

Medical Inventory Management

College Name:

BISHOP APPASAMY COLLEGE OF ARTS AND SCIENCE.

College Code: BRU3G.

TEAM ID: NM2025TMID20459

TEAM MEMBERS:

Team Leader Name: Adithya R

Email: adithyavar6@gmail.com

Team Member 1: Prabhu S

Email: prabhu7339201211@gmail.com

Team Member 2: Jeeva P

Email: jeevapandi2006@gmail.com

Team Member 3: Sridhar M

Email: sridharmanikandan51@gmail.com

INTRODUCTION:

The Medical Inventory Management System is a comprehensive Salesforce application designed to streamline and manage various operational aspects of the medical inventory. It can efficiently maintain supplier details, manage purchase orders, track product details and transactions, and monitor expiry dates of products, thereby improving operational efficiency, data accuracy, and reporting capabilities.

Are you new to Salesforce? Not sure exactly what it is, or how to use it? Don't know where you should start on your learning journey? If you've answered yes to any of these questions, then you're in the right place. This module is for you.

Welcome to Salesforce! Salesforce is game-changing technology, with a host of productivity-boosting features, that will help you sell smarter and faster. As you work toward your badge for this module, we'll take you through these features and answer the question, "What is Salesforce, anyway?".

What Is Salesforce?

Salesforce is your customer success platform, designed to help you sell, service, market, analyze, and connect with your customers.

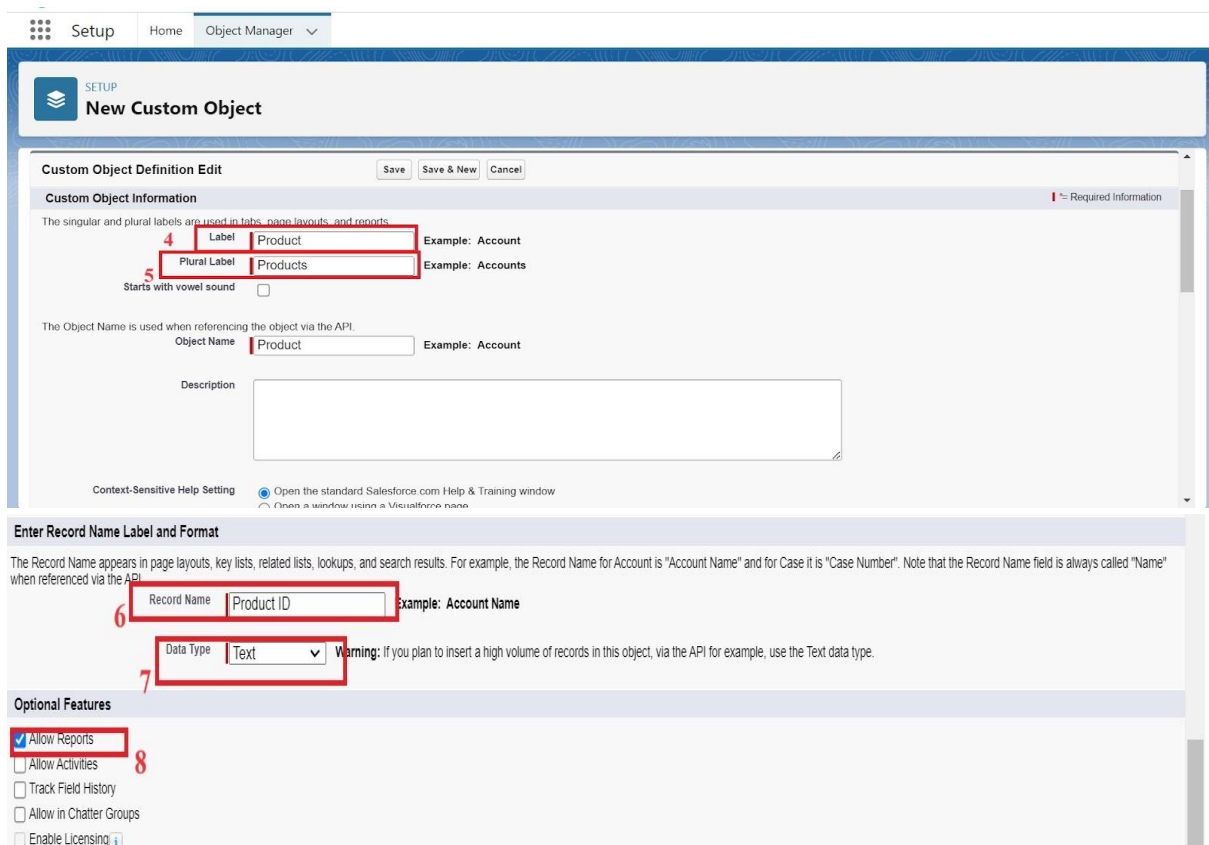
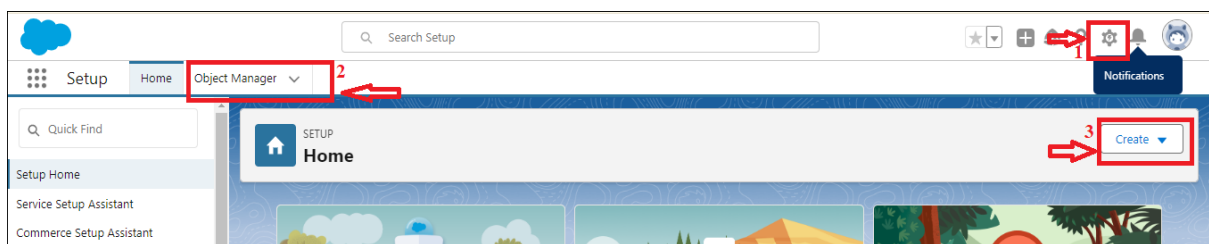
Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud. So what does that really mean? Well, before Salesforce, your contacts, emails, follow-up tasks, and prospective deals might have been organized something like this:

