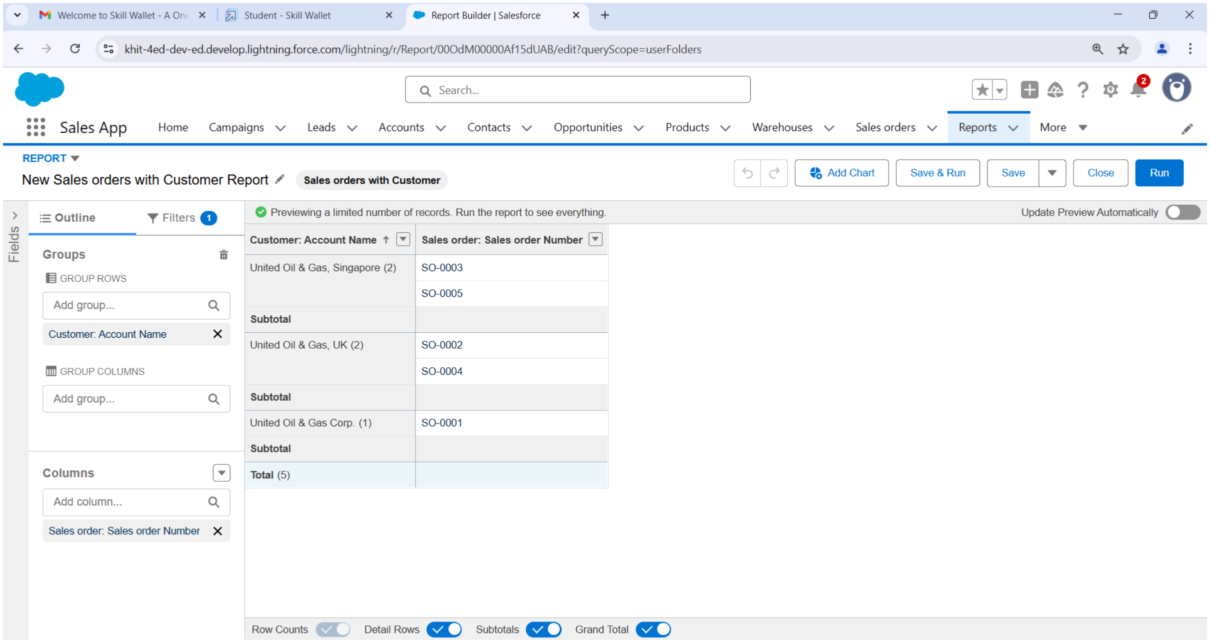
- Click on App Launcher on left side of screen.
- Search Sales App & click on it.
- Click on Sales Order Tab.
- Click on any record name. you can see the details of the Sales Order



