

PROJECT TITLE:

Medical Inventory Management

College Name:

Sri Vasavi College Self Finance Wing, Erode

College Code:

bru17

Team ID: NM2025TMID23361

TEAM MEMBERS:

Team Leader Name: SIVACHANDRAN S

Email: shivask0332@gmail.com

Team Member: HARIHARAN M

Email: harithiva1234@gmail.com

Team Member: SHRIL S

Email: johnpalshril408@gmail.com

Team Member: MADHUMITHA R

Email: madhued06@gmail.com

Medical Inventory Management Project Documentation

1. Project Overview

The **Medical Inventory Management System** is a Salesforce application designed to streamline the process of managing medical inventory. It helps track suppliers, products, purchase orders, order items, and inventory transactions in a single platform.

This system improves:

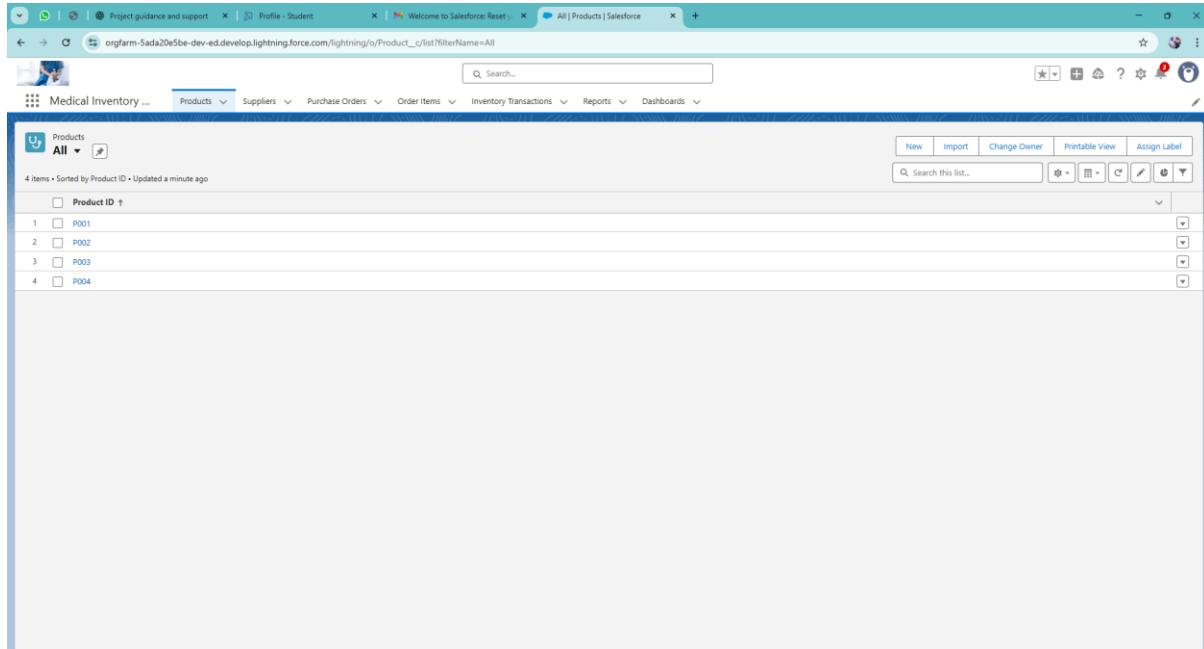
- Supplier tracking
- Stock level monitoring
- Order management
- Expiry date tracking
- Report generation and analytics

2. Objectives

The main objectives of this project are:

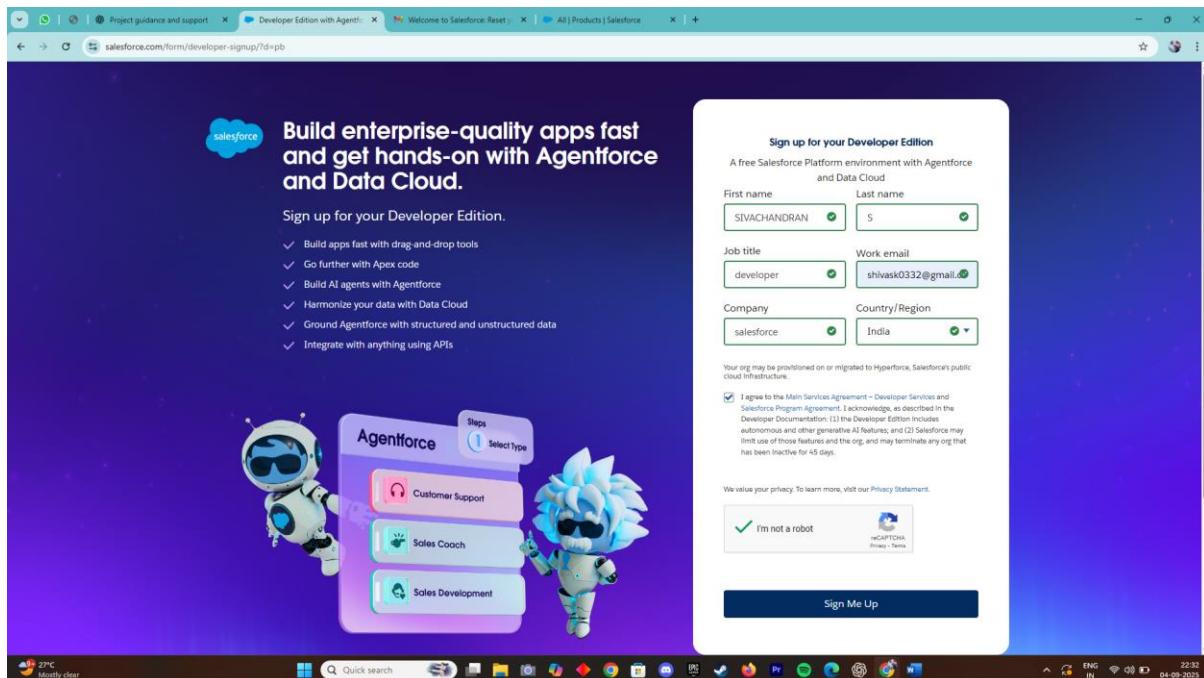
1. To efficiently manage and track **medical products** and their **stock levels**.
2. To streamline the **purchase order process** by linking suppliers and products.
3. To ensure **real-time monitoring** of stock and expiry dates.
4. To generate **insightful reports and dashboards** for decision-making.
5. To automate repetitive tasks using Salesforce tools like **flows and validation rules**.

Medical Inventory Management App.



3.Created a developer org in salesforce.

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



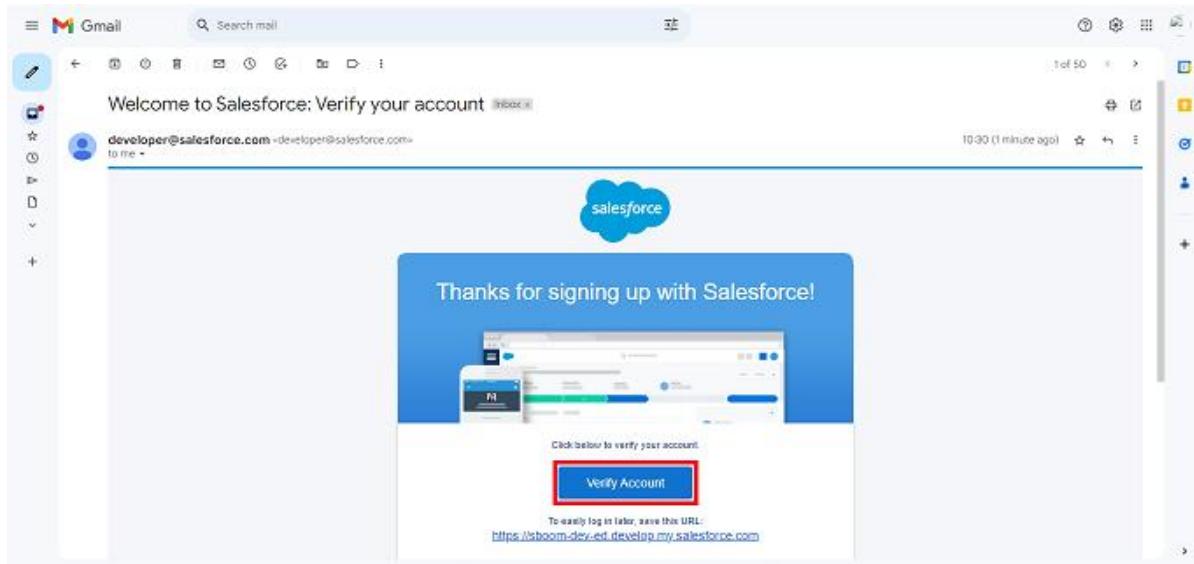
1. First name & Last name
2. Email

3. Role : Developer
4. Company : College Name
5. County : India
6. Postal Code : pin code

Click on sign me up after filling these.

4.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.

The screenshot shows the 'Change Your Password' page in Salesforce. It prompts the user to enter a new password for the email lead@sb.oom, specifying requirements: 8 characters, 1 letter, and 1 number. The 'New Password' field contains 'Good' and the 'Confirm New Password' field contains 'Match'. Below these, a 'Security Question' dropdown is set to 'In what city were you born?' and the 'Answer' field contains 'asdfghjkl'. A large red box highlights the entire form area, and a blue box highlights the 'Change Password' button at the bottom.