<https://youtu.be/r9EX3IGde5k>

Objects

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



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 ⁹

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

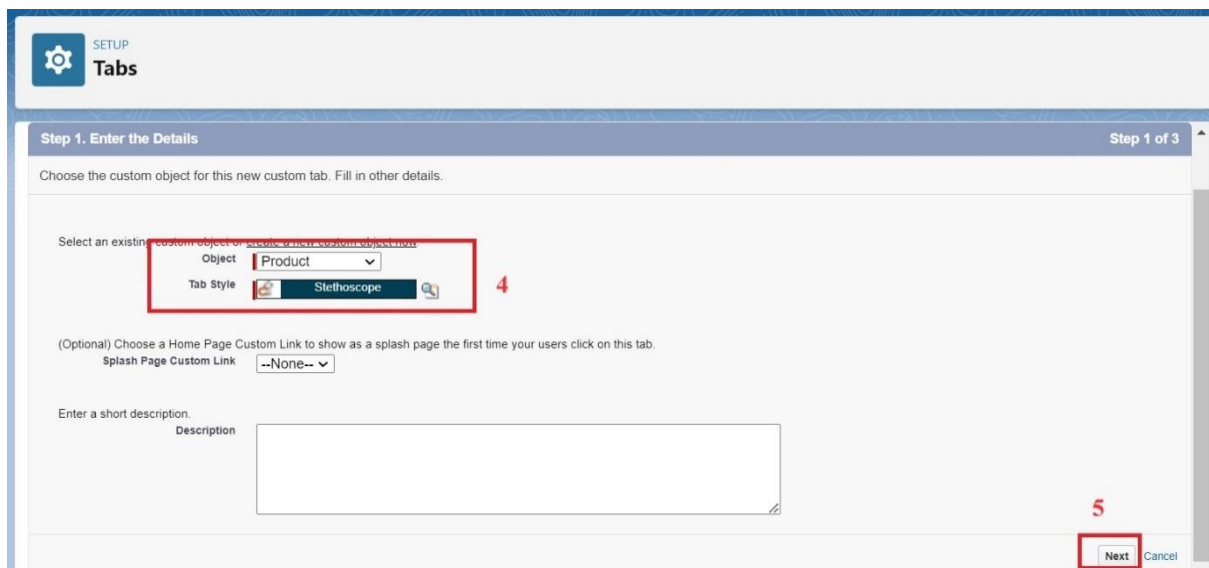
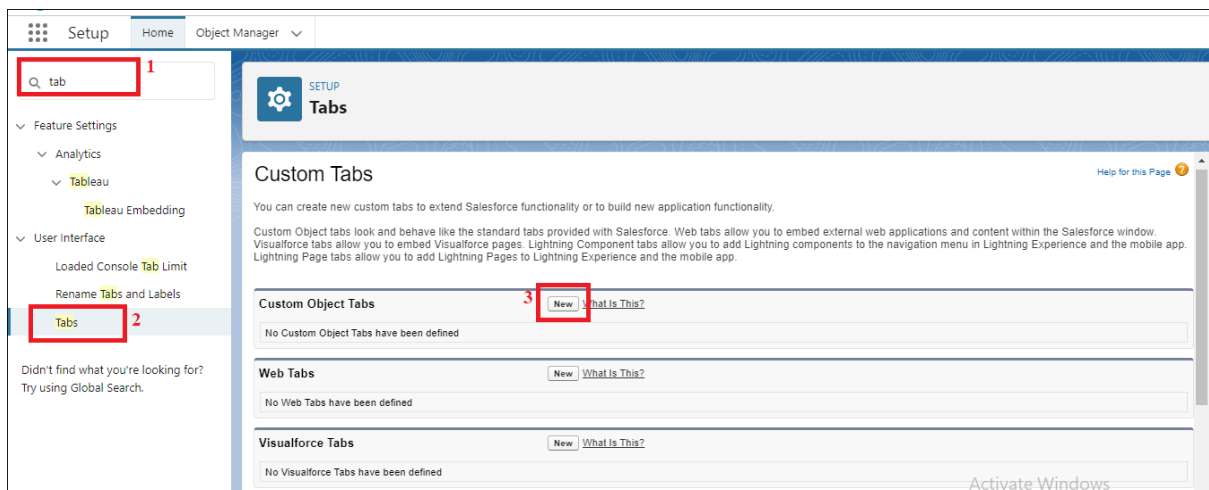
10

Activate Windows
Go to Settings to activate Windows.

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

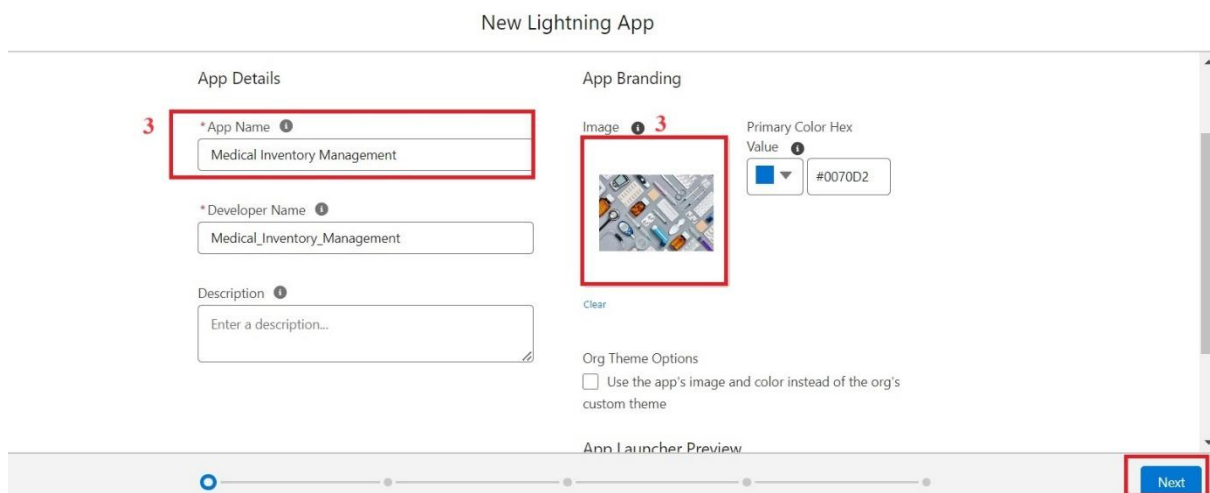
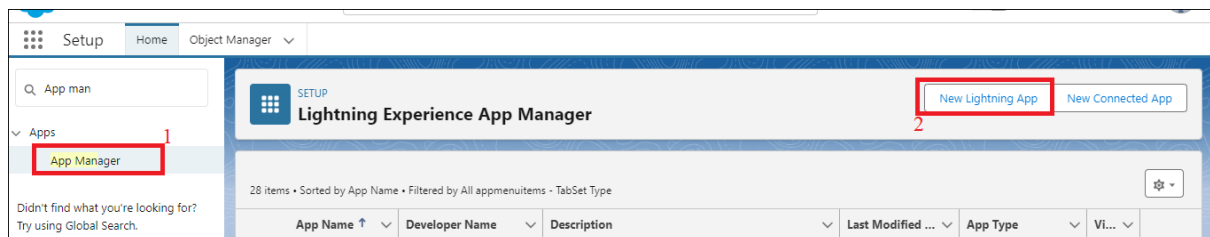
TABS:

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



LIGHTNING APP

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.



Navigation Items

Choose the items to include in the app, and arrange the order in which they appear. Users can personalize the navigation to add or move items, but users can't remove or rename the items that you add. Some navigation items are available only for phone or only for desktop. These items are dropped from the navigation bar when the app is viewed in a format that the item doesn't support.

Available Items

Create

Dash

Dashboards

Selected Items

- Products
- Purchase Orders
- Order Items
- Inventory Transactions
- Suppliers
- Reports

6

New Lightning App

User Profiles

Choose the user profiles that can access this app.

Available Profiles

system

System Administrator

Selected Profiles

Back

Save & Finish

7

8

FIELDS:

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.

The screenshot shows the Salesforce Setup interface. In the top navigation bar, the 'Setup' tab is selected. Below it, the 'Object Manager' tab is highlighted with a red box and labeled '2'. The main content area displays the 'Object Manager' page for the 'Product' object. A table lists various objects, with the 'Product' custom object highlighted by a red box and labeled '3'.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Fulfillment Order Product	FulfillmentOrderLineItem	Standard Object			
Opportunity Product	OpportunityLineItem	Standard Object			
Order Product	OrderItem	Standard Object			
Product	Product__c	Custom Object		18/06/2024	✓
Product	Product2	Standard Object			
Product Attribute	ProductAttribute	Standard Object			

The screenshot shows the Salesforce Setup interface for the 'Product' object. The 'Fields & Relationships' tab is highlighted with a red box and labeled '4'. The main content area displays the 'Fields & Relationships' page for the 'Product' object. A table lists the fields, with the 'Product ID' field highlighted by a red box.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Product ID	Name	Text(80)		✓

<input type="radio"/> Geolocation	clicking Send an Email. Note that custom email addresses cannot be used for mass emails.
<input type="radio"/> Number	Allows users to define locations. Includes latitude and longitude components, and can be used to calculate distance.
<input type="radio"/> Percent	Allows users to enter any number. Leading zeros are removed.
<input type="radio"/> Phone	Allows users to enter a percentage number, for example, "10" and automatically adds the percent sign to the number.
<input type="radio"/> Picklist	Allows users to enter any phone number. Automatically formats it as a phone number.
<input type="radio"/> Picklist (Multi-Select)	Allows users to select a value from a list you define.
<input checked="" type="radio"/> Text	Allows users to select multiple values from a list you define.
<input type="radio"/> Text Area	Allows users to enter any combination of letters and numbers.
<input type="radio"/> Text Area (Long)	Allows users to enter up to 255 characters on separate lines.
<input type="radio"/> Text Area (Rich)	Allows users to enter up to 131,072 characters on separate lines.
<input type="radio"/> Text (Encrypted) ⓘ	Allows users to enter formatted text, add images and links. Up to 131,072 characters on separate lines.
<input type="radio"/> Time	Allows users to enter any combination of letters and numbers and store them in encrypted form.
<input type="radio"/> URL	Allows users to enter a local time. For example, "2:40 PM", "14:40", "14:40:00", and "14:40:50.600" are all valid times for this field.
	Allows users to enter any valid website address. When users click on the field, the URL will open in a separate browser window.
	<div>Next Cancel</div>

Step 2. Enter the details

Step 2 of 4

PreviousNextCancel

Field LabelProduct Name ⓘ

Please enter the maximum length for a text field below.

Length255

Field NameProduct ⓘ

Description

Help Text ⓘ

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

Unique☐ Do not allow duplicate values

☒ Treat "ABC" and "abc" as duplicate values (case insensitive)

☐ Treat "ABC" and "abc" as different values (case sensitive)

External ID☐ Set this field as the unique record identifier from an external system

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

798

PAGE LAYOUTS:

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.

Save Quick Save Preview As... Cancel Undo Redo Layout Properties

Fields

Buttons
Quick Actions
Mobile & Lightning Actions
Expanded Lookups
Related Lists
Report Charts

Quick Find Field Name

Section	Last Modified By	Product ID
Blank Space	Minimum Stock Level	Product Name
Created By	Owner	Unit Price
Current Stock Level	Product Description	

Information (Header visible on edit only)

Product ID	Sample Text	Unit Price	₹123.45
Product Name	Sample Text	Current Stock Level	12,420
Product Description	Sample Text	Minimum Stock Level	21,114
		Owner	Sample Text

System Information (Header visible on edit only)

Created By Sample Text Last Modified By Sample Text

COMPACT LAYOUTS:

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.

The screenshot shows the Salesforce 'Compact Layouts' configuration page for the 'Product' object. The interface is divided into several sections:

- Compact Layouts Table:** A table with columns: LABEL, API NAME, PRIMARY, MODIFIED BY, and LAST MODIFIED. It currently shows one item, 'System Default', with API NAME 'SYSTEM' and PRIMARY status checked. A red box labeled '3' highlights the 'New' button in the top right corner.
- Enter Compact Layout Information:** This section contains two input fields: 'Label' (containing 'Product Compact Layout') and 'Name' (containing 'Product_Compact_Layout'). A red box labeled '4' highlights both fields.
- Select Compact Layout Fields:** This section has two columns: 'Available Fields' and 'Selected Fields'.
 - Available Fields:** Includes 'Created By', 'Last Modified By', 'Minimum Stock Level', 'Owner', and 'Product ID'.
 - Selected Fields:** Includes 'Product Name', 'Unit Price', and 'Current Stock Level'. A red box labeled '5' highlights this column.
 - Buttons:** Between the columns are 'Add' and 'Remove' buttons. To the right of the 'Selected Fields' column are buttons for 'Top', 'Up', 'Down', and 'Bottom'.
- Save Section:** At the bottom, there is a red box labeled '6' highlighting the 'Save' button, next to a 'Cancel' button.