Reports

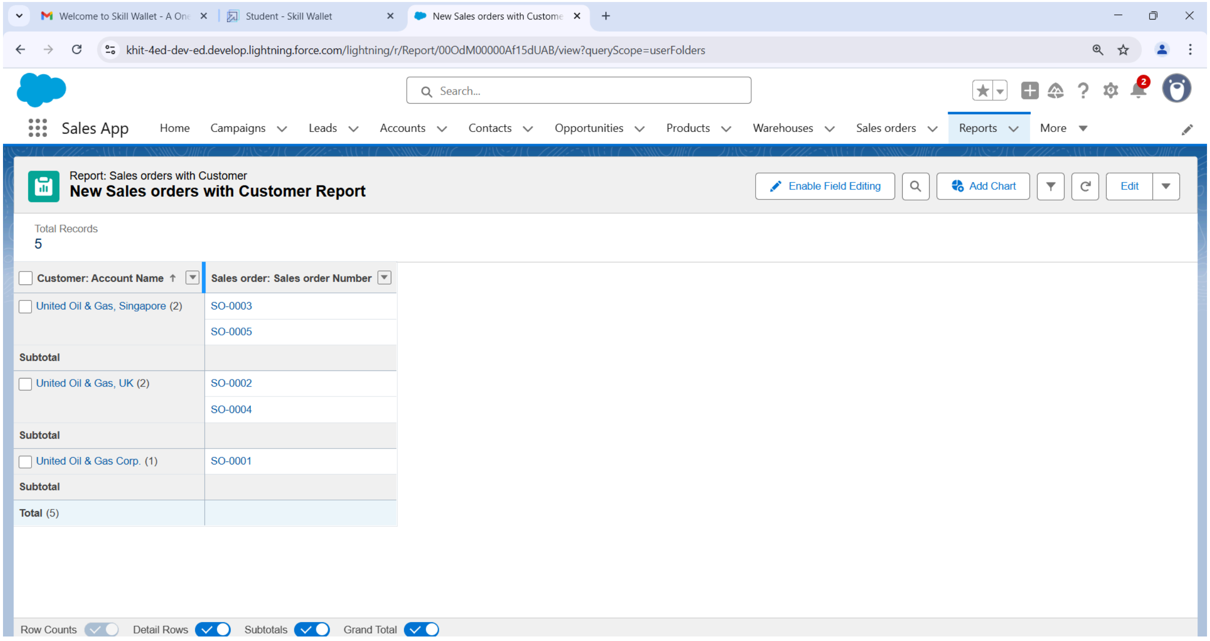
Create Report

- Click App Launcher and
- Select Sales App
- Click reports tab
- Click New Report.
- Click the report type as Sales order with customer Click Start report.
- Customize your report, in group rows select - Customer Account Name
- Click refresh
- Click save and run
- Give report name – New Sales orders with Customer Report
- Click Save



View Report

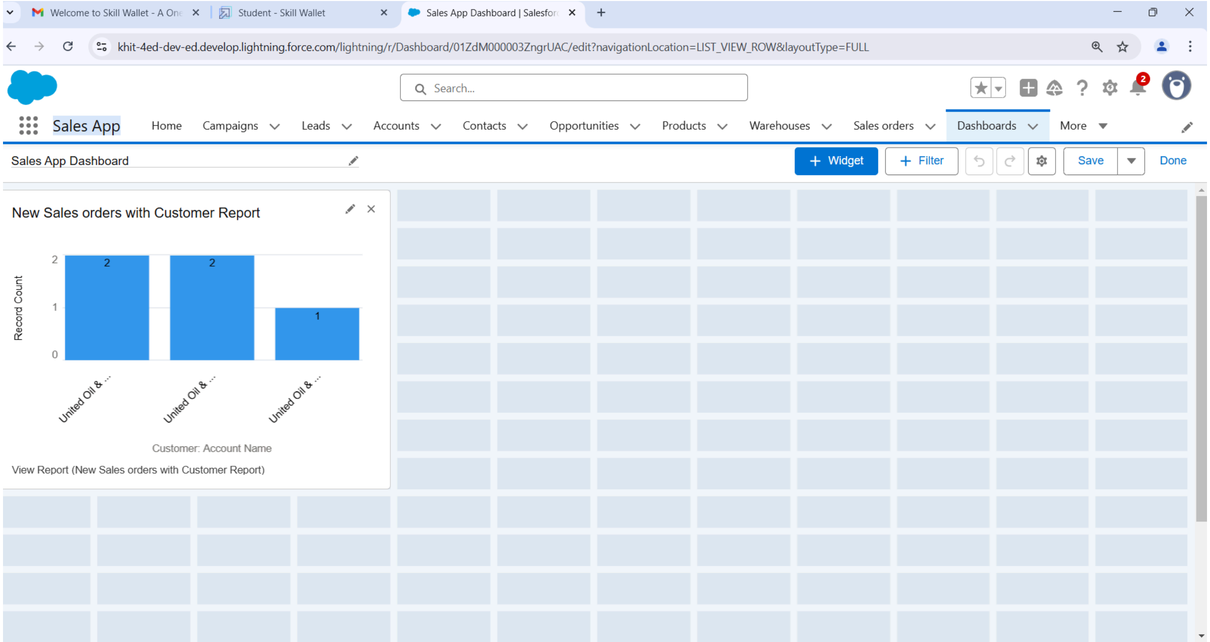
- Click on App Launcher on left side of screen.
- Search “Sales App” & click on it.
- Click on Reports Tab.
- Click on New Sales orders with Customer Report and see records.



