

Team Details

S.NO	NAME	ROLL.NO	MAIL ID
1	RAGAVI S	2322KC41	2322kc41@kgcas.com
2	RAMYA R	2322KC42	2322kc42@kgcas.com
3	RITHIGA V	2322KC43	2322kc43@kgcas.com
4	SACHIN M	2322KC44	2322kc44@kgcas.com

SALESFORCE

CRM Application for Jewel Management - (Developer)

The Jewel Inventory System is a comprehensive software Solution designed to streamline and manage the inventory and sales processes of a jewellery store or a jewellery manufacturer.

The system aims to provide an efficient and user-friendly solution to track and control the inventory of various jewellery items, maintain accurate records, and facilitate seamless sales transactions.

What you'll learn

1. Real Time Salesforce Project
2. Data Modelling
3. Creating an Application
4. User Interface Customization
5. Object & Relationship in Salesforce
6. Formula fields and Validation rules.
7. Field Dependencies
8. Record Types
9. Cross object formula fields.
10. Conditional formatting.
11. Flows
12. Email alerts and email templates
13. Reports & Dashboards

Salesforce

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 organised something like this: <https://youtu.be/r9EX3lGde5k>

Creating Developer Account

Creating a developer org in salesforce.

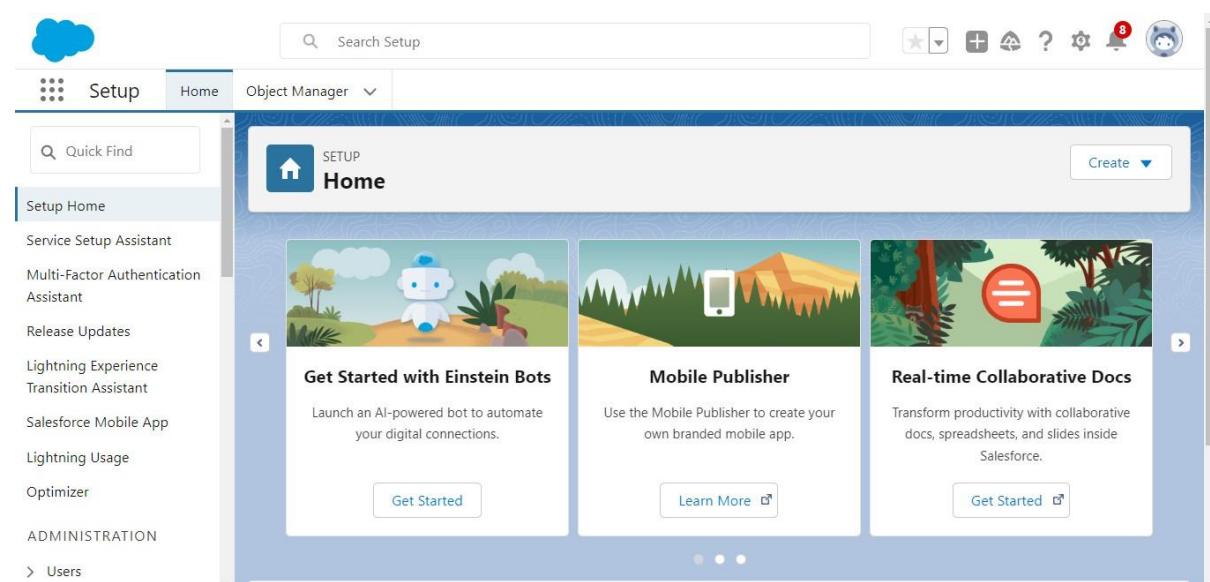
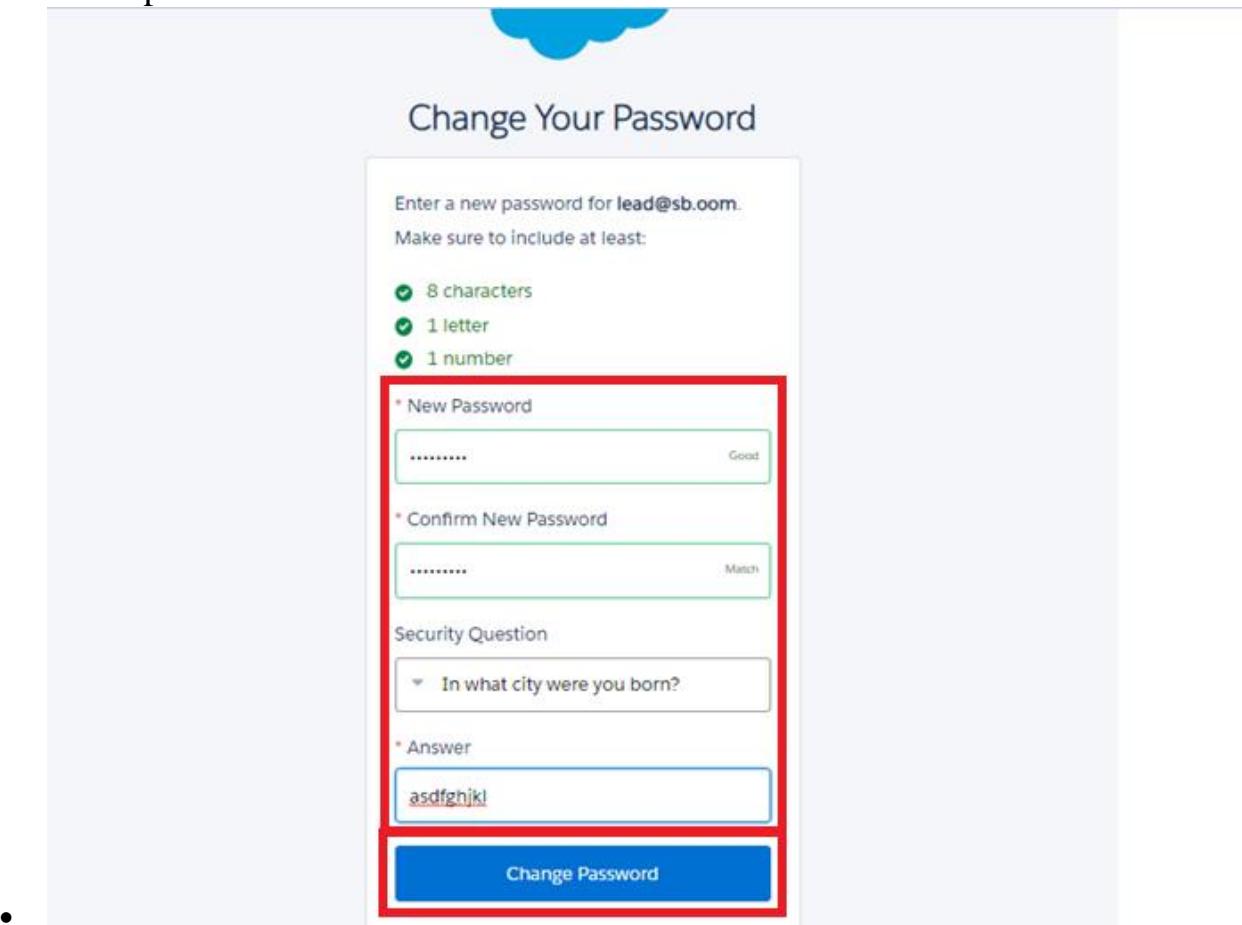
1. Go to <https://developer.salesforce.com/signup>
2. On the sign up form, enter the following details :