At the bottom of the page, there is a note: 'Use SHIFT + click to select adjacent fields. Use CTRL + click to select an assortment of fields.'

Product Compact Layouts

Compact Layout Assignment

Save

Cancel

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

9

10

VALIDATION RULES:

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.

Purchase Order Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit [Save] [Save & New] [Cancel]

Rule Name: Expected_Delivery_Date_Validation 3

Active: ☒ 4

Description:

Error Condition Formula ⓘ = Required Information

Example: Discount_Percent__c>0.30 More Examples...
Display an error if Discount is more than 30%
If this formula expression is **true**, display the text defined in the Error Message area

Insert Field Insert Operator

(Expected_Delivery_Date__c - Order_Date__c)> 7 5

Functions: -- All Function Categories --
ABS
ACOS
ADDMONTHS
AND
ASCII
ASIN

Error Message

Example: Discount percent cannot exceed 30%

This message will appear when Error Condition formula is **true**

Error Message: The Expected Delivery Date should not exceed 7 days. 6

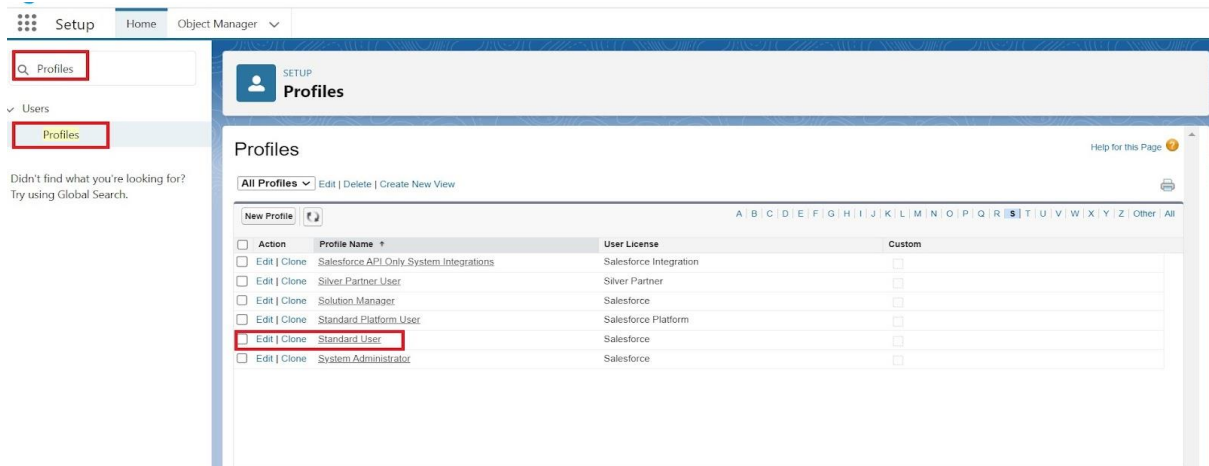
This error message can either appear at the top of the page or below a specific field on the page

Error Location: ☒ Top of Page ☐ Field ⓘ 7

[Save] [Save & New] [Cancel] 8

PROFILES:

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.



Clone Profile

Enter the name of the new profile.

You must select an existing profile to clone from.

Existing Profile	Standard User
User License	Salesforce
Profile Name	<input type="text" value="Inventory Manager"/>

2. While still on the profile page, then click Edit.
3. Select the Custom App settings as default for the Medical Inventory Management.

SETUP Profiles

Custom App Settings ! = Required Information

	Visible	Default		Visible	Default
All Tabs (standard__AllTabSet)	<input checked="" type="checkbox"/>	<input type="radio"/>	Sales (standard__LightningSales)	<input checked="" type="checkbox"/>	<input type="radio"/>
Analytics Studio (standard__Insights)	<input checked="" type="checkbox"/>	<input type="radio"/>	Sales (standard__Sales)	<input checked="" type="checkbox"/>	<input type="radio"/>
App Launcher (standard__AppLauncher)	<input checked="" type="checkbox"/>	<input type="radio"/>	Sales Console (standard__LightningSalesConsole)	<input checked="" type="checkbox"/>	<input type="radio"/>
Bolt Solutions (standard__LightningBolt)	<input checked="" type="checkbox"/>	<input type="radio"/>	Salesforce Chatter (standard__Chatter)	<input checked="" type="checkbox"/>	<input type="radio"/>
Community (standard__Community)	<input checked="" type="checkbox"/>	<input type="radio"/>	Salesforce Scheduler Setup (standard__LightningScheduler)	<input type="checkbox"/>	<input type="radio"/>
Content (standard__Content)	<input checked="" type="checkbox"/>	<input type="radio"/>	Sample Console (standard__ServiceConsole)	<input type="checkbox"/>	<input type="radio"/>
Data Manager (standard__DataManager)	<input checked="" type="checkbox"/>	<input type="radio"/>	Service (standard__Service)	<input checked="" type="checkbox"/>	<input type="radio"/>
Digital Experiences (standard__SalesforceCMS)	<input checked="" type="checkbox"/>	<input type="radio"/>	Service Console (standard__LightningService)	<input checked="" type="checkbox"/>	<input type="radio"/>
Lightning Usage App (standard__LightningInstrumentation)	<input checked="" type="checkbox"/>	<input type="radio"/>	Site.com (standard__Sites)	<input checked="" type="checkbox"/>	<input type="radio"/>
Marketing CRM Classic (standard__Marketing)	<input checked="" type="checkbox"/>	<input type="radio"/>	Subscription Management (standard__RevenueCloudConsole)	<input checked="" type="checkbox"/>	<input type="radio"/>
Medical Inventory Management (Medical_Inventory_Management)	<input type="checkbox"/>	<input checked="" type="radio"/>	WDC (standard__Work)	<input checked="" type="checkbox"/>	<input type="radio"/>
Queue Management (standard__QueueManagement)	<input checked="" type="checkbox"/>	<input type="radio"/>			

4. Scroll down to Custom Object Permissions and Give access permissions as mentioned in the below diagram.

Custom Object Permissions

	Basic Access				Data Administration	
	Read	Create	Edit	Delete	View All	Modify All
Inventory Transactions	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Order Items	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Products	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

	Basic Access				Data Administration	
	Read	Create	Edit	Delete	View All	Modify All
Purchase Orders	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Suppliers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

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.

