Medical Inventory Management

College Name: KPR COLLEGE OF ARTS SCIENCE AND RESEARCH

College Code: bruaz

TEAM ID:

NM2025TMID21391

TEAM MEMBERS:

Team Leader Name: Preethi R

Email: 23bda047@kprcas.ac.in

Team Member1: Laksmiga V

Email: 23bda030@kprcas.ac.in

Team Member 2: Sangami Preetha T

Email: 23bda054@kprcas.ac.in

Team Member 3: Shreyas S

Email: 23bda058@kprcas.ac.in

Team Member 4: Suba Priya S

Email: 23bda063@kprcas.ac.in

INTRODUCTION

Project Overview :

This project is a comprehensive Salesforce application to streamline and manage various operational aspects of medical inventory. The system aims to efficiently maintain supplier details, manage purchase orders, track product details and transactions, and monitor the expiry dates of products. Maintain detailed records of suppliers, including contact information. Catalog product information, including descriptions, stock levels. Monitor and track product expiry dates to avoid using expired items. Comprehensive reports to track supplier performance, and purchase orders.

Objectives :

- **Streamline Supplier Management:** Centralize and maintain supplier records to enhance communication.
- **Enhance Inventory Control:** Accurately catalog and track products and stock levels in real time.
- **Automate Expiry Monitoring:** Proactively track expiry dates with automated alerts to reduce waste.
- **Optimize Purchase Orders:** Create a seamless system for generating and tracking purchase orders.
- **Improve Operational Transparency:** Provide comprehensive reports for clear visibility into performance and trends.
- **Ensure Data Accuracy and Compliance:** Establish a robust system for tracking all transactions.
- **Increase Operational Efficiency:** Reduce manual data entry and administrative overhead through automation.

Student Outcomes:

- **Hands-on Experience with Inventory Automation:** Students gain practical skills in configuring Salesforce to manage suppliers, products, and purchase orders.
- **Understanding of the Project Lifecycle in a Business Context:** Students learn the complete end-to-end process, from requirements to deployment of a Salesforce application.
- **Enhanced Analytical and Problem-Solving Skills:** Students develop the ability to identify operational challenges and design automated solutions within the Salesforce platform.
- **Improved Collaboration Skills:** Students gain experience working as a team to coordinate key tasks like requirement analysis and system development.
- **Industry-Relevant Exposure:** Students get exposure to real-world uses of Salesforce in healthcare and supply chain sectors.

System Requirements :

Hardware Requirements:

• Computer with min/sum 4 GB RAM, Dual-core processor

• Stable internet connection

Software Requirements:

• Salesforce Developer Edition Org

• Modern Web Browser (e.g., Google Chrome, Firefox)

Phase 1: Requirement Analysis & Planning :

Utilizing Salesforce, our project streamlines medical inventory management, from supplier details and product tracking to expiry date monitoring, ensuring operational efficiency and data accuracy.

Phase 2: Salesforce Development – Backend & Configurations :

Milestone 1- Salesforce Account

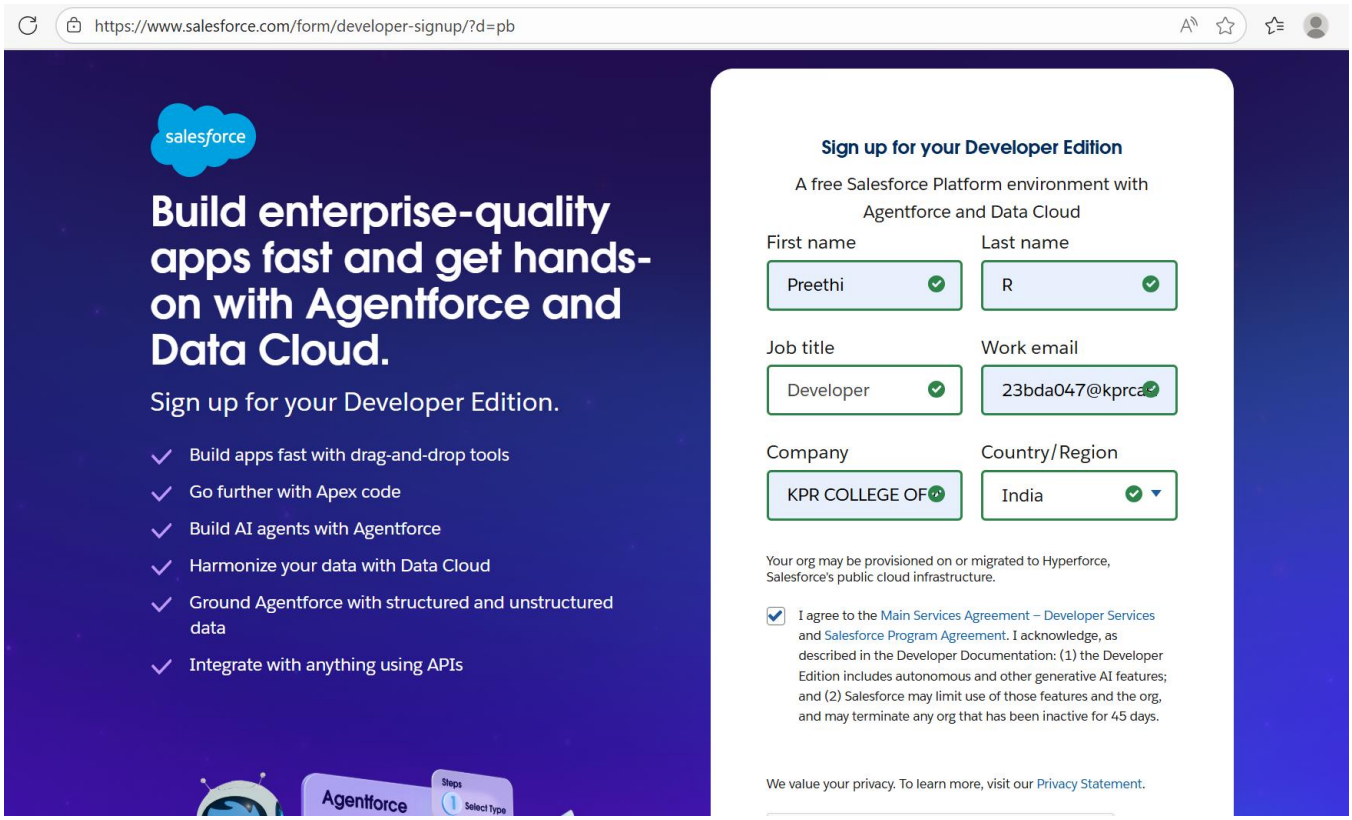
Activity 1: Creating Developer Account

Creating a developer org in salesforce.

1. Go to https://developer.salesforce.com/signup
2. On the sign up form, enter the following details :
    1. First name & Last name
    2. Email
    3. Role : Developer
    4. Company : College Name
    5. County : India
    6. Postal Code : pin code
    7. Username : should be a combination of your name and company

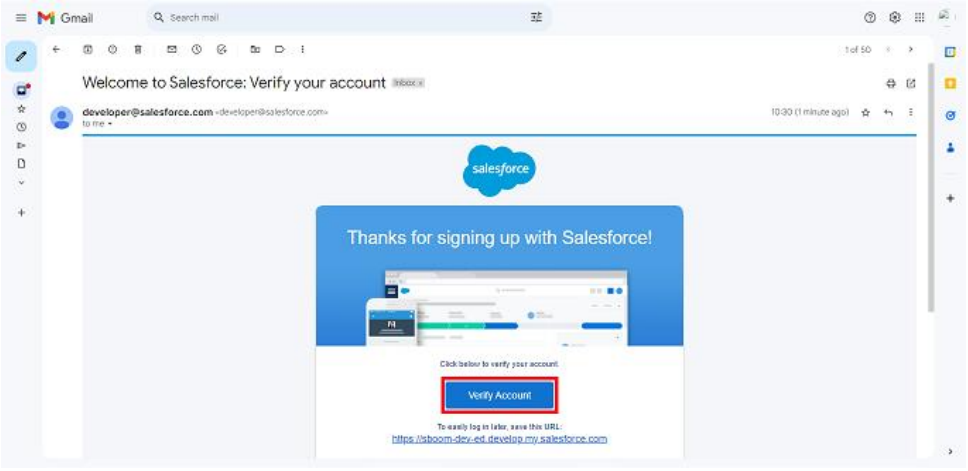
This need not be an actual email id, you can give anything in the format : username@organization.com

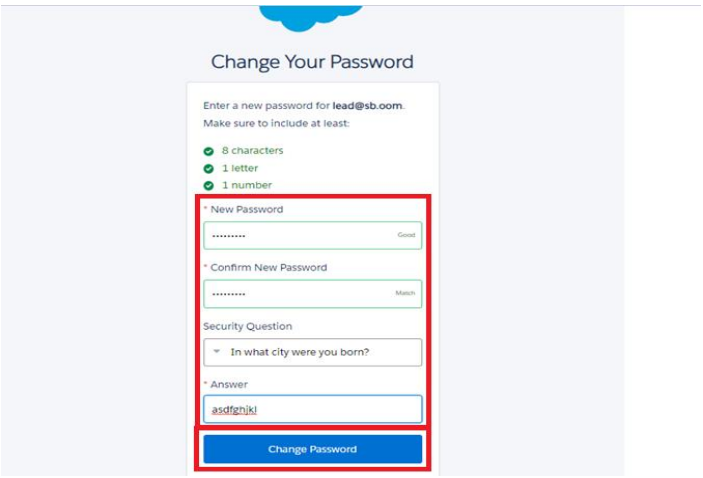
Click on sign me up after filling these.

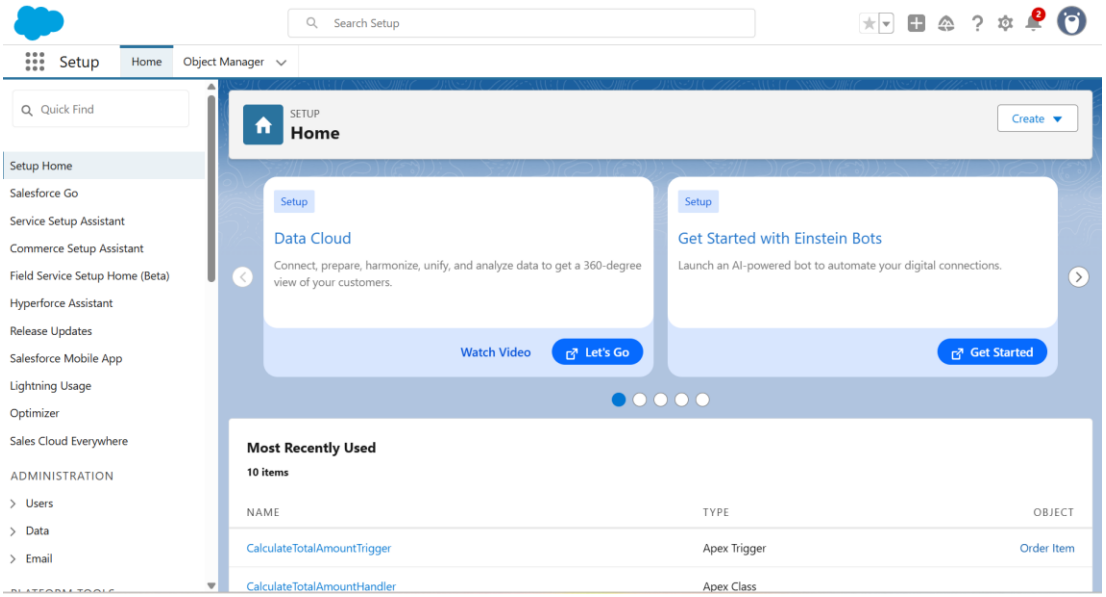


Activity 2: Account Activation

1. 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.
2. Click on Verify Account
3. Give a password and answer a security question and click on change password.
4. Then you will redirect to your salesforce setup page.







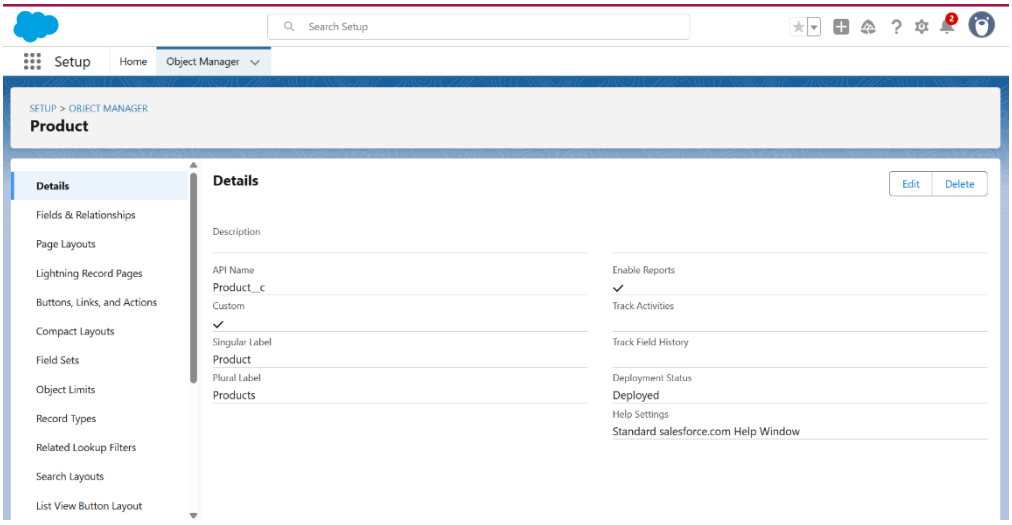
Milestone 2- Objects

Activity 1:

Creating a Product Object

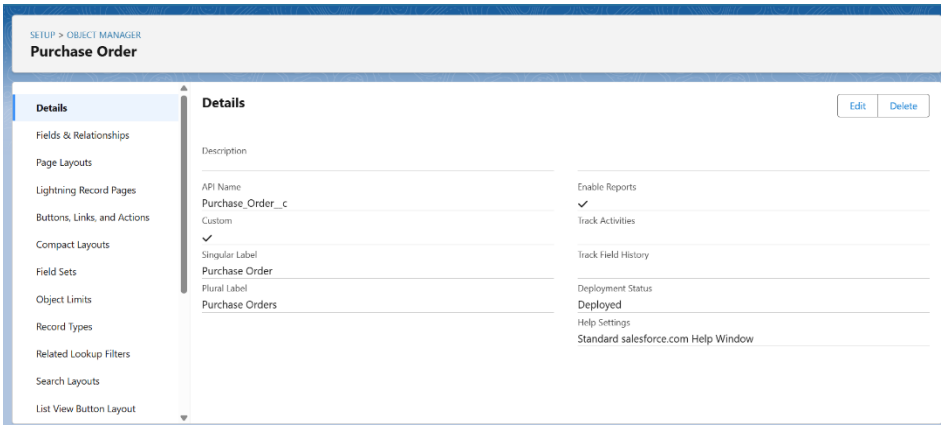
1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.
4. Enter the label name as Product
5. Enter Plural label name as Products