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.

Account Activation

1. Go to the inbox of the email that you used while signing up. Click on the Reset Password to activate your account. The email may take 5-10mins.
 - o Click on Reset Password
 - o Give a password and answer a security question and click on change password.



- Then you will redirect to your salesforce setup page.

Object

What Is an Object?

Salesforce objects are database tables that permit you to store data that is specific to an organization. What are the types of Salesforce objects

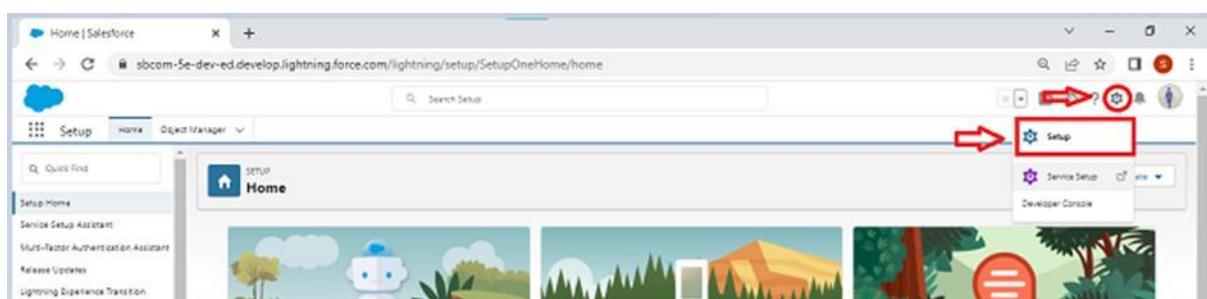
Salesforce objects are of two types:

- Standard Objects: Standard objects are the kind of objects that are provided by salesforce.com such as users, contracts, reports, dashboards, etc.
- Custom Objects: Custom objects are those objects that are created by users. They supply information that is unique and essential to their organization. They are the heart of any application and provide a structure for sharing data.

Use Case:

Creating an object in Salesforce organisation is essential for efficient data management and process automation. By defining custom objects, businesses can structure and store data specific to their needs, enabling streamlined workflows, personalized reporting, and enhanced user experiences. Objects serve as the foundation for organizing and leveraging critical information within Salesforce.

To Navigate to Setup page:



Click on gear icon >> click setup.

Create Jewel Customer Object

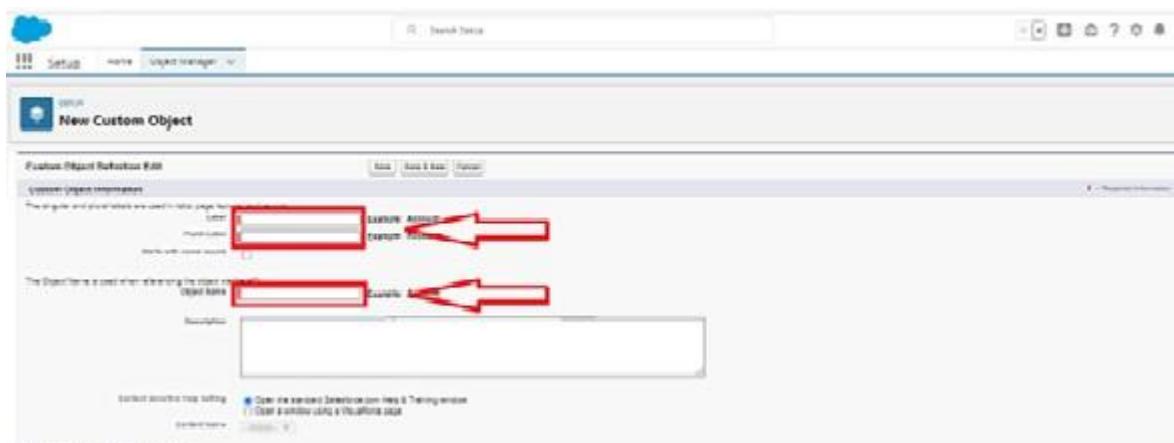
The purpose of creating a Jewel Customer custom object is to store and manage information about Customer.

To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.



- Enter the label name : Jewel Customer
- Plural label name : Jewel Customers



- Enter the label name : Jewel Customer

Enter Record Name Label and Format

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

Record Name	Customer	Example: Account Name
Data Type	Text	

Optional Features

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

Object Classification

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

Allow Sharing
 Allow Bulk API Access
 Allow Streaming API Access

Deployment Status

In Development
 Deployed

- Plural label name : Jewel Customers
- Click on Allow reports.
- Allow search and click Save.

Create Item Object

The purpose of creating a Item object is to manage the inventory of gold and silver items.

To create an object:

- From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
 - Enter the label name >> Item
 - Plural label name >> Items
 - Enter Record Name Label and Format
 - Record Name >> Item Id
 - Data Type >> Auto Number
 - Display Format >> Item-{00}
 - Starting Number >> 1
- Click on Allow reports.
- Allow search >> Save.

Note: Create 3 more objects with label names as Customer Order, Price, Billing
 (Use “Auto Number” as a data type for Customer Order, Price, Billing).

Tabs

What is Tab: A tab is like a user interface that is used to build records for objects and to view the records in the objects.

Types of Tabs:

- Custom Tabs
 - Custom object tabs are the user interface for custom applications that you build in salesforce.com. They look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.