Dashboards

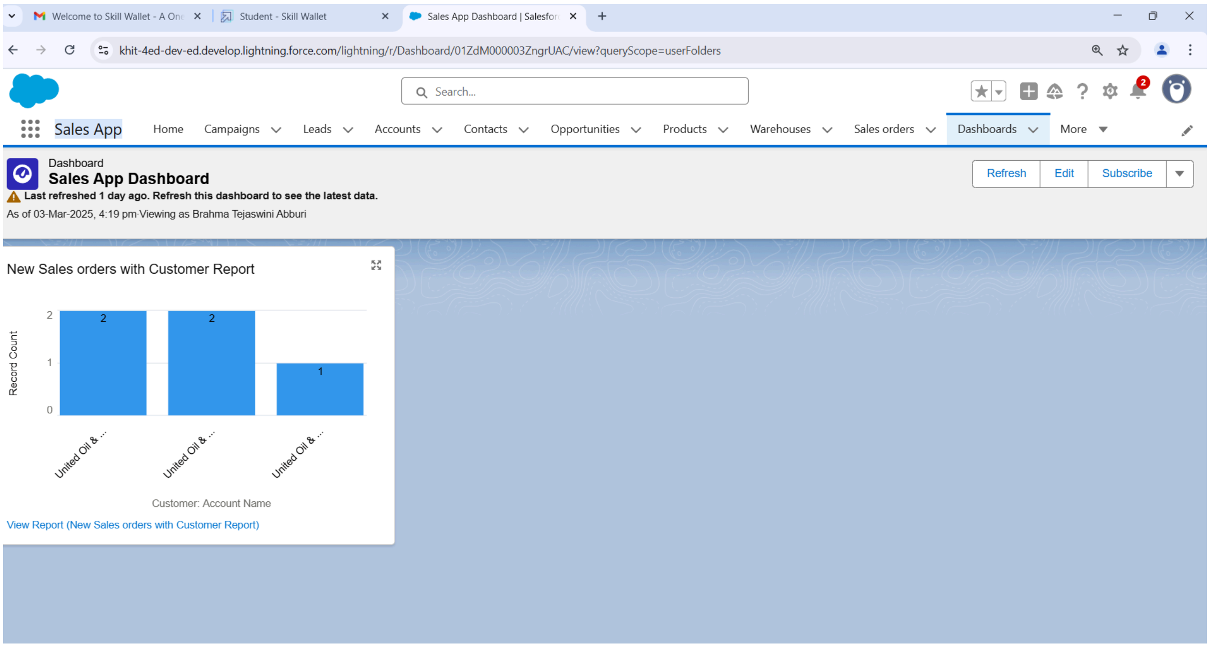
Create Dashboard

- Click on Dashboards tab from the “Sales App” application,
- Click on new dashboard
- Give name- Sales App Dashboard
- Click create
- Give your dashboard a name and click on +component.
- Select the New Sales orders with Customer Report which you created.
  For the data visualization select any of the chart, table etc as your wish.
- Click add
- Click save



View Dashboard

- Click on App Launcher on left side of screen.
- Search Sales App & click on it.
- Click on App Launcher on left side of screen.
- Search Sales App & click on it.