6. Enter Record Name as Product ID
7. Select Data Type as Text.
8. Select Allow reports.
9. Select Allow search.
10. Click on Save and New



Creating a Purchase Order Object

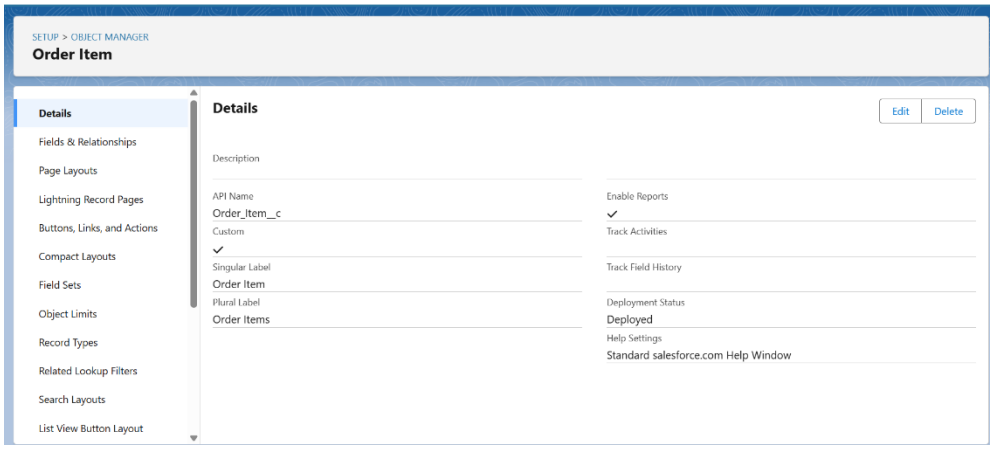
1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.
4. Enter the label name as Purchase Order
5. Enter Plural label name as Purchase Orders
6. Enter Record Name as Purchase Order ID
7. Select Data Type as Text.
8. Select Allow reports.
9. Select Allow search.
10. Click on Save and New



Creating a Order Item Object

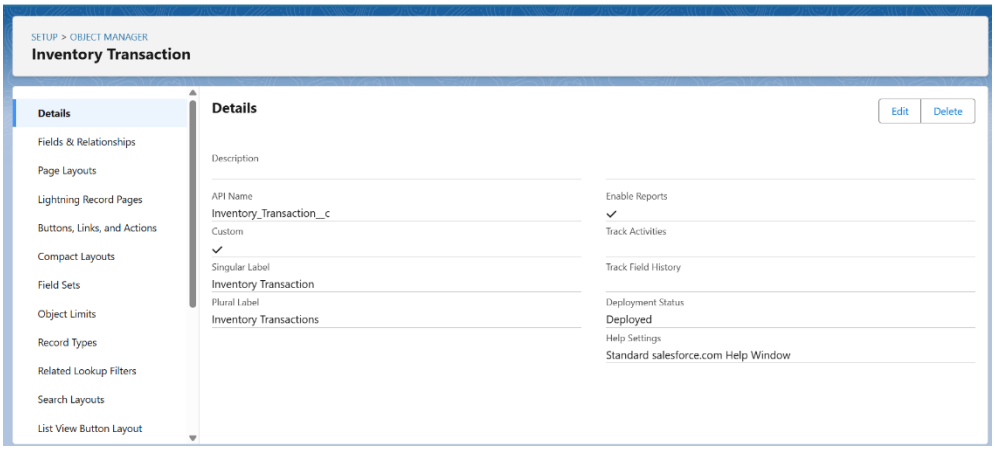
1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.
4. Enter the label name as Order item
5. Enter Plural label name as Order items
6. Enter Record Name as Order Item ID
7. Select Data Type as Text.
8. Select Allow reports.
9. Select Allow search.
10. Click on Save and New

-



Creating a Inventory Transaction Object

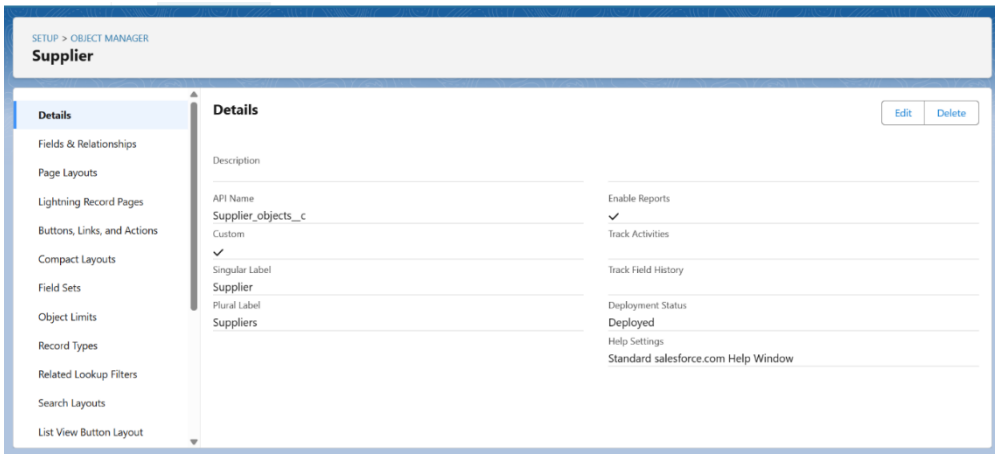
1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.
4. Enter the label name as Inventory Transaction
5. Enter Plural label name as Inventory Transactions
6. Enter Record Name as Inventory Transaction ID
7. Select Data Type as Text.
8. Select Allow reports.
9. Select Allow search.
10. Click on Save and New



Creating a Supplier Object

1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.
4. Enter the label name as Supplier
5. Enter Plural label name as Suppliers
6. Enter Record Name as Supplier ID
7. Select Data Type as Text.

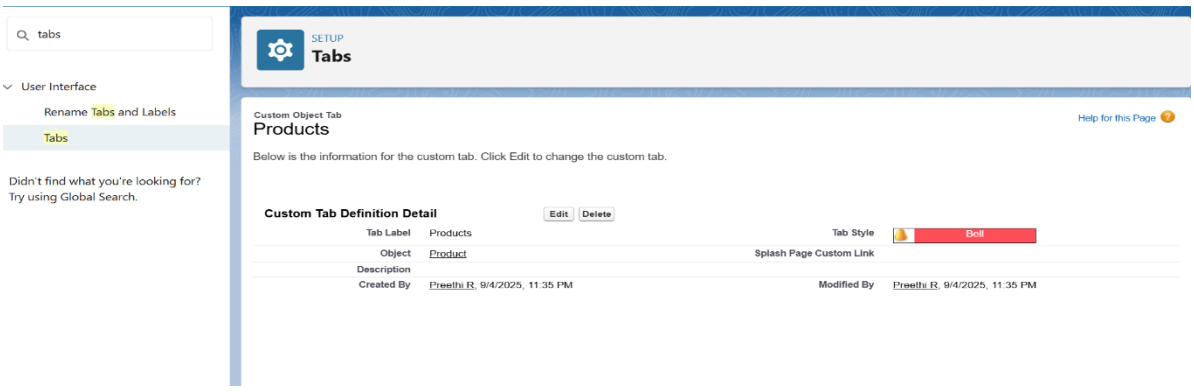
8. Select Allow reports.
9. Select Allow search.
10. Click on Save and New



Milestone 3- Tabs

Activity 1: Creating a tab for Product Object

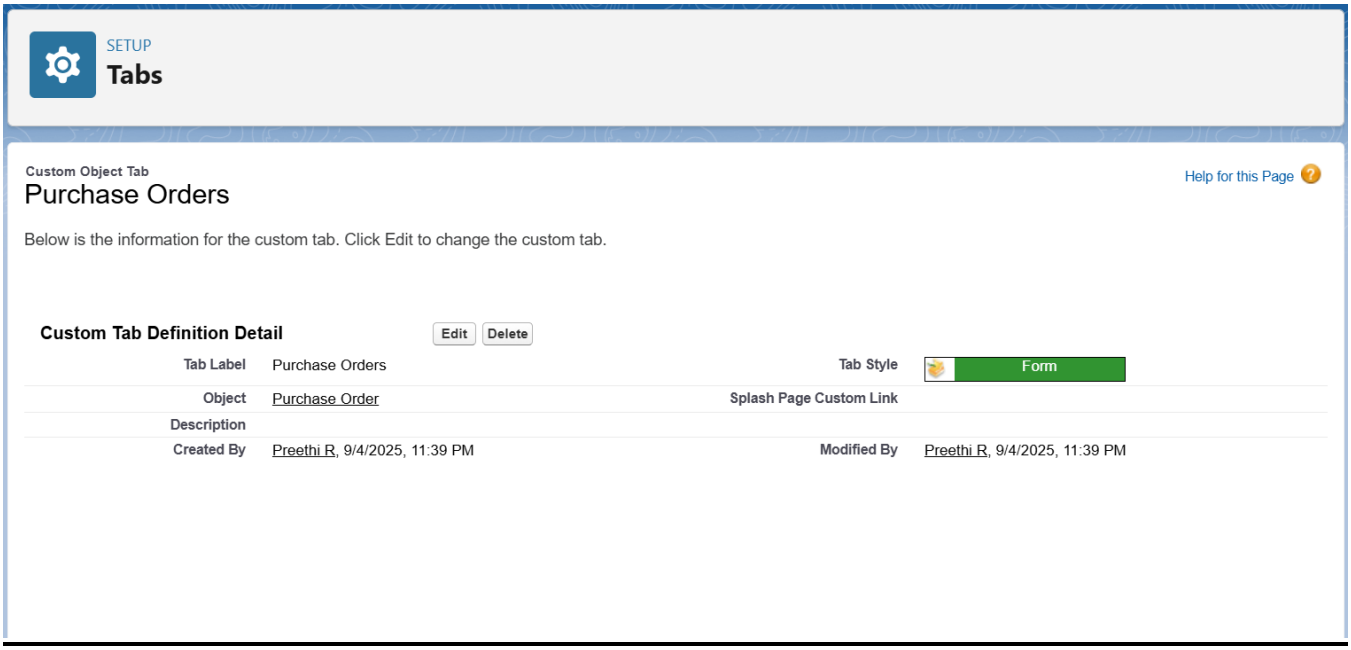
1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).
4. Select Object(Product) >> Select the tab style
5. Click on Next >> (Add to profiles page) keep it as default >> Click on Next (Add to Custom App) uncheck the include tab .
6. Make sure that the Append tab to user's existing personal customizations is checked.
7. Click save



Activity 2: Creating Remaining Tabs

Creating Purchase Orders Tab

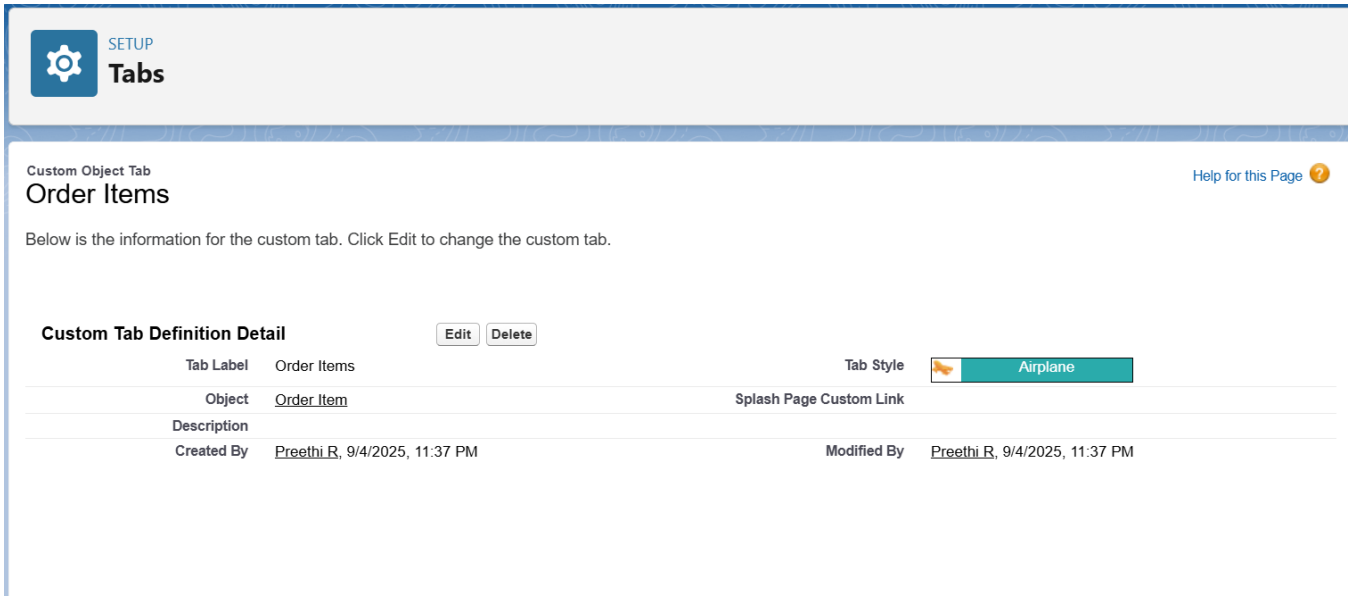
1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).
4. Select Object(Purchase Order) >> Select the tab style
5.Click on Next >> (Add to profiles page) keep it as default >> Click on
6.Next (Add to Custom App) uncheck the include tab .
7.Make sure that the Append tab to user's existing personal customizations is checked.
8.Click save



Creating Order Items Tab

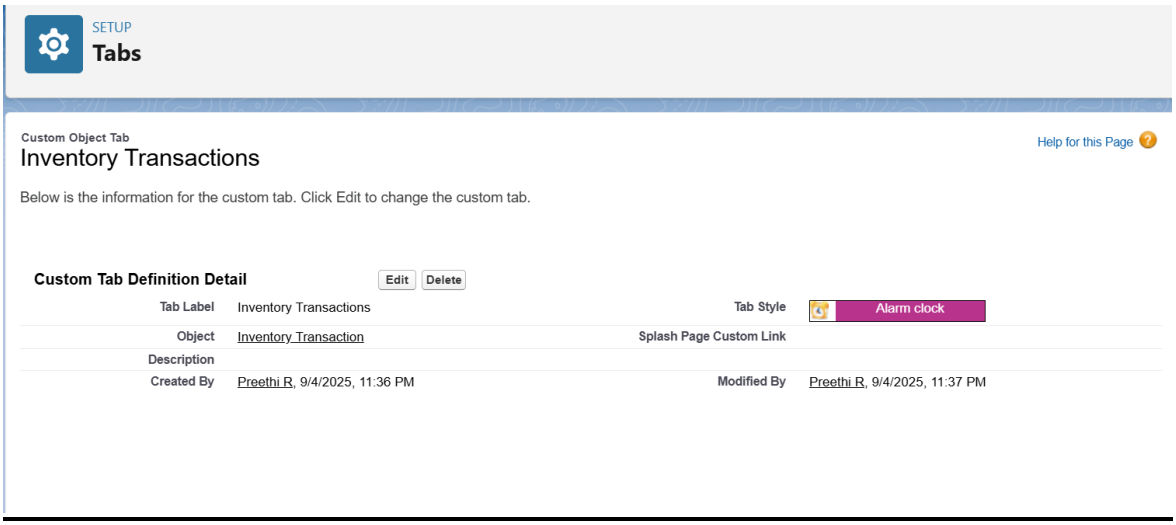
1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).
4. Select Object(Order Item) >> Select the tab style

5.Click on Next >> (Add to profiles page) keep it as default >> Click on
6.Next (Add to Custom App) uncheck the include tab .
7.Make sure that the Append tab to user's existing personal customizations is checked.
8.Click save.



