

# MEDICAL INVENTORY MANAGEMENT

---

## Project Overview

The Medical Inventory Management System is designed to streamline the process of managing medicines, equipment, and other medical supplies in hospitals, clinics, and pharmacies. The system helps in tracking stock levels, monitoring expiry dates, generating purchase orders, and reducing wastage due to overstocking or understocking. It ensures efficient inventory control, cost savings, and timely availability of critical medical items.

## Problem Statement

Healthcare facilities often struggle with manual inventory management systems, leading to challenges such as:

- Overstocking or shortage of medicines.
- Difficulty in tracking expiry dates and batch numbers.
- Lack of real-time updates on available stock.
- Human errors in manual record-keeping.
- Inefficient procurement and supply chain management.

These issues can directly impact patient care, increase operational costs, and result in financial losses.

## Objectives

1. To develop a system that automates inventory tracking of medicines and medical equipment.
2. To provide real-time stock status, including alerts for low stock and near-expiry items.

3. To generate reports on inventory usage, purchase history, and supplier performance.
4. To reduce human errors and improve accuracy in inventory records.
5. To ensure cost-effectiveness and better decision-making in procurement.

## Student Outcomes

By completing this project, students will be able to:

- Apply database design and management skills in real-world applications.
- Develop problem-solving and analytical thinking skills.
- Gain experience in system design, implementation, and testing.
- Understand the role of automation in healthcare management.
- Work collaboratively and communicate effectively through project documentation.

## Scope of the Project

- In-Scope Features:
  - User authentication (Admin, Pharmacist, Staff).
  - Add, update, and delete medical items.
  - Stock monitoring with real-time updates.
  - Expiry date alerts and notifications.
  - Purchase order generation and supplier details management.
  - Reporting and analytics (daily, monthly, yearly).

- Out-of-Scope Features (Future Enhancements):
  - Integration with hospital billing systems.
  - Mobile app for staff accessibility.
  - Barcode/RFID-based inventory tracking.
  - AI-driven demand forecasting.

# System Requirements

## *Hardware Requirements*

- Processor: Intel i3 or above
- RAM: 4 GB minimum
- Hard Disk: 500 GB
- Monitor: 15" LED/LCD
- Input Devices: Keyboard, Mouse
- Network: Stable internet connection (for cloud-based system)

## *Software Requirements*

- Operating System: Windows 10 / Linux / MacOS
- Database: MySQL / PostgreSQL / Oracle
- Backend: Java / Python / PHP / Node.js
- Frontend: HTML, CSS, JavaScript, React/Angular (optional)
- Server: Apache Tomcat / XAMPP / Node.js
- IDE/Tools: VS Code, Eclipse, or IntelliJ IDEA

# Milestone 1-Salesforce Account

Introduction:

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.

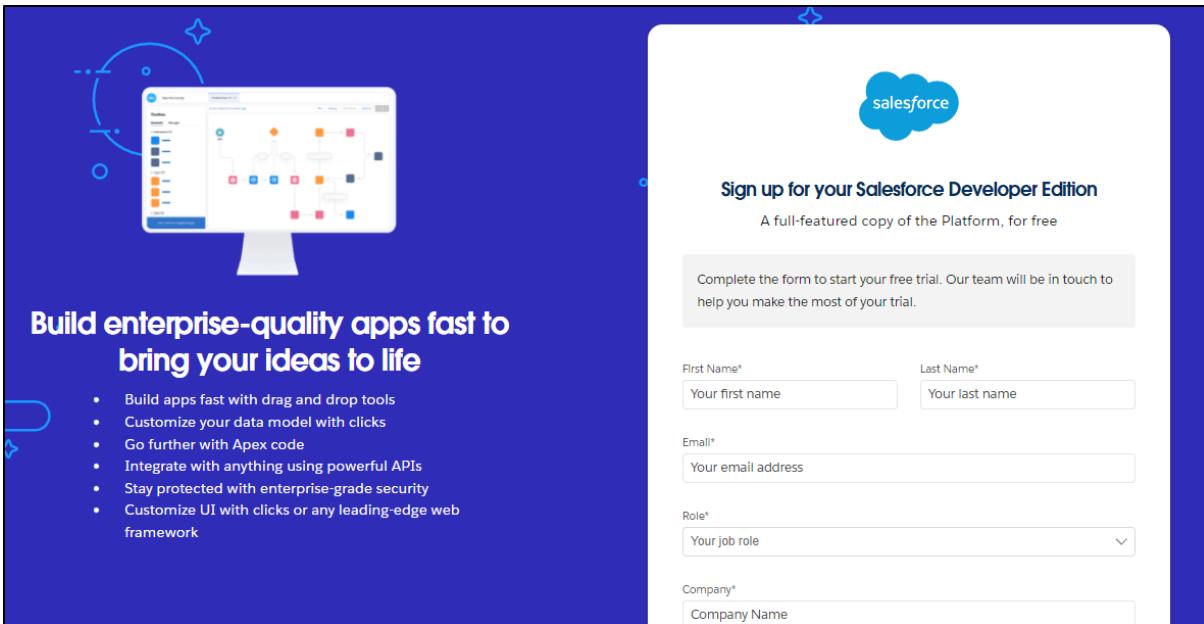
Salesforce has everything you need to run your business from anywhere. Using standard products and features, you can manage relationships with prospects and customers, collaborate and engage with employees and partners, and store your data securely in the cloud. So what does that really mean? Well, before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

<https://youtu.be/r9EX3lGde5k>

## 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 :



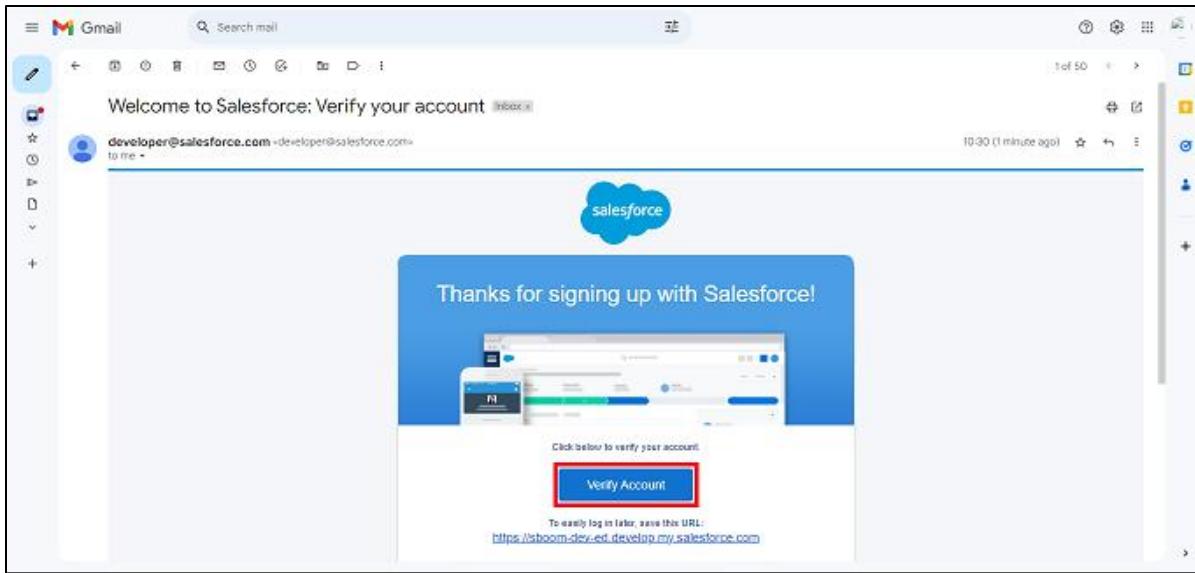
1. First name & Last name
2. Email
3. Role : Developer
4. Company : College Name
5. County : India
6. Postal Code : pin code
7. Username : should be a combination of your name and company

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

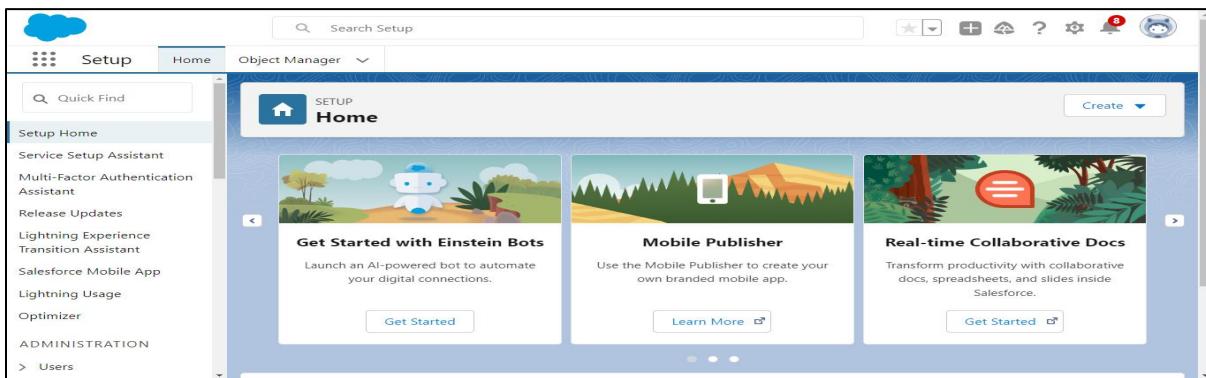
1. Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.



2. Click on Verify Account
3. Give a password and answer a security question and click on change password.

A screenshot of the "Change Your Password" page in the Salesforce Setup. The page title is "Change Your Password". It asks for a new password that includes at least 8 characters, 1 letter, and 1 number. The "New Password" field contains "....." and is labeled "Good". The "Confirm New Password" field also contains "....." and is labeled "Match". Below these are fields for "Security Question" (set to "In what city were you born?") and "Answer" (containing "asdfghjkl"). A red box highlights the "Change Password" button at the bottom.

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



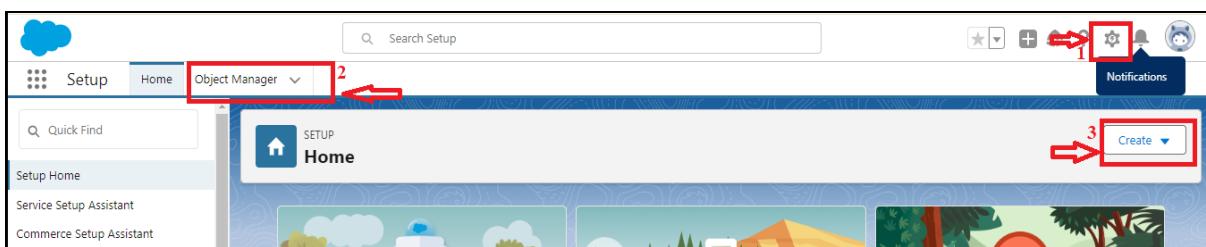
# Milestone 2- Objects

In Salesforce, objects are database tables that allow you to store data specific to your organization.

## Activity 1: Creating a Product Object

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



The screenshot shows the 'Custom Object Definition Edit' page. At the top, there are three buttons: 'Save', 'Save & New', and 'Cancel'. Below that, the 'Custom Object Information' section has a note: 'The singular and plural labels are used in tabs, page layouts, and reports.' There is a checkbox for 'Starts with vowel sound'. The 'Label' field contains 'Product' (labeled '4') and the 'Plural Label' field contains 'Products' (labeled '5'). Both fields have 'Example: Account' listed next to them. Below these fields, there is a note: 'The Object Name is used when referencing the object via the API.' The 'Object Name' field contains 'Product' and the 'Example' field contains 'Account'. There is also a 'Description' field with a large text input area. At the bottom, there are two 'Context-Sensitive Help Setting' options: 'Open the standard Salesforce.com Help & Training window' (radio button selected) and 'Open a window using a Visualforce page'.

**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.

6 Record Name  Example: Account Name  
 7 Data Type  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

**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

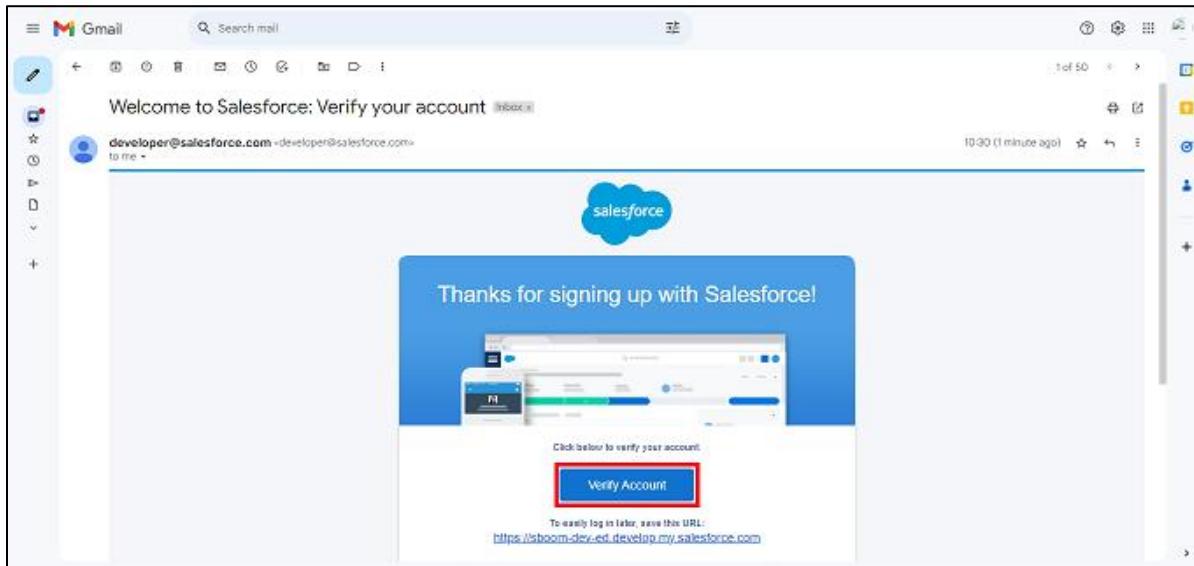
10

Activate Windows  
[Go to Settings to activate Windows.](#)

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

## 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.

The screenshot shows the 'Change Your Password' page in the Salesforce setup interface. At the top, it says 'Change Your Password'. Below that, it asks to enter a new password for 'lead@sb.oom' and specifies requirements: '8 characters', '1 letter', and '1 number'. A red box highlights the password input fields: '\* New Password' and '\* Confirm New Password'. Below these are 'Security Question' and 'Answer' fields. The 'Change Password' button at the bottom is also highlighted with a red box.

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

The screenshot shows the Salesforce Setup Home page. The left sidebar includes links like 'Setup Home', 'Service Setup Assistant', 'Multi-Factor Authentication Assistant', 'Release Updates', 'Lightning Experience Transition Assistant', 'Salesforce Mobile App', 'Lightning Usage', and 'Optimizer'. The main area displays three cards: 'Get Started with Einstein Bots', 'Mobile Publisher', and 'Real-time Collaborative Docs'. Each card has a 'Get Started' or 'Learn More' button. The top navigation bar includes 'Search Setup', 'Home', 'Object Manager', and various icons.

## Milestone 3- Tabs

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.

## 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

The top screenshot shows the 'Tabs' section of the Setup menu. A red box highlights the 'Tabs' link in the sidebar under 'Customize' (step 1). Another red box highlights the 'New' button in the 'Custom Object Tabs' list (step 3). The bottom screenshot shows the 'Step 1. Enter the Details' page for creating a new custom tab. A red box highlights the 'Object' dropdown set to 'Product' (step 4). A red box highlights the 'Next Step' button at the bottom right (step 5).

## Activity 2: Creating Remaining Tabs

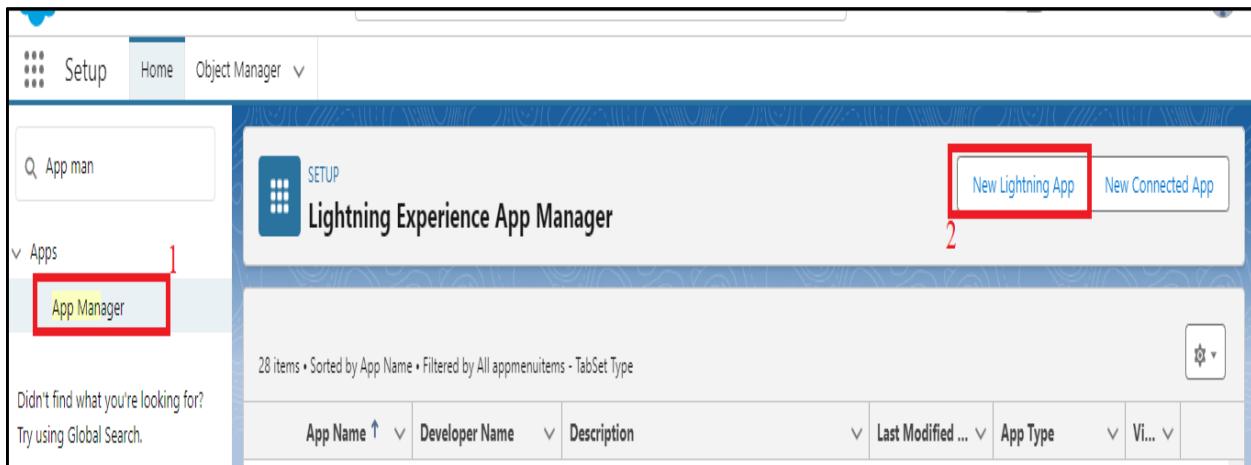
1. Now create the Tabs for the remaining Objects, they are “Purchase Order, Order Item, Inventory Transaction, Supplier”.
2. Follow the same steps as mentioned in Activity -1 .

## Milestone 4- The Lightning App

A Lightning App in Salesforce is a collection of items that work together to serve a particular function for the end-users. These items can include standard and custom objects, tabs, utilities, and other productivity tools. Lightning Apps are designed to provide a more intuitive and efficient user experience compared to traditional Salesforce apps.

## 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.



### New Lightning App

#### App Details

**3** \*App Name Medical Inventory Management

\*Developer Name Medical\_Inventory\_Management

Description Enter a description...

#### App Branding

Image

Primary Color Hex  
Value #0070D2

Org Theme Options  
 Use the app's image and color instead of the org's custom theme

Next

### 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

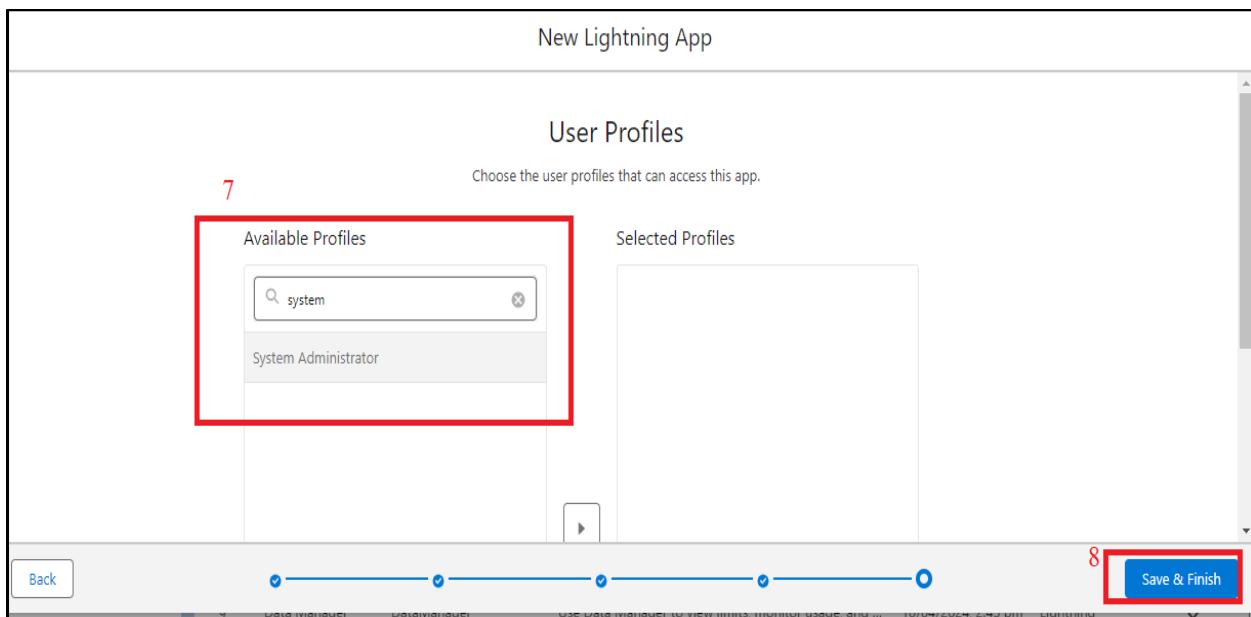
Dashboard

Selected Items

	Products
	Purchase Orders
	Order Items
	Inventory Transactions
	Suppliers
	Reports

Object	Field Name	Data Type
<b>Product</b>	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
<b>Purchase Order</b>	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)
<b>Order Item</b>	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)
<b>Inventory Transaction</b>	Transaction ID(Standard)	Text
	Purchase Order ID	Lookup(Purchase Order)
	Transaction Date	Date
	Transaction Type	Picklist
	Total Order Cost	Formula(Currency)
<b>Supplier</b>	Supplier ID(Standard)	Text
	Supplier Name	Text
	Contact Person	Text
	Phone Number	Phone
	Email	Email
	Address	TextArea



## Milestone 5- FieldsActivity 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.

**Fields & Relationships**

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)	✓	

Allows users to enter any combination of letters and numbers.

Allows users to enter up to 255 characters on separate lines.

Allows users to enter up to 131,072 characters on separate lines.

Allows users to enter formatted text, add images and links. Up to 131,072 characters on separate lines.

Allows users to enter any combination of letters and numbers and store them in encrypted form.

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.

**Next** Cancel

Step 2. Enter the details

Step 2 of 4

Previous Next Cancel

Field Label  7

Please enter the maximum length for a text field below.

Length  7

Field Name  7

Description

Help Text

Required  Always require a value in this field in order to save a record 8

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 8

## 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.

6

<input type="radio"/> Geolocation	Allows users to define locations. Includes latitude and longitude components, and can be used to calculate distance.
<input type="radio"/> Number	Allows users to enter any number. Leading zeros are removed.
<input type="radio"/> Percent	Allows users to enter a percentage number, for example, '10' and automatically adds the percent sign to the number.
<input type="radio"/> Phone	Allows users to enter any phone number. Automatically formats it as a phone number.
<input type="radio"/> Picklist	Allows users to select a value from a list you define.
<input type="radio"/> Picklist (Multi-Select)	Allows users to select multiple values from a list you define.
<input type="radio"/> Text	Allows users to enter any combination of letters and numbers.
<input checked="" type="radio"/> Text Area	Allows users to enter up to 255 characters on separate lines.
<input type="radio"/> Text Area (Long)	Allows users to enter up to 131,072 characters on separate lines.
<input type="radio"/> Text Area (Rich)	Allows users to enter formatted text, add images and links. Up to 131,072 characters on separate lines.
<input type="radio"/> Text (Encrypted) <small>i</small>	Allows users to enter any combination of letters and numbers and store them in encrypted form.
<input type="radio"/> Time	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.
<input type="radio"/> URL	Allows users to enter any valid website address. When users click on the field, the URL will open in a separate browser window.

Next Cancel

Step 2. Enter the details

Step 2 of 4

Previous Next Cancel

Field Label	<input type="text" value="Product Description"/> 7	8
Field Name	<input type="text" value="Product_Description"/> 7	
Description		
Help Text		
Required	<input type="checkbox"/> Always require a value in this field in order to save a record	
Auto add to custom report type	<input checked="" type="checkbox"/> Add this field to existing custom report types that contain this entity	
Default Value	Show Formula Editor	
<small>Use formula syntax: Enclose text and picklist value API names in double quotes : ("the_Text"), include numbers without quotes : (25), show percentages as decimals: (0.10), and express date calculations in the standard format: (Today() + 7). To reference a field from a Custom Metadata type record use: \$CustomMetadataType__mdt.RecordAPIName.Field__c</small>		

## 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.

Step 2. Enter the details

Step 2 of 4

Previous Next Cancel

Field Label	<input type="text" value="Current Stock Level"/> 5	7
<small>Please enter the length of the number and the number of decimal places. For example, a number with a length of 8 and 2 decimal places can accept values up to "12345678.90".</small>		
Length	<input type="text" value="18"/> 6	Number of digits to the left of the decimal point
Decimal Places	<input type="text" value="0"/>	Number of digits to the right of the decimal point
Field Name	<input type="text" value="Current_Stock_Level"/> 6	
Description		
Help Text		
Required	<input type="checkbox"/> Always require a value in this field in order to save a record	
Unique	<input type="checkbox"/> Do not allow duplicate values	
External ID	<input type="checkbox"/> Set this field as the unique record identifier from an external system	

# 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.

Step 2. Enter the details

Field Label: Unit Price 5

Please enter the length of the number and the number of decimal places. For example, a number with a length of 8 and 2 decimal places can accept values up to "12345678.90".

Length: 16 6 Number of digits to the left of the decimal point  
Field Name: Unit\_Price 7 Number of digits to the right of the decimal point

Description:

Help Text:

Required  Always require a value in this field in order to save a record 7  
Auto add to custom report type  Add this field to existing custom report types that contain this entity 8

8. Click on Next, Next and Save.

# Activity 5 : Creating Lookup Relationship in Purchase Order Object

A Lookup relationship is a type of relationship in Salesforce that connects two objects together based on a field known as the Lookup field. It establishes a relationship between a child object and a parent object, allowing the child object to reference the parent 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.

Specify the type of information that the custom field will contain.

**Data Type**

Select one of the data types below.

None Selected  
 Auto Number  
 Formula  
 Roll-Up Summary i  
 **Lookup Relationship** 4  
 Master-Detail Relationship  
 External Lookup Relationship

Creates a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other 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 ownership and sharing of a detail record are determined by the master record.
- When a user deletes the master record, all detail records are deleted.
- 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 popup 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.

**Next** Cancel

Purchase Order New Relationship Help for this Page ?

**Step 2. Choose the related object** Step 2

Select the other object to which this object is related.

**Related To**  5

**Previous** **Next** Cancel 6

**Previous** **Next** Cancel

Field Label  7

Field Name  9

Description

Help Text

Child Relationship Name  8

Required  Always require a value in this field in order to save a record 8

What to do if the lookup record is deleted?  
 Clear the value of this field. You can't choose this option if you make this field required.  
 Don't allow deletion of the lookup record that's part of a lookup relationship.

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

## 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.

Step 2. Enter the details

Step 2 of 4

Previous Next Cancel

Field Label  5

Field Name  6

Description

Help Text

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

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

Default Value

Use formula syntax: Enclose text and picklist value API names in double quotes: ("the\_text"), include numbers without quotes : (25), show percentages as decimals: (0.10), and express date calculations in the standard format: (Today() + 7). To reference a field from a Custom Metadata type record use: \$CustomMetadata.Type\_\_mdt RecordAPIName.Field\_\_c

## 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.

Data Type

Select one of the data types below.

None Selected

Auto Number A system-generated sequence number that uses a display format you define. The number is automatically incremented for each new record.

Formula A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.

Roll-Up Summary 4 A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.

Purchase Order  
New Custom Field

Step 2. Enter the details Step 2 of 5

Field Label  5

Field Name

Description

Help Text

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

Previous Next Cancel

Purchase Order  
New Custom Field

Step 3. Define the summary calculation Step 3 of 5

Select Object to Summarize

Master Object Purchase Order  
Summarized Object Order Items 6

Required Information

Select Roll-Up Type

COUNT 7

SUM  
 MIN  
 MAX

Field to Aggregate —None--

Filter Criteria

All records should be included in the calculation  
 Only records meeting certain criteria should be included in the calculation

## 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.

**Step 2. Choose output type**

Field Label  5 Field Name

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

**Formula Return Type**

None Selected Select one of the data types below.

Checkbox Calculate a boolean value  
Example: `TODAY() > CloseDate`

Currency 6 Calculate a dollar or other currency amount and automatically format the field as a currency amount.  
Example: `Gross Margin = Amount - Cost_c`

Date Calculate a date, for example, by adding or subtracting days to other dates.  
Example: `Reminder Date = CloseDate - 7`

Date/Time Calculate a date/time, for example, by adding a number of hours or days to another date/time.  
Example: `Next = NOW() + 1`

Number Calculate a numeric value.  
Example: `Fahrenheit = 1.8 * Celsius_c + 32`

Percent Calculate a percent and automatically add the percent sign to the number.

**Order Item** New Custom Field

**Step 3. Enter formula**

Enter your formula and click Check Syntax to check for errors. Click the Advanced Formula subtab to use additional fields, operators, and functions.

Example: `Gross Margin = Amount - Cost_c` [More Examples...](#)

Insert Field  Functions

`Unit Price (Currency) =` 7

ABS  
ACOS  
ADDMONTHS  
AND  
ASCII  
ASIN

**Quick Tips** 8

- Getting Started
- Operators & Functions

## 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.

The screenshot shows a formula editor interface. At the top, there are tabs for "Simple Formula" and "Advanced Formula", with "Simple Formula" selected. Below the tabs is a toolbar with "Insert Field" and "Insert Operator". The main area contains the formula: "Amount (Currency) = Quantity\_Received\_\_c \* Unit\_Price\_\_c". A red box highlights the entire formula entry field, and a red number "7" is placed to its right. To the right of the formula is a "Functions" panel titled "All Function Categories" with a scroll bar. The panel lists several functions: ABS, ACOS, ADDMONTHS, AND, ASCII, ASIN, with "Insert Selected Function" at the bottom.

## 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.

The screenshot shows the "Step 2. Enter the details" screen for creating a picklist field. The title bar says "Step 2 of 4" and has "Previous", "Next", and "Cancel" buttons. The "Field Label" input field is set to "Transaction Type" and is highlighted with a red box, with a red number "5" to its right. Below it is a "Values" section with two radio button options: "Use global picklist value set" (unchecked) and "Enter values, with each value separated by a new line" (checked). A red box highlights the "Enter values" option, and a red number "6" is placed to its right. The "Values" input field contains the text "Receipt  
Issue  
Adjustment". At the bottom of the screen, there are three checkboxes: "Display values alphabetically, not in the order entered" (unchecked), "Use first value as default value" (unchecked), and "Restrict picklist to the values defined in the value set" (checked). The "Field Name" input field is set to "Transaction\_Type" and the "Description" input field is empty. The status bar at the bottom right says "OneDrive - Personal Online".

# 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.

The screenshot shows the 'Step 2. Enter the details' dialog box for creating a new field. The 'Field Label' is set to 'Phone Number' (highlighted with a red box, labeled 5). The 'Field Name' is 'Phone\_Number'. The 'Required' checkbox is checked (highlighted with a red box, labeled 6). The 'Always require a value in this field in order to save a record' checkbox is also checked. The 'Auto add to custom report type' checkbox is checked. The 'Next' button is highlighted with a red box (labeled 7). A note at the bottom explains formula syntax: 'Use formula syntax: Enclose text and picklist value API names in double quotes - ("the\_Text") , include numbers without quotes - (25), show percentages as decimals (.010), and express date calculations in the standard format: (Today() + 7). To reference a field from a Custom Metadata type record use: \$CustomMetadataType\_\_mdt RecordAPIName Field\_\_c'.

# 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.

Step 2. Enter the details Step 2 of 4

Field Label  5

Field Name  6

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  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

Default Value

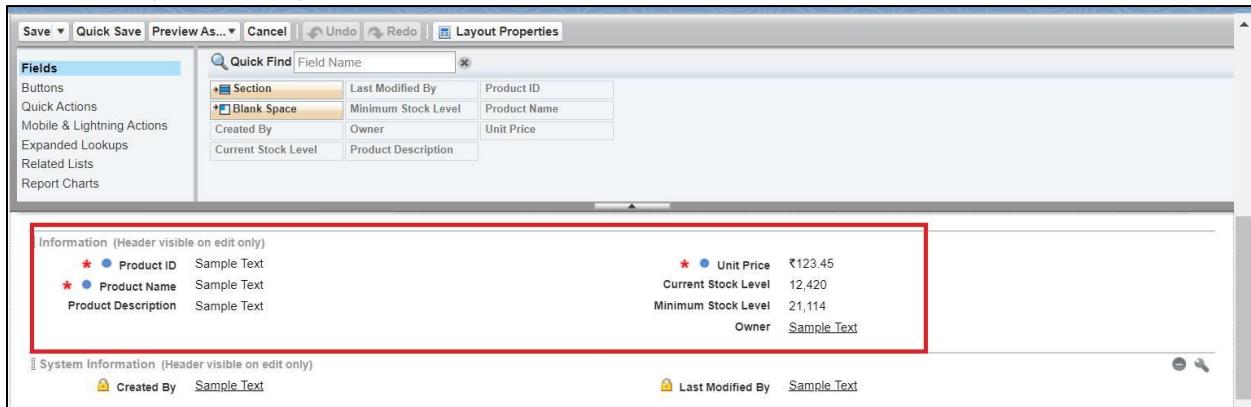
Use formula syntax. Enclose text and picklist value API names in double quotes : ("the\_text"), include numbers without quotes

# Milestone 6 -Editing of Page Layouts

Page layouts in Salesforce are used to customize the organization, structure, and content of pages for viewing and editing records. They determine which fields, related lists, and custom links are visible to users, as well as the order and grouping of those elements.

## 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

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	Field	Description
Section	Expected Delivery...	Owner
Blank Space	Last Modified By	Purchase Order ID
Actual Delivery Date	Order Count	Supplier ID
Created By	Order Date	Total Order Cost

Information (Header visible on edit only)

- \* Purchase Order ID Sample Text
- \* Supplier ID Sample Text
- \* Order Date 07/07/2024
- Expected Delivery Date 07/07/2024
- Actual Delivery Date 07/07/2024
- Order Count 36,243
- Total Order Cost ₹123.45
- Owner Sample Text

System Information (Header visible on edit only)

Created By Sample Text Last Modified By Sample Text

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

SETUP > OBJECT MANAGER  
Order Item

Details

Fields & Relationships

**Page Layouts**

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

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

Quick Find Field Name

Section	Field	Description
Section	Last Modified By	Quantity Ordered
Blank Space	Order Item ID	Quantity Received
Amount	Product ID	Unit Price
Created By	Purchase Order ID	

Information (Header visible on edit only)

- \* Order Item ID Sample Text
- \* Purchase Order ID Sample Text
- Amount ₹123.45

Product details

Field	Value	Value
Product ID	Sample Text	Quantity Ordered 23,712
Unit Price	₹123.45	Quantity Received 33,407

System Information (Header visible on edit only)

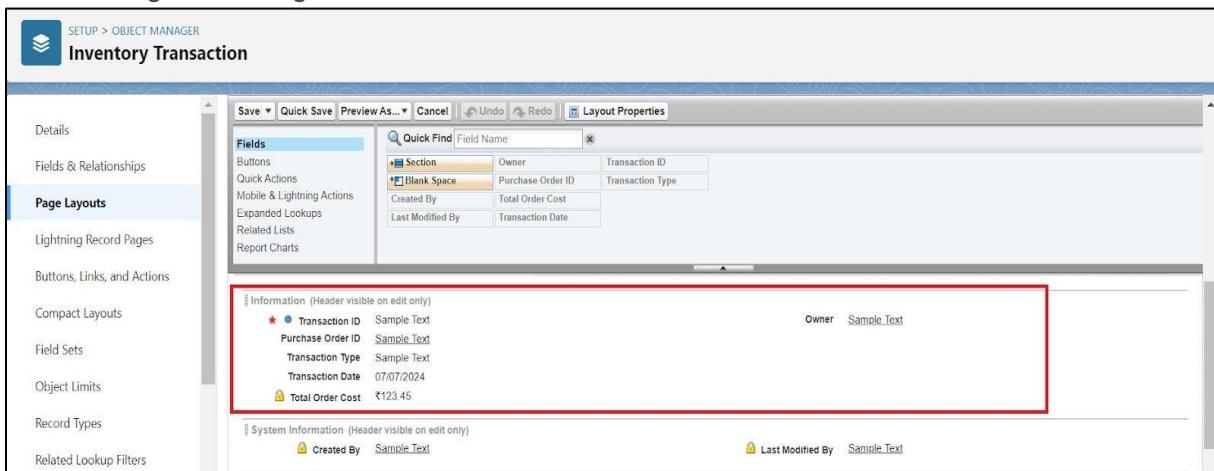
Created By Sample Text Last Modified By Sample Text

Custom Links (Header visible on edit only)

4. Click Save.

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

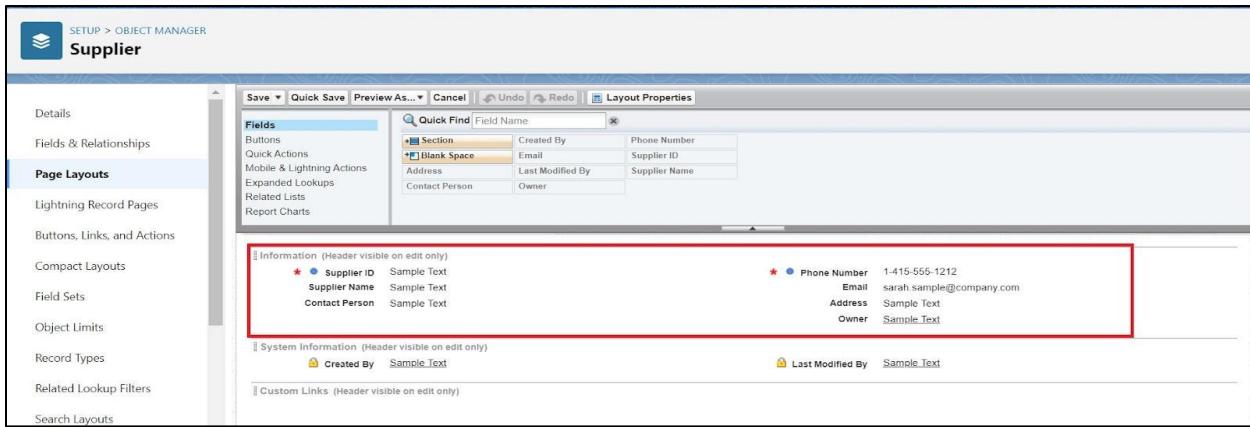
1. Go to setup >> click on Object Manager >> type object name(Inventory Transaction) in quick find box >> click on the Inventory Transaction object >> Page Layouts.
2. Click on the Inventory Transaction Layout
3. Drag and Arrange the field as shown below



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



- Click Save.

## Milestone 7 - Compact Layouts

Compact layouts display a record's key fields at a glance, providing important information quickly without needing to open the record.

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

- Go to setup >> click on Object Manager >> type object name(Product) in quick find box >> click on the Product object
- Click on Compact Layouts in the sidebar .
- Click on New.
- Enter the Label as "Product Compact Layout".
- Select the Compact Layout Fields : Select Product name, Unit Price, Current Stock Level.
- Click Save.
- Click Compact Layout Assignment.
- Click Edit Assignment.
- Choose "Product Compact Layout" from the dropdown.
- Click Save.

SETUP > OBJECT MANAGER

## Product

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
**Compact Layouts** 2
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts

**Compact Layouts**  
1 Items, Sorted by Label

LABEL	API NAME	PRIMARY	MODIFIED BY	LAST MODIFIED
System Default	SYSTEM	✓		

**Enter Compact Layout Information** ! = Required Information

Label	<input type="text" value="Product Compact Layout"/> <span style="border: 2px solid red; padding: 2px;">3</span>
Name	<input type="text" value="Product_Compact_Layout"/> <span style="border: 1px solid red; padding: 2px;">4</span>

**Select Compact Layout Fields**

Available Fields

- Created By
- 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.

6

**Product Compact Layouts**

### Compact Layout Assignment

**Primary Compact Layout**

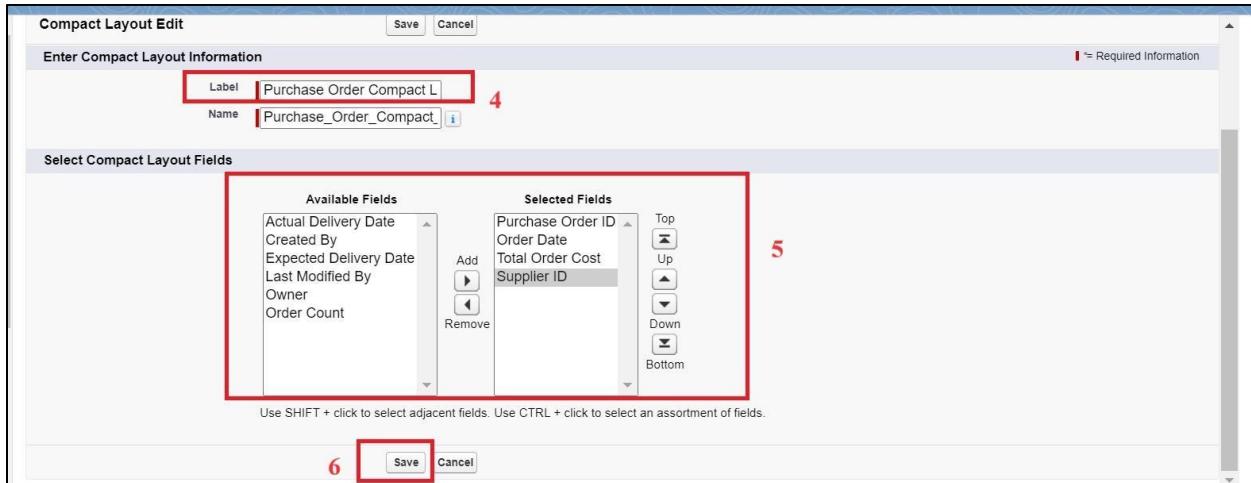
Select the compact layout to use when this object's records appear as list items in the mobile app.

**Primary Compact Layout:**  9

10

## 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.



Purchase Order Compact Layouts

## Compact Layout Assignment

10 **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:** Purchase Order Compact Layout ✓ 9

**Save** **Cancel**

## Milestone 8 - Validation Rules

Validation rules in Salesforce are used to ensure data integrity by preventing users from saving invalid data in records. They consist of a formula or expression that evaluates the data in one or more fields and return a value of true or false. When the rule's criteria are met (i.e., the expression evaluates to true), an error message is displayed, and the user is prevented from saving the record until the issue is resolved.

### 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 :

$(\text{Expected\_Delivery\_Date\_c} - \text{Order\_Date\_c}) > 7$

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**

Rule Name: Expected\_Delivery\_Date\_Validation 3

Active:  4

Description:

**Error Condition Formula**

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

F = Required Information

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.

**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

8 Save Save & New Cancel

## Milestone 9 - Profiles:

Profiles in Salesforce are fundamental to the platform's security model, defining what users can do within the organization. Profiles control a user's permissions to objects, fields, tabs, apps, and other settings. Each user in Salesforce must be assigned a profile, and the profile assigned to a user determines what they can see and do in the system.

## 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.

The screenshot shows the Salesforce Setup interface under the Profiles section. A search bar at the top left contains the text 'Profiles'. Below it, a sidebar lists 'Users' and 'Profiles', with 'Profiles' being the active tab and highlighted with a red box. A message below the sidebar says, 'Didn't find what you're looking for? Try using Global Search.' The main content area is titled 'Profiles' and shows a table of profiles. One row, 'Edit | Clone Standard User', is highlighted with a red box. The table includes columns for Action, Profile Name, User License, and Custom.

## 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   

**Save**

**Cancel**

2. While still on the profile page, then click Edit.

3. Select the Custom App settings as default for the Medical Inventory Management.

The screenshot shows the Salesforce Setup interface under the Custom App Settings section. A table lists various apps with columns for Visible and Default. The 'Medical Inventory Management (Medical\_Inventory\_Management)' app is highlighted with a red box and has its 'Default' checkbox checked.

	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 <small>i</small>	Modify All <small>i</small>
Inventory Transactions	<input checked="" type="checkbox"/>					
Order Items	<input checked="" type="checkbox"/>					
Products	<input checked="" type="checkbox"/>					
Purchase Orders	<input checked="" type="checkbox"/>					
Suppliers	<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	<input type="text" value="Never expires"/>
Enforce password history	<input type="text" value="3 passwords remembered"/>
Minimum password length	<input type="text" value="8"/>
Password complexity requirement	<input type="text" value="Must include alpha and numeric characters"/>
Password question requirement	<input type="text" value="Cannot contain password"/>
Maximum invalid login attempts	<input type="text" value="10"/>
Lockout effective period	<input type="text" value="15 minutes"/>
Obscure secret answer for password resets	<input type="checkbox"/>
Require a minimum 1 day password lifetime	<input type="checkbox"/>
Don't immediately expire links in forgot password emails	<input type="checkbox"/> <small>i</small>
<input type="button" value="Save"/> <input type="button" value="Save &amp; New"/> <input type="button" value="Cancel"/>	

## 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.

Profile Edit

Name: Purchase Manager

User License: Salesforce

Description:

Custom Profile: ✓

**Custom App Settings**

	Visible	Default		Visible	Default
All Tabs (standard__AltTabSet)	<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 checked="" 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.

Basic Access				Data Administration		
	Read	Create	Edit	Delete	View All	Modify All
Inventory Transactions	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Order Items	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Products	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Purchase Orders	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Suppliers	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input 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	<input type="checkbox"/>
Require a minimum 1 day password lifetime	<input type="checkbox"/>
Don't immediately expire links in forgot password emails	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

Save Save & New Cancel

## Milestone 10 - Roles

Roles in Salesforce are used to control record-level access and define the hierarchy of an organization, determining the level of visibility and sharing of records among users. Roles work in conjunction with profiles to provide a robust security model. While profiles control what actions users

can perform (object and field permissions), roles control which records users can see based on their position in the hierarchy.

## 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.

The screenshot shows the Salesforce Setup interface. In the top left, there's a navigation bar with 'Setup', 'Home', and 'Object Manager'. Below it, a sidebar has sections like 'Users' (with 'Roles' highlighted by a red box) and 'Feature Settings' (with 'Sales' expanded, showing 'Contact Roles on Contracts' and 'Contact Roles on Opportunities'). The main content area is titled 'Understanding Roles' with the sub-section 'Territory-based Sample'. It shows a hierarchical tree of roles: Executive Staff (CEO - President, CFO - VP, Sales), Eastern Sales Director (Sales Director of E. Sales), International Sales Director (Sales Director of Int'l Sales), Western Sales Director (Sales Director of W. Sales), and three sales rep levels (CA Sales Rep, NY Sales Rep, MA Sales Rep). Each node has a tooltip describing its permissions. At the bottom right of the main area is a 'Set Up Roles' button.

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.

The screenshot shows the 'Role Edit' page for creating a new role. The title is 'Role Edit' and the sub-section is 'New Role'. The form fields are as follows:
 

- Label:** Inventory Manager
- Role Name:** Inventory\_Manager
- This role reports to:** SVP, Sales & Marketing
- Role Name as displayed on reports:** (empty field)

 At the bottom are three buttons: 'Save' (highlighted by a red box), 'Save & New', and 'Cancel'.

## Milestone 12 - Permission Sets

Permission Sets in Salesforce are a powerful tool to extend user permissions beyond what is defined in their profiles. They allow administrators to grant additional access to various tools and functions

without altering the user's profile. Permission sets are particularly useful for providing specialized permissions to specific users without the need to create multiple profiles.

## Activity 1 : Create a Permission Set.

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

Action	Permission Set Label	Description	License
<input type="checkbox"/>	Action	Allows access to the store. Lets users see products and categories, make...	B2B Buyer Permission Set One Seat
<input type="checkbox"/>	Clone	Includes all Buyer capabilities, and allows access to manage carts and or...	B2B Buyer Manager Permission Set One Seat
<input type="checkbox"/>	Clone	Allows integration user to access features specific to C360 High Scale Flow	Cloud Integration User
<input type="checkbox"/>	Clone	Denotes that the user is a Sales Cloud or Service Cloud user.	CRM User
<input type="checkbox"/>	Clone	Allow access to commerce admin features	Commerce Admin Permission Set License Seat
<input type="checkbox"/>	Contact Center Admin	Manage Service Cloud Voice contact centers that use Amazon Connect a...	Service Cloud Voice User
<input type="checkbox"/>	Contact Center Admin (Partner Telephony)	Manage Service Cloud Voice contact centers that use your preferred tele...	Service Cloud Voice User (Partner Telephony)
<input type="checkbox"/>	Contact Center Agent	Access agent features in Service Cloud Voice contact centers that use A...	Service Cloud Voice User
<input type="checkbox"/>	Contact Center Agent (Partner Telephony)	Access agent features in Service Cloud Voice contact centers that use yo...	Service Cloud Voice User (Partner Telephony)
<input type="checkbox"/>	Contact Center Supervisor	Access supervisor features in Service Cloud Voice contact centers that us...	Service Cloud Voice User
<input type="checkbox"/>	Contact Center Supervisor (Partner Telephony)	Access supervisor features in Service Cloud Voice contact centers that us...	Service Cloud Voice User (Partner Telephony)
<input type="checkbox"/>	Cloud	Allows integration user to access entities specific to Remote Data Cloud.	Cloud Integration User
<input type="checkbox"/>	Cloud	Cloud Integration User	Salesforce
<input type="checkbox"/>	DeliveryEstimationServicePermSet		
<input type="checkbox"/>	DeliveryProfileManager		

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

Enter permission set information

Label	Purchase Manager Create Access	Save Cancel
API Name	Purchase_Manager	= Required Information
Description		
Session Activation Required	<input type="checkbox"/>	

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

**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**

Full Name	Alias	Username	Role	Active	Profile
Annapurna Gurram	AGurr	medicalinventory@sb.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	System Administrator
Chatter Expert	Chatter	chatty.00dd0000058bqlua.yrgohck7wjvo@chatter.salesforce.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Chatter Free User
Integration User	integ	integration@00dd0000058bqlua.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Analytics Cloud Integration User
<b>John PurchaseM</b>	jpurc	john@purchasem.com	Purchasing Manager	<input checked="" type="checkbox"/>	Purchase Manager
Security User	sec	insightssecurity@00dd0000058bqlua.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Analytics Cloud Security User

Next

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

... > PERMISSION SET 'PURCHASE MANAGER CREATE ACCESS' > MANAGE ASSIGNMENT EXPIRATION

## 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
-------	--------	---------	---------	-------------

**Selected Users**

Full Name	Role	Profile	Active	User License	Expires On
John PurchaseM	Purchasing Manager	Purchase Manager	<input checked="" type="checkbox"/>	Salesforce	Never Expires

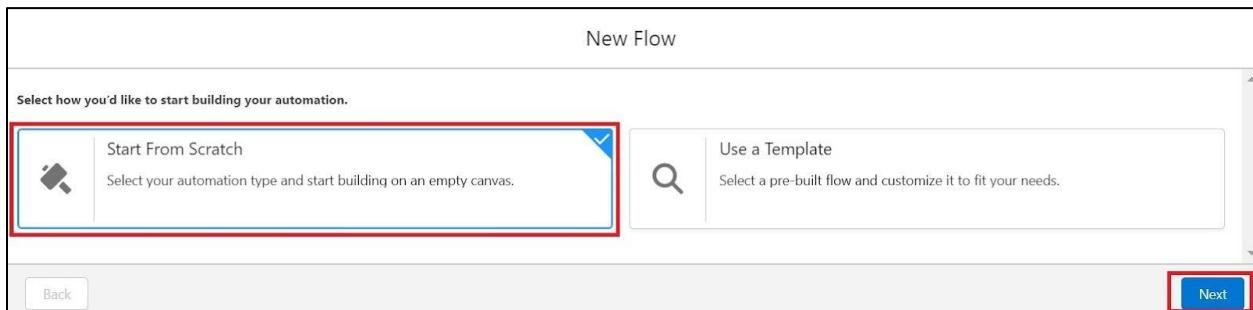
Back Assign

# Milestone 13 - Flows

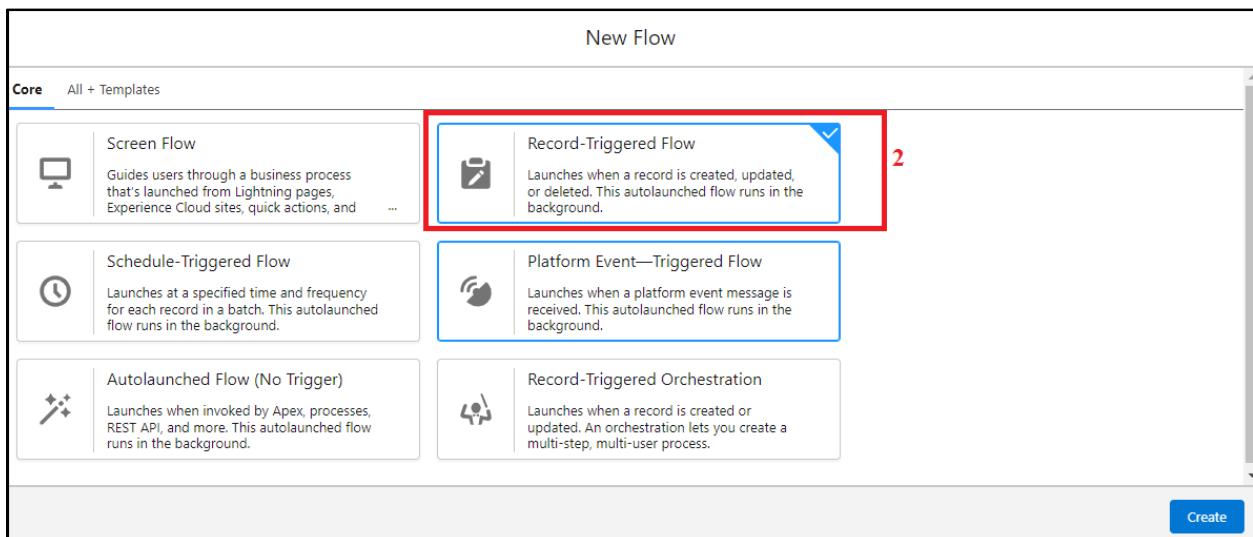
Flows in Salesforce, part of the Lightning Flow product, are powerful automation tools that help you collect data and perform actions in your Salesforce environment. Flows can be used to automate business processes, guide users through tasks, and integrate with external systems. They are highly versatile and can be configured to meet a wide range of business requirements without the need for custom code.

## 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

 Configure Start X

### 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**

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

6

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}

The screenshot shows the configuration of a 'Get Records' component. The 'Label' field is set to 'Get Purchase Record' (highlighted by a red box, step 8). The 'API Name' field is set to 'Get\_Purchase\_Record'. The 'Object' field is set to 'Purchase Order' (highlighted by a red box, step 9). The 'Condition Requirements' dropdown is set to 'All Conditions Are Met (AND)'. A single condition is defined: 'Field Id Operator Equals Value \$Record > Record ID'. A red box highlights this entire condition section (step 10).

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.

The screenshot shows the configuration for how to store record data. Under 'How Many Records to Store', 'Only the first record' is selected (highlighted by a red box). Under 'How to Store Record Data', 'Choose fields and let Salesforce do the rest' is selected (highlighted by a red box). Under 'Select Purchase Order Fields to Store in Variable', 'ID' is listed under 'Field'. Another field, 'Order\_Date\_\_c', is added and highlighted by a red box. A red box also highlights the '+ Add Field' button.

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: Assignment

\* API Name: Assignment\_1

Description:

**Set Variable Values**

Each variable is modified by the operator and value combination.

Variable	Operator	Value
ActualDeliveryDate	Equals	\$Record > 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**

\* 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

**Info** 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

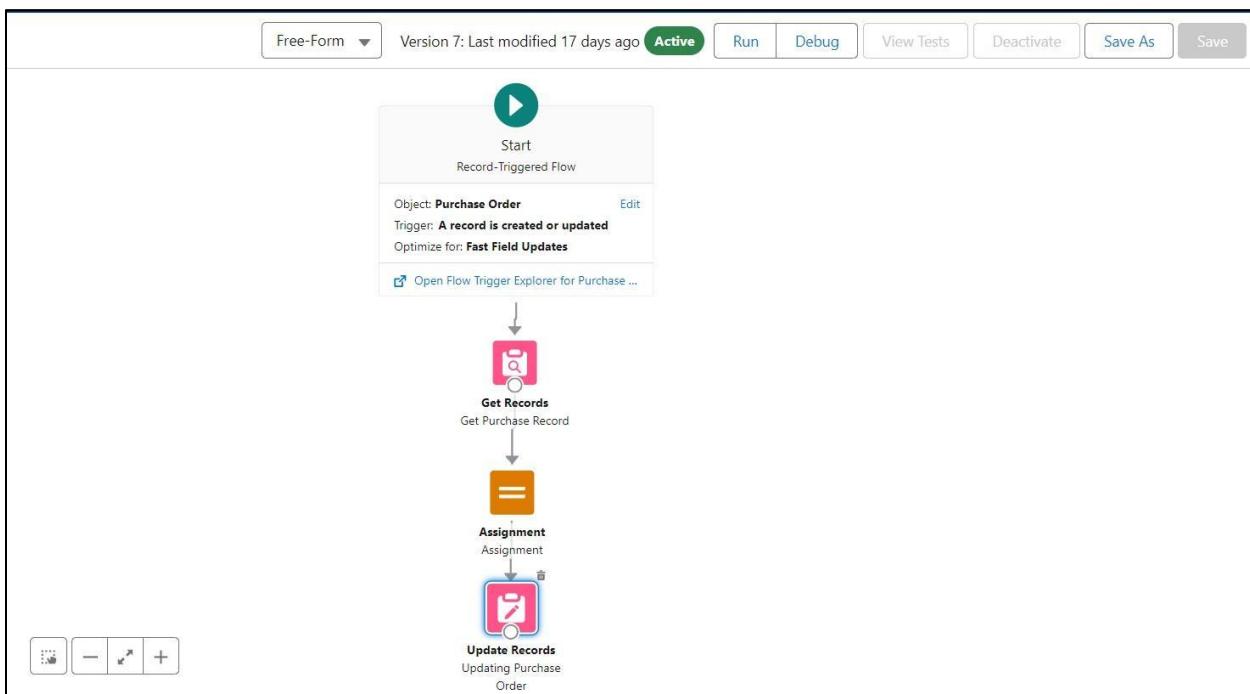
None—Always Update Record

**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.



# Milestone 14 - Triggers

Triggers in Salesforce are pieces of Apex code that execute before or after specific data manipulation events on Salesforce records, such as insertions, updates, deletions, and undeletions. They are powerful tools for automating complex business logic and ensuring data integrity by enforcing custom validation rules and workflows that cannot be achieved through declarative tools alone.

## 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.

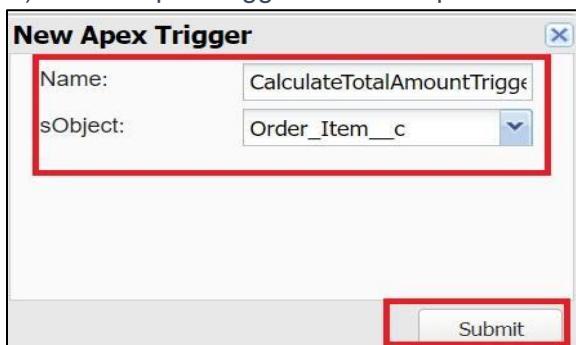
ii)Click on Developer Console: Click on the "Developer Console" option from the Setup menu. This will open the Developer Console in a new browser tab or window.

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.update, Trigger.delete, 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

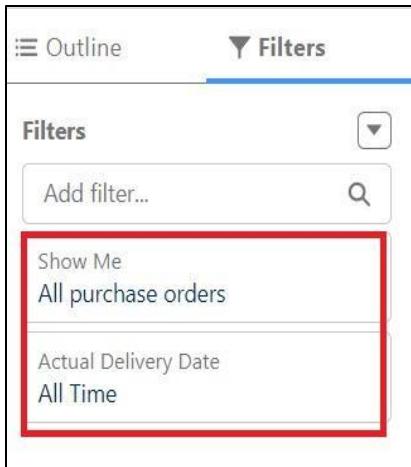
Reports in Salesforce provide a powerful way to visualize and analyze data stored in your Salesforce organization. They allow users to create, customize, and share different types of reports based on data from standard and custom objects. Reports help organizations make informed decisions by providing insights into key metrics, trends, and performance indicators.

## 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.

The screenshot shows the 'Create Report' page in Salesforce. On the left, there's a sidebar with categories like 'Recently Used' and 'All'. Under 'All', there are links for 'Accounts & Contacts', 'Opportunities', 'Customer Support Reports', 'Leads', 'Campaigns', 'Activities', and 'Contracts and Orders'. The main area has a search bar labeled 'Select a Report Type' with 'Purchase' typed in. Below it, a list of 'Report Type Name' and 'Category' is shown, with 'Purchase Orders' selected. To the right, a 'Details' panel is open for the 'Purchase Orders' report, showing its status as a 'Standard Report Type' and a large blue 'Start Report' button, which is highlighted with a red rectangle. The 'Details' panel also lists 'Fields (17)', 'Created By You' (with a note about no recent activity), and 'Created By Others'.

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”)

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

### 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 software interface for 'Medical Inventory ...'. The top navigation bar includes links for Products, Purchase Orders, Order Items, Inventory Transactions, Suppliers, Reports (selected), and Dashboards. A search bar is at the top right. The main content area is titled 'Report: Purchase Orders' and 'Purchase Orders based on Suppliers'. It shows summary statistics: Total Records (5), Total Order Count (14), and Total Order Cost (₹26,325.00). Below this is a detailed table:

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
<b>Total (5)</b>		<b>14</b>	<b>₹26,325.00</b>

At the bottom, there are buttons for Row Counts, Detail Rows, Subtotals, and Grand Total.

## 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

The screenshot shows the 'Filters' section of a reporting tool. At the top, there are tabs for 'Outline' and 'Filters'. Below the tabs is a search bar with the placeholder 'Add filter...' and a magnifying glass icon. A red box highlights a section titled 'Show Me' containing the text 'All purchase orders'. Below this, another red box highlights the 'Actual Delivery Date' dropdown set to '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 the 'Complete Purchase Details Report' interface. At the top, there is a navigation bar with 'Medical Inventory ...' and several dropdown menus. Below the navigation is a toolbar with 'Save & Run' (highlighted with a red box), 'Save', 'Close', and 'Run' buttons. To the right of the toolbar is a checkbox for 'Update Preview Automatically' (also highlighted with a red box). The main area contains a report table with the following structure:

Supplier ID	Actual Delivery Date	Purchase Order: Purchase Order ID	Product ID: Product ID	Order Count	Product ID: Product Name	Quantity Received	Amount
Supplier-001 (12)	18/06/2024 (2)	Purchase-0002 (2)	Gen-0001	2	Syringes	50	₹250.00
			Cap-0001	2	Dolo 650	150	₹3,000.00
				2		200	₹3,250.00
				2		200	₹3,250.00
				Subtotal			
	22/06/2024 (3)	Purchase-0001 (3)	Gen-0001	3	Syringes	5	₹25.00
			Gen-0001	3	Syringes	10	₹50.00
			Cap-0001	3	Dolo 650	100	₹2,000.00
				3		115	₹2,075.00
				Subtotal			
	23/06/2024 (3)	Purchase-0003 (3)	Syr-0001	3	Calpol 120mg Syrup	100	₹4,000.00
			Cap-0001	3	Dolo 650	50	₹1,000.00
			Gen-0001	3	Syringes	400	₹2,000.00
				3		550	₹7,000.00
				Subtotal			
	11/07/2024 (4)	Purchase-0004 (4)	Syr-0001	4	Calpol 120mg Syrup	100	₹4,000.00
			IV-0001	4	Saline	50	₹2,500.00
			Cap-0001	4	Dolo 650	100	₹2,000.00
			Gen-0001	4	Sunrise	200	₹1,000.00

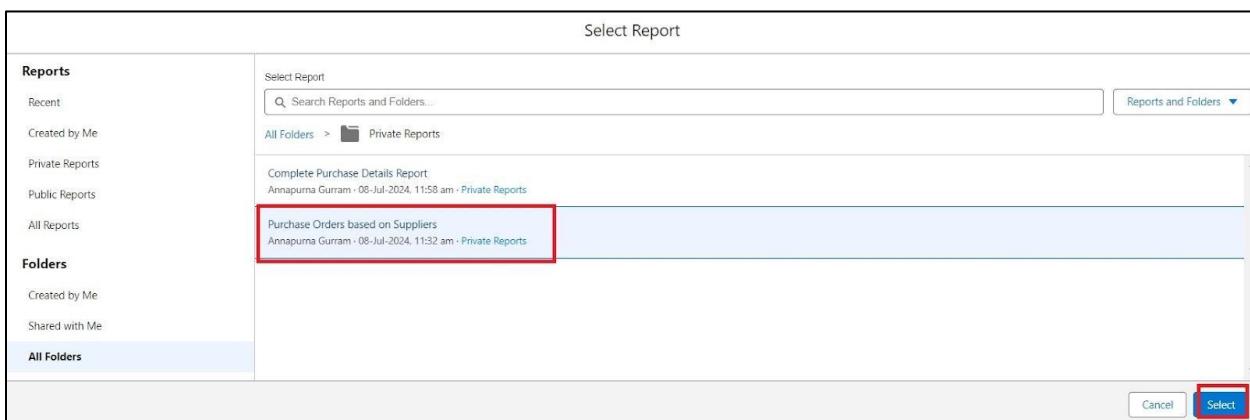
On the left side, there are sections for 'Groups' and 'Columns' with various filter and column selection options. At the bottom of the report area, there are checkboxes for 'Row Counts', 'Detail Rows', 'Subtotals', and 'Grand Total'.

## Milestone 16 - Dashboards

Dashboards in Salesforce are dynamic visual representations of key metrics and data from reports, providing a consolidated view of organizational performance and trends. They are powerful tools for monitoring real-time data, tracking progress towards goals, and gaining actionable insights at a glance. Dashboards consist of components such as charts, tables, metrics, and gauges that display data from underlying reports.

## 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.



Add Widget

**Report**

Purchase Orders based on Suppliers X

Use chart settings from report i

**Display As**



**Value**

Sum of Total Order Cost

**Sliced By**

Supplier ID

**Preview**

**Purchase Orders based on Suppliers**

Sum of Total Order Cost



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 

**Purchase Orders based on Suppliers**

Sum of Total Order Cost



Supplier ID

Supplier-001

Supplier-002

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

Cancel
Add

## 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