4. Then you will redirect to your salesforce setup page.

The screenshot shows the Salesforce Setup Home page. The left sidebar includes links for Service Setup Assistant, Multi-Factor Authentication Assistant, Release Updates, Lightning Experience Transition Assistant, Salesforce Mobile App, Lightning Usage, Optimizer, and Administration. The main content area displays three cards: 'Get Started with Einstein Bots' (Launch an AI-powered bot to automate your digital connections), 'Mobile Publisher' (Use the Mobile Publisher to create your own branded mobile app), and 'Real-time Collaborative Docs' (Transform productivity with collaborative docs, spreadsheets, and slides inside Salesforce). A 'Create' button is located in the top right corner of the main content area.

5. Objects Created

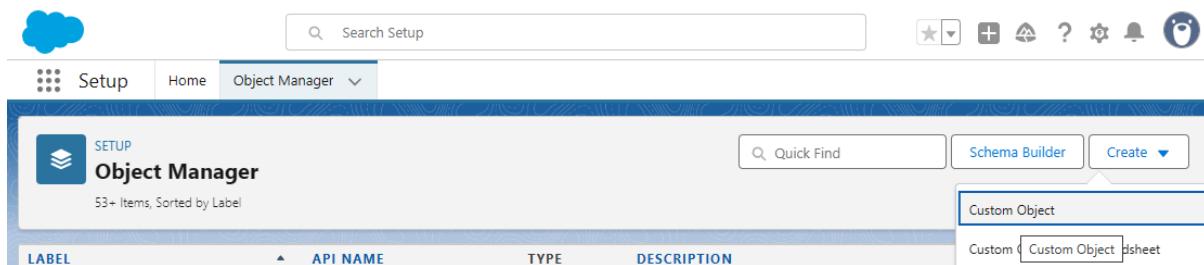
The following **custom objects** were created to manage the system:

Object Name	Purpose
Supplier	To store supplier details like name, contact person, phone, and address.

Object Name	Purpose
Product	To store product information like unit price, current stock level, and minimum stock level.
Purchase Order	To record order details from suppliers.
Order Item	To link products with each purchase order and track quantities and costs.
Inventory Transaction	To track stock movement (Inbound/Outbound).
In Salesforce, objects are database tables that allow you to store data specific to your organization.	

To create an 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



New Custom Object

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles. [Tell me more](#) | [Don't show this message again](#)

Custom Object Definition Edit

Custom Object Information

The singular and plural labels are used in tabs, page layouts, and reports.

Label: Product **Example:** Account

Plural Label: Products **Example:** Accounts

Starts with vowel sound:

The Object Name is used when referencing the object via the API.

Object Name: Product **Example:** Account

Description:

Context-Sensitive Help Setting: Open the standard Salesforce.com Help & Training window
 Open a window using a Visualforce page

Content Name:

Enter Record Name Label and Format

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name: Product ID **Example:** Account Name

Data Type: Text **Warning:** If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

Optional Features

- Allow Reports
- Allow Activities
- Track Field History
- Allow in Chatter Groups
- Enable Licensing

Object Classification

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more](#).

- Allow Sharing
- Allow Bulk API Access
- Allow Streaming API Access

Deployment Status

- In Development
- Deployed

Search Status

When this setting is enabled, your users can find records of this object type when they search. [Learn more](#).

Allow Search

Object Creation Options (Available only when custom object is first created)

- Add Notes and Attachments related list to default page layout
- Launch New Custom Tab Wizard after saving this custom object

In the same way Created Purchase Order, Order Item, Inventory Transaction and Supplier objects.

6.Tabs Created

In Salesforce, tabs are used to make the data stored in objects accessible to users through the user interface. Tabs are a fundamental part of the Salesforce interface, providing a way to navigate to different objects and records.

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

Custom Tabs

You can create new custom tabs to extend Salesforce functionality or to build new application functionality. Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external web applications and content within the Salesforce window. Visualforce tabs allow you to embed Visualforce pages. Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app. Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.

Category	Action	Description
Custom Object Tabs	New	What Is This?
Web Tabs	New	What Is This?
Visualforce Tabs	New	What Is This?
Lightning Component Tabs	New	What Is This?
Lightning Page Tabs	New	What Is This?

Edit Custom Object Tab Products

Fill in the fields below to define the custom tab.

Custom Tab Definition Edit

Custom Object Tab Information

Tab Label	Products
Object	Product
Tab Style	Stethoscope

(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.
Splash Page Custom Link:

Enter a short description.

Description:

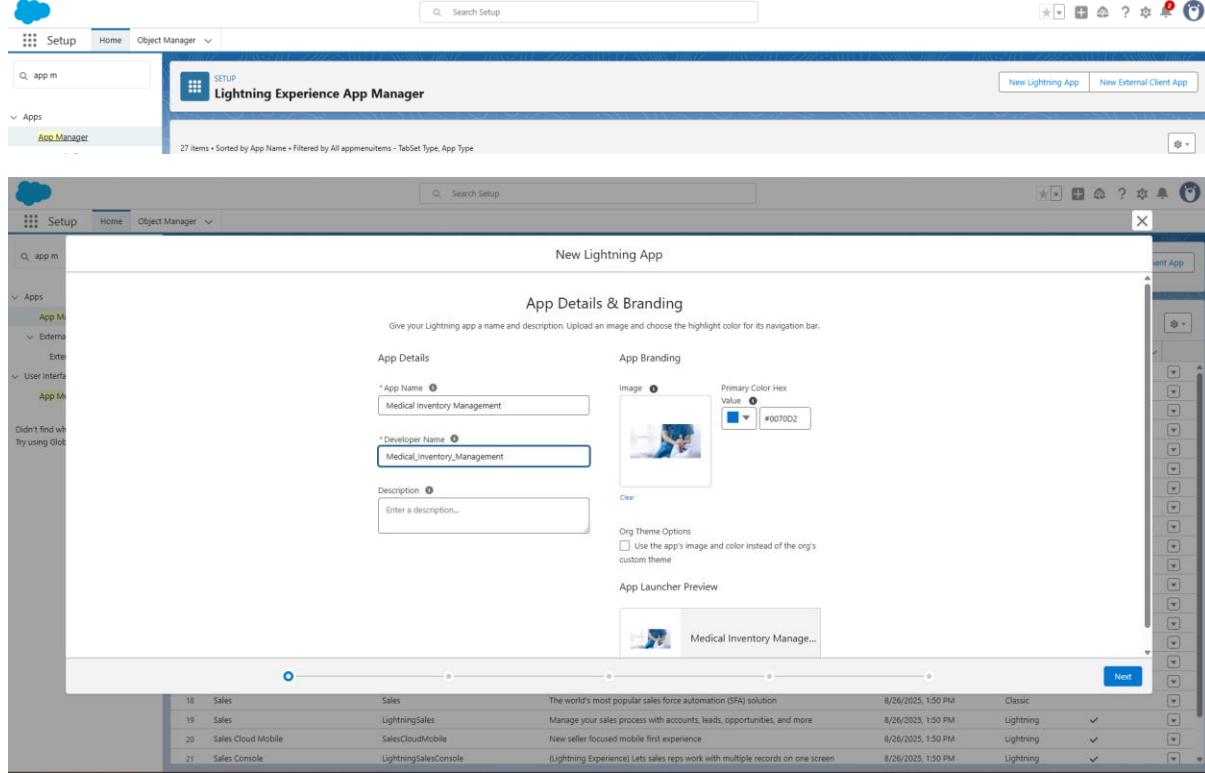
1. Now created the Tabs for the remaining Objects, they are “Purchase Order, Order Item, Inventory Transaction, Supplier”.
2. Followed the same steps as mentioned in above.

The screenshot shows the Salesforce Setup interface under the 'Tabs' section. The left sidebar has 'User Interface' expanded, with 'Tabs' selected. The main area is titled 'Custom Tabs' and contains sections for 'Custom Object Tabs', 'Web Tabs', 'Visualforce Tabs', 'Lightning Component Tabs', and 'Lightning Page Tabs'. Under 'Custom Object Tabs', there is a table listing tabs for 'Inventory Transactions', 'Order Items', 'Products', 'Purchase Orders', and 'Suppliers'. Each row includes 'Edit' and 'Delete' buttons, a 'Label' column, a 'Tab Style' column showing a blue square icon, and a 'Description' column which is empty. The 'Web Tabs', 'Visualforce Tabs', 'Lightning Component Tabs', and 'Lightning Page Tabs' sections all indicate 'No [Type] Tabs have been defined'.