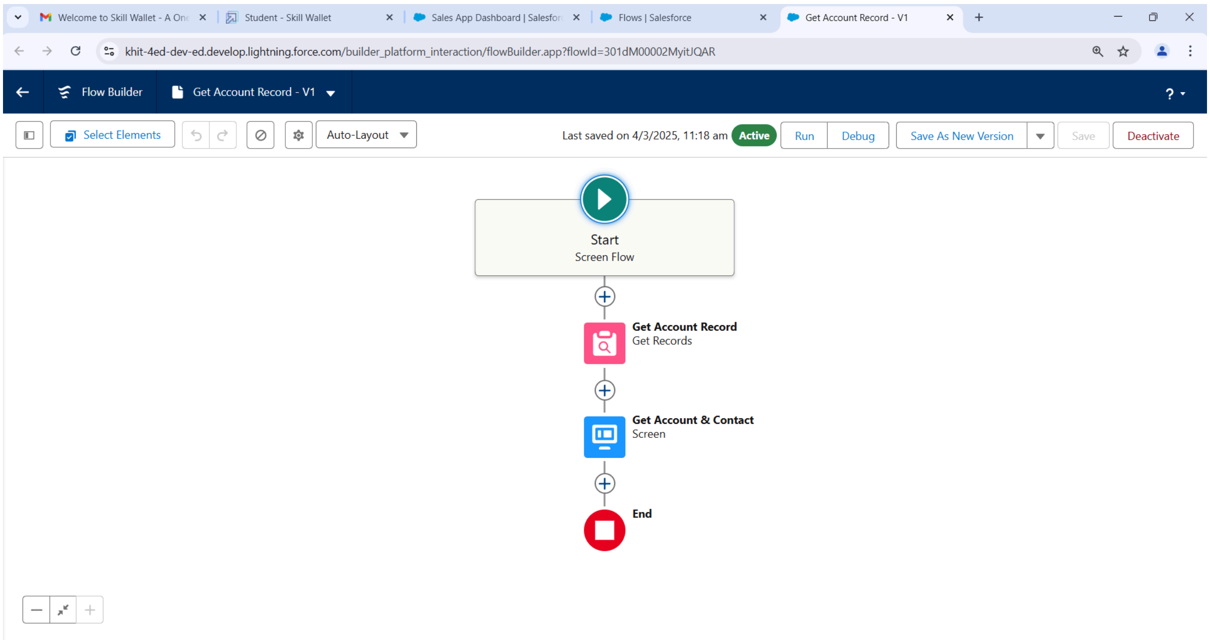


Flows

Create Flow

- o Click on setup gear
- o In quick find search for Flows
- o Choose new flow option at right side of the page
- o Now select screen flow as a new flow
- o Left side corner of the page you can find a toggle click on that and select a new resource.
- o and select resource type has variable
- o Give api name as Recordid
- o and select data type as Text
- o At bottom for Availability outside the flow check the box as Available for Input
- o Click on done
- o Now below the start button click on add element and choose Get Records
- o Now give the label name as Get Account Record
- o For Get record of object choose object as - Account
- o For Filter account records condition requirements are - All conditions are met
- o Field- Account id Operator- equals Value- Recordid (variable which we had created)
- o For how many records to share - Only the first record
- o How to store record data- Automatically stores all fields.
- o Click on done.
- o Now again add the element below the Get account record and select Screen as your element
- o Now again add the element below the Get account record and select Screen as your element
- o Give the label name as Get Account & Contact
- o Left side in the component section search for Name and drag it to the screen
- o Give the api name as Name
- o Now drag Email from component section and move it to the screen
- o Give the Api name- Email
- o Required - {! $ GlobalConstant.True}

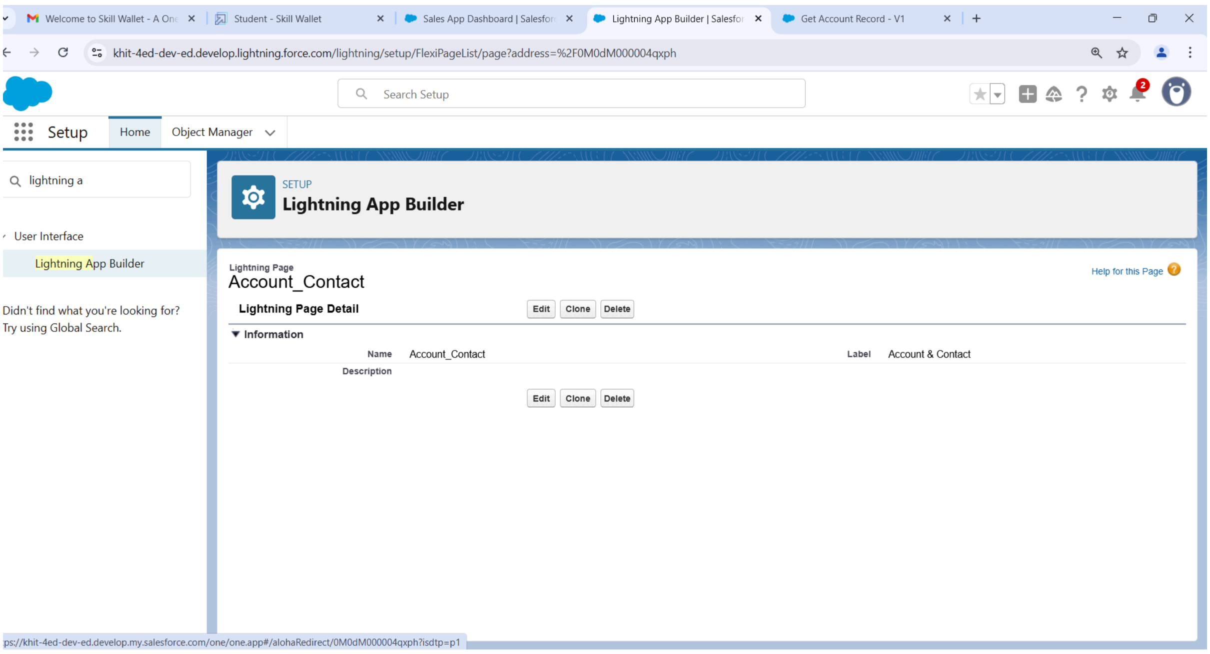
- Now drag the Phone from component to screen below the email
- Give the Api name as - Phone
- Required- {! $ Global Constant. True}
- Now Drag Address from component section to screen
- Give the Api name as - Address
    - City Value- {! Get _ Account _ Record. Shipping City}
    - Country Value- {! Get _ Account_ Record. Billing Country}
    - Postal code- {! Get _ Account_ Record. Shipping Postal Code}
    - State/province value- {! Get _ Account _ Record. Shipping State}
    - Street Value- {! Get_ Account _Record. Shipping Street}
- Click on done and save it. Give the label name as Get Account & Contact