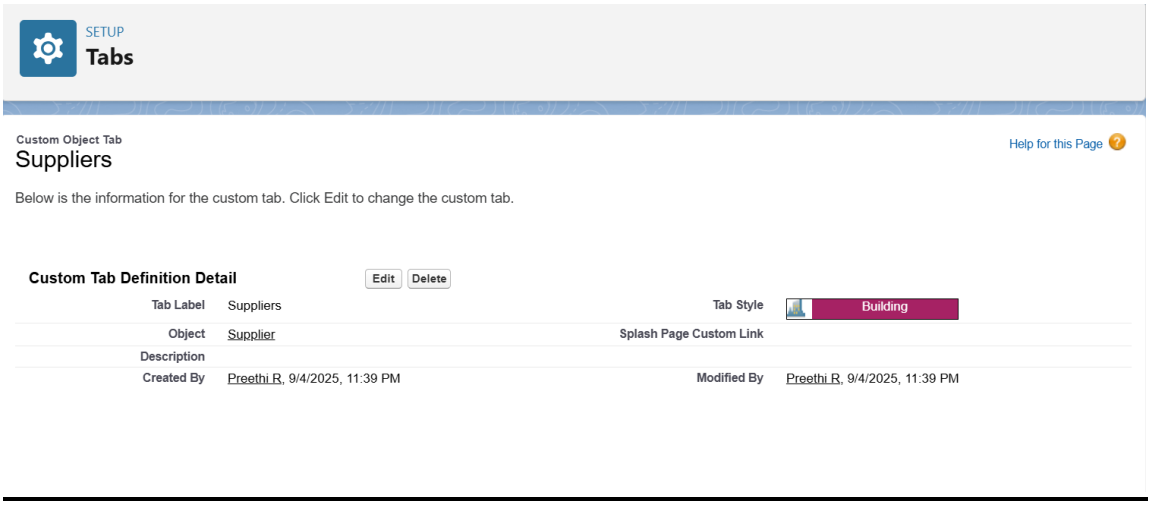
Creating Inventory Transactions Tab

1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).
4. Select Object(Inventory Transaction) >> Select the tab style
5.Click on Next >> (Add to profiles page) keep it as default >> Click on
6.Next (Add to Custom App) uncheck the include tab .
7.Make sure that the Append tab to user's existing personal customizations is checked.
8.Click save.



Creating Suppliers Tab

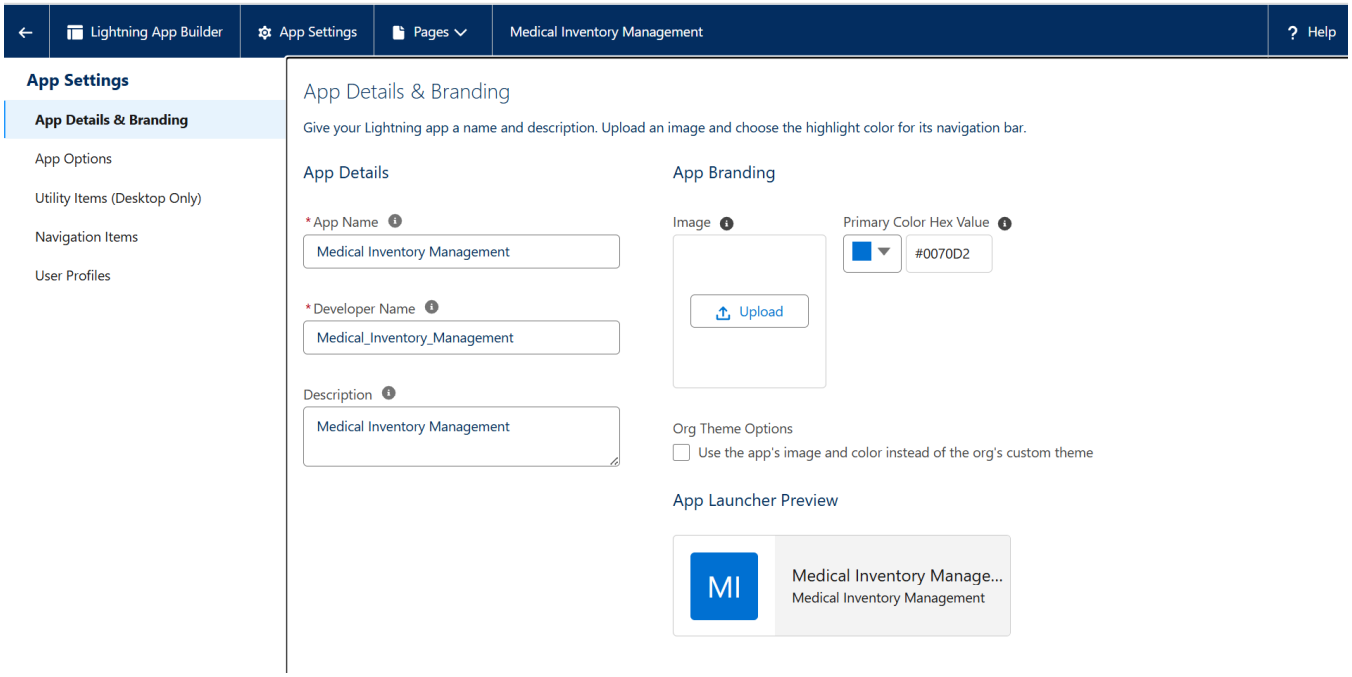
1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).
4. Select Object(Supplier) >> Select the tab style
5. Click on Next >> (Add to profiles page) keep it as default >> Click on
6. Next (Add to Custom App) uncheck the include tab .
7. Make sure that the Append tab to user's existing personal customizations is checked.
8. Click save.

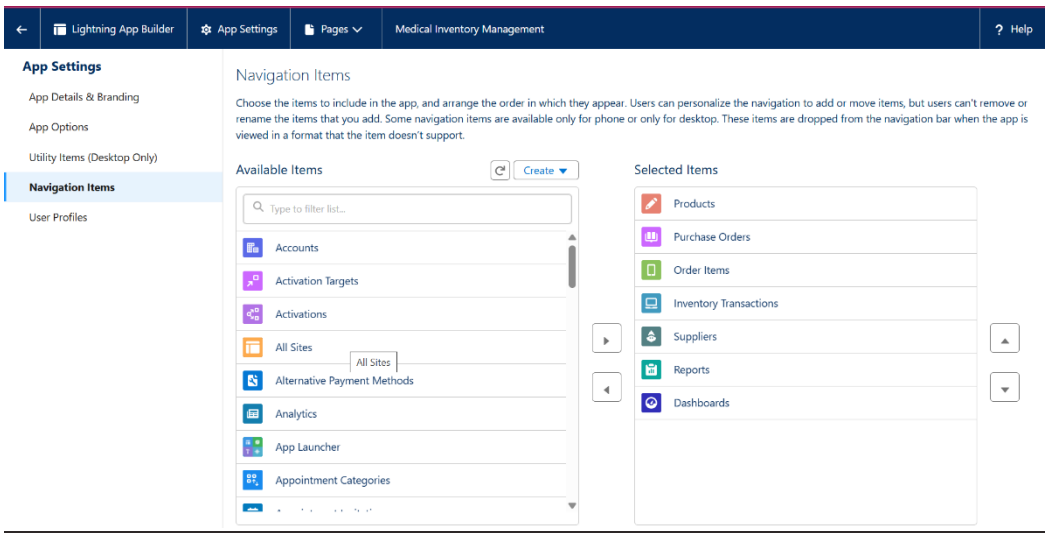


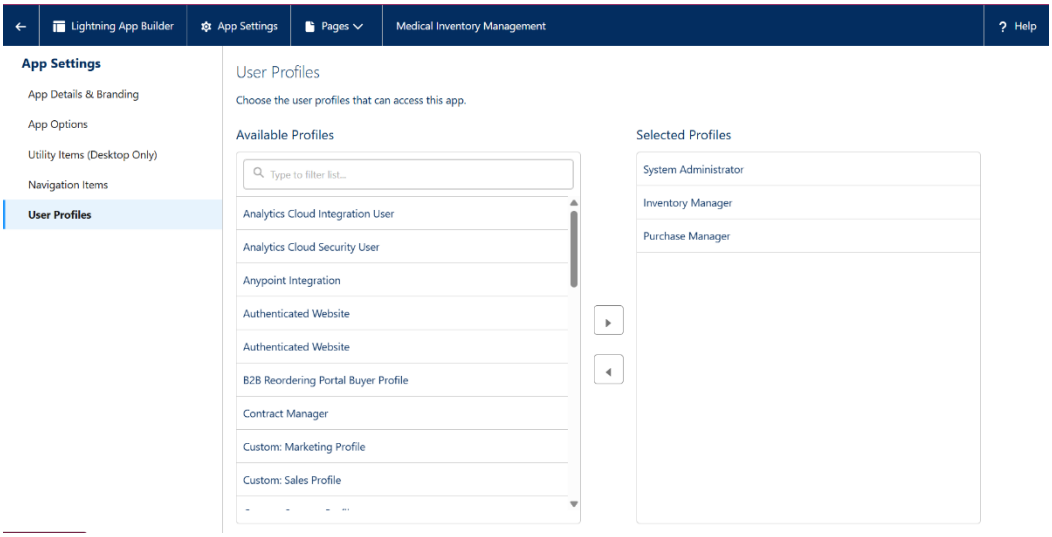
Milestone 4- The Lightning App

Activity 1: Create a Lightning App for Medical Inventory Management

1. From Setup, enter App Manager in the Quick Find and select App Manager.
2. Click New Lightning App.
3. Enter Medical Inventory Management as the App Name >> Click on upload image and add an image related to Medical Inventory 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 Products, Purchase Orders, Order Items, Inventory Transactions, Suppliers, Reports, and Dashboards and move them to Selected Item and Click Next.
7. From Available Profiles, select System Administrator and move it to Selected Profiles.
8. Click Save & Finish.







Milestone 5- Fields

Activity 1: Creating a Text Field in Product Object

To create fields in an 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 Product custom object.
4. Select Fields & Relationships from the left navigation
5. Click on New
6. Select Text field, click Next
7. Enter Field Label as “Product Name” and Length 255.
8. Select Required Field.
9. Click Next, Next, then Save & New.

Activity 2: Creating a TextArea Field in Product Object

To create fields in an 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 Product custom object.
4. Select Fields & Relationships from the left navigation
5. Click on New
6. Select TextArea field, click Next
7. Enter Field Label as “Product Description” .
8. Click Next, Next, then Save & New.

Activity 3: Creating a Number Field in Product object

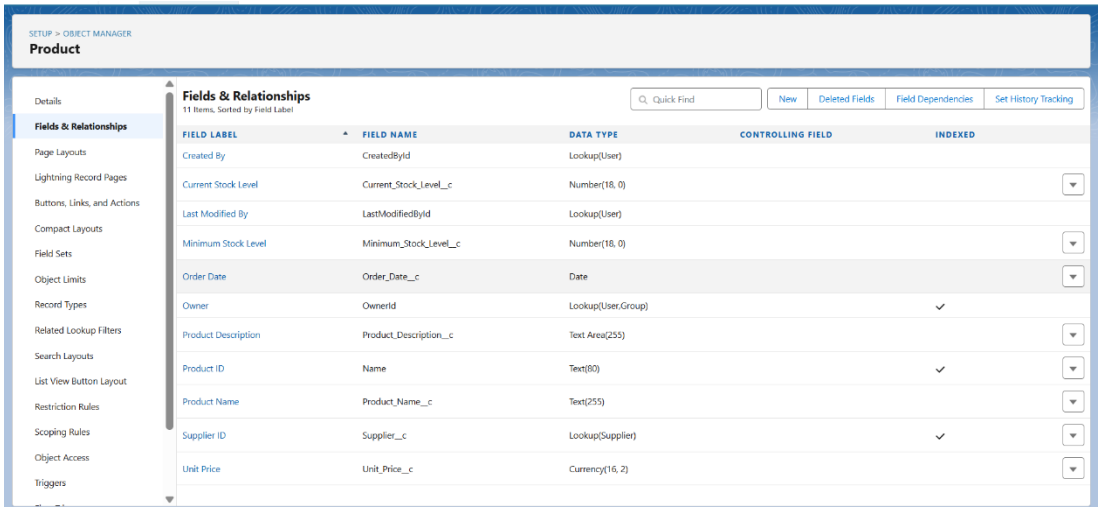
To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Product) in quick find box >> click on the Product custom object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Number” and click Next.
5. Enter Field Label as “ Current Stock Level”.
6. Length - 18, Decimal Places - 0.
7. Click on Next, Next and Save.

Activity 4: Creating a Currency Field in Product object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Product) in quick find box >> click on the Product custom object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Currency” and click Next.
5. Enter Field Label as “ Unit Price”.
6. Length - 16, Decimal Places - 2.
7. Select Required Field.
8. Click on Next, Next and Save.



Activity 5 : Creating Lookup Relationship in Purchase Order Object

To Create a relationship from Purchase Order to Supplier .

1. Go to the Setup page >> click on Object manager >> type object name(Purchase Order) in the quick find bar >> click on the Purchase Order object.
2. Click on Fields & Relationship
3. Click on New.

4. Select “Lookup relationship” as data type and click Next.
5. Select the related object “ Supplier”.
6. Click on Next.
7. Give Field Label as “Supplier ID” .
8. Select Required Field.
9. Click on Next , Next, Next , Save.