7. Created 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.



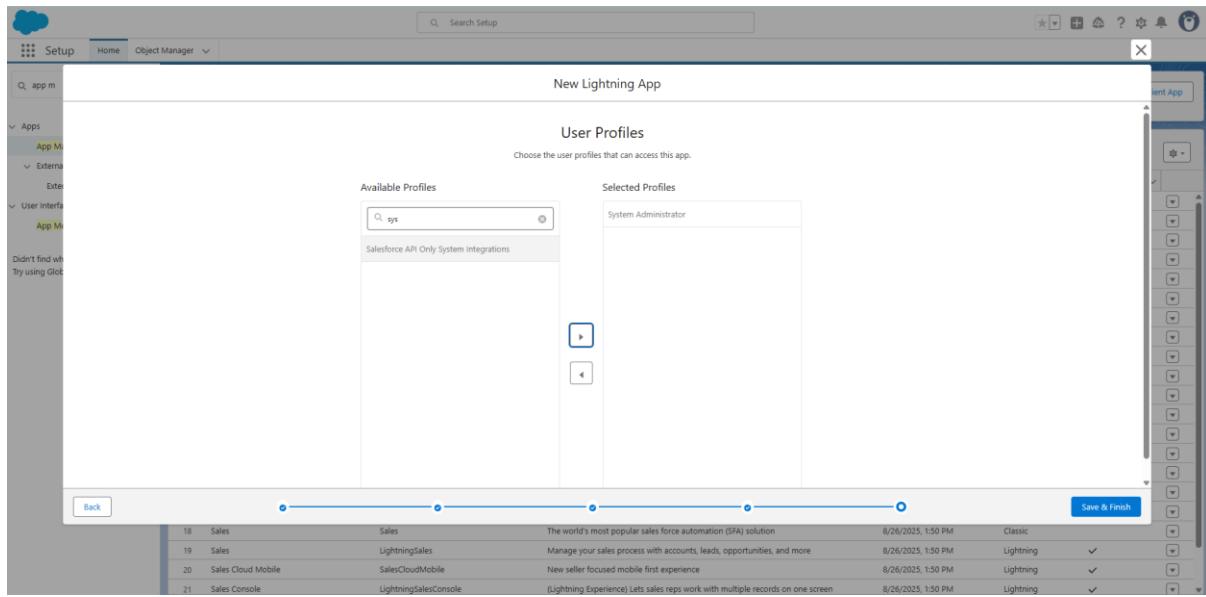
Available Items

- Accounts
- Activation Targets
- Activations
- All Sites
- Alternative Payment Methods
- Analytics
- App Launcher
- Appointment Categories
- Appointment Invitations
- Approval Requests
- Approval Submission Details
- Approval Submissions
- Approval Work Items
- Asset Action Sources
- Asset Actions
- Asset State Periods

Selected Items

■ Products
■ Suppliers
■ Purchase Orders
■ Order Items
■ Inventory Transactions
■ Reports
■ Dashboards

▲
▼



8. Created a Text Field in Product 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.

Object Manager						
LABEL	API NAME	TYPE	DESCRIPTION	LAST_MODIFIED	DEPLOYED	
Purchase Order	Purchase_Order__c	Custom Object		9/5/2025	✓	
Product	Product__c	Custom Object		9/4/2025	✓	
Supplier	Supplier__c	Custom Object		9/4/2025	✓	
Inventory Transaction	Inventory_Transaction__c	Custom Object		9/4/2025	✓	
Order Item	Order_Item__c	Custom Object		9/4/2025	✓	

Setup | Home | Object Manager | Product

SETUP > OBJECT MANAGER

Product

Fields & Relationships

Details

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

Object Access

Triggers

Flow Triggers

Validation Rules

Conditional Field Formatting

Fields & Relationships

11 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Current Stock Level	Current_Stock_Level__c	Number(18, 0)		
Expiry Date	Expiry_Date__c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
Minimum Stock Level	Minimum_Stock_Level__c	Number(18, 0)		
Owner	OwnerId	Lookup(User, Group)		✓
Product Description	Product_Description__c	Text Area(255)		
Product ID	Product_ID__c	Text(255)		
Product ID	Name	Text(80)		✓
Product Name	Product_Name__c	Text(255)		
Unit Price	Unit_Price__c	Currency(16, 2)		

Setup | Home | Object Manager | Product

SETUP > OBJECT MANAGER

Product

Fields & Relationships

Details

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

Object Access

Triggers

Flow Triggers

Validation Rules

Conditional Field Formatting

Fields & Relationships

- Lookup Relationship
- Master-Detail Relationship
- External Lookup Relationship

Creates a relationship from this object to another object. The relationship field allows users to click on a lookup icon to select a value from a pop-up list. The master object is the source of the values in the list.

Creates a special type of parent-child relationship between this object (the child, or "detail") and another object (the parent, or "master") where:

- The relationship field is required on all detail records.
- The relationship field and any of its detail fields are determined by the master record.
- You can use relationships to define the scope of all detail records as defined.
- You can create rollup summary fields on the master record to summarize the detail records.

 The relationship field allows users to click on a lookup icon to select a value from a pop-up list. The master object is the source of the values in the list.

Creates a relationship that links this object to an external object whose data is stored outside the Salesforce org.

Setup | Home | Object Manager | Product

SETUP > OBJECT MANAGER

Product

Fields & Relationships

Details

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

Object Access

Triggers

Flow Triggers

Validation Rules

Conditional Field Formatting

New Custom Field

Step 2. Enter the details

Step 2 of 4

Previous Next Cancel

Field Label: **Product Name**

Please enter the maximum length for a text field below:

Length: **255**

Field Name: **Product_Name**

Description:

Help Text:

Required: Always require a value in this field in order to save a record

Unique: Do not allow duplicate values

External ID: Treat "ABC" and "abc" as duplicate values (case insensitive)

External ID: Treat "ABC" and "abc" as different values (case sensitive)

Auto add to custom report type: Add this field to existing custom report types that contain this entity

Default Value: **Show Formula Editor**

Use Simple Editor: Enter text and populate values with ranges in double quotes (""). Use `$(id)` to include numbers without quotes. Use single quotes for values in `$(id)`, and express date values in the standard format: `$(date)`. To reference a field from a Custom Metadata type record use: `$(CustomMetadataType__RecordName.Field__c)`

Previous Next Cancel

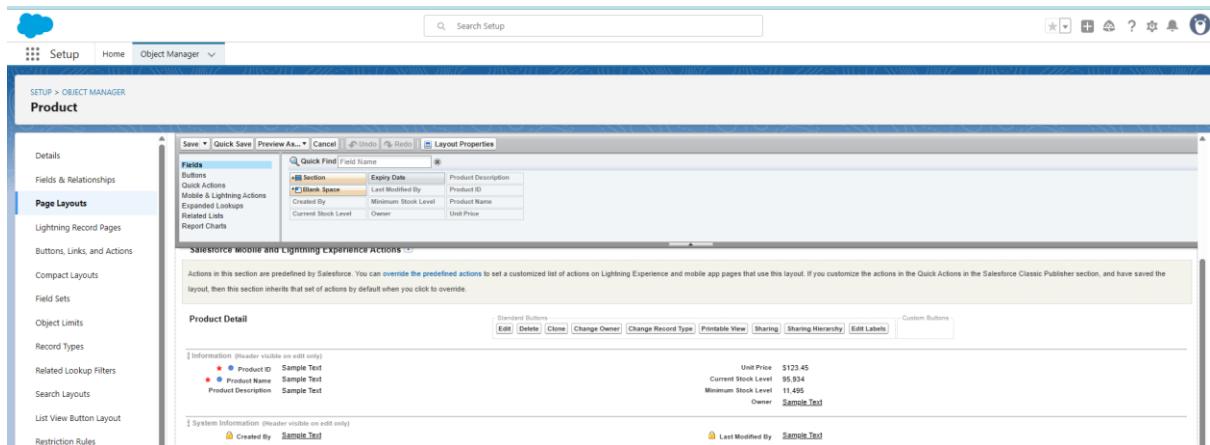
Field Added in Object:

Object	Field Name	Data Type
Product	Product ID(Standard)	Text
	Product Name	Text
	Product Description	Text Area
	Minimum Stock Level	Number(18, 0)
	Current Stock Level	Number(18, 0)
	Unit Price	Currency(16, 2)
	Expiry Date	Date
Purchase Order	Purchase Order ID(Standard)	Text
	Supplier ID	Lookup(Supplier)
	Order Date	Date
	Expected Delivery Date	Date
	Actual Delivery Date	Date
	Order Count	Roll-Up Summary (COUNT Order Item)
	Total Order Cost	Currency(16, 2)
Order Item	Order Item ID(Standard)	Text
	Product ID	Lookup(Product)
	Purchase Order ID	Master-Detail(Purchase Order)
	Quantity Ordered	Number(18, 0)

	Quantity Received	Number(18, 0)
	Unit Price	Formula(Currency)
	Amount	Formula(Currency)
Inventory Transaction	Transaction ID(Standard)	Text
	Purchase Order ID	Lookup(Purchase Order)
	Transaction Date	Date
	Transaction Type	Picklist
	Total Order Cost	Formula(Currency)
Supplier	Supplier ID(Standard)	Text
	Supplier Name	Text
	Contact Person	Text
	Phone Number	Phone
	Email	Email
	Address	TextArea

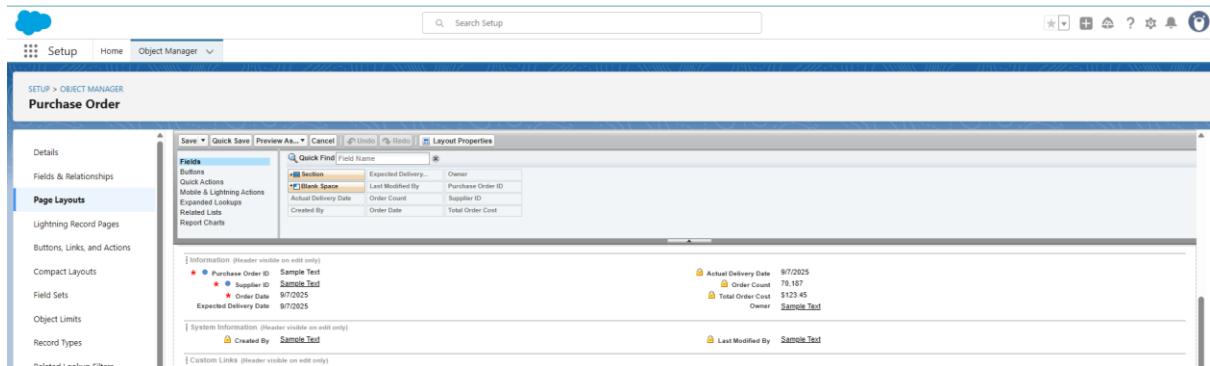
9. Edited a Page Layout in Product Object

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.