To Create Lightning home page

- Click on setup gear.
- Now search for lightning App builder.
- And select Newoption
- In create a new lightning page select Home page.
- Select Next
- Give the label name - Account & Contact

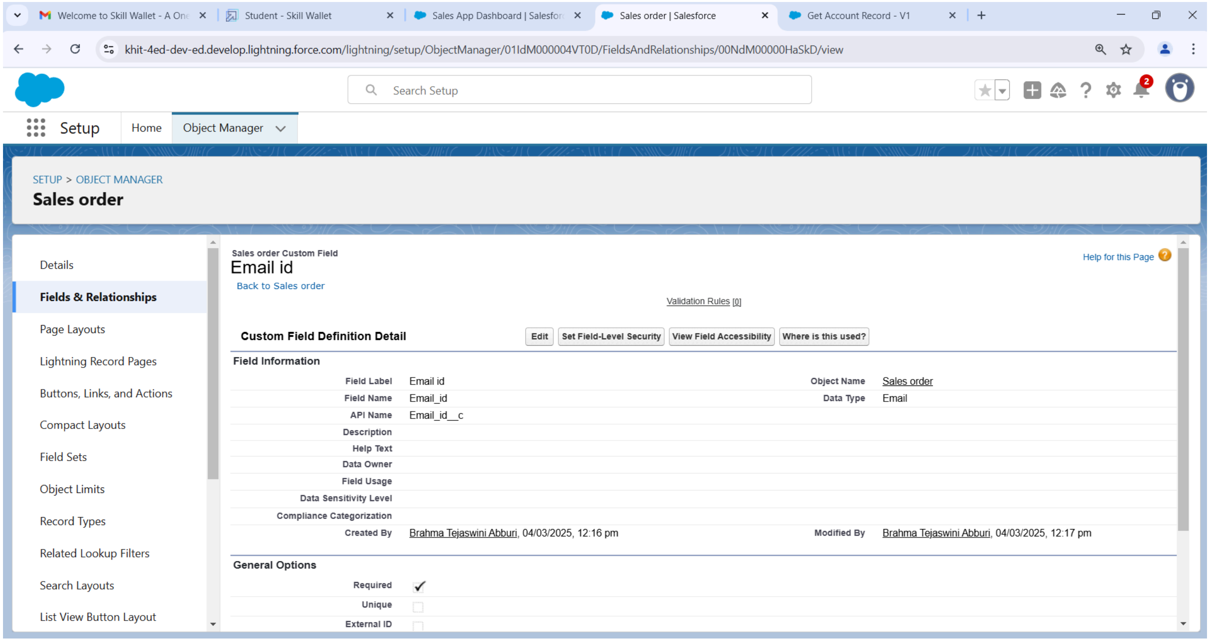
- Choose a standard home page.
- Now in the component section select flow and drag down it to Corner of the page.
- At the right side select the flow Get Account & Contact
- at the rightside top of the page click on Save.
- You will get the populate notification and click on activate.
- you will get an activation pop up select App and profile.
- Select Sales app in lightning app selection.
- In profiles select System administrator, Standard user, Standard platform user.
- Save it.
- Now click on app launcher and search for Sales App
- At the right sidecorner you can find a Pencil icon to personalize navigation click on that.
- Click on add more items and in available items click on all and search for home.
- Move the home page to top and click on save