Password Policies

User passwords expire in **Never expires**

Enforce password history **3 passwords remembered**

Minimum password length **8**

Password complexity requirement **Must include alpha and numeric characters**

Password question requirement **Cannot contain password**

Maximum invalid login attempts **10**

Lockout effective period **15 minutes**

Obscure secret answer for password resets ☐

Require a minimum 1 day password lifetime ☐

Don't immediately expire links in forgot password emails ☐

Save Save & New Cancel

ROLES:

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

Setup Home Object Manager

roles

Users

Roles

Feature Settings

Sales

Contact Roles on Contracts

Contact Roles on Opportunities

Service

Case Teams

Case Team Roles

Contact Roles on Cases

Didn't find what you're looking for? Try using Global Search.

SETUP Roles

Understanding Roles

Set up your Role Hierarchy to control how your organization reports on and accesses data.

Sample Role Hierarchy

View other sample Role Hierarchies: Territory-based Sample

Executive Staff

CEO, President, CFO, VP, Sales

Western Sales Director

Eastern Sales Director

International Sales Director

Western Sales Rep

Eastern Sales Rep

International Sales Rep

Set Up Roles

Don't show this page again

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.

SETUP Roles

Role Edit

New Role

Role Edit

Label

Purchasing Manager

Role Name

Purchasing_Manager

This role reports to

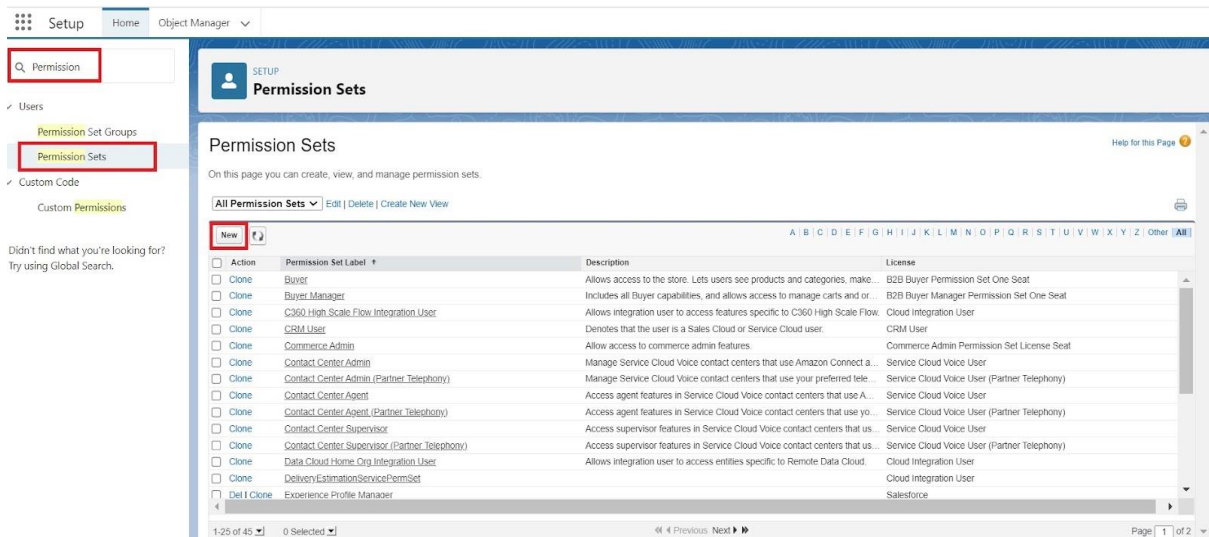
SVP, Sales & Marketing

Role Name as displayed on reports

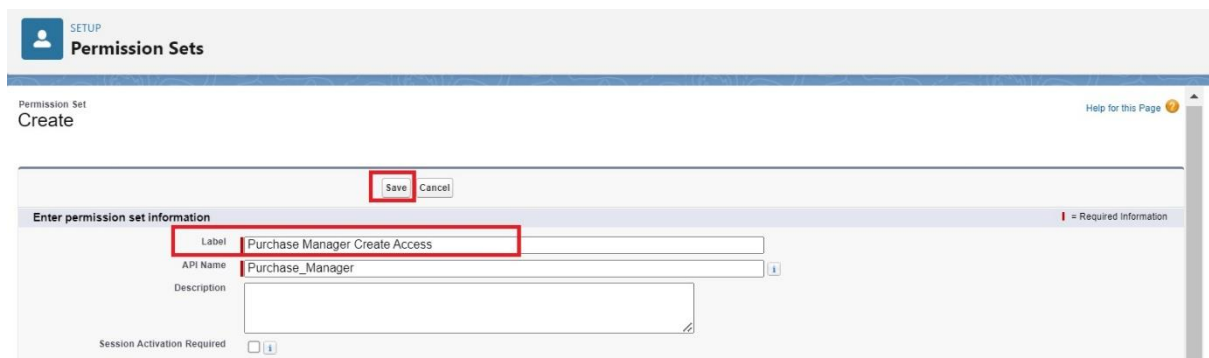
Save Save & New Cancel

PERMISSION SETS:

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.

SETUP
Permission Sets

Permission Set
Purchase Manager Create Access

Find Settings... | Clone Delete Edit Properties Manage Assignments View Summary

Permission Set Overview > Object Settings Order Items

Order Items Save Cancel

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	<input type="checkbox"/>
Modify All	<input type="checkbox"/>

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.

... > PERMISSION SET 'PURCHASE MANAGER CREATE ACCESS' > MANAGE ASSIGNMENT EXPIRATION
Purchase Manager Create Access

Select Users to Assign

Active Users 1 item selected Search this list...