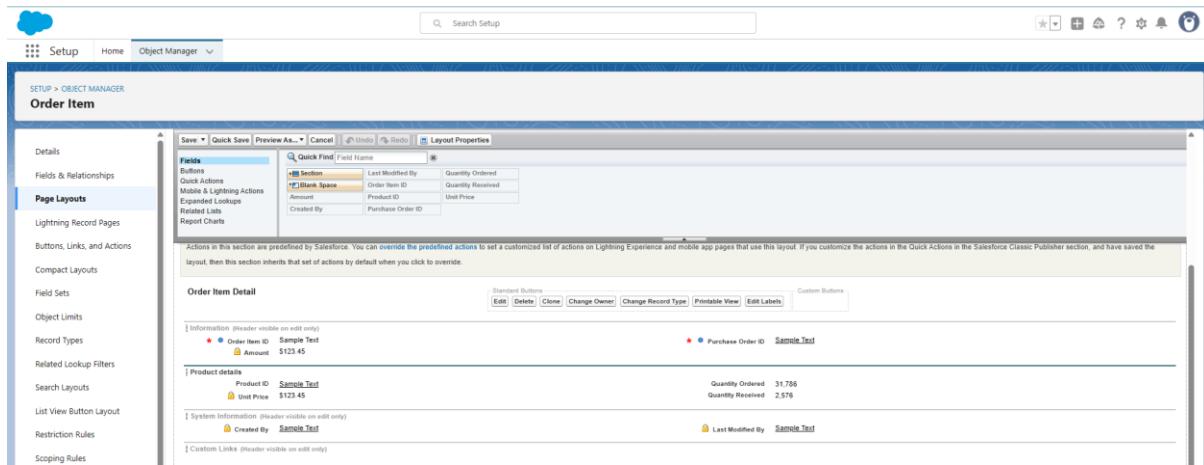


4. Click on Save.

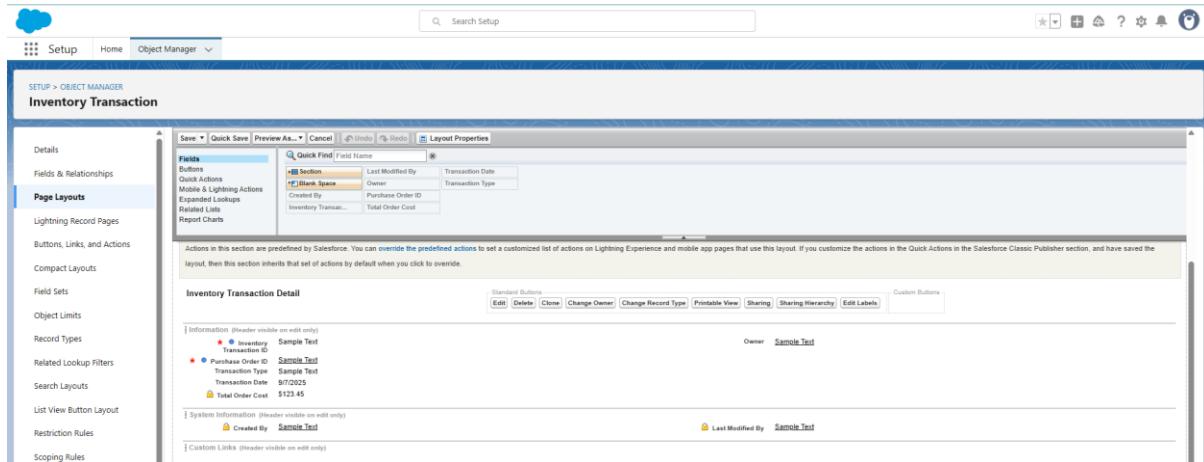
9.1 Edited a Page Layout in Purchase Order Object



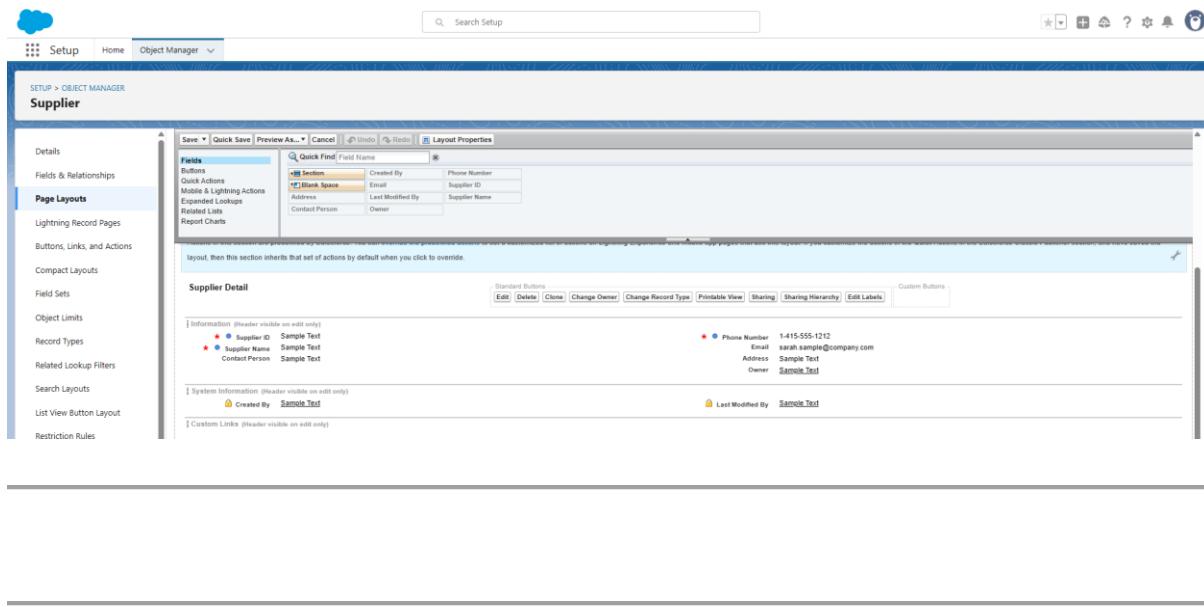
9.2 Edited a Page Layout in Order Item Object



9.3 Edited a Page Layout in Inventory Transaction Object



9.4 Edited a Page Layout in Supplier Object



10. Created 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.

Setup Home Object Manager

SETUP > OBJECT MANAGER Product

Compact Layouts 2 Items, Sorted by Label

LABEL	API NAME	PRIMARY	MODIFIED BY	LAST MODIFIED
Product Compact Layout	Product_Compact_Layout	✓	SIVACHANDRAN S	9/6/2025, 11:21 AM
System Default	SYSTEM			

Details Fields & Relationships Page Layouts Lightning Record Pages Buttons, Links, and Actions Compact Layouts Field Sets

Search Setup Quick Find New Compact Layout Assignment Compact Layout

Setup Home Object Manager

SETUP > OBJECT MANAGER Product

Compact Layouts Edit Product Compact Layout Help for this Page

Product Compact Layout

Compact layouts are used in the mobile app and some Chatter feed items to display a record's key fields at a glance. You can select and prioritize up to ten fields for the compact layout, but the number of fields that display may vary based on the device's screen, which record page is being viewed, and the permissions of the user.

Enter Compact Layout Information

Label: Product Compact Layout
Name: Product_Compact_Layout

Select Compact Layout Fields

Available Fields: Created By, Expiry Date, Last Modified By, Minimum Stock Level, Owner, Product ID
Selected Fields: Product Name, Unit Price, Current Stock Level

Top ▲ Up ▲ Down ▼ Bottom

Use SHIFT + click to select adjacent fields. Use CTRL + click to select an assortment of fields.

Save Cancel

Setup Home Object Manager

SETUP > OBJECT MANAGER Product

Compact Layouts Assignment

Product Compact Layouts Compact Layout Assignment Help for this Page

Compact Layout Assignment

Primary Compact Layout

Select the compact layout to use when this object's records appear as list items in the mobile app.

Primary Compact Layout: Product Compact Layout

Save Cancel

10.1 Created a Compact Layout to a Purchase Order Object

The image contains two screenshots of the Salesforce Setup interface, specifically the Object Manager for the Purchase Order object.

Screenshot 1: Edit Purchase Order Compact Layout - Purchase Order Compact Layout

This screenshot shows the "Edit Purchase Order Compact Layout" page. The "Label" field is set to "Purchase Order Compact Layout" and the "Name" field is set to "Purchase_Order_Compact_L". The "Select Compact Layout Fields" section lists available fields like Actual Delivery Date, Created By, Expected Delivery Date, Last Modified By, Order Count, and Owner. The "Selected Fields" section includes Purchase Order ID, Order Date, Total Order Cost, and Supplier ID. A vertical arrangement tool on the right allows moving fields between sections.

Screenshot 2: Purchase Order Compact Layouts - Compact Layout Assignment

This screenshot shows the "Compact Layout Assignment" page. It lists the "Primary Compact Layout" as "Purchase Order Compact Layout".

11. Create an Expected Delivery Date Validation rule 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 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 :
$$(\text{Expected_Delivery_Date_c} - \text{Order_Date_c}) > 7$$

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.