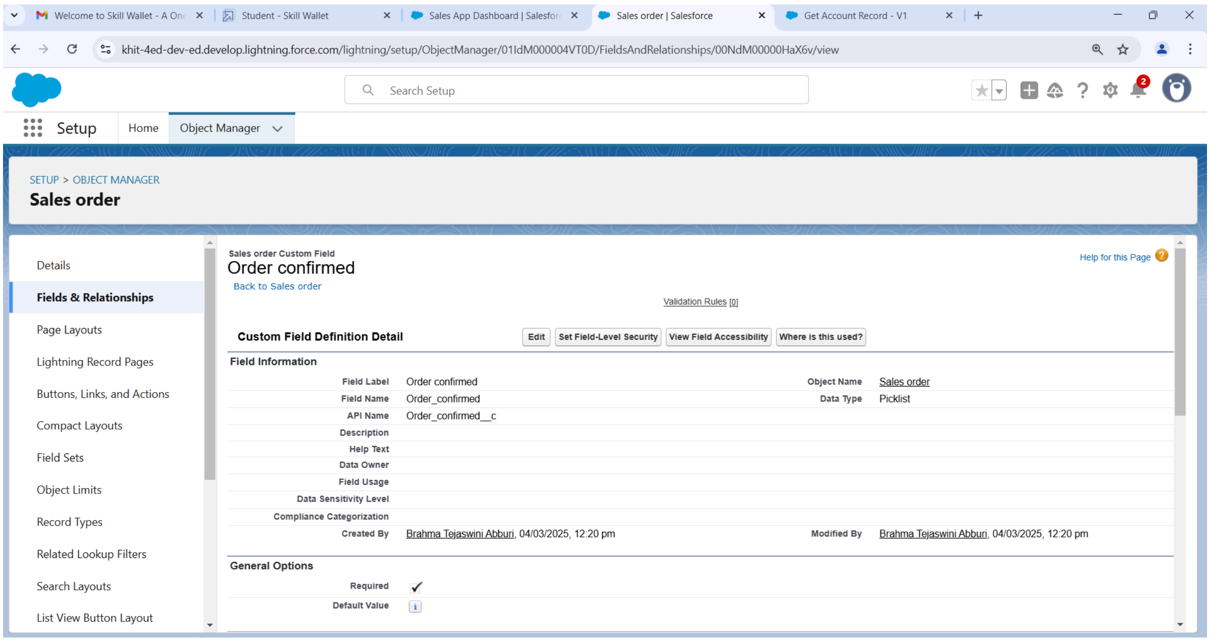
To send an email alert to the customer once order is confirmed

- Navigate to setup click on object manager
- Navigate to setup click on object manager
- Select sales order as object
- click on fields and relationships

- Select Email id field and click on editThan in general options select Required Field and save it



- Now Create a new field Order confirmed
- On the same object sales order create a Pikclist field
- Give the label name as Order confirmed.
- And in Values give 1) Yes 2) No
- Make it as Required field.