	Full Name ↑	Alias	Username	Role	Active	Profile
<input type="checkbox"/>	Annapurna Gurram	AGurr	medicalinventory@sb.com		<input checked="" type="checkbox"/>	System Administrator
<input type="checkbox"/>	Chatter Expert	Chatter	chatty.00ddl0000058bqluaa.yrgohck7wjyo@chatter.salesforce.com		<input checked="" type="checkbox"/>	Chatter Free User
<input type="checkbox"/>	Integration User	integ	integration@00ddl0000058bqluaa.com		<input checked="" type="checkbox"/>	Analytics Cloud Integration User
<input checked="" type="checkbox"/>	John PurchaseM	jpurc	john@purchasem.com	Purchasing Manager	<input checked="" type="checkbox"/>	Purchase Manager
<input type="checkbox"/>	Security User	sec	insightssecurity@00ddl0000058bqluaa.com		<input checked="" type="checkbox"/>	Analytics Cloud Security User

Cancel Next

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



Purchase Manager Create Access

Select an Expiration Option For Assigned Users

☒ No expiration date ⓘ

☐ Specify the expiration date

1 Day

1 Week

30 Days

60 Days

Custom Date

ⓘ Time Zone

Select a time zone...

Selected Users

Full Name	Role	Profile	Active	User License	Expires On
John PurchaseM	Purchasing Manager	Purchase Manager	✓	Salesforce	Never Expires

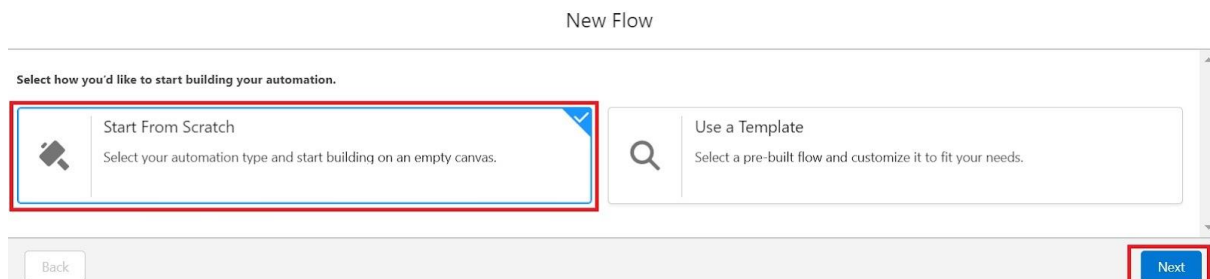
Cancel

Back

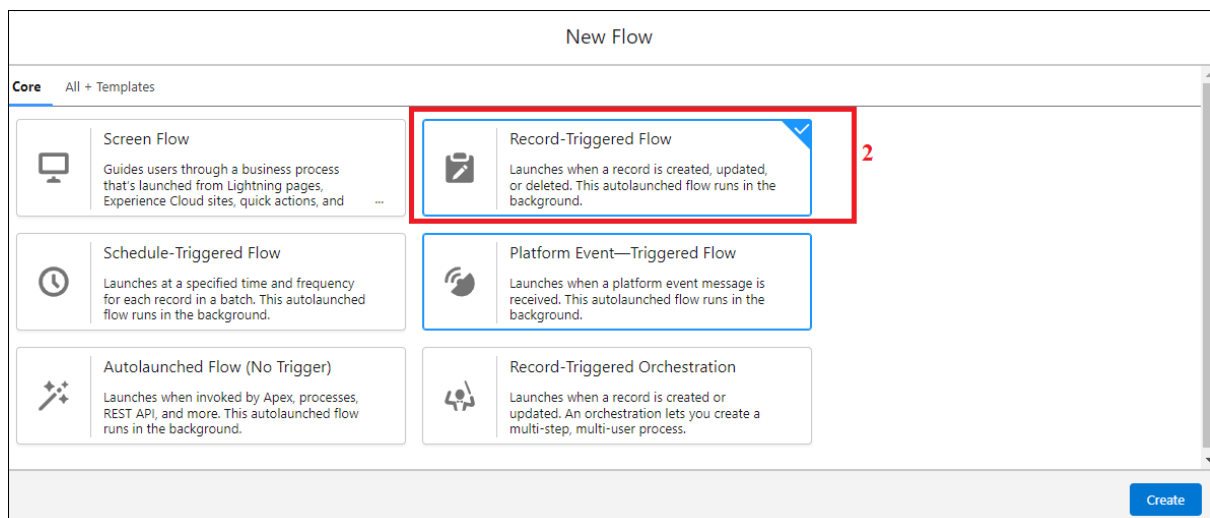
Assign

FLOWS:

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

Configure Start

Select Object

Select the object whose records trigger the flow when they're created, updated, or deleted.

* Object

Purchase Order 3

Configure Trigger

* Trigger the Flow When:

☐ A record is created
☐ A record is updated
☒ A record is created or updated 4
☐ A record is deleted

5. Set Entry Conditions : None

6. Select Fast Field Updates and click on Done

Set Entry Conditions

Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Condition Requirements

None 5

* Optimize the Flow for:

Fast Field Updates

Update fields on the record that triggers the flow to run. This high-performance flow runs *before* the record is saved to the database.

Actions and Related Records 6

Update any record and perform actions, like send an email. This more flexible flow runs *after* the record is saved to the database.

☐ Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed

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: Get Purchase Record **8**

*API Name: Get_Purchase_Record

Description:

Get Records of This Object

*Object: Purchase Order **9**

Filter Purchase Order Records **10**

Condition Requirements: All Conditions Are Met (AND)

Field: Id Operator: Equals Value: \$Record > 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

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: ID

Field: Order_Date__c

+ 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

The screenshot shows the 'Assignment' configuration window. At the top, there's a title bar with an orange icon and the word 'Assignment'. Below it, there are two input fields: '* Label' with the value 'Assignment' and '* API Name' with the value 'Assignment_1'. A 'Description' text area is below these. The main section is titled 'Set Variable Values' with a note: 'Each variable is modified by the operator and value combination.' Below this, there are two rows of configuration, each with a 'Variable', 'Operator', and 'Value' field. The first row has 'ActualDeliveryDate' as the variable, 'Equals' as the operator, and '\$Record > Order Date' as the value. The second row has 'ActualDeliveryDate' as the variable, 'Add' as the operator, and '3' as the value. Each row has a trash icon to its right. At the bottom, there is a '+ Add Assignment' button.