The screenshot shows the 'Validation Rule Edit' screen for a 'Purchase Order' object. The 'Rule Name' is set to 'Expected_Delivery_Date_Validation'. The 'Active' checkbox is checked. The 'Error Condition Formula' field contains the formula: `|Expected_Delivery_Date__c - Order_Date__c| > 7`. The 'Error Message' field contains the message: 'The Expected Delivery Date should not exceed 7 days.' The 'Error Location' is set to 'Top of Page'. The sidebar on the left shows various setup categories like Details, Fields & Relationships, Page Layouts, etc., with 'Validation Rules' selected.

12. Created an Inventory Manager Profile

The first screenshot shows the 'Profiles' screen with a list of existing profiles. A new profile named 'Custom' is being created. The second screenshot shows the 'Clone Profile' screen where a new profile named 'Inventory Manager' is being cloned from the 'Standard User' profile.

12.1 Created an Purchase Manager Profile

The screenshot shows the Salesforce Setup page with the following sections:

- Custom Object Permissions**: A grid where you can define permissions for various objects like Inventory Transactions, Order Items, Products, Purchase Orders, and Suppliers across basic access (Read, Create, Edit, Delete) and data administration (View All Records, Modify All Records, View All Fields).
- Session Settings**: Set session times out after 2 hours of inactivity and specify session security levels required at login.
- Password Policies**: Configure password expiration (Never expires), history (3 passwords remembered), length (8 characters), complexity (Must include alpha and numeric characters), and other requirements like requiring a minimum 1 day password lifetime and locking accounts after 10 invalid logins.

13. Created a Purchasing Manager Role.

The screenshot shows the Salesforce Roles setup page with the following sections:

- Understanding Roles**: A diagram illustrating the role hierarchy. At the top is the **Executive Staff** role, which includes the **CEO**, **President**, and **CFO, VP, Sales**. Below them are the **Western Sales Director**, **Eastern Sales Director**, and **International Sales Director**. Each of these oversees multiple **Sales Rep** roles: Western Sales Reps (CA Sales Rep, OR Sales Rep), Eastern Sales Reps (NY Sales Rep, MA Sales Rep), and International Sales Reps (Asia Sales Rep, European Sales Rep).
- Help for this Page**: A link to additional documentation.

The screenshot shows the Role Edit page for the **Purchasing Manager** role:

- Role Edit**: The page title.
- Label**: **Purchasing Manager**.
- Role Name**: **Purchasing_Manager**.
- This role reports to**: **SVP, Sales & Marketing**.
- Role Name as displayed on reports**: An empty input field.
- Save**, **Save & New**, and **Cancel** buttons at the bottom.

13.1 Created a Inventory Manager Role.

The top screenshot displays the 'Roles' page in the Salesforce Setup. It features a sidebar with navigation links like 'Users', 'Sales', 'Service', and 'Case Teams'. The main content area is titled 'Understanding Roles' and contains a 'Sample Role Hierarchy' diagram for 'Territory-based Sample'. The diagram shows a hierarchy from 'Executive Staff' down to 'Western Sales Rep'. The bottom screenshot shows the 'Role Edit' page for the 'Inventory Manager' role. It includes fields for 'Label' (set to 'Inventory Manager'), 'Role Name' (set to 'Inventory_Manager'), and 'This role reports to' (set to 'SVP. Sales & Marketing').

14. Created a Permission Set.

1. Go to setup >> type Permission in quick find box >> Select Permission Set >> click on New.

The screenshot shows the 'Permission Sets' page in the Salesforce Setup. It lists several permission sets such as 'Customer Data Platform', 'Customer Data Cloud for Marketing', and 'Customer Data Cloud for Marketing Data Aware Specialist'. At the top left, there is a 'New' button with a dropdown menu. The page also includes a search bar and navigation links for 'All Permission Sets', 'Edit', and 'Create New View'.

2. Enter Label as Purchase Manager Create Access >> Click on Save.

Permission Set Create

Enter permission set information

Label: Purchase Manager Create Access

API Name: Purchase_Manager_Create_Access

Description:

Session Activation Required:

3. From Object Settings >> Select Order Item >> Enable for both Tab Available and Visible >> Enable Read and Create in Object Permissions >> Click on Save.

Setup Home Object Manager

Permission Set

Purchase Manager Create Access

Object Settings Order Items

Tab Settings

Available	Visible
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Object Permissions

Permission Name	Enabled
Read	<input checked="" type="checkbox"/>
Create	<input checked="" type="checkbox"/>
Edit	<input type="checkbox"/>
Delete	<input type="checkbox"/>
View All Records	<input type="checkbox"/>
Modify All Records	<input type="checkbox"/>
View All Fields	<input type="checkbox"/>

4. Navigate to the Permission Set detail page >> Click Manage Assignments >> Click Add Assignments >> Select the user SIVACHANDRAN S to assign the permission set to and click Next.

Permission Set

Purchase Manager Create Access

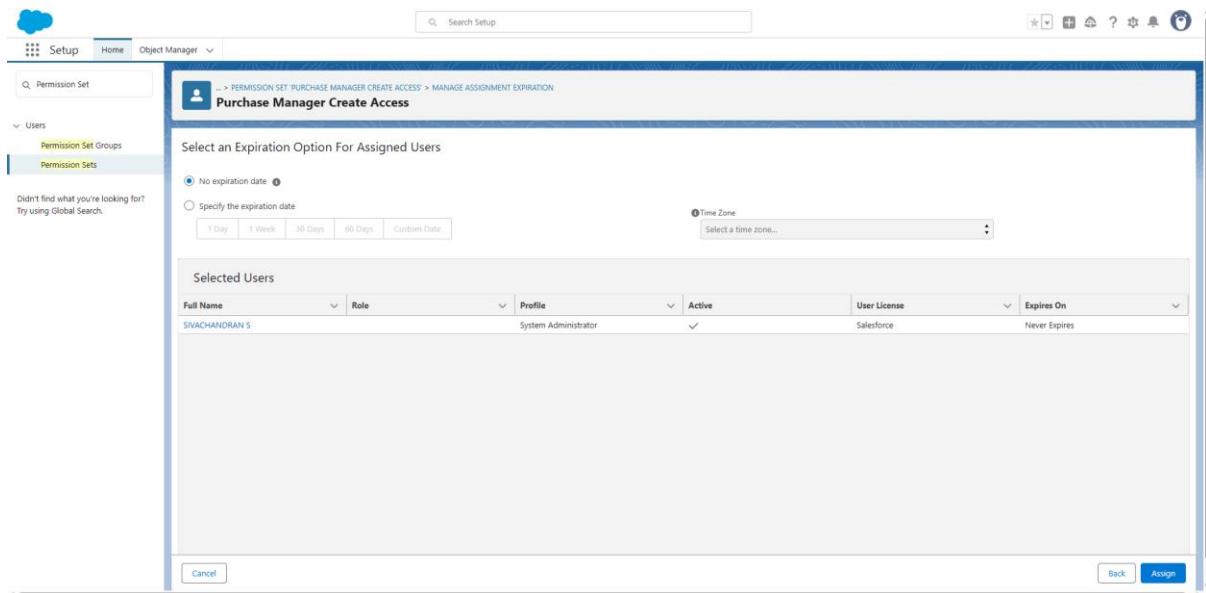
Select Users to Assign

All Users

Full Name	Alias	Username	Role	Active	Profile
Chatter Expert	Chatter	chatty.00dgk00000a2zrluaf-eblurzziwzp@chatter.salesforce.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Chatter Free User
Integration User	integ	integration@00dgk00000a2zrluaf.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Analytics Cloud Integration User
OrgFarm EPIC	OEPIC	epic.c2b788d28a58@orgfarm.salesforce.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	System Administrator
Security User	sec	insightssecurity@00dgk00000a2zrluaf.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Analytics Cloud Security User
SIVACHANDRAN S	shi	shivask0332956@agentforce.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	System Administrator