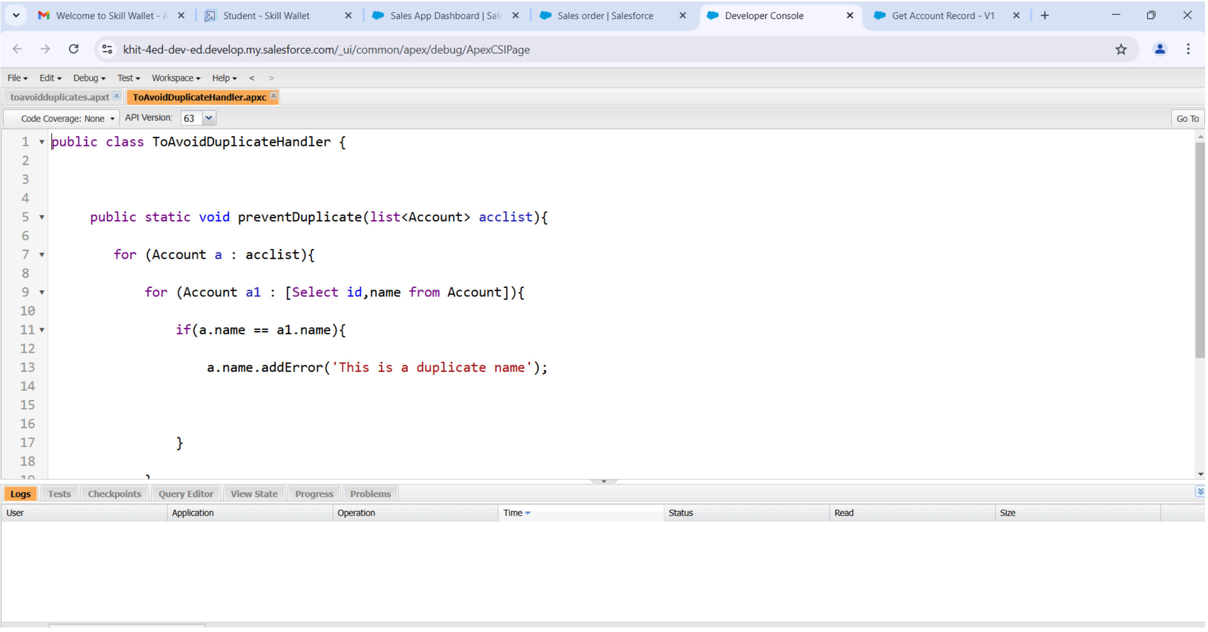
5.Testing and Validation

Triggers

Trigger on account to prevent Duplicate Name

- Click on Setup and select developer console
- Click on file and than New
- Select Apex Class give the name as Toavoidduplicateshandler

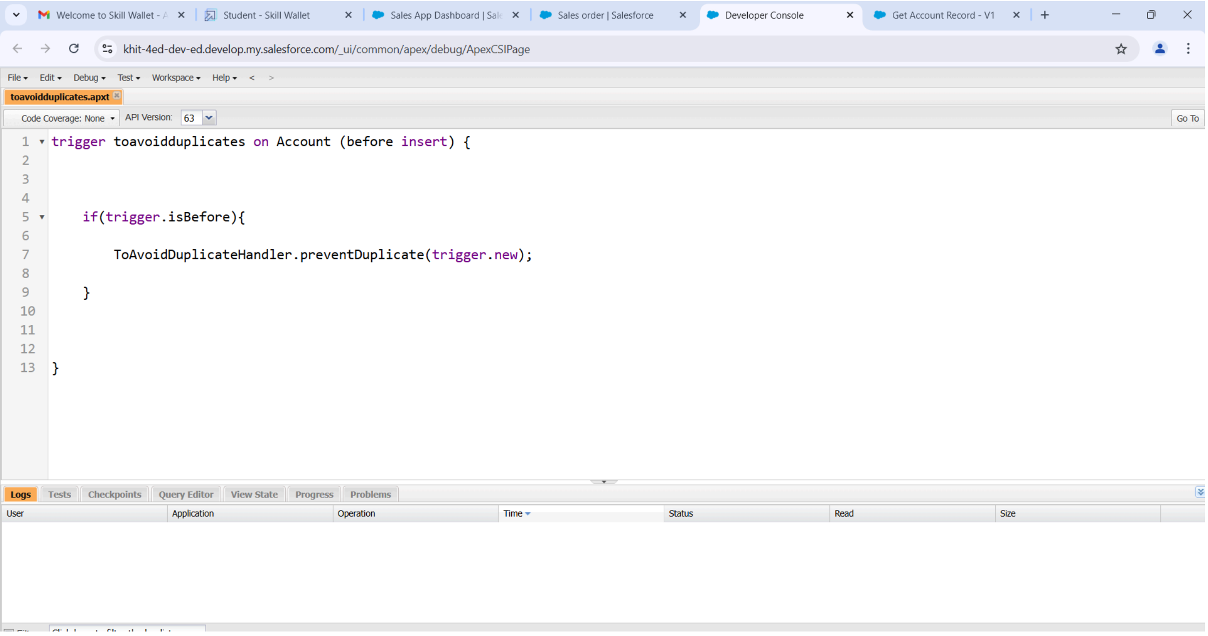
```
 public class ToAvoidDuplicateHandler {


   public static void preventDuplicate(list<Account> acclist){

     for (Account a : acclist){

         for (Account a1 : [Select id,name from Account]){SS

             if(a.name == a1.name){

                 a.name.addError('This is a duplicate name');


             }

         }

     }


   }

}
```