Variable	Operator	Value
ActualDeliveryDate	Equals	\$Record > Order Date
ActualDeliveryDate	Add	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}

 Update Records ×

*** 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



 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

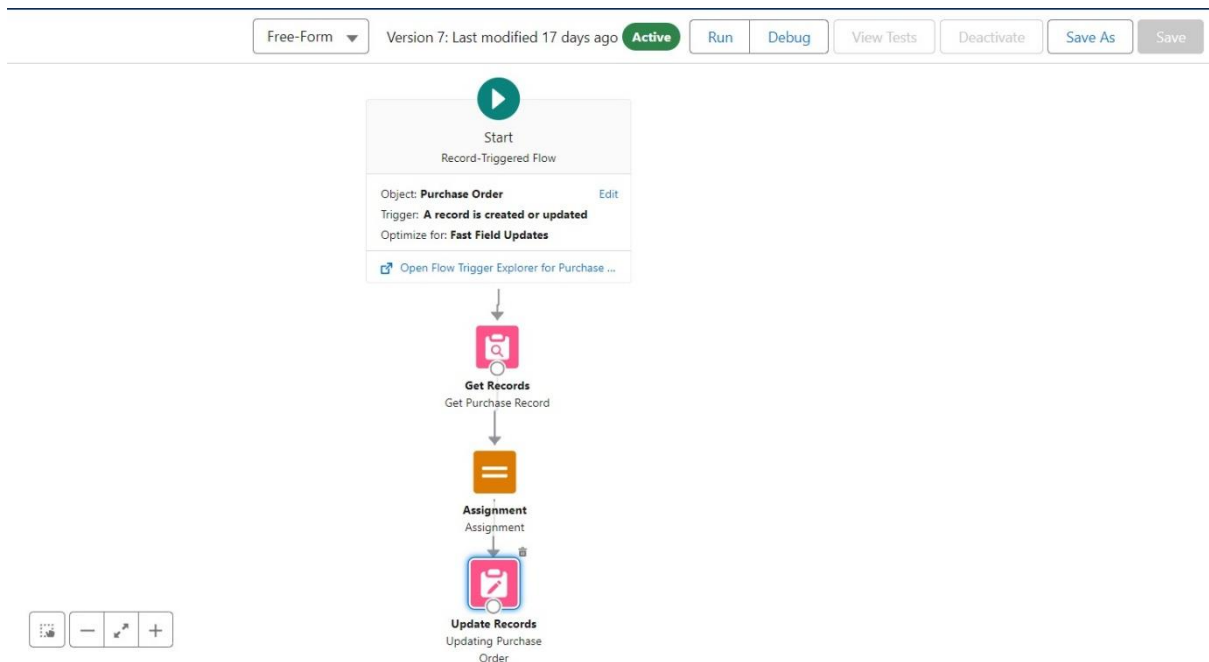
Field	Value
Actual_Delivery_Date__c	←  ActualDeliveryDate × 

[+ Add Field](#)

24. Click Done

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

26. Activate the flow.



TRIGGERS:

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.

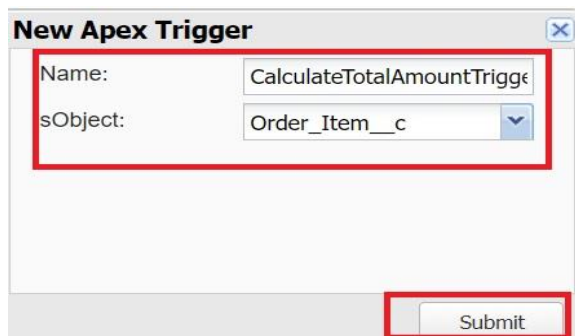
ii) Click on Developer Console: Click on the "Developer Console" option from the Setup menu. This will open the Developer Console in a new browser tab or window.

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.



The screenshot shows the 'New Apex Trigger' dialog box. The 'Name' field contains 'CalculateTotalAmountTrigg€' and the 'sObject' field contains 'Order_Item__c'. A red box highlights these two fields. The 'Submit' button is located at the bottom right and is also highlighted with a red box.