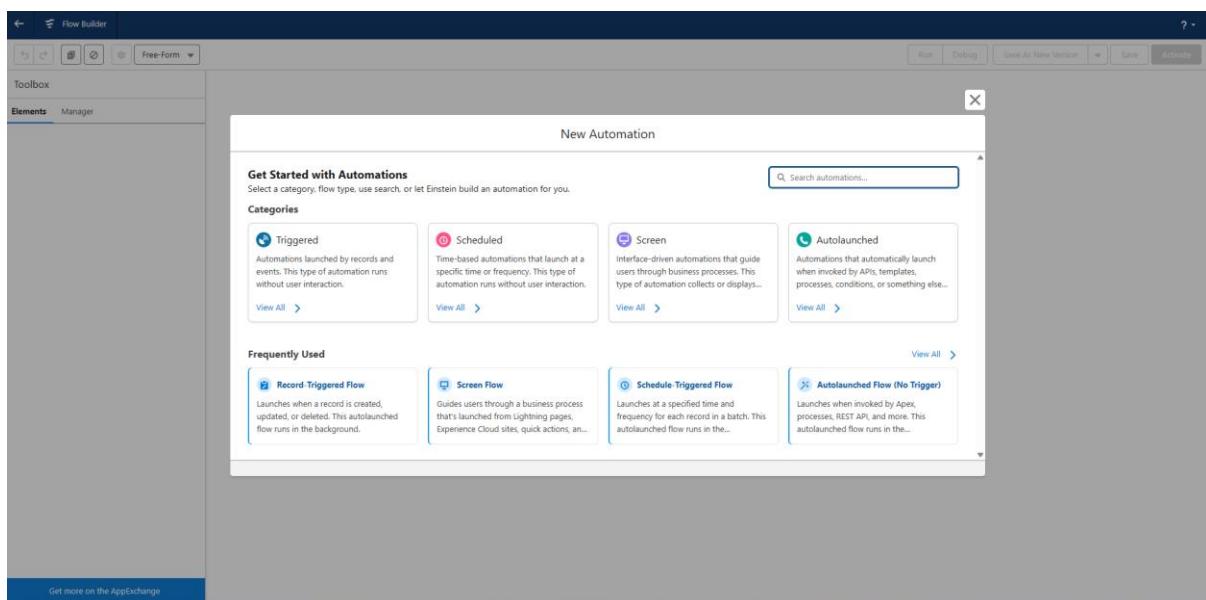
Cancel Next

5. Select No Expiration date >> Click on Assign.



15. Created Flow to update the Actual Delivery Date.

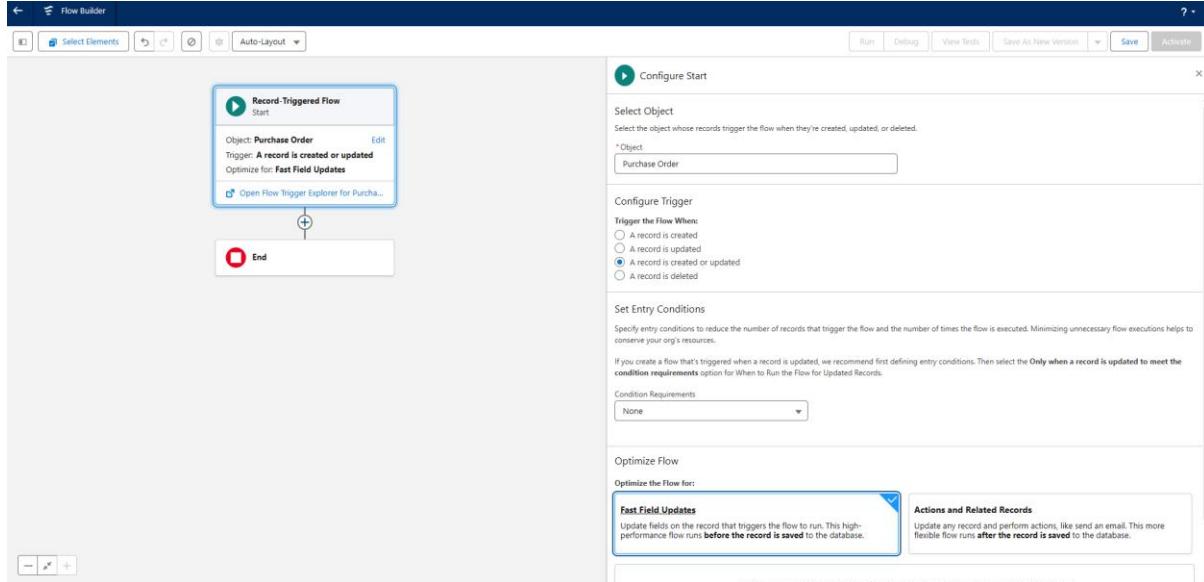
1. Go to setup >> type Flow in quick find box >> Click on the Flow and Select the New Flow >> Start From Scratch .
2. Select the record Triggered flow.Click on create.



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}

 Get Records

* Label

* API Name

Description

Get Records of This Object

* Object

Filter Purchase Order Records

Condition Requirements

Field	Operator	Value
Aa Record ID	Equals	Aa ...urchase_Order__c > Record ID

[+ Add Condition](#)

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.

How Many Records to Store

- Only the first record
- All records, up to a specified limit
- All records

How to Store Record Data

- Automatically store all fields
- Choose fields and let Salesforce do the rest
- Choose fields and assign variables (advanced)

Select Purchase Order Fields to Store in Variable

Field

Field



[+ Add Field](#)

13. In the Flow Builder, click on the Manager tab on the left-hand side >> Click on New Resource >> In the Resource Type dropdown, select Variable.
14. Enter API name as ActualDeliveryDate >> Select Data type as Date >> Click on Done.
15. From the Toolbox drag and drop Assignment element.
16. Enter the label as “Assignment”.
17. Set Variable Values:
 - a) Variable : {!ActualDeliveryDate}
Operator : Equals
Value : {!\$Record.Order_Date_c}
 - b) Variable : {!ActualDeliveryDate}
Operator : Add
Value : 3

Assignment

* Label * API Name i

Assignment Assignment

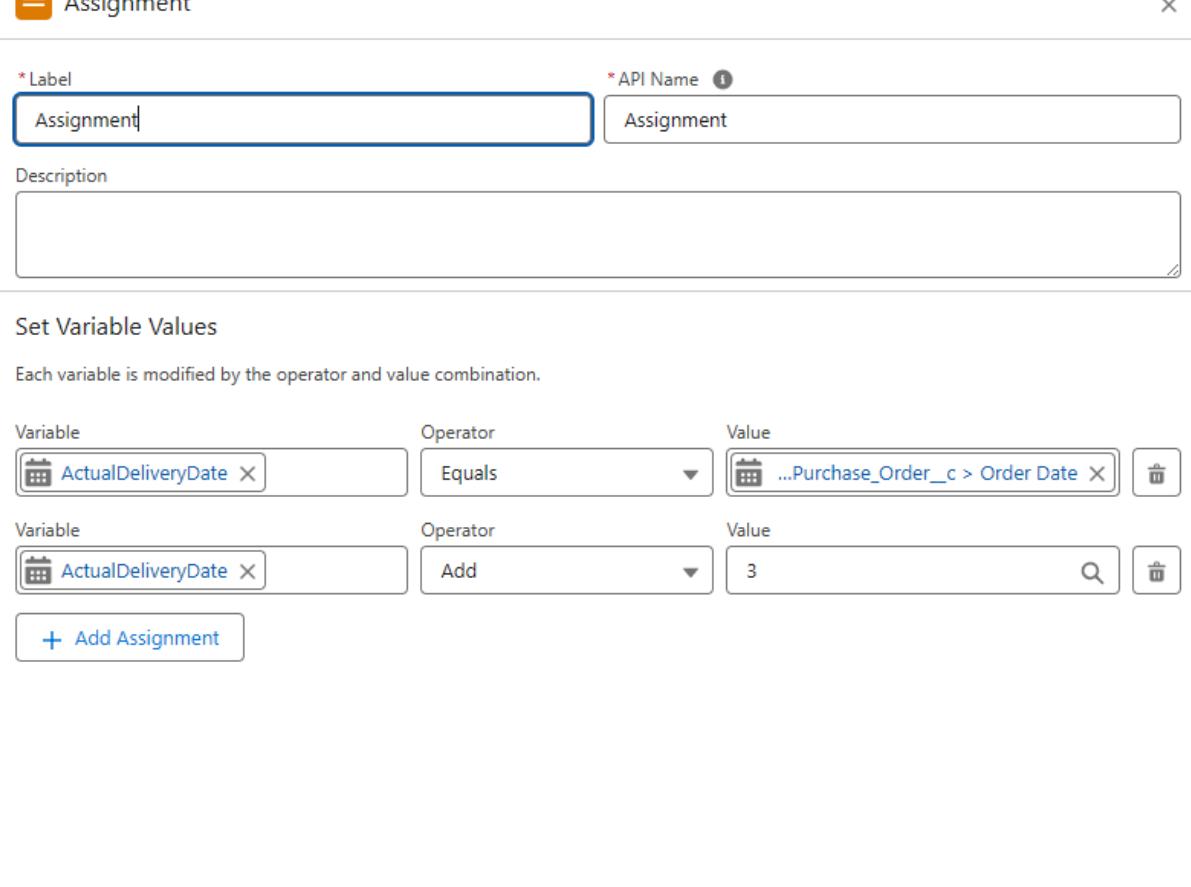
Description

Set Variable Values

Each variable is modified by the operator and value combination.

Variable	Operator	Value
ActualDeliveryDate	Equals	...Purchase_Order_c > Order Date
ActualDeliveryDate	Add	3

[+ Add Assignment](#)



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}

Update Records

* Label: Updating Purchasing Order * API Name: Updating_Purchasing_Order

Description:

* How to Find Records to Update and Set Their Values

- Use the purchase order record that triggered the flow
- Update records related to the purchase order record that triggered the flow
- Use the IDs and all field values from a record or record collection
- Specify conditions to identify records, and set fields individually

i Because this flow runs *before* a record is saved, you can only update the record that triggered the flow to run. To update other records, configure the trigger to run the flow *after* the record is saved.