Activity 6: Creating a Date Field in Purchase Order object

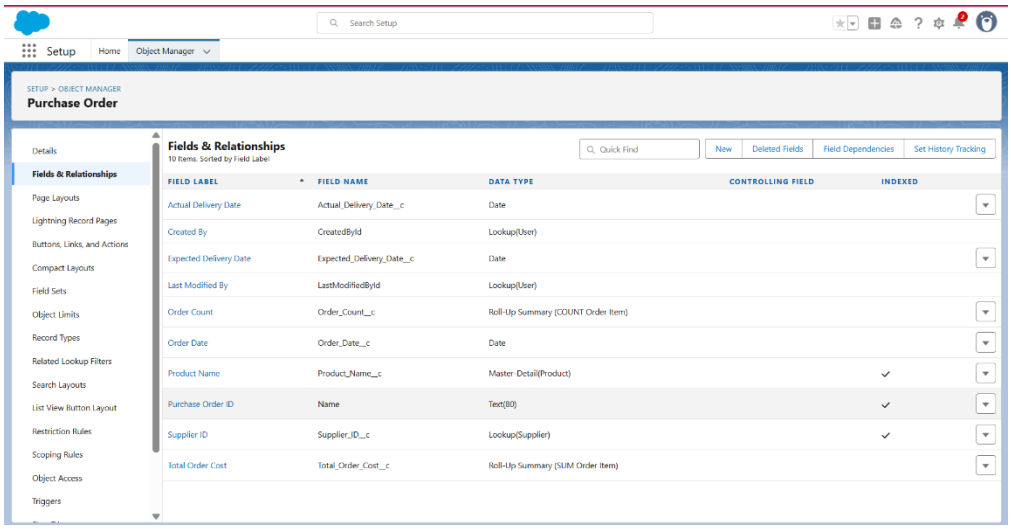
To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Purchase Order) in quick find box>> click on the Purchase Order object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Date” and click Next.
5. Enter Field Label as “ Order Date”.
6. Click on Next, Next and Save.

Activity 7: Creating a Roll-Up Summary Field in Purchase Order object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Purchase Order) in quick find box>> click on the Purchase Order object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Roll-Up Summary” and click Next.
5. Enter Field Label as “ Order Count”.
6. Choose the Summarized Object as “Order Items”.
7. For Select Roll-Up Type select “Count”.
8. Click on Next, Next and Save.



Activity 8: Creating a Unit Price Formula Field in Order Item object

To create fields in an object:

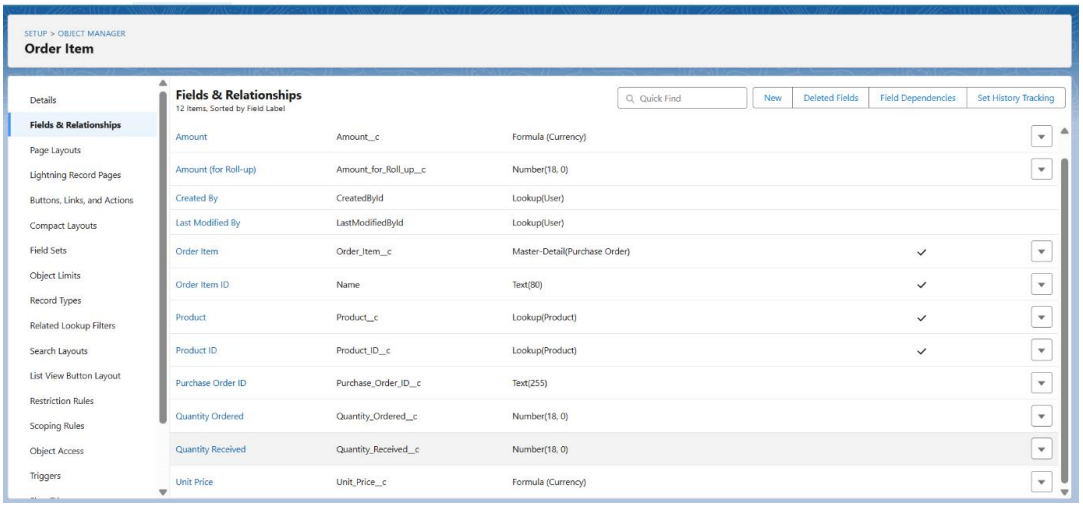
1. Go to setup >> click on Object Manager >> type object name(Order Item) in quick find box >> click on the Order Item object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Formula” and click Next.
5. Enter field label Unit Price.
6. Select formula return type Currency, Click Next
7. Create and insert Advance formula: Product_ID__r.Unit_Price__c
8. Click Next, Next, then Save.

Activity 9: Creating a Amount Formula Field in Order Item object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Order Item) in quick find box >> click on the Order Item object.
2. Now click on “Fields & Relationships”
3. Click on New.

4. Select Data type as “Formula” and click Next.
5. Enter field label Amount.
6. Select formula return type Currency, Click Next
7. Create and insert Advance
   formula: Quantity_Received__c * Unit_Price__c
8. Click Next, Next, then Save.



Activity 10: Creating a Picklist Field in Inventory Transaction Object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Inventory Transaction) in quick find box>> click on the Inventory Transaction Object.
2. Now click on “Fields & Relationships” .
3. Click on New.
4. Select Data type as “Picklist” and click Next.
5. Enter Field Label as “Transaction Type”.
6. In values select “Enter values, with each value separated by a new line" and enter values as shown below.

   Receipt

   Issue

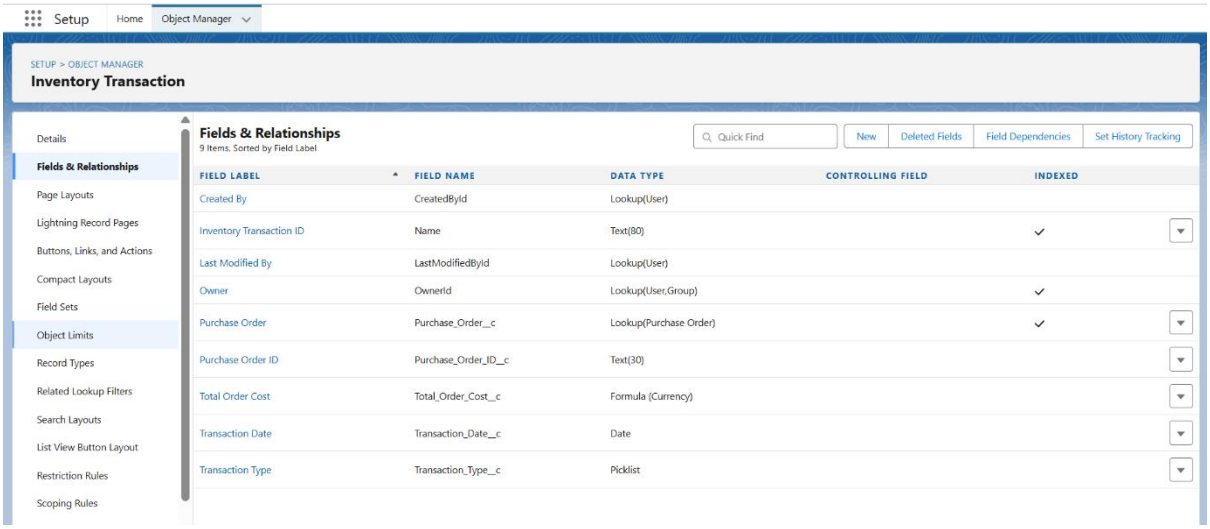
Adjustment

7. Click on Next, Next and Save.

Activity 11: Creating a Total Order Cost Formula Field in Inventory Transaction object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Inventory Transaction) in quick find box >> click on the Order Item object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Formula” and click Next.
5. Enter field label Total Order Cost.
6. Select formula return type Currency, Click Next
7. Create and insert Advance formula: Purchase_Order_ID__r.Total_Order_Cost__c
8. Click Next, Next, then Save.



Activity 12: Creating a Phone Field in Supplier object

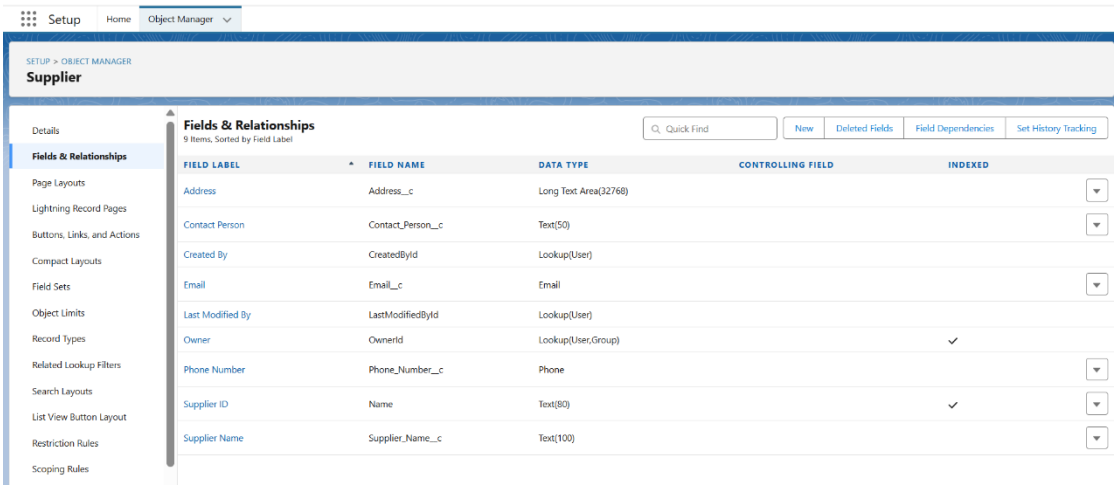
To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Supplier) in quick find box>> click on the Supplier object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Phone” and click Next.
5. Enter the Field Label as “ Phone Number”.
6. Select Required Field.
7. Click on Next, Next and Save.

Activity 13: Creating a Email Field in Supplier object

To create fields in an object:

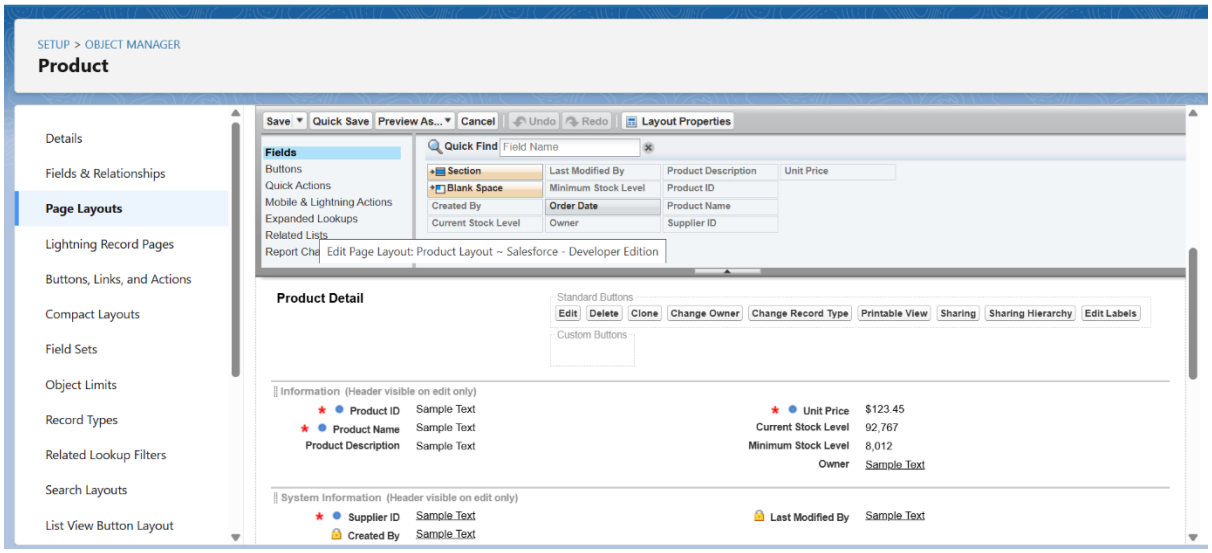
1. Go to setup >> click on Object Manager >> type object name(Supplier) in quick find box>> click on the Supplier object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Email” and click Next.
5. Enter the Field Label as “ Email”.
6. Click on Next, Next and Save.



Milestone 6 -Editing of Page Layouts

Activity 1: To edit a Page Layout in Product Object

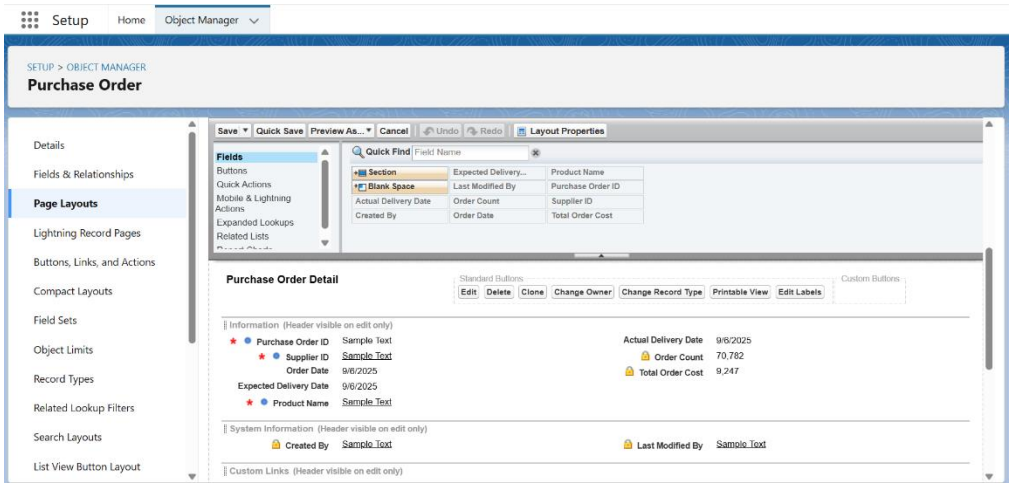
1. Go to setup >> click on Object Manager >> type object name(Product) in quick find box >> click on the Product object >> Page Layouts .
2. Click on the Product Layout.
3. Drag and Arrange the field as shown below.
4. Click on Save.