- Web Tabs
 - Web Tabs are custom tabs that display web content or applications embedded in the salesforce.com window. Web tabs make it easier for your users to quickly access content and applications they frequently use without leaving the salesforce.com application.
- Visualforce Tabs
 - Visualforce Tabs are custom tabs that display a Visualforce page. Visualforce tabs look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.
- Lightning Component Tabs
 - Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app.
- Lightning Page Tabs
 - Lightning Page Tabs let you add Lightning Pages to the mobile app navigation menu. Lightning Page tabs don't work like other custom tabs. Once created, they don't show up on the All Tabs page when you click the Plus icon that appears to the right of your current tabs. Lightning Page tabs also don't show up in the Available Tabs list when you customize the tabs for your apps.

Use Case:

Creating Objects and storing Jewels data is the very first step in the requirements they want. Now to access the stored data by an Owner(Gold Smith) in the organisation Admin needs to create Tabs. By designing a dedicated Tab, businesses can improve user experience, simplify navigation, and provide quick access to critical information, enhancing productivity and ensuring efficient utilisation of Salesforce's capabilities.

Creating a Custom Tab

To create a Tab:(Customer)

- Go to setup page >> type Tabs in Quick Find bar >> click on tabs >> New (under custom object tab)

Custom Tabs

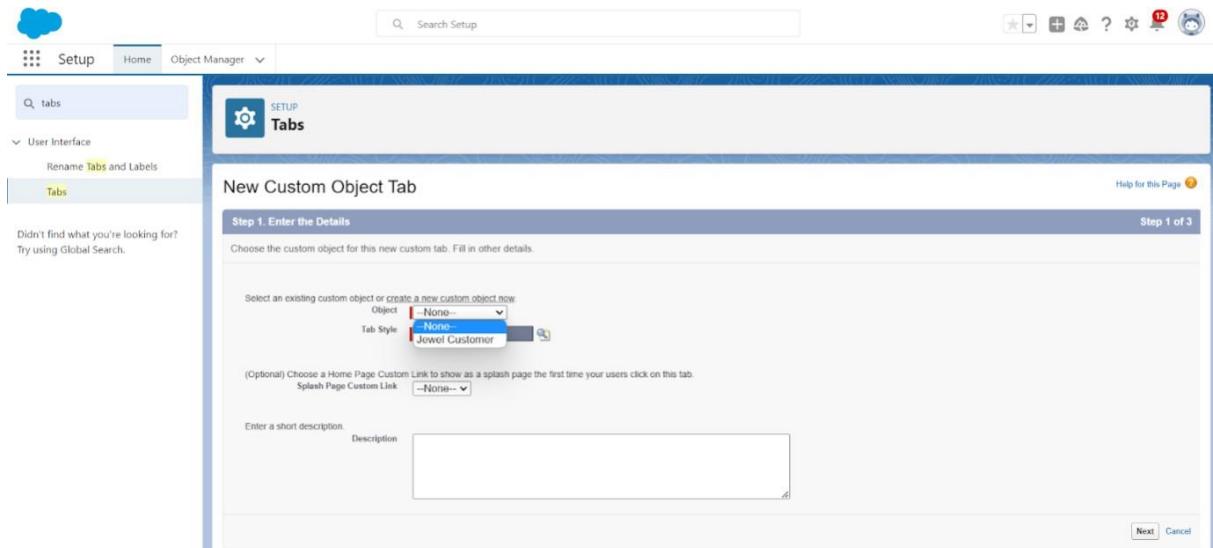
You can create new custom tabs to extend Salesforce functionality or to build new application functionality.

Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external pages. Lightning Component tabs allow you to add Lightning components to the navigation bar. You can also allow users to add Lightning Pages to Lightning Experience and the mobile app.

The screenshot shows two sections of the Salesforce Setup interface:

- Custom Object Tabs:** A table with one row showing "No Custom Object Tabs have been defined". It includes a "New" button (which is highlighted with a red box) and a "What Is This?" link.
- Web Tabs:** A table with one row showing "No Web Tabs have been defined". It includes a "New" button and a "What Is This?" link.

- Select Object(Jewel Customer) >> Select any tab style >> Next (Add to profiles page) keep it as default >> Next (Add to Custom App) keep it as default >> Save.



To create a Tab:(Item)

- Go to setup page >> type Tabs in Quick Find bar >> click on tabs >> New (under custom object tab)
- Select Object(Item) >> Select the tab style >> Next (Add to profiles page) keep it as default >> Next (Add to Custom App) keep it as default >> Save.

Note: Now create tabs for Customer Order, Price, Billing objects.

The Lightning App

An app is a collection of items that work together to serve a particular function. In Lightning Experience, Lightning apps gives users access to sets of objects, tabs, and other items all in one convenient bundle in the navigation bar. Lightning apps let you brand your apps with a custom color and logo. You can even include a utility bar and Lightning page tabs in your Lightning app. Members of your org can work more efficiently by easily switching between apps.

Use Case:

Well done you have reached close to your requirement by creating the objects to store the organization's data. Making a database for an organization is just not enough to reach out the requirements, the task is how the users at the organization can access the objects you have created for them. As an Admin for the organization it's your duty to make sure every user of the organization is able to access the data modelling structure.

Types :

1. App Page

- Designed for an entire app.
- Can include dashboards, reports, or custom components.

2. Home Page

- Acts as a personalized landing page for users.
- Can show tasks, performance metrics, assistant, and announcements.