Set Filter Conditions

Condition Requirements to Update Record

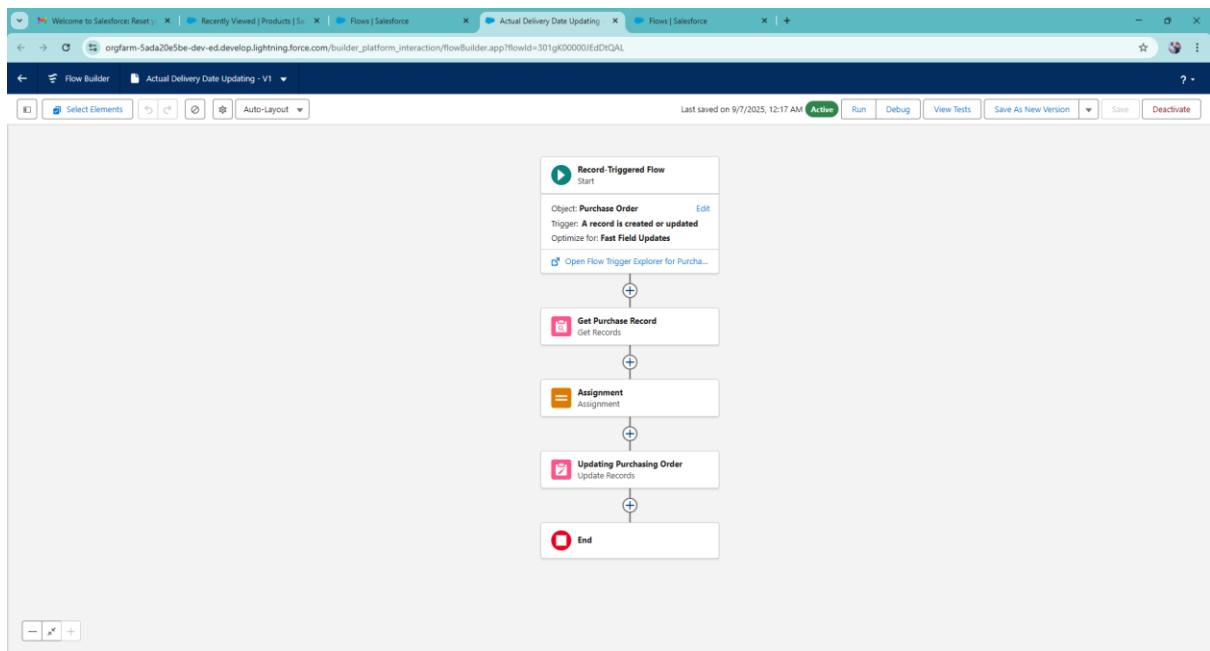
None—Always Update Record ▾

Set Field Values for the Purchase Order Record

Field	Value
Actual Delivery Date X	ActualDeliveryDate X

+ Add Field

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



16. Created 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.

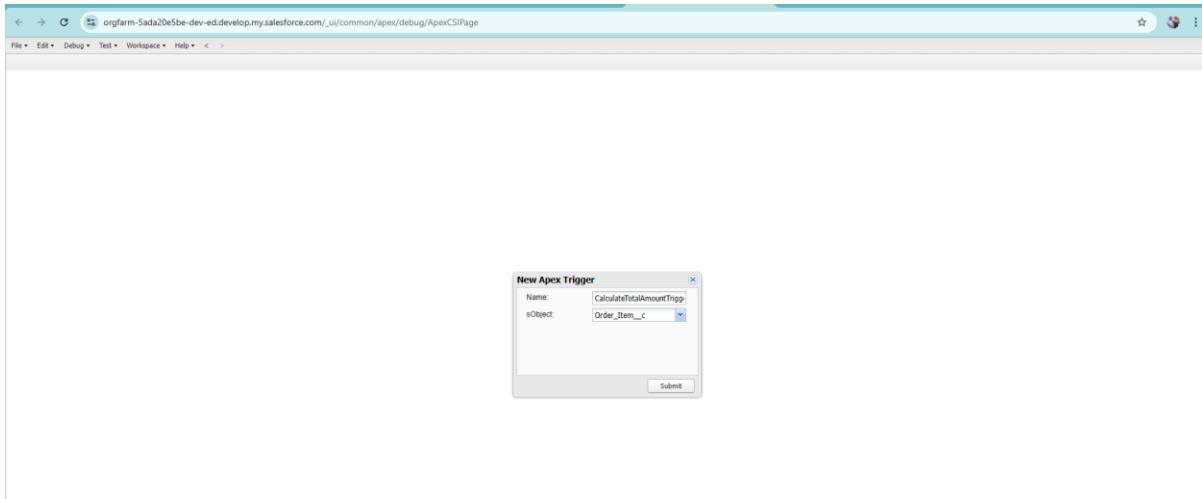
ii) Click on Developer Console: Click on the "Developer Console" option from the Setup menu. This will open the Developer Console in a new browser tab or window.