Activity 2: To edit a Page Layout in Purchase Order Object

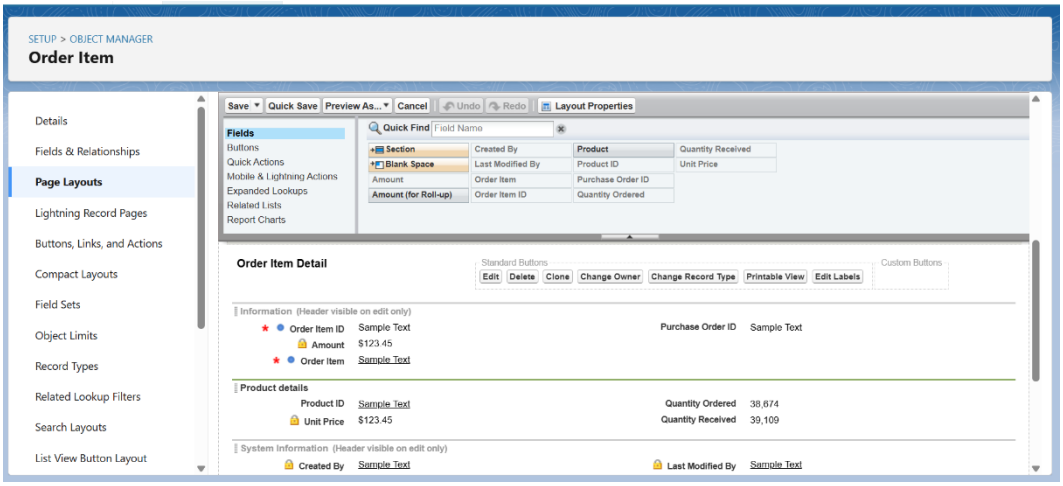
1. Go to setup >> click on Object Manager >> type object name(Purchase Order) in quick find box >> click on the Purchase Order object >> Page Layouts.
2. Click on the Purchase Order Layout
3. Drag and Arrange the field as shown below
4. Click on field Order Date >> click on settings >> select Required and save it.
5. Click on field Total Order Cost >> click on settings >> select Read Only and save it.

6. Click Save.



Activity 3: To edit a Page Layout in Order Item Object

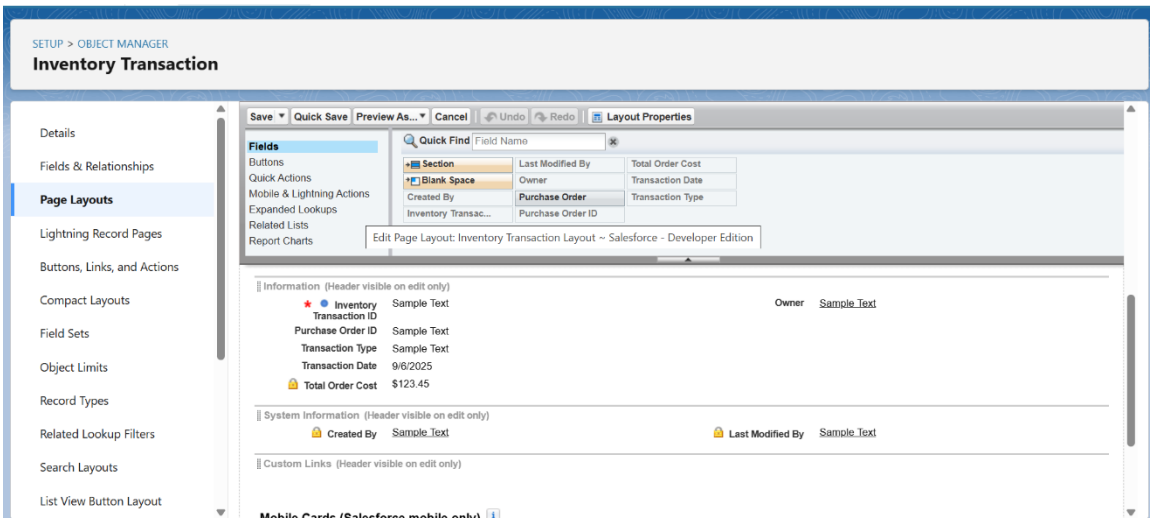
1. Go to setup >> click on Object Manager >> type object name(Order Item) in quick find box >> click on the Order Item object >> Page Layouts.
2. Click on the Order Item Layout
3. Drag and Arrange the field as shown below.
4. Click Save.



Activity 4: To edit a Page Layout in Inventory Transaction Object

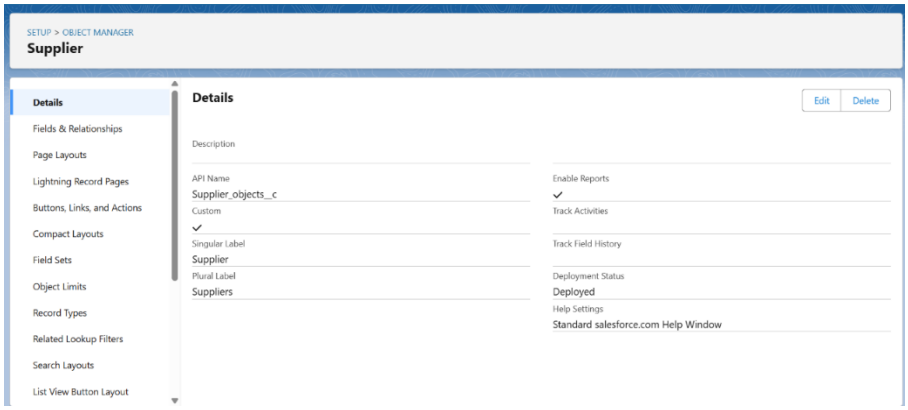
1. Go to setup >> click on Object Manager >> type object name(Inventory Transaction) in quick find box >> click on the Inventory Transaction object >> Page Layouts.
2. Click on the Inventory Transaction Layout

3. Drag and Arrange the field as shown below
4. Click Save.



Activity 5: To edit a Page Layout in Supplier Object

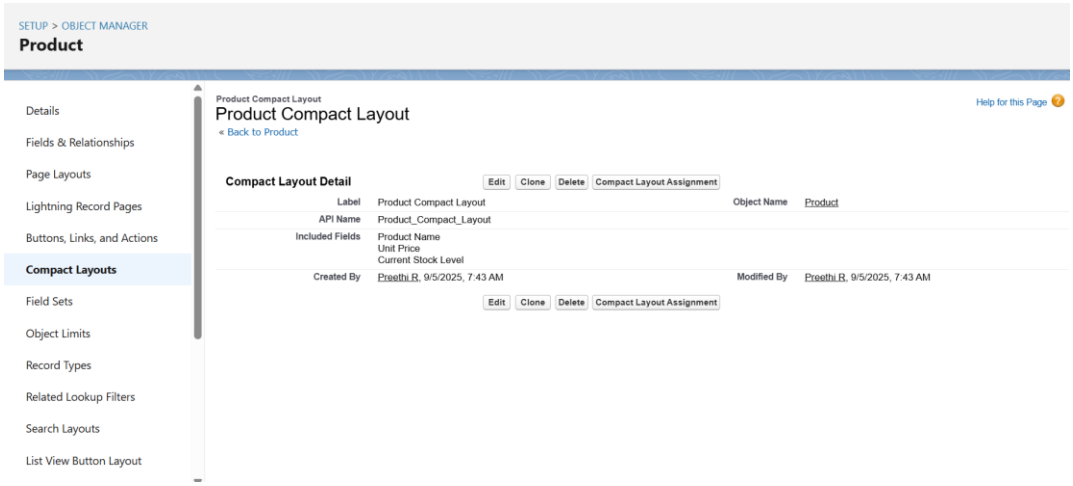
1. Go to setup >> click on Object Manager >> type object name(Supplier) in quick find box >> click on the Supplier object >> Page Layouts.
2. Click on the Supplier Layout
3. Drag and Arrange the field as shown below
4. Click Save.



Milestone 7 - Compact Layouts

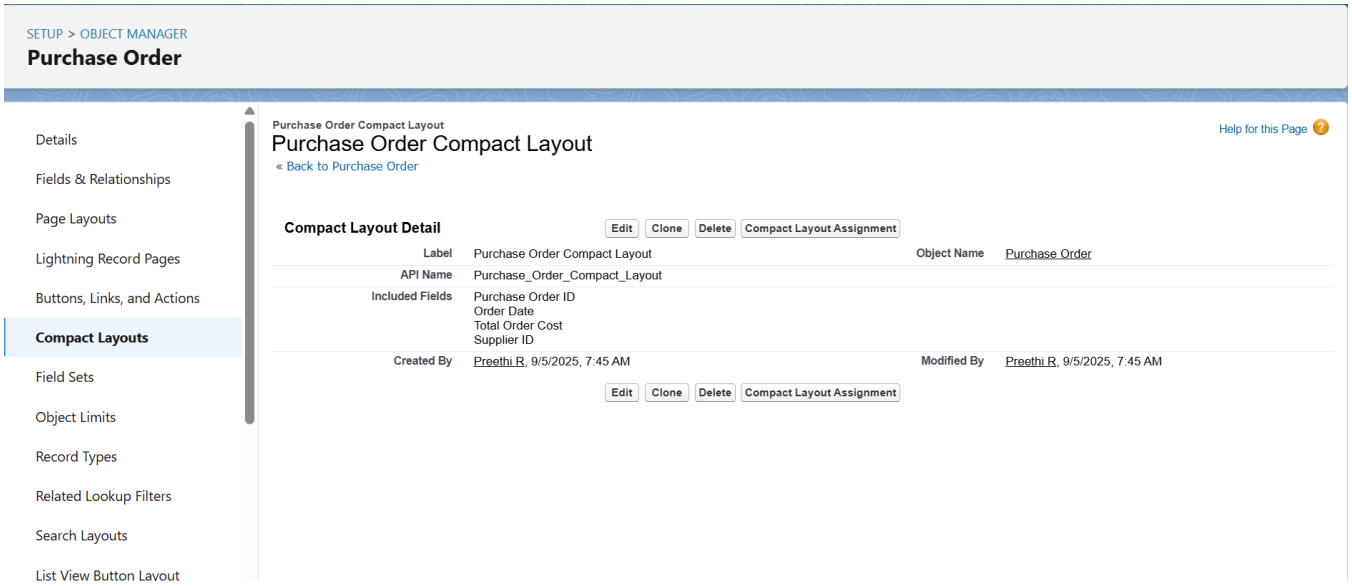
Activity 1: To create a Compact Layout to a Product Object

1. Go to setup >> click on Object Manager >> type object name(Product) in quick find box >> click on the Product object
2. Click on Compact Layouts in the sidebar .
3. Click on New.
4. Enter the Label as “Product Compact Layout”.
5. Select the Compact Layout Fields : Select Product name, Unit Price, Current Stock Level.
6. Click Save.
7. Click Compact Layout Assignment.
8. Click Edit Assignment.
9. Choose "Product Compact Layout" from the dropdown.
10. Click Save.



Activity 2: To create a Compact Layout to a Purchase Order Object

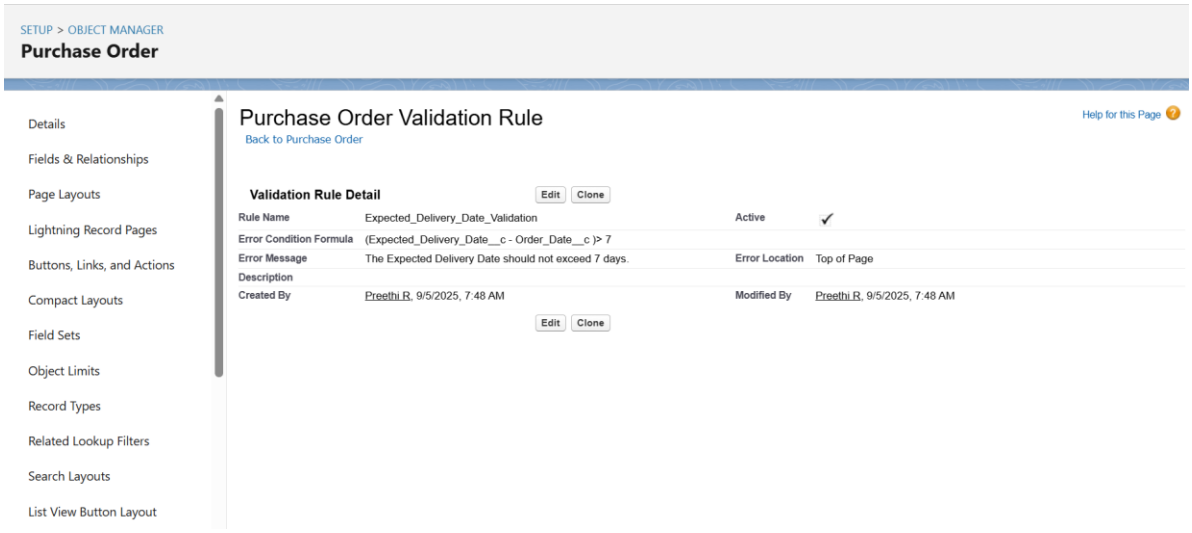
1. Go to setup >> click on Object Manager >> type object name(Purchase Order) in quick find box >> click on the Purchase Order object
2. Click on Compact Layouts in the sidebar .
3. Click on New.
4. Enter the Label as “Purchase Order Compact Layout”.
5. Select the Compact Layout Fields : Select Purchase Order ID, Order Date, Total Order Cost, Supplier ID.
6. Click Save.
7. Click Compact Layout Assignment.
8. Click Edit Assignment.
9. Choose "Purchase Order Compact Layout" from the dropdown.
10. Click Save.



Milestone 8 - Validation Rules

Activity 1: To create an Expected Delivery Date Validation rule to a Employee Object

1. Go to setup >> click on Object Manager >> type object name(Purchase Order) in quick find box>> click on the Purchase Order object
2. Click on the validation rule >> click on New.
3. Enter the Rule name as “Expected Delivery Date Validation”.
4. Select Active
5. Insert the Error Condition Formula as : (Expected_Delivery_Date__c - Order_Date__c)> 7
6. Enter the Error Message as “The Expected Delivery Date should not exceed 7 days.”.
7. Select the Error location as Top of Page
8. Click Save.