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);
}

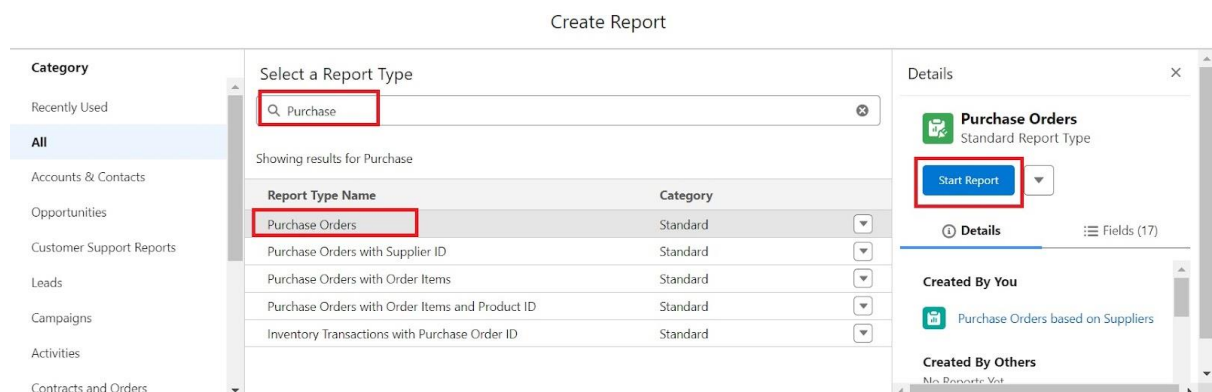
// Update Purchase Orders if there are any changes
if (!purchaseToUpdate.isEmpty()) {
    update purchaseToUpdate;
}
}
}
}

```

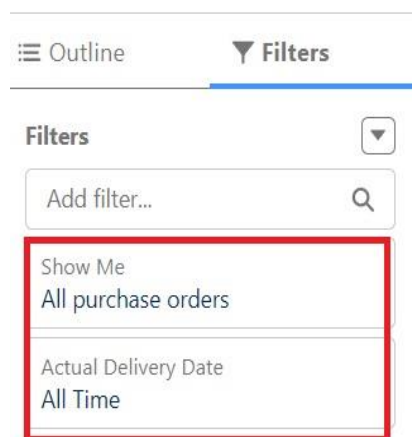
Save it.

REPORTS:

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”)

The screenshot shows the 'Medical Inventory ...' app interface. The 'Reports' tab is selected, and the report 'Purchase Orders based on Suppliers' is chosen. The 'Save & Run' button is highlighted with a red box. The 'Update Preview Automatically' checkbox is also highlighted with a red box. The 'Fields' panel on the left shows the 'GROUP ROWS' section with 'Supplier ID' and 'Purchase Order: Purchase Order ID' selected. The 'GROUP COLUMNS' section has 'Add group...' and 'Add column...' buttons. The 'Columns' section shows '# Order Count' and '# Total Order Cost' selected. The main table displays the following data:

Supplier ID	Purchase Order: Purchase Order ID	Order Count	Total Order Cost
Supplier-001 (4)	Purchase-0001 (1)	3	₹2,075.00
	Purchase-0002 (1)	2	₹3,250.00
	Purchase-0003 (1)	3	₹7,000.00
	Purchase-0004 (1)	4	₹9,500.00
Supplier-002 (1)	Purchase-0005 (1)	2	₹4,500.00
Total (5)		14	₹26,325.00

At the bottom, the 'Row Counts' checkbox is checked, and the 'Grand Total' checkbox is also checked.

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.

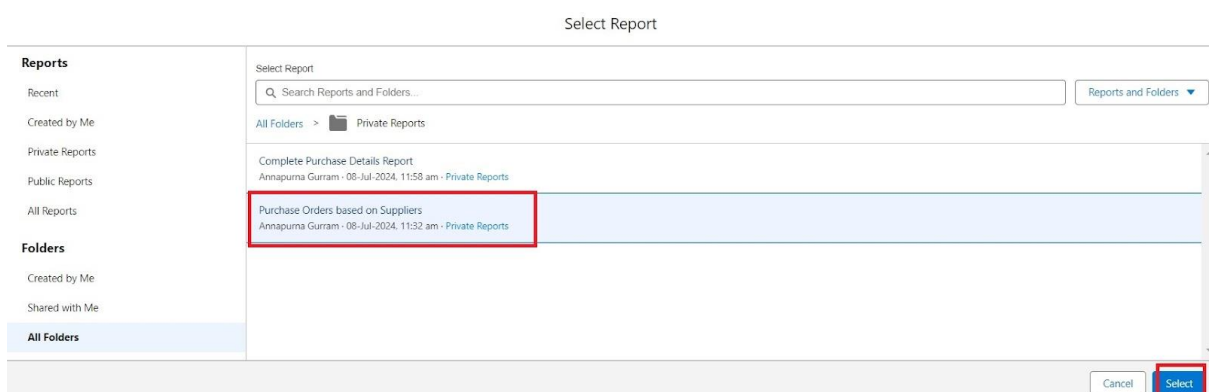
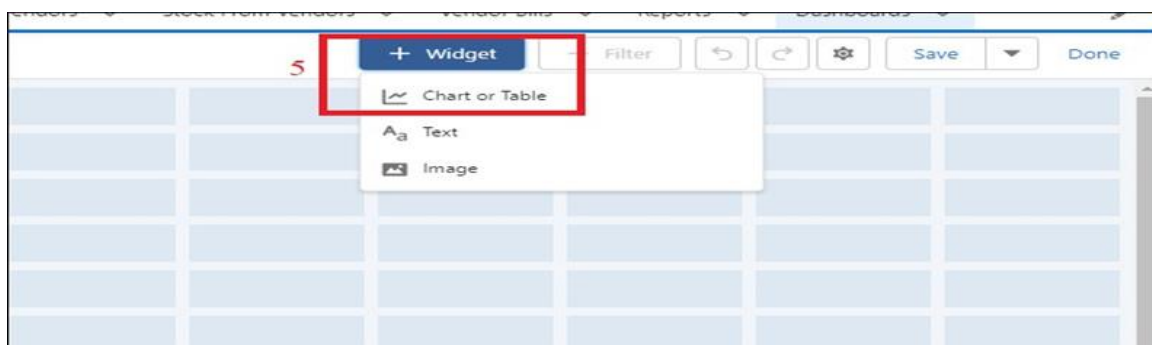
The screenshot shows the 'Medical Inventory ...' app interface. The 'Reports' tab is selected, and the report 'Purchase Orders based on Suppliers' is viewed. The 'Enable Field Editing' button is highlighted with a red box. The 'Add Chart' button is also highlighted with a red box. The 'Edit' button is highlighted with a red box. The main table displays the following data:

Supplier ID	Purchase Order: Purchase Order ID	Order Count	Total Order Cost
Supplier-001 (4)	Purchase-0001 (1)	3	₹2,075.00
	Purchase-0002 (1)	2	₹3,250.00
	Purchase-0003 (1)	3	₹7,000.00
	Purchase-0004 (1)	4	₹9,500.00
Supplier-002 (1)	Purchase-0005 (1)	2	₹4,500.00
Total (5)		14	₹26,325.00

At the bottom, the 'Row Counts' checkbox is checked, and the 'Grand Total' checkbox is also checked.

DASHBOARDS:

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.








Add Widget





Report


Purchase Orders based on Supplie ✕

☐ Use chart settings from report i

Display As







Value

Sum of Total Order Cost

Sliced By

Supplier ID

Display Units

Preview

Purchase Orders based on Suppliers

Sum of Total Order Cost

₹4.5k

₹26k

₹22k

Supplier ID

Supplier-001

Supplier-002

View Report (Purchase Orders based on Suppliers)

Cancel

Add

Add Widget

Title

Purchase Orders based on Suppliers

Subtitle


Footer

Legend Position


Right

Widget Theme

☐ Light (Dashboard default)



☒ Dark



Preview

Purchase Orders based on Suppliers

Sum of Total Order Cost

₹4.5k

₹26k

₹22k

Supplier ID

Supplier-001

Supplier-002

View Report (Purchase Orders based on Suppliers)

Cancel

Add