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;
}

```

```
}
```

```
}
```

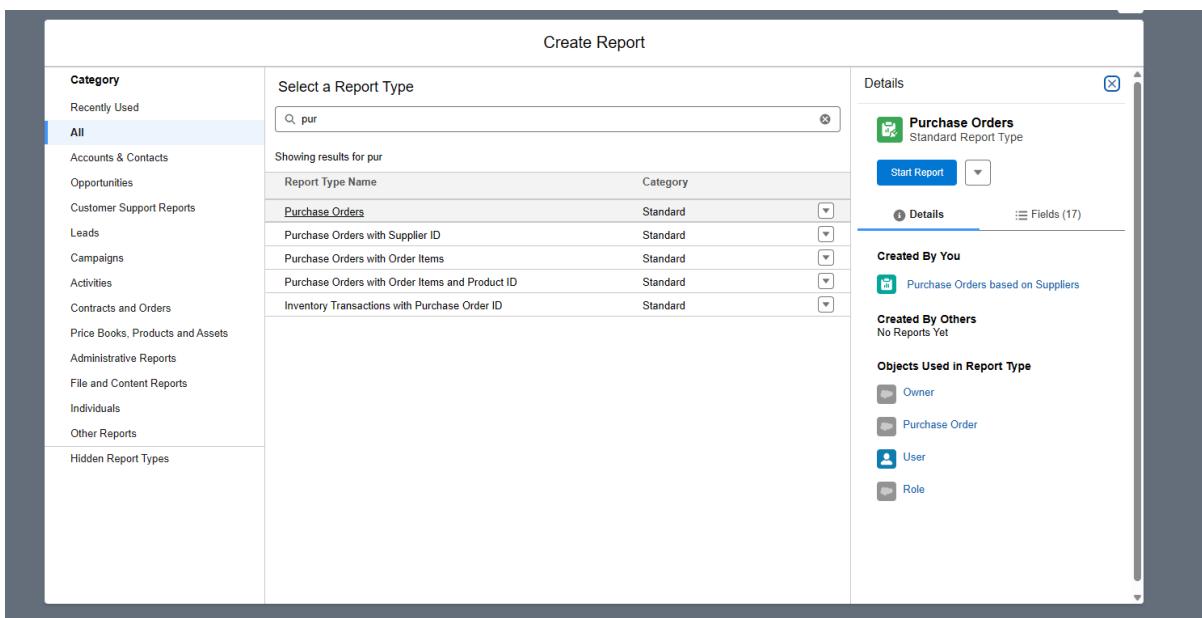
```
}
```

```
}
```

Save all.

17. Created 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.



Create Report

Category	Report Type Name	Category
Recently Used	Showing results for pur	
All	Purchase Orders	Standard
Accounts & Contacts	Purchase Orders with Supplier ID	Standard
Opportunities	Purchase Orders with Order Items	Standard
Customer Support Reports	Purchase Orders with Order Items and Product ID	Standard
Leads	Inventory Transactions with Purchase Order ID	Standard
Campaigns		
Activities		
Contracts and Orders		
Price Books, Products and Assets		
Administrative Reports		
File and Content Reports		
Individuals		
Other Reports		
Hidden Report Types		

Details

Purchase Orders
Standard Report Type
Start Report

Created By You
Purchase Orders based on Suppliers

Created By Others
No Reports Yet

Objects Used in Report Type

- Owner
- Purchase Order
- User
- Role

6. Click on Filters and select as follows and click on Apply

Filters

Add filter...

Show Me
All purchase orders

Actual Delivery Date
All Time

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”)

Purchase Orders based on Suppliers / Purchase Orders

Supplier ID	Purchase Order: Purchase Order ID	Order Count	Total Order Cost
SUP001 (1)	P0001 (1)	1	\$1,000.00
SUP002 (1)	P0002 (1)	1	\$900.00
Total (2)		2	\$1,900.00

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.

The screenshot shows a Salesforce Lightning interface with a top navigation bar containing tabs like 'Welcome to Salesforce: Reset!', 'Recently Viewed | Products | ...', 'Flows | Salesforce', 'Purchase Orders based on Sup...', 'Developer Console', 'Flows | Salesforce', and a '+' button. Below the navigation is a header with a user profile picture, a search bar, and various icons. The main content area is titled 'Report: Purchase Orders' and 'Purchase Orders based on Suppliers'. It shows a summary table with three rows: 'Total Records' (2), 'Total Order Count' (2), and 'Total Total Order Cost' (\$1,960.00). Below this is a detailed table with columns for 'Supplier ID', 'Purchase Order', 'Purchase Order ID', 'Order Count', and 'Total Order Cost'. The data shows two entries: one for SUP001 (PO001) with a count of 1 and a cost of \$1,000.00, and another for SUP002 (PO002) with a count of 1 and a cost of \$960.00. A 'Grand Total' row at the bottom shows a value of \$1,960.00. At the bottom of the report area, there are several checkboxes for 'Row Counts', 'Detail Rows', 'Subtotals', and 'Grand Total'.

Total Records	Total Order Count	Total Total Order Cost	
2	2	\$1,960.00	
Supplier ID + Purchase Order: Purchase Order ID + Order Count + Total Order Cost			
SUP001 (1)	PO001 (1)	1	\$1,000.00
SUP002 (1)	PO002 (1)	1	\$960.00
Total (2)		2	\$1,960.00

17.1 Created 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

The screenshot shows a software interface for filtering data. At the top, there are two tabs: "Outline" and "Filters". The "Filters" tab is active. Below the tabs is a search bar with the placeholder "Add filter...". Underneath the search bar are two filter boxes. The first box contains the text "Show Me All purchase orders". The second box contains the text "Actual Delivery Date All Time".

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

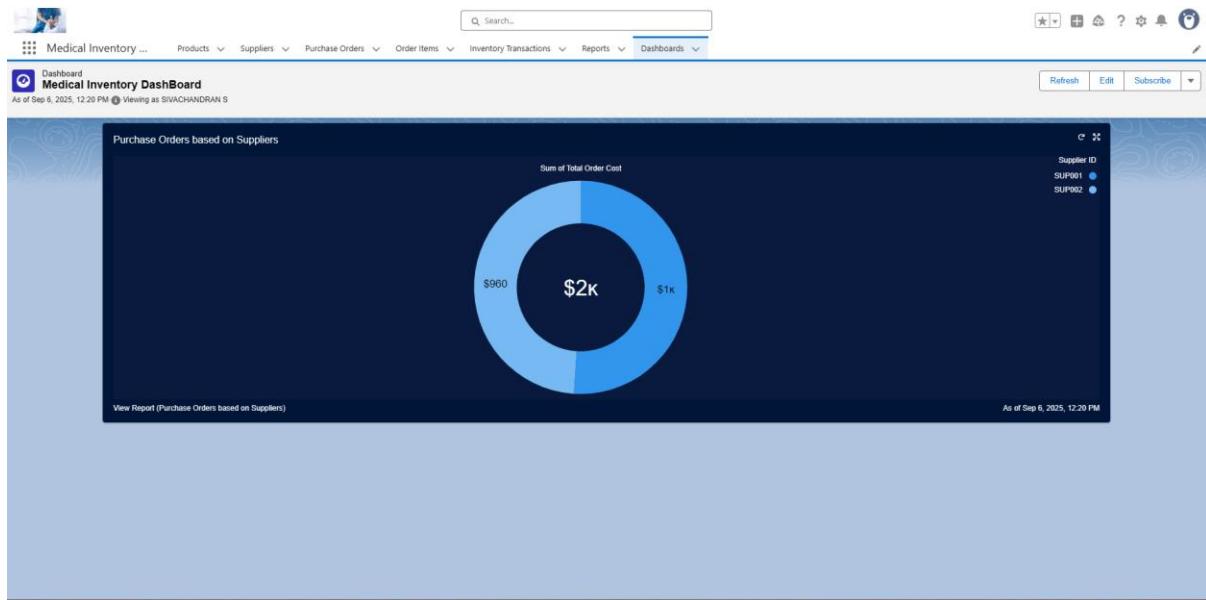
The screenshot shows a report preview titled "Purchase Orders with Order Items and Product ID". The report includes a header with fields like "Supplier ID", "Actual Delivery Date", "Purchase Order: Purchase Order ID", "Product ID: Product ID", "Order Count", "Product ID: Product Name", "Quantity Received", and "Amount". The main body of the report displays data for two purchase orders. Purchase Order 0001 (SUP001) has one item: Fever Tabled (01tgk000005TAFe) with an order count of 1 and a quantity received of 20. Purchase Order 0002 (SUP002) has one item: Cough Syrup (01tgk000005TTYR) with an order count of 1 and a quantity received of 8. Both purchase orders have a subtotal of \$1,960.00.

Supplier ID	Actual Delivery Date	Purchase Order: Purchase Order ID	Product ID: Product ID	Order Count	Product ID: Product Name	Quantity Received	Amount
SUP001 (1)	9/9/2025 (1)	PO001 (1)	01tgk000005TAFe	1	Fever Tabled	20	\$1,960.00
			Subtotal			20	\$1,960.00
			Subtotal			20	\$1,960.00
SUP002 (1)	9/9/2025 (1)	PO002 (1)	01tgk000005TTYR	1	Cough Syrup	8	\$980.00
			Subtotal			8	\$980.00
			Subtotal			8	\$980.00
			Total (2)			28	\$1,960.00

18. Created 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.

The screenshot shows the 'Dashboards' tab selected in the top navigation bar. A modal window titled 'Select Report' is open, displaying a list of recent reports. On the left, there's a sidebar with 'Reports' and 'Folders' sections. The main area shows a grid of dashboard slots and a floating menu with options: '+ Widget', '+ Filter', 'Save', and 'Done'. The 'Select Report' modal has a search bar and a list of two reports: 'Complete Purchase Details Report' and 'Purchase Orders based on Suppliers'. At the bottom right of the modal are 'Cancel' and 'Select' buttons.



19. Testing Approach

To ensure the app works as intended:

- Step 1:** Created sample suppliers, products, purchase orders, and transactions.
- Step 2:** Verified validations by entering incorrect data.
- Step 3:** Checked automation by adding inbound transactions to see if stock updated correctly.
- Step 4:** Reviewed reports and dashboards to ensure accurate totals.

The image contains two screenshots of the "Medical Inventory" application interface. Both screenshots show a "Recently Viewed" list with a "New" button, "Import" button, "Change Owner" button, and "Assign Label" button. A search bar and filter icons are also present.

Top Screenshot (Products):

Recently Viewed

4 items • Updated a few seconds ago

Product ID
P004
P003
P002
P001

Bottom Screenshot (Suppliers):

Recently Viewed

2 items • Updated a few seconds ago

Supplier ID
SUP002
SUP001

Purchase Orders

Recently Viewed

2 items • Updated a few seconds ago

- Purchase Order ID
 - 1 P002
 - 2 P001

Order Items

Recently Viewed

2 items • Updated a few seconds ago

- Order Item ID
 - 1 OI002
 - 2 OI001

Inventory Transactions

Recently Viewed

2 items • Updated a few seconds ago

- Inventory Transaction ID
 - 1 T002
 - 2 T001

Report: Purchase Orders with Order Items and Product ID

Complete Purchase Details Report

Total Records	Total Order Count	Total Quantity Received	Total Amount																																																																
2	2	90	\$6,600.00																																																																
<table border="1"> <thead> <tr> <th>Supplier ID</th> <th>Actual Delivery Date</th> <th>Purchase Order: Purchase Order ID</th> <th>Product ID: Product ID</th> <th>Order Count</th> <th>Product ID: Product Name</th> <th>Quantity Received</th> <th>Amount</th> </tr> </thead> <tbody> <tr> <td>SUP001 (1)</td> <td>9/6/2025 (1)</td> <td>P001 (1)</td> <td>01gk000005TAFe</td> <td>1</td> <td>Fever Tablet</td> <td>60</td> <td>\$3,600.00</td> </tr> <tr> <td></td> <td></td> <td></td> <td>Subtotal</td> <td></td> <td></td> <td>60</td> <td>\$3,600.00</td> </tr> <tr> <td></td> <td></td> <td></td> <td>Subtotal</td> <td></td> <td></td> <td>60</td> <td>\$3,600.00</td> </tr> <tr> <td>SUP002 (1)</td> <td>9/6/2025 (1)</td> <td>P002 (1)</td> <td>01gk000005TTYR</td> <td>1</td> <td>Cough Syrup</td> <td>30</td> <td>\$3,600.00</td> </tr> <tr> <td></td> <td></td> <td></td> <td>Subtotal</td> <td></td> <td></td> <td>30</td> <td>\$3,600.00</td> </tr> <tr> <td></td> <td></td> <td></td> <td>Subtotal</td> <td></td> <td></td> <td>30</td> <td>\$3,600.00</td> </tr> <tr> <td>Total (2)</td> <td></td> <td></td> <td></td> <td>2</td> <td></td> <td>90</td> <td>\$6,600.00</td> </tr> </tbody> </table>				Supplier ID	Actual Delivery Date	Purchase Order: Purchase Order ID	Product ID: Product ID	Order Count	Product ID: Product Name	Quantity Received	Amount	SUP001 (1)	9/6/2025 (1)	P001 (1)	01gk000005TAFe	1	Fever Tablet	60	\$3,600.00				Subtotal			60	\$3,600.00				Subtotal			60	\$3,600.00	SUP002 (1)	9/6/2025 (1)	P002 (1)	01gk000005TTYR	1	Cough Syrup	30	\$3,600.00				Subtotal			30	\$3,600.00				Subtotal			30	\$3,600.00	Total (2)				2		90	\$6,600.00
Supplier ID	Actual Delivery Date	Purchase Order: Purchase Order ID	Product ID: Product ID	Order Count	Product ID: Product Name	Quantity Received	Amount																																																												
SUP001 (1)	9/6/2025 (1)	P001 (1)	01gk000005TAFe	1	Fever Tablet	60	\$3,600.00																																																												
			Subtotal			60	\$3,600.00																																																												
			Subtotal			60	\$3,600.00																																																												
SUP002 (1)	9/6/2025 (1)	P002 (1)	01gk000005TTYR	1	Cough Syrup	30	\$3,600.00																																																												
			Subtotal			30	\$3,600.00																																																												
			Subtotal			30	\$3,600.00																																																												
Total (2)				2		90	\$6,600.00																																																												

Row Counts Detail Rows Subtotals Grand Total

Medical Inventory

Medical Inventory DashBoard

As of Sep 6, 2025, 12:20 PM Viewing as SIVACHANDRAN S

Purchase Orders based on Suppliers

Sum of Total Order Cost

\$900 \$1k \$2k

Supplier ID
SUP001
SUP002

[View Report \(Purchase Orders based on Suppliers\)](#)

As of Sep 6, 2025, 12:20 PM

20. Conclusion

The **Medical Inventory Management System** successfully demonstrates how Salesforce can be used to manage medical products, suppliers, and stock efficiently.

By using custom objects, validation rules, flows, reports, and dashboards, the application provides a complete solution for inventory management with real-time tracking and analytics.