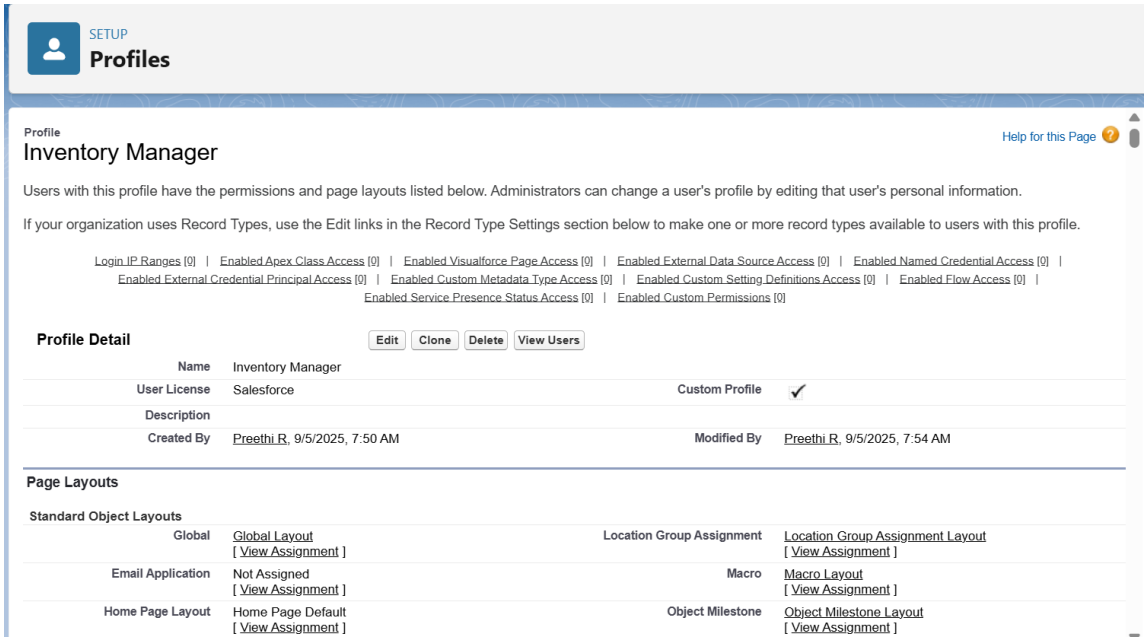


Milestone 9 - Profiles

Activity 1: To create an Inventory Manager Profile

1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Standard User) >> enter profile name (Inventory Manager) >> Save.
2. While still on the profile page, then click Edit.

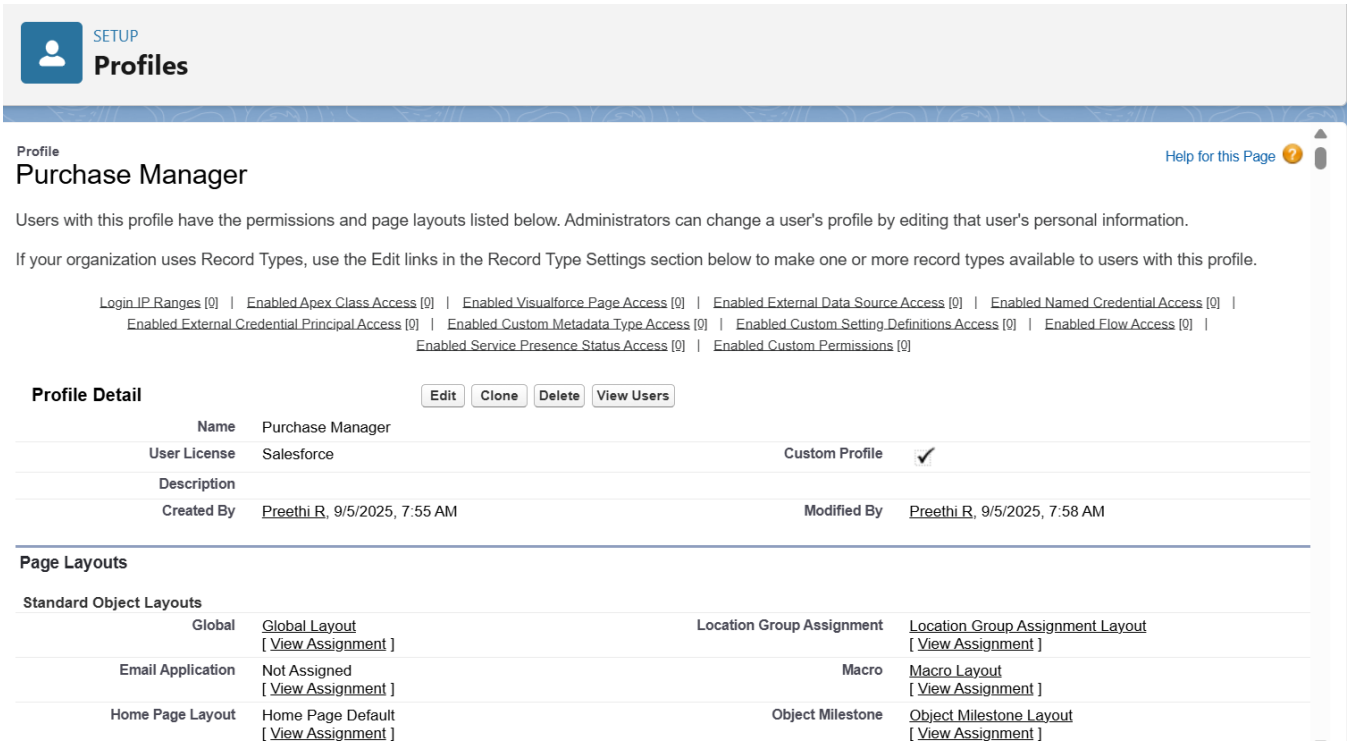
3. Select the Custom App settings as default for the Medical Inventory Management.
4. Scroll down to Custom Object Permissions and Give access permissions as mentioned in the below diagram.
5. Change the password policies as mentioned :
6. User passwords expire in should be “ never expires ”.
7. Minimum password length should be “ 8 ”, and click save.



Activity 2: To create an Purchase Manager Profile

1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Standard User) >> enter profile name (Purchase Manager) >> Save.
2. While still on the profile page, then click Edit.
3. Select the Custom App settings as default for the Medical Inventory Management.
4. Scroll down to Custom Object Permissions and Give access permissions as mentioned in the below diagram.

5. Change the password policies as mentioned :
6. User passwords expire in should be “ never expires ”.
7. Minimum password length should be “ 8 ”, and click save.



Milestone 10 - Roles

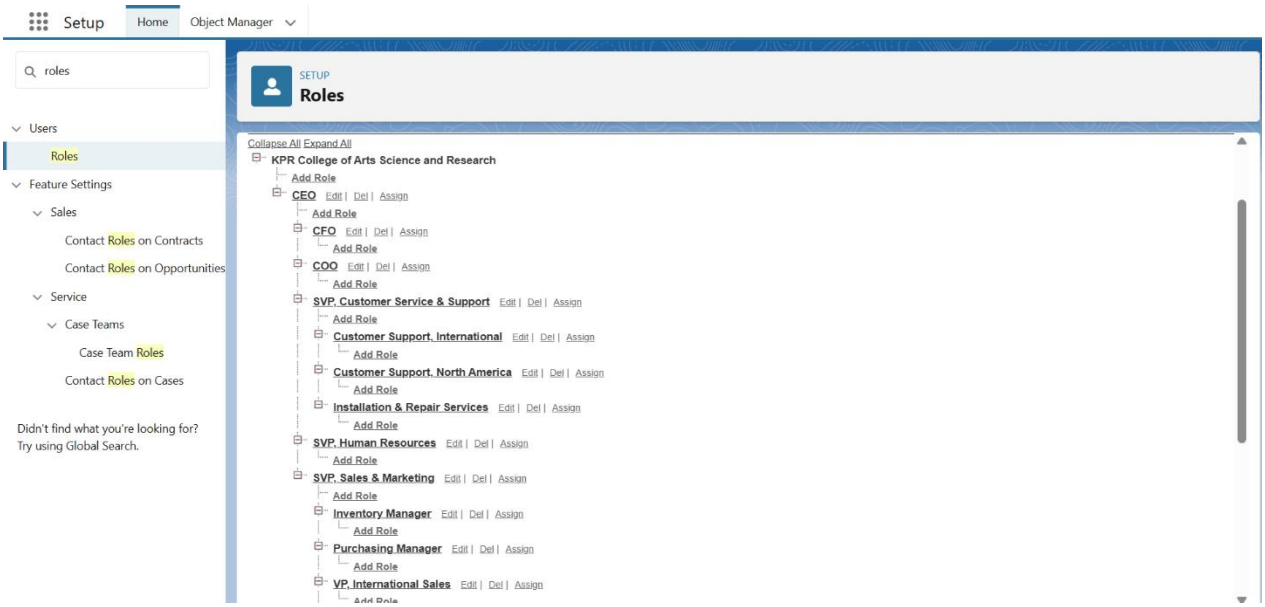
Activity 1 : Create a Purchasing Manager Role.

1. Go to quick find >> Search for Roles >> click on Set Up Roles.
2. Click on Expand All and click on add role under SVP, Sales & Marketing role.
3. Give Label as “Purchasing Manager” and Role name gets auto populated. Then click on Save.

Activity 2 : Create a Purchasing Manager Role.

1. Go to quick find >> Search for Roles >> click on Set Up Roles.

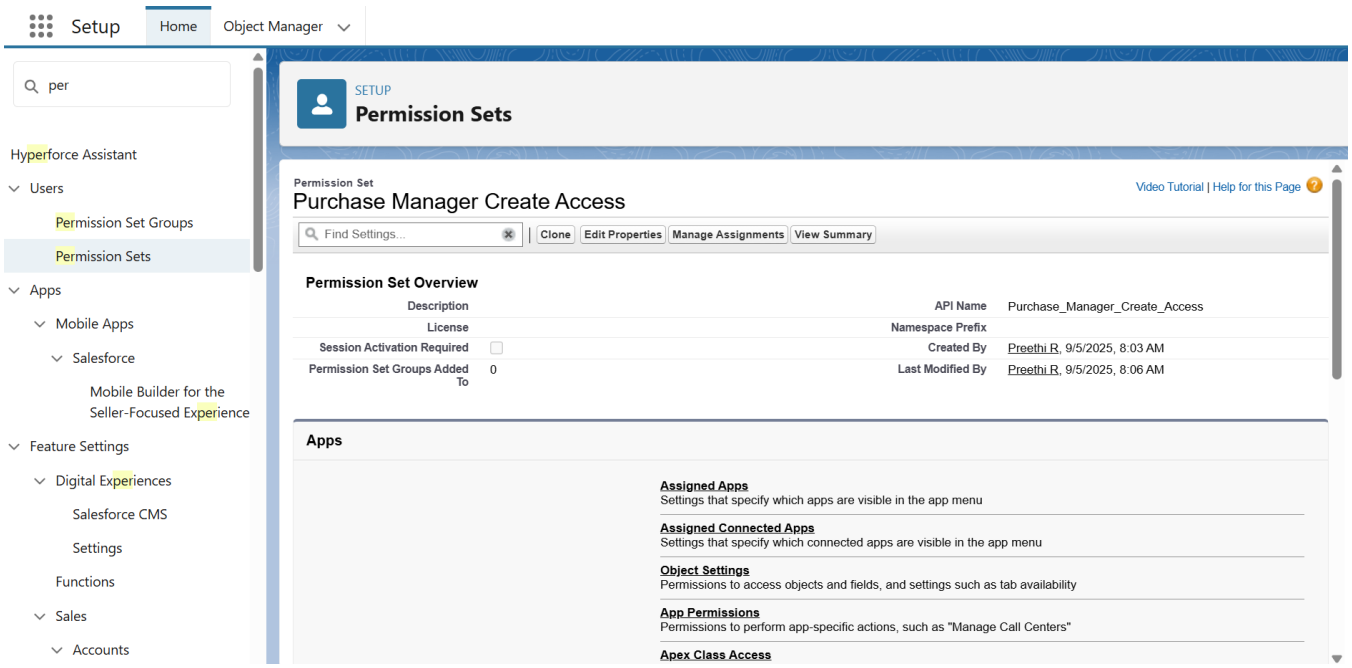
2. Click on Expand All and click on add role under SVP, Sales & Marketing role.
3. Give Label as “Inventory Manager” and the Role name gets auto populated. Then click on Save.



Milestone 12 - Permission Sets

Activity 1 : Create a Permission Set.

1. Go to setup >> type Permission in quick find box >> Select Permission Set >> click on New.
2. Enter Label as Purchase Manager Create Access >> Click on Save.
3. From Object Settings >> Select Order Item >> Enable for both Tab Available and Visible >> Enable Read and Create in Object Permissions >> Click on Save.
4. Navigate to the Permission Set detail page >> Click Manage Assignments >> Click Add Assignments >> Select the user John PurchaseM to assign the permission set to and click Next.
5. Select No Expiration date >> Click on Assign.



Milestone 13 - Flows

Activity 1 : Create Flow to update the Actual Delivery Date.

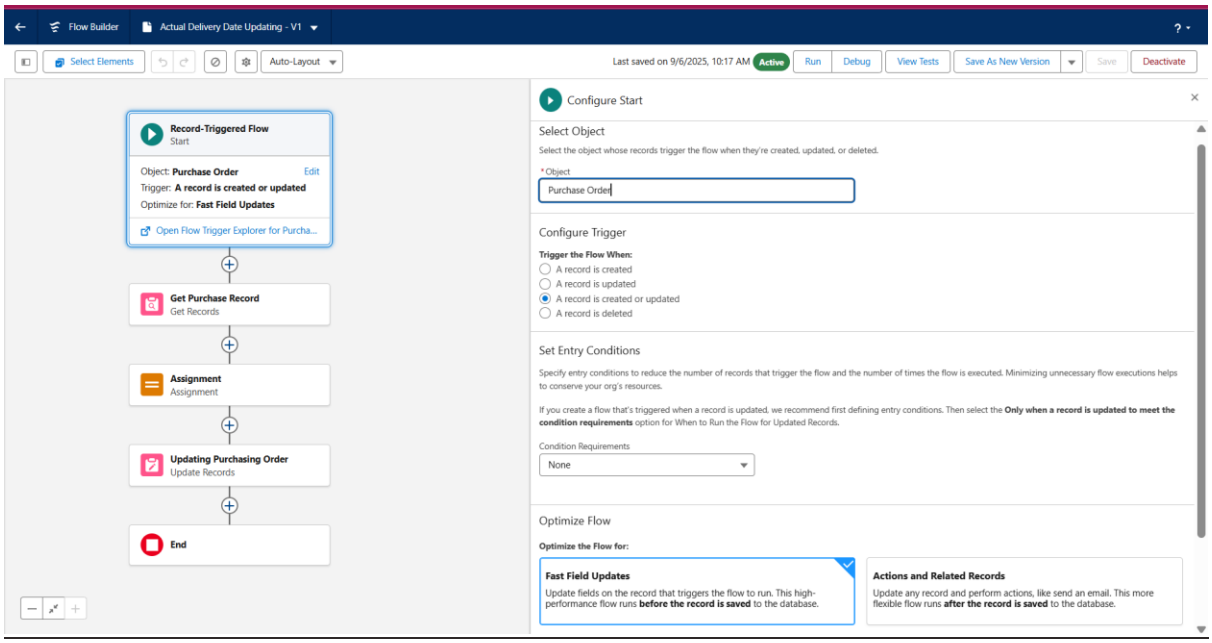
1. Go to setup >> type Flow in quick find box >> Click on the Flow and Select the New Flow >> Start From Scratch .
2. Select the record Triggered flow.Click on create.
3. Under Object select “Purchase Order”
4. Select A record is created or updated
5. Set Entry Conditions : None
6. Select Fast Field Updates and click on Done
7. Under the record trigger flow click on the “+” icon and select Get Records.
8. Enter Label as “ Get Purchase Record ”.
9. For Object select Purchase Order.
10. For Condition Requirements , select All Conditions are Met(AND) For the first condition select as follows:

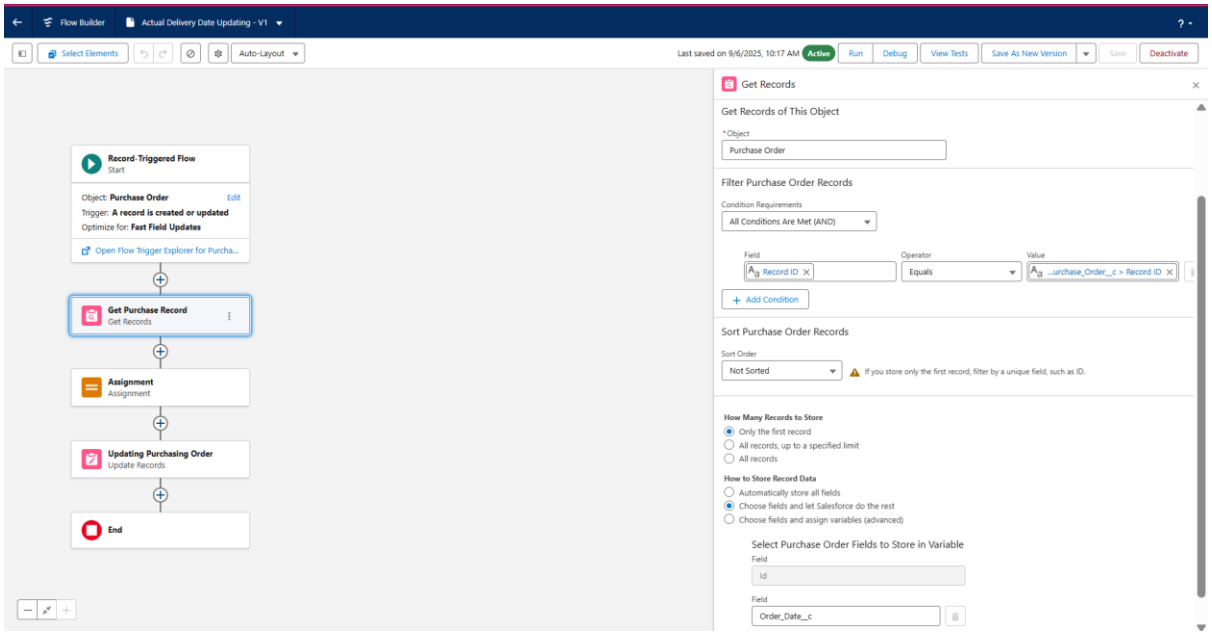
    Field: Id

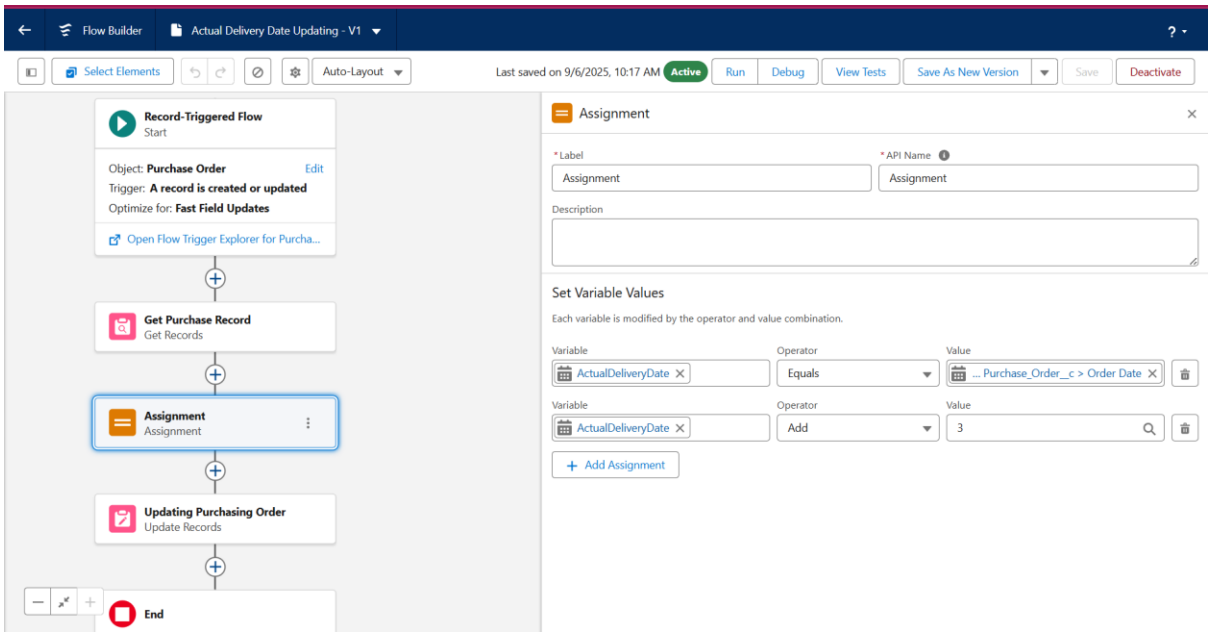
Operator: Equals
Value: {!$Record.Id}

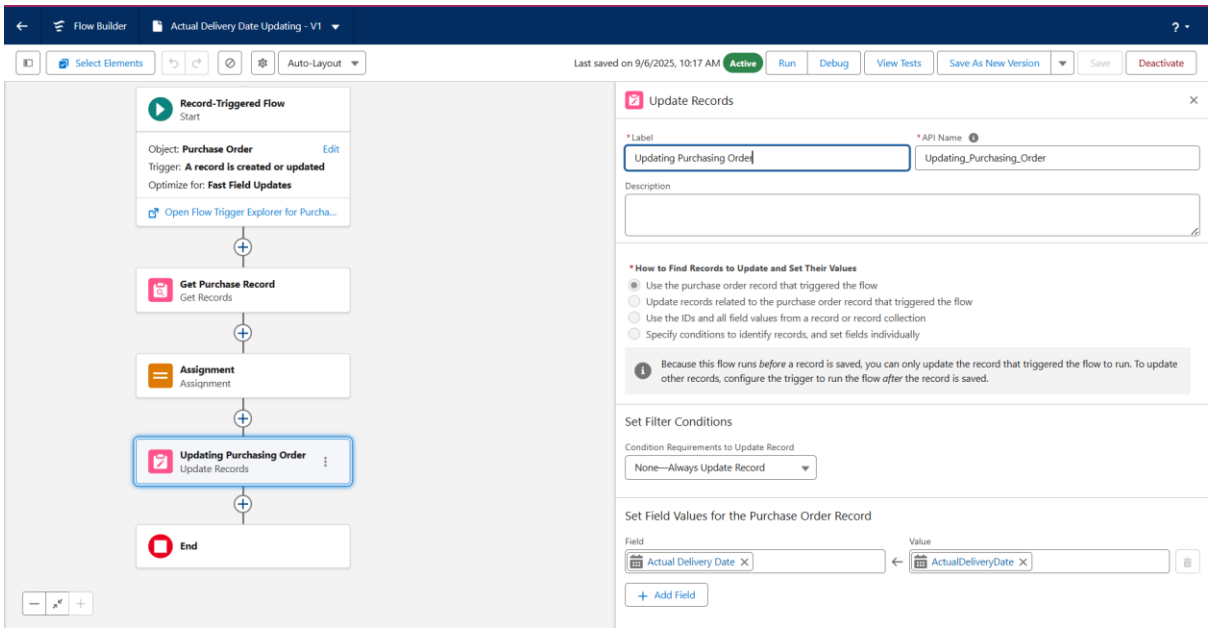
11. For How many Records to store Select Only the First Record.
12. For How to Store Record Data select Choose fields and let Salesforce do the rest. Select Field: Order_Date__c. Click on Done.
13. In the Flow Builder, click on the Manager tab on the left-hand side >>
14. Click on New Resource >> In the Resource Type dropdown, select Variable.
15. Enter API name as ActualDeliveryDate >> Select Data type as Date >> Click on Done.
16. From the Toolbox drag and drop Assignment element.
17. Enter the label as "Assignment".
18. Set Variable Values:
    a) Variable : {!ActualDeliveryDate}
       Operator : Equals
       Value : {!$Record.Order_Date__c}
    b) Variable : {!ActualDeliveryDate}
       Operator : Add
       Value : 3
18. Click Done
19. From the Toolbox drag and drop Update Records element and connect to the Assignment element.
20. Enter the label as "Updating Purchasing Order".
21. How to Find Records to Update and Set Their Values : Use the Purchase Order record that triggered the flow
22. Set Filter Conditions : None -Always Update Record
23. Set Field Values for the Trip Record as
    Field : Actual_Delivery_Date__c
    Value : {!ActualDeliveryDate}
24. Click Done

25. Save the flow as “Actual Delivery Date Updating”.
26. Activate the flow.









Milestone 14 - Triggers

Activity 1 : Create a Trigger to Calculate total amount on Order Item.

Step 1 : Login to Salesforce:

Log in to your Salesforce account with administrative privileges.

Step 2:
i)Navigate to Setup: Once logged in, click on the gear icon ?? (Setup) located at the top-right corner of the page. This will open the Setup menu.

ii)Click on Developer Console: Click on the "Developer Console" option from the Setup menu. This will open the Developer Console in a new browser tab or window.

Step 3:
i) In the Developer Console window, go to the top menu and click on "File".

ii)Select New: From the dropdown menu under "File", select "New".

iii)Choose Apex Trigger: This will open a new Apex Trigger editor tab.

Create an Apex Trigger:

```
trigger CalculateTotalAmountTrigger on Order_Item__c (after insert, after update, after delete, after undelete) {
    // Call the handler class to handle the logic
    CalculateTotalAmountHandler.calculateTotal(Trigger.new, Trigger.old, Trigger.isInsert, Trigger.isUpdate, Trigger.isDelete, Trigger.isUndelete);
}
```

Step 4:
i) In the Developer Console window, go to the top menu and click on "File".

ii)Select New: From the dropdown menu under "File", select "New".

iii)Choose Apex Class: Name it as CalculateTotalAmountHandler

```
public class CalculateTotalAmountHandler {

    // Method to calculate the total amount for Purchase Orders based on related Order Items
    public static void calculateTotal(List<Order_Item__c> newItems, List<Order_Item__c> oldItems, Boolean isInsert, Boolean isUpdate, Boolean isDelete, Boolean isUndelete) {

        // Collect Purchase Order IDs affected by changes in Order_Item__c records
        Set<Id> parentIds = new Set<Id>();

        // For insert, update, and undelete scenarios
        if (isInsert || isUpdate || isUndelete) {
            for (Order_Item__c ordItem : newItems) {
                parentIds.add(ordItem.Purchase_Order_Id__c);
            }
        }


        // For update and delete scenarios
        if (isUpdate || isDelete) {
            for (Order_Item__c ordItem : oldItems) {
                parentIds.add(ordItem.Purchase_Order_Id__c);
            }
        }

        // Calculate the total amounts for affected Purchase Orders
        Map<Id, Decimal> purchaseToUpdateMap = new Map<Id, Decimal>();
```

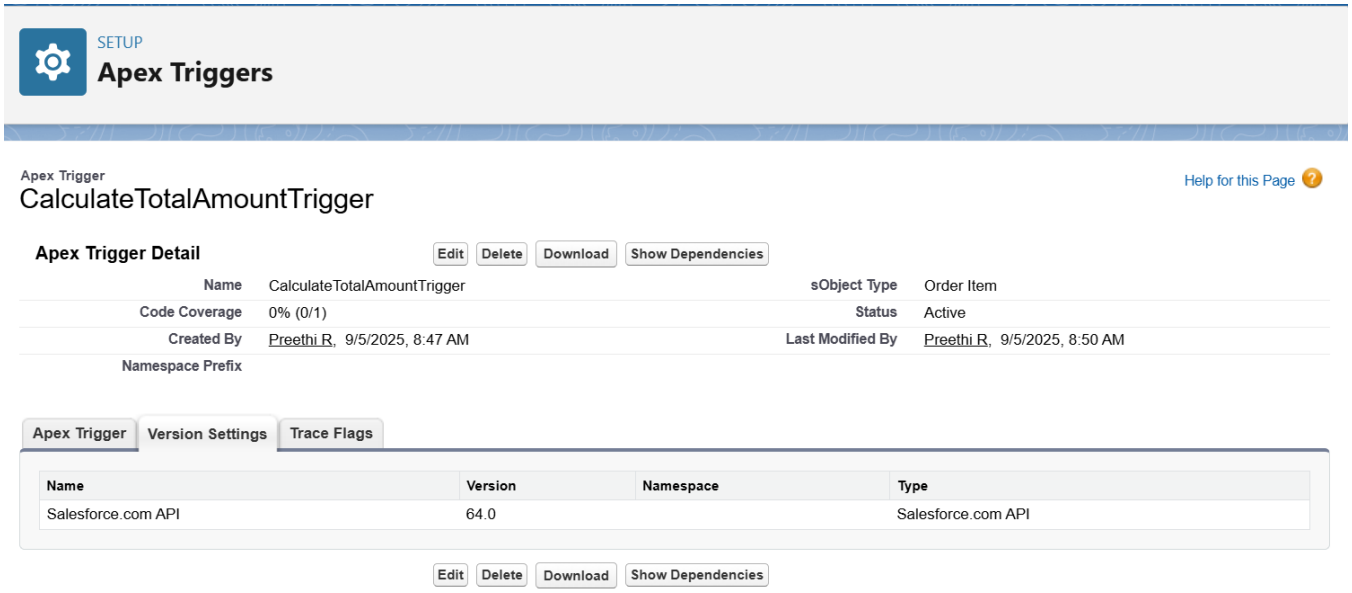
```
    if (!parentIds.isEmpty()) {
        // Perform an aggregate query to sum the Amount__c for each
Purchase Order
        List<AggregateResult> aggrList = [
            SELECT Purchase_Order_Id__c, SUM(Amount__c) totalAmount
            FROM Order_Item__c
            WHERE Purchase_Order_Id__c IN :parentIds
            GROUP BY Purchase_Order_Id__c
        ];
        // Map the result to Purchase Order IDs
        for (AggregateResult aggr : aggrList) {
            Id purchaseOrderId = (Id)aggr.get('Purchase_Order_Id__c');
            Decimal totalAmount = (Decimal)aggr.get('totalAmount');
            purchaseToUpdateMap.put(purchaseOrderId, totalAmount);
        }

        // Prepare Purchase Order records for update
        List<Purchase_Order__c> purchaseToUpdate = new
List<Purchase_Order__c>();
        for (Id purchaseOrderId : purchaseToUpdateMap.keySet()) {
            Purchase_Order__c purchaseOrder = new Purchase_Order__c(Id =
purchaseOrderId, Total_Order_cost__c =
purchaseToUpdateMap.get(purchaseOrderId));
            purchaseToUpdate.add(purchaseOrder);
        }

        // Update Purchase Orders if there are any changes
        if (!purchaseToUpdate.isEmpty()) {
            update purchaseToUpdate;
        }
```

```
        }
    }
}
```

Save it.