Trigger

- Click on Setup and select developer console
- Click on file and than New
- Select Apex trigger give the name as Toavoidduplicates
- Sobject as Account.

```
trigger toavoidduplicates on Account (before insert) {

    if(trigger.isBefore){
        ToAvoidDuplicateHandler.preventDuplicate(trigger.new);



    }

}
```



6.Key Scenarios Addressed by Salesforce in the Implementation Project

Here's a simplified breakdown of how Salesforce can be used in a retail management application implementation:

1. Customer Management

Problem: Managing customer information.

Solution: Salesforce helps track customer details, purchase history, and preferences to improve service and engagement.

2. Sales and Order Management

Problem: Handling sales and order processes.

Solution: Salesforce automates sales tracking, helps manage orders, and ensures that sales teams can easily process customer transactions.

3. Inventory Management

Problem: Keeping track of stock levels.

Solution: Salesforce can track product availability and manage stock, ensuring that retailers don't run out of popular items.

4. Customer Support

Problem: Offering good post-purchase support.
Solution: Salesforce helps manage customer service requests and supports faster issue resolution.

5. Marketing Campaigns

Problem: Running marketing campaigns to attract customers.
Solution: Salesforce automates email campaigns, social media promotions, and sends targeted offers based on customer preferences.

6. Reports and Insights

Problem: Accessing data for better decisions.
Solution: Salesforce provides easy-to-read reports and dashboards to analyze sales, customer behavior, and campaign performance.

7. Multi-Channel Sales

Problem: Managing sales across different channels (online, in-store, mobile).
Solution: Salesforce connects all sales channels, providing a single view of sales, no matter where they happen.

8. Personalized Recommendations

Problem: Recommending products to customers.
Solution: Salesforce uses AI to suggest products to customers based on their past shopping behavior.

9. Loyalty Programs

Problem: Managing customer loyalty rewards.
Solution: Salesforce helps track loyalty points and run promotions for repeat customers.

10. Vendor Management

Problem: Managing suppliers and inventory restocking.
Solution: Salesforce connects with suppliers, making it easier to track orders and deliveries.

11. Omni-Channel Support

Problem: Providing support through multiple communication channels (phone, email, chat).
Solution: Salesforce helps agents manage customer support from any channel in one place.

12. Mobile Access for Sales Teams

Problem: Sales teams needing data on-the-go.
Solution: Salesforce allows sales teams to access important data from their mobile devices.

13. Team Collaboration

Problem: Teams needing to collaborate.
Solution: Salesforce offers tools for employees to communicate and share information easily across teams.

14. Data Security

Problem: Keeping customer data secure and compliant.
Solution: Salesforce ensures customer data is protected and complies with regulations like GDPR.

7.Conclusion

In conclusion, implementing Salesforce for the retail management application has led to significant improvements across various areas. Customer data is now easily accessible, enabling more personalized service and enhancing customer satisfaction. The sales process has been streamlined through automation, making order management faster and more efficient. Inventory tracking has been optimized, preventing stockouts and overstocking. Customer support has become quicker and more effective, improving the overall customer experience. Marketing campaigns are more targeted, leading to higher engagement and better results. Real-time reports and data insights have empowered the business to make smarter decisions regarding sales and inventory. Sales channels, including online, in-store, and mobile, are now seamlessly integrated, providing a smooth shopping experience for customers. AI-driven product recommendations have boosted sales and personalized the shopping journey. Loyalty programs are easier to manage, fostering repeat business and customer retention. Finally, Salesforce has ensured secure global operations with support for multiple currencies and compliance with data protection regulations. Overall, Salesforce has significantly enhanced efficiency, customer service, and growth within the retail business.