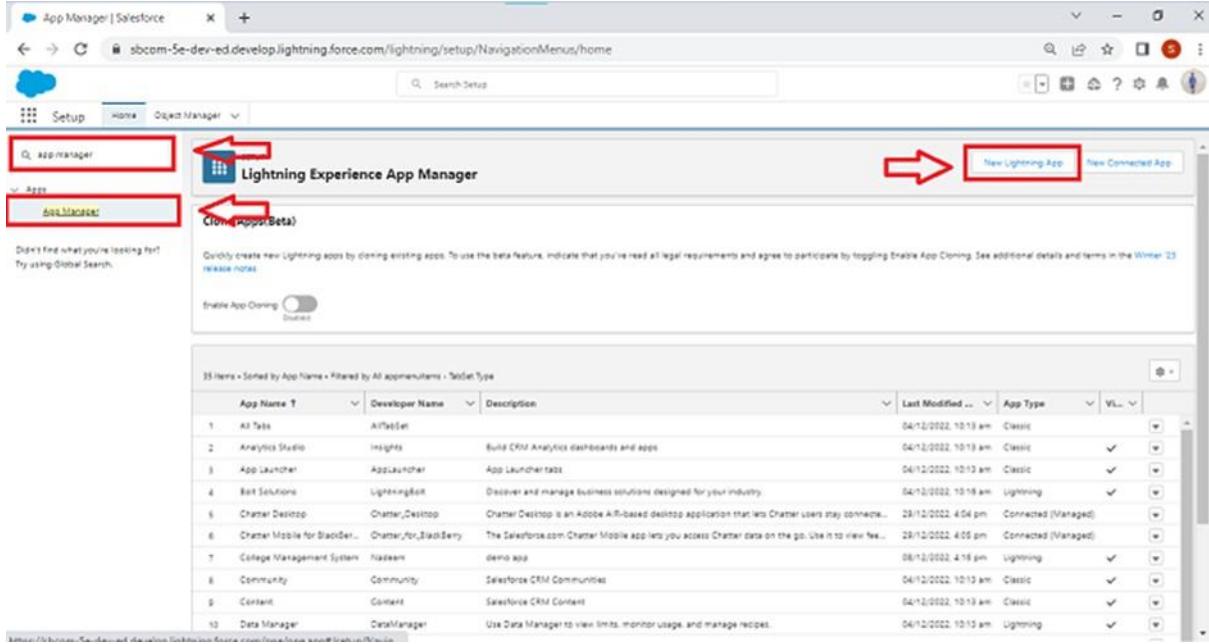
3. Record Page

- Used to customize the layout of a specific object's record.
- Example: For an Opportunity record, you can display related contacts, deal stages, and activities in a customized format.

Create a Lightning App

To create a lightning app page:

- Go to setup page >> search “app manager” in quick find >> select “app manager” >>



The screenshot shows the Salesforce App Manager interface. At the top, there are tabs for 'Setup', 'Home', and 'Object Manager'. A red box highlights the 'App Manager' tab. Below the tabs, there's a search bar and a 'Clone (Appx Beta)' button. To the right of the clone button is a red arrow pointing to a 'New Lightning App' button. Another red box highlights the 'New Lightning App' button. The main area displays a table of existing apps, with 35 items listed. The columns include 'App Name', 'Developer Name', 'Description', 'Last Modified', 'App Type', and 'VLR'. The table lists various standard Salesforce apps like 'AI Test', 'Analytics Studio', 'App Launcher', etc.

- click on New lightning App.

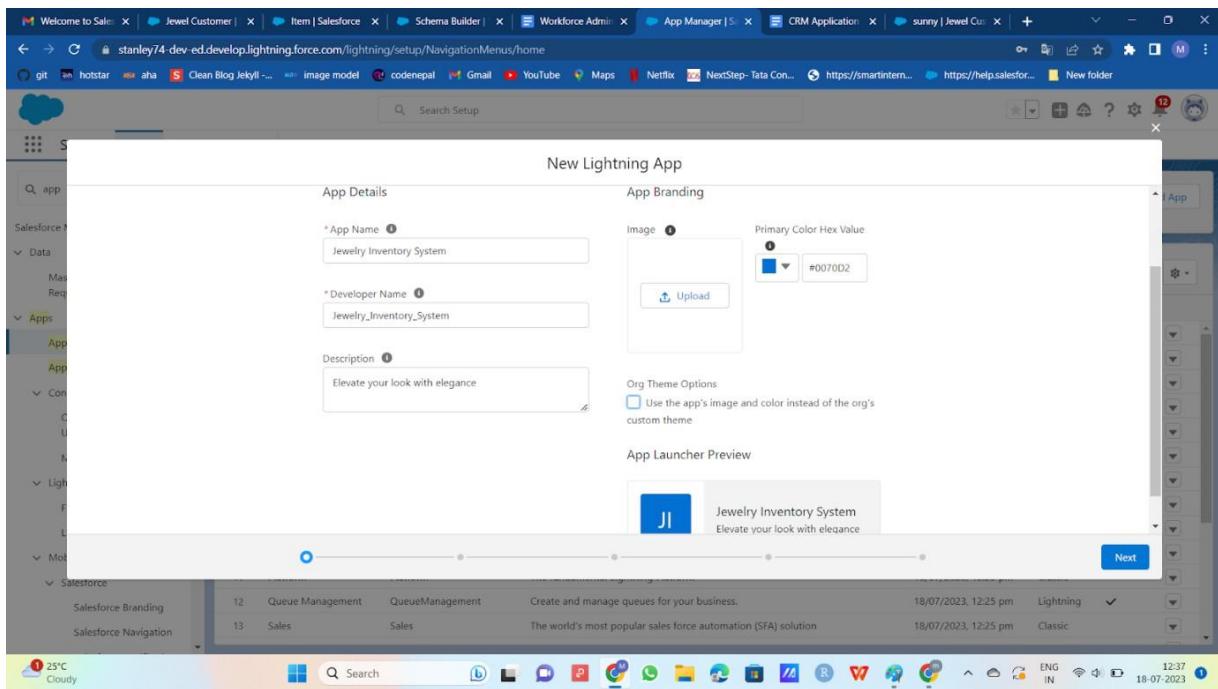
Fill the app name in app details and branding as follow App Name : Jewellery Inventory System.

Developer Name : This will auto populated

Description : Elevate your look with elegance

Image : optional (if you want to give any image you can otherwise not mandatory)

Primary colour hex value : keep this default.



1. Then click Next >> (App option page) Set Navigation Style as Console

App Options

Navigation Items

Supported Form Factors

Navigation >> Next.

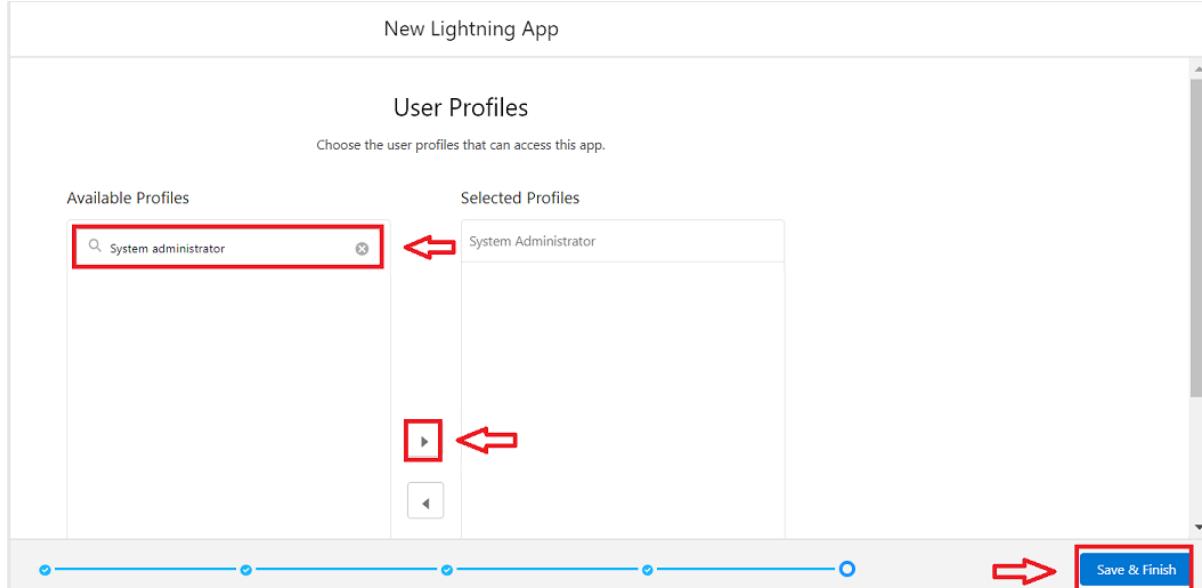
- (Utility Items) keep it as default >> Next.
- To Add Navigation Items:

Navigation Items

Available Items

Selected Items

- Search for the item in the (JewelCustomer,Item,CustomerOrder,Price,Billing,Reports, Dashboard) from the search bar and move it using the arrow button ? Next? Next.
- To Add User Profiles:



Search profiles (System administrator) in the search bar >> click on the arrow button >> save & finish.

Fields

When we talk about Salesforce, Fields represent the data stored in the columns of a relational database. It can hold any valuable information that you require for a specific object. Hence, the overall searching, deletion, and editing of the records become simpler and quicker.

Types of Fields :

- Standard Fields
- Custom Fields

Standard Fields:

As the name suggests, the Standard Fields are the predefined fields in Salesforce that perform a standard task. The main point is that you can't simply delete a Standard Field until it is a non-required standard field. Otherwise, users have the option to delete them at any point from the application freely. Moreover, we have some fields that you will find common in every Salesforce application. They are,

- >> Created By
- >> Owner
- >> Last Modified
- >> Field Made During object Creation

Custom Fields:

On the other side of the coin, Custom Fields are highly flexible, and users can change them according to requirements. Moreover, each organiser or company can use them if necessary. It means you need not always include them in the records, unlike Standard fields. Hence, the final decision depends on the user, and he can add/remove Custom Fields of any given form.

Use Case:

Now it's time for you to think out of the box for your organisation. You have successfully created the database objects for the organisation but now all eyes turn on you as you have to define what sort of information the objects store which you have created. As a life saver of your organisation you come up with the idea of creating fields to store different types of data.

Creating Lookup Relationship

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 between Jewel Customer & Customer Order Objects.

1. Go to the setup page >> click on object manager >> type object name(Customer Order) in the quick find bar >> click on the object.
2. Click on fields & relationship >> click on New.
3. Select “Lookup relationship” as data type and click Next.
4. Select the related object “ Jewel Customer ”.
5. Give Field Label as “Customer” and click Next.
6. Next >> Next >> Save.

Creating a Master-Detail Relationship

Master-detail relationship is a type of relationship between two objects where the master object controls certain behaviours and settings of the detail object. Here are a few use cases that demonstrate the use of master-detail relationships

Creating Master-Detail Relationship between Item & Customer Order Object.

To Create a Master-Detail relationship :

1. Go to the setup page >> click on object manager >> type object name(Customer Order) in the quick find bar >> click on the object.
2. Click on fields & relationships >> click on New.
3. Select “Master-Detail relationship” as data type and click Next.
4. Select the related object “ Item ”.
5. Give Field Label as “Item” and click Next.Next>> Next >> Save.

Creating Text Field in Jewel Customer Object

To create fields in an object:

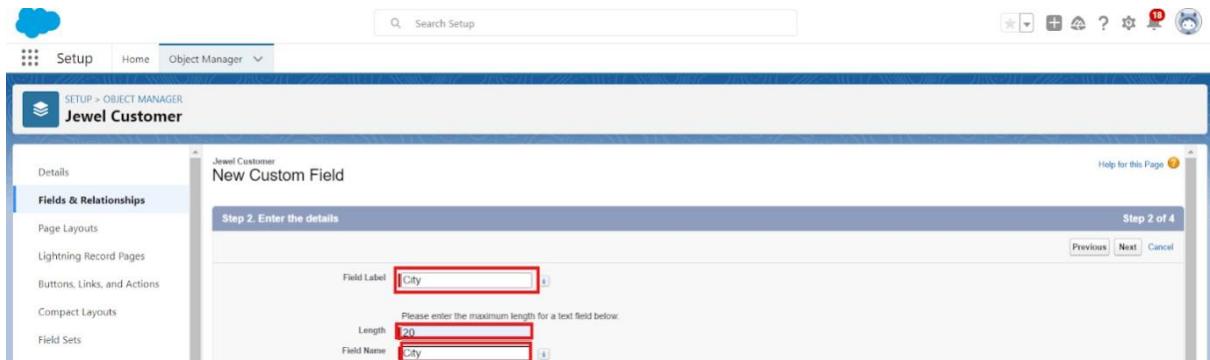
1. Go to setup >> click on Object Manager >> type object name(Jewel

The screenshot shows the Salesforce Object Manager. At the top, there's a search bar with 'jewel' typed into it. Below the search bar, the 'Object Manager' tab is selected. A red arrow points to the 'jewel' entry in the search results. In the main list, there is one item: 'Jewel Customer'. Another red arrow points to the 'Jewel Customer' label. The table columns are labeled: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. The 'Jewel Customer' row has 'Jewel_Customer__c' in the API Name column, 'Custom Object' in the Type column, and '7/18/2023' in the Last Modified column. The Deployed column has a checkmark.

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Jewel Customer	Jewel_Customer__c	Custom Object		7/18/2023	✓

Customer) in quick find bar >> click on the object.

- Now click on “Fields & Relationships” >> New
- Select Data type as “Text”.

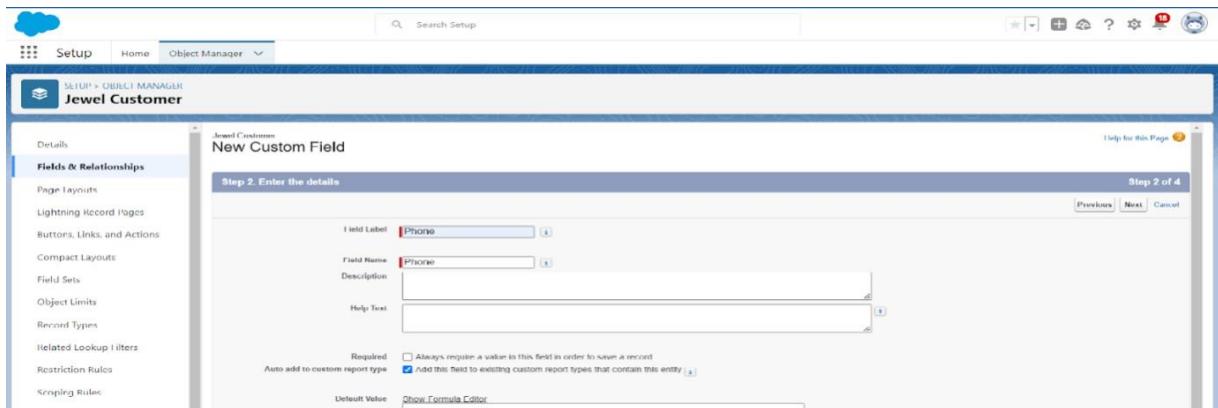


- Click on Next
0. Fill the above as following:
 - o Field Label: City
 - o Length : 20
 - o Field Name : gets auto generated
 - o Click on Next >> Next >> Save and new.

Creating the Phone field in object Jewel Customer

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Jewel Customer) in quick find bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as “Phone” and click Next.
4. Given the Field Label as “ Phone”.



1. Field Name will be auto populated, and click on Next >> Next >> Save & new.

Creating the Email field in object Jewel Customer

To create fields in an object:

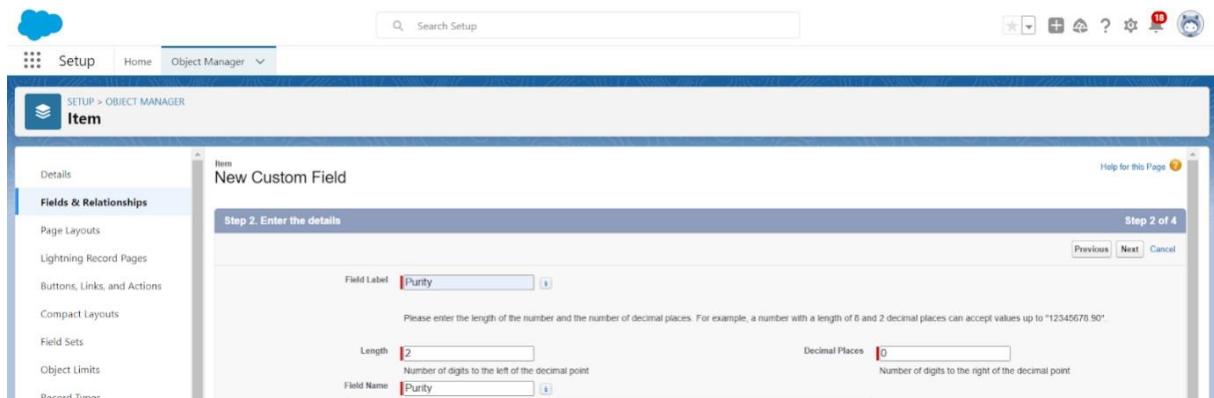
- Go to setup >> click on Object Manager >> type object name(Jewel Customer) in quick find bar >> click on the object.
- Now click on “Fields & Relationships” >> New
- Select Data type as “Email” and click Next.

- Given the Field Label as “ Email”.
- Field Name will be auto populated, and click on Next >> Next >> Save.

Creating the number field in Item object

To create fields in an object:

- Go to setup >> click on Object Manager >> type object name(Item) in quick find bar? click on the object.
- Now click on “Fields & Relationships” >> New
- Select Data type as “Number” and click Next.

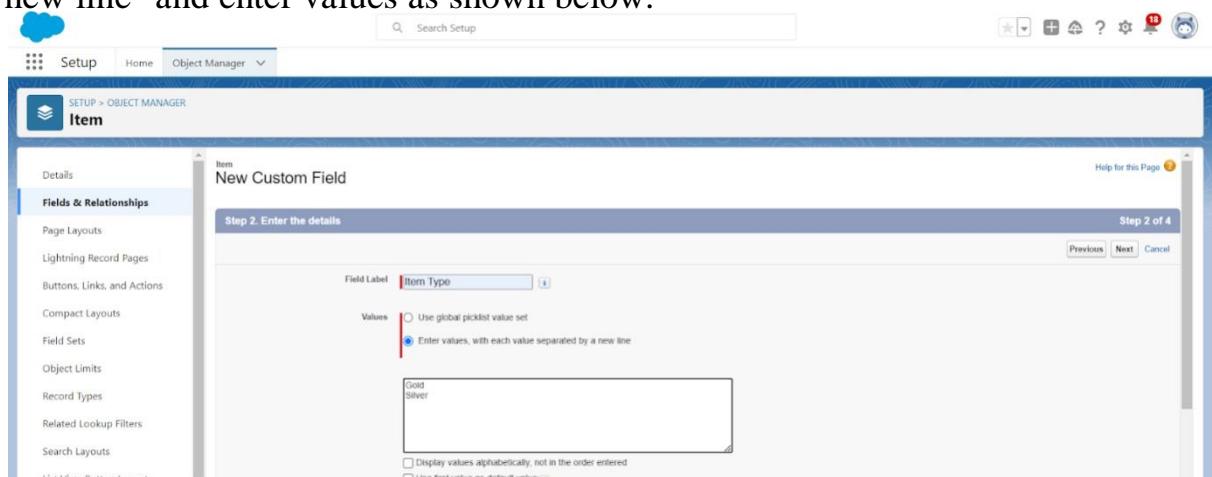


- Given the Field Label as “ Purity” and length as “ 2 ”.
- Field Name will be auto populated, and click on Next >> Next >> Save.

Creating Picklist Field in Item Object

To create fields in an object:

- Go to setup >> click on Object Manager >> type object name(Item) in quick find bar>> click on the object.
- Now click on “Fields & Relationships” >> New.
- Select Data type as “Picklist” and click Next.
- Enter Field Label as “Item Type”.
- In values select “Enter values(Gold,Silver), with each value separated by a new line” and enter values as shown below.

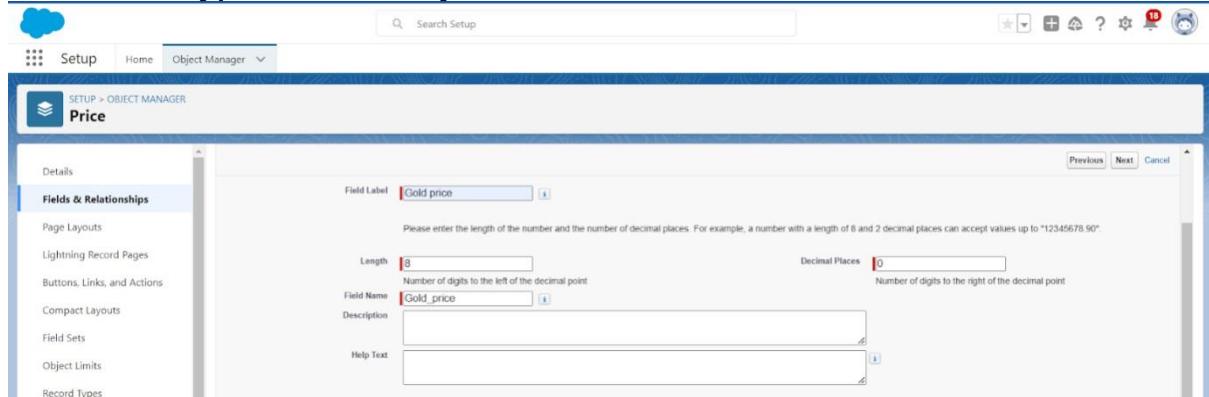


6. Click Next? Next ?Next ?Save .

Creating Currency Field in Price Object

To create fields in an object:

- Go to setup >> click on Object Manager >> type object name(Price) in quick find bar >> click on the object.
- Now click on “Fields & Relationships” >> New.
- Select Data type as “Currency” and click Next.



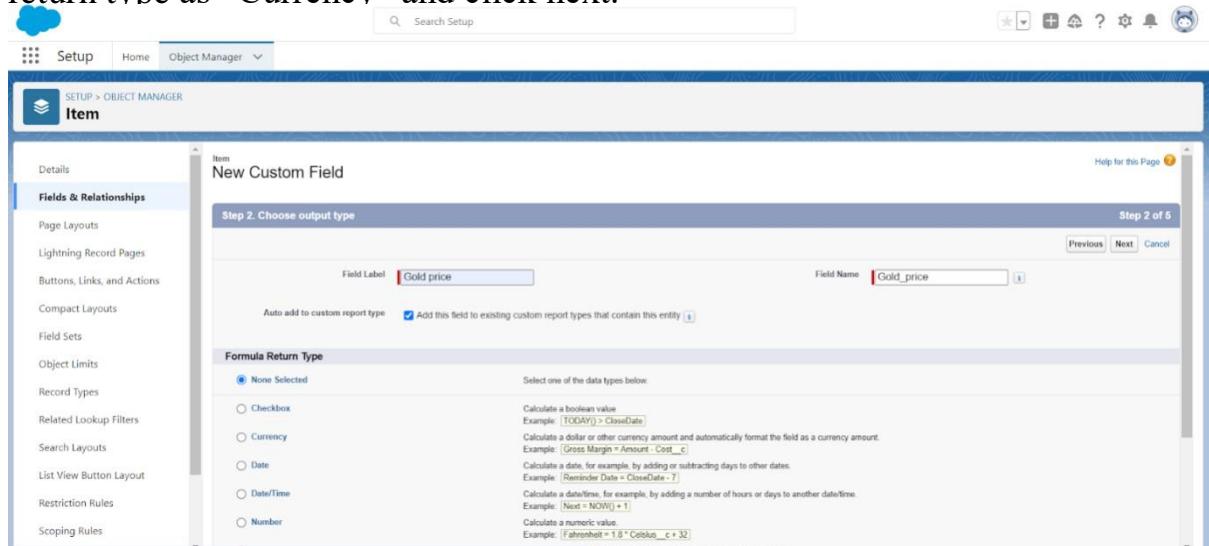
- Enter Field Label as “Gold Price” and length as “ 8”and decimal 0.Field name will be auto generated.
- Click Next >> Next >> Next >>Save .

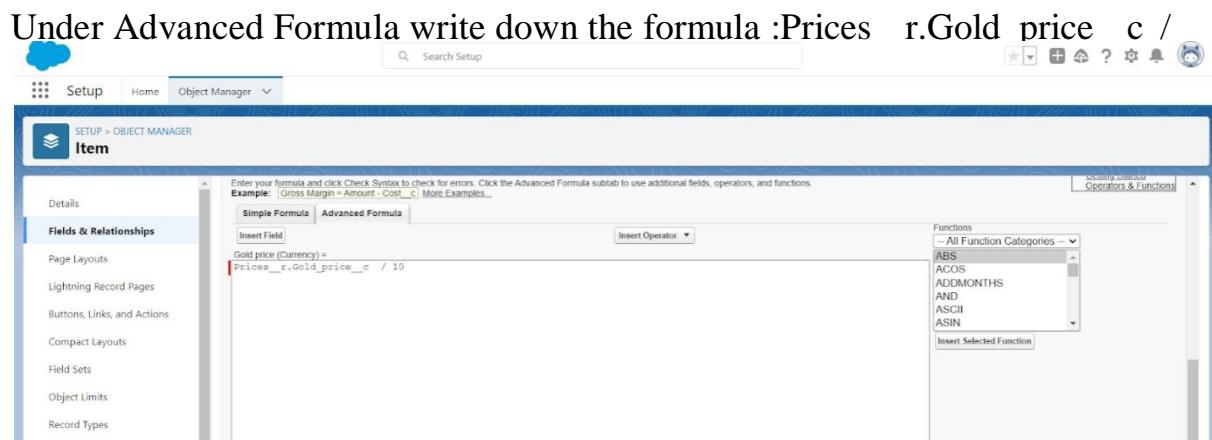
Creating Formula Field(Cross Object) in Item Object

To create fields in an object:

(Note:Create a Lookup Relationship in Item Object to Price Object with Field Name:Prices)

1. Go to setup >> click on Object Manager >> type object name(Item) in quick find bar? click on the object.
2. Now click on “Fields & Relationships” >> New.
3. Select Data type as “Formula” and click Next.
4. Give Field Label and Field Name as “Gold Price” and select formula return type as “Currency” and click next.





click “Check Syntax” and Next >> Next >> Save & New.

Creating Remaining Fields in Objects

Now create the remaining fields using the data types mentioned.

s.no	Object name	Fields	
1	Jewel Customer	Field Name	Data
		type State	Text(20)
		Street	Text(20)
		Country	Text(18)
		Zip/Postal code	Text(6)

2	Price	Silver Price	Currency (Length=8, Decimal=5)
---	-------	--------------	-----------------------------------

3	Item	Field Label:Customer Name	Lookup Relationship
		with Jewel Customer Object Ornament	Text(20)
		Weight	Number (Length=8, Decimal=5)
		Stone Weight	Number (Length=5, Decimal=5)
		Percentage	Number (Length=2, Decimal=0)

Stone/Other Price	Currency (Length=8, Decimal=2)
Expected Days Of Return	Picklist 1-3 Days 4-5 Days 6-7 Days 8-10 Days

		Priority	Picklist
			Low Medium High Critical
	Silver Price	Formul	
		a (Return Type:Number) (Decimal=3)	(Prices_r.Silver_price_c / 1000)
	Purity Gold Price	Formul	
		a (Return Type:Currency) (Decimal=2)	

		$((\text{Prices_r.Gold_price_c} * \text{Purity_c}) / 24) / 10$
Total Weight	Formul a (Return Type:Number) (Decimal=3)	$(\text{Weight_c} - \text{Stone_weight_c})$
Amount	Formul a (Return Type:Currency) (Decimal=3)	$\text{IF}(\text{ISPICKVAL}(\text{Item_Type_c}, \text{"Gold"}), \text{Total_weight_c} * \text{Purity_Gold_price_c}, \text{Total_weight_c} * \text{Silver_price_c})$
KDM	Formul a (Return Type:Currency) (Decimal=0)	$(\text{Amount_c} * \text{Percentage_c}) / 100$
Making Charges	Formul a (Return Type:Currency) (Decimal=0)	

		<pre>IF(ISPICKVAL(Item_Type_c , "Gold"), Weight_c * 300 , Weight_c * 10)</pre>
--	--	--

4	Customer Order	<table border="1"> <tr> <td>Order Status</td><td>Picklist</td></tr> <tr> <td></td><td> <ul style="list-style-type: none"> Started Not Started On Hold Completed Not Completed </td></tr> </table>	Order Status	Picklist		<ul style="list-style-type: none"> Started Not Started On Hold Completed Not Completed
Order Status	Picklist					
	<ul style="list-style-type: none"> Started Not Started On Hold Completed Not Completed 					

5	Now create the remaining fields using the data types mentioned.		
	s.no	Object name	Fields

		Jewel Customer											
1			<table border="1"> <thead> <tr> <th>Field Name</th><th>Data</th></tr> </thead> <tbody> <tr> <td>type State</td><td>Text(20)</td></tr> <tr> <td>Street</td><td>Text(20)</td></tr> <tr> <td>Country</td><td>Text(18)</td></tr> <tr> <td>Zip/Postal code</td><td>Text(6)</td></tr> </tbody> </table>	Field Name	Data	type State	Text(20)	Street	Text(20)	Country	Text(18)	Zip/Postal code	Text(6)
Field Name	Data												
type State	Text(20)												
Street	Text(20)												
Country	Text(18)												
Zip/Postal code	Text(6)												
2	Price		<table border="1"> <tr> <td>Silver Price</td><td>Currency (Length=8, Decimal =5)</td></tr> </table>	Silver Price	Currency (Length=8, Decimal =5)								
Silver Price	Currency (Length=8, Decimal =5)												
3	Item		<table border="1"> <thead> <tr> <th>Field Label:Customer Name</th><th>Lookup Relationship with Jewel Customer Object</th></tr> </thead> <tbody> <tr> <td>Ornament</td><td>Text(20)</td></tr> <tr> <td>Weight</td><td>Number (Length=8,Decimal=5)</td></tr> <tr> <td>Stone Weight</td><td>Number</td></tr> </tbody> </table>	Field Label:Customer Name	Lookup Relationship with Jewel Customer Object	Ornament	Text(20)	Weight	Number (Length=8,Decimal=5)	Stone Weight	Number		
Field Label:Customer Name	Lookup Relationship with Jewel Customer Object												
Ornament	Text(20)												
Weight	Number (Length=8,Decimal=5)												
Stone Weight	Number												

		(Length=5,Decimal=5)
	Percentage	Number (Length=2,Decimal=0)
	Stone/Other Price	Currency (Length=8,Decimal=2)
	Expected Days Of Return	<p>Picklist</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> 1-3 Days 4-5 Days 6-7 Days 8-10 Days </div>
	Priority	<p>Picklist</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> Low Medium High Critical </div>
	Silver Price	<p>Formul a (Return Type:Number) (Decimal=3)</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> $(\text{Prices_r.Silver_price_c} / 1000)$ </div>
	Purity Gold Price	<p>Formul a (Return Type:Currency) (Decimal=2)</p>

		$((\text{Prices_r.Gold_price_c} * \text{Purity_c}) / 24) / 10$
Total Weight	Formul a (Return Type:Number) (Decimal=3)	$(\text{Weight_c} - \text{Stone_weight_c})$
Amount	Formul a (Return Type:Currency) (Decimal=3)	$\text{IF}(\text{ISPICKVAL}(\text{Item_Type_c}, "Gold"), \text{Total_weight_c} * \text{Purity_Gold_price_c}, \text{Total_weight_c} * \text{Silver_price_c})$
KDM	Formul a (Return Type:Currency) (Decimal=0)	$(\text{Amount_c} * \text{Percentage_c}) / 100$

		Making Charges	Formula a (Return Type:Currency) (Decimal=0)
			IF(ISPICKVAL(Item_Type_c , "Gold"), Weight_c * 300 , Weight_c * 10)

	Customer Order	Order Status	Picklist Started Not Started On Hold Completed Not Completed

5	Billi n g	Field Relationship	Lookup Label:Item with Item Object