Milestone 15 - Reports

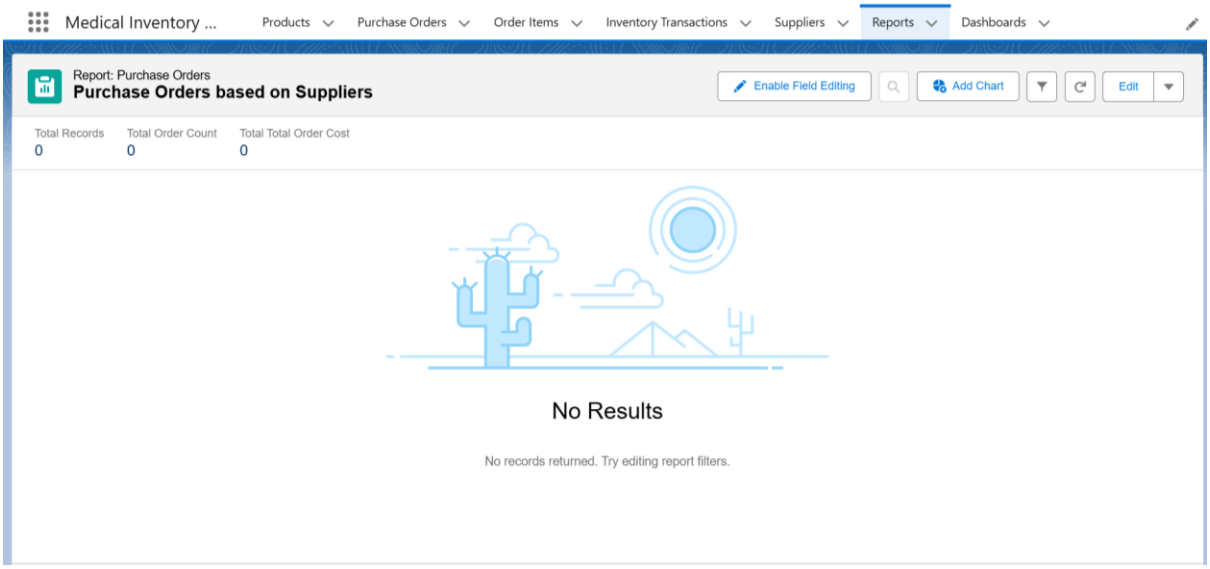
Activity 1: Create a Purchase Orders based on Suppliers(Summary) Report

1. Click App Launcher
2. Select Medical Inventory Management App
3. Click on Reports tab
4. Click on New Report.

5. Click the report type as Purchase Orders Click Start report.
6. Click on Filters and select as follows and click on Apply
7. Customize your report, in group rows select – Supplier ID, Purchase Order: Purchase Order ID, for columns Order Count, Total Order Cost (In this way we are making a Summary Report).
8. Click save and run
9. Give report name – Purchase Orders based on Suppliers.
10. Click Save

NOTE: In this report you can see your all record of the object you selected for reporting

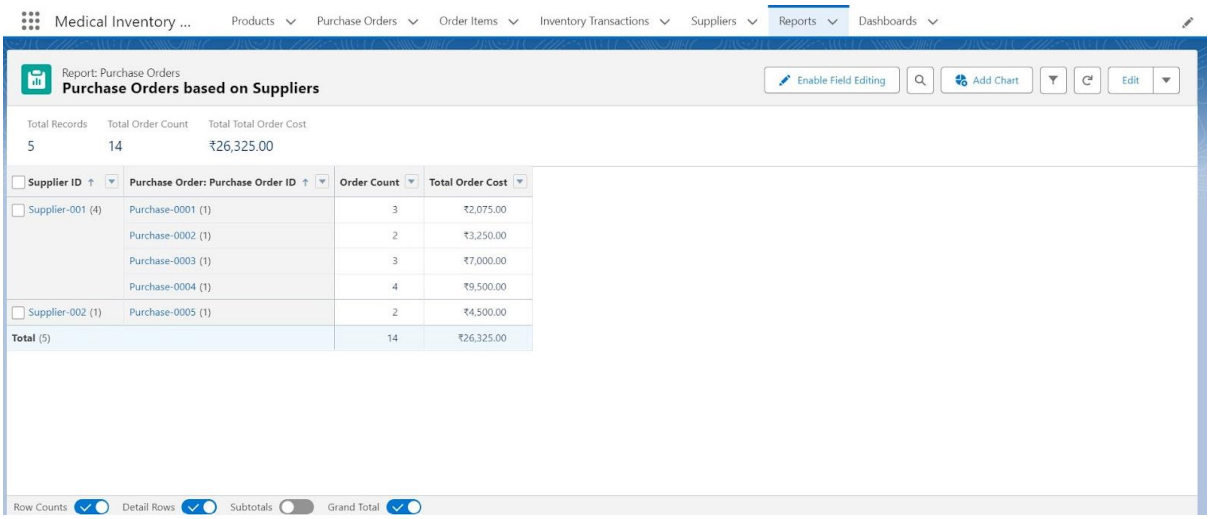
(What you selects in "Select a report type option")



View Report

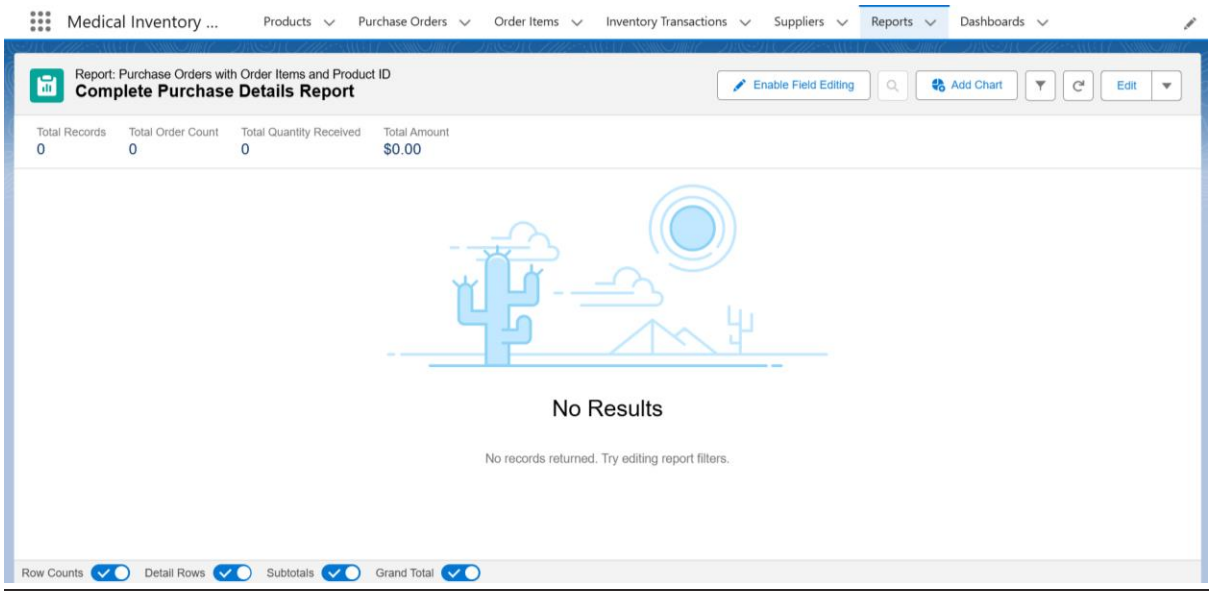
1. Click on App Launcher on the left side of the screen.
2. Search Medical Inventory Management App & click on it.

3. Click on Reports Tab.
4. Click on Purchase Orders based on Suppliers and see records.



Activity 2: Create a Complete Purchase Details Report

1. Click App Launcher
2. Select Medical Inventory Management App
3. Click on Reports tab
4. Click on New Report.
5. Click the report type as Purchase Orders with Order Items and Product ID >> Click Start report.
6. Click on Filters and select as follows and click on Apply
7. Customize your report, in group rows select – Supplier ID, Actual Delivery Date, Purchase Order: Purchase Order ID, for columns Product ID : Product ID, Product ID : Product Name, Order Count, Quantity Received, Amount (In this way we are making a Summary Report).
8. Click save and run
9. Give report name – Complete Purchase Details Report
10. Click Save



Milestone 16 - Dashboard

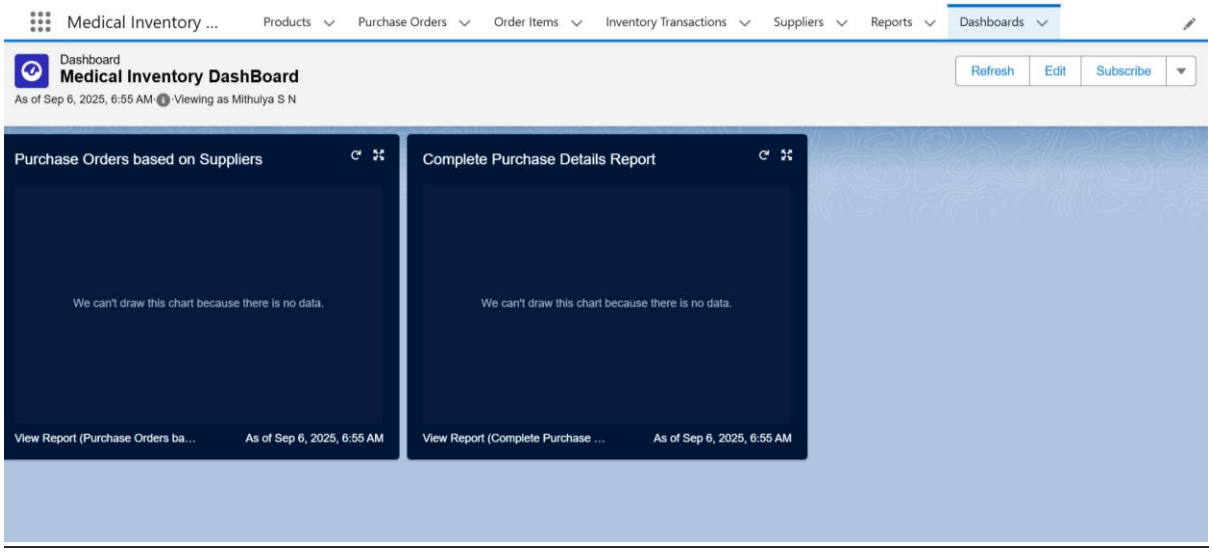
Activity 1: - Create Dashboard

1. Click on the Dashboards tab from the Medical Inventory Management application.
2. Click on the new dashboard.
3. Give name - Medical Inventory DashBoard
4. Click create
5. Click on +widget
6. Select the Purchase Orders based on Suppliers Report
7. For the data visualization select any of the charts, tables etc. as per your choice/requirement
8. Click add.
9. Click save.

Activity 2: View Dashboard

1. Click on App Launcher on the left side of the screen.
2. Search Medical Inventory Management & click on it.

3. Click on Dashboard Tab.
4. Click on Medical Inventory DashBoard see graph view of records



Conclusion :

From a developer's standpoint, creating a **medical inventory management** application with Salesforce is highly valuable. This project provides an opportunity to build a system that automates logistics, offers real-time visibility into stock levels, and generates crucial data for regulatory compliance. The solution empowers healthcare professionals and supply chain teams to prevent product shortages, minimize waste, and ensure the timely availability of essential supplies.

However, the development journey is not without its complexities. It involves tackling challenges like managing highly specific and dynamic product data, integrating with specialized external systems, maintaining strict data security standards to protect sensitive health information, and guaranteeing a robust and scalable architecture that can support critical operations without interruption.

Ultimately, a well-engineered medical inventory system acts as a vital technological link, enhancing operational efficiency and directly supporting patient care. For developers, it's a chance to apply technical skills to solve real-world problems in a high-stakes environment, all while taking on the responsibility of ensuring the solution's security, reliability, and long-term viability.

Project Achievements :

- **Operational Streamlining:** Successfully consolidated supplier management, purchase order processing, and product tracking, boosting efficiency.
- **Real-Time Inventory Control:** Implemented real-time tracking of product stock levels and transaction histories, providing immediate visibility and an audit trail.
- **Enhanced Safety and Compliance:** Developed an automated alert system to flag products nearing expiration, helping prevent the use of expired items.
- **Improved Decision Making:** Created dynamic reports and dashboards that offer valuable insights into supplier performance and purchasing trends.
- **Increased Data Transparency:** Centralized all critical data on Salesforce, establishing a single source of truth that enhances collaboration and provides clear visibility.

Student Learning Outcome :

- **Hands-on Experience:** Students will gain practical skills in Salesforce by configuring objects, automating workflows, and managing real-time inventory.

- **Project Lifecycle Understanding:** Students will learn the end-to-end process of a Salesforce project, from requirements to deployment.
- **Analytical Skills:** Students will develop the ability to identify challenges, design solutions, and troubleshoot issues within the system.
- **Collaboration Skills:** Students will gain experience working in a team on tasks such as requirement gathering, development, and testing.
- **Industry Exposure:** Students will be exposed to real-world applications of Salesforce in a medical context, preparing them for future career opportunities.

Future Scope :

- **Mobile Integration:** Develop a mobile app for on-the-go inventory management.
- **Advanced Analytics & AI-Powered Forecasting:** Use AI to predict inventory needs and minimize waste.
- **Integration with E-commerce & Supplier Portals:** Integrate with supplier portals for automated reordering.
- **Enhanced Barcode & QR Code Scanning:** Enhance the system with barcode and QR code scanning for faster audits.
- **IoT & Sensor Integration:** Integrate IoT sensors for real-time monitoring of sensitive supplies.
- **Multi-Location Management:** Expand the system to manage multiple warehouse locations.
- **Vendor Performance Scoring:** Implement a scoring system to evaluate vendor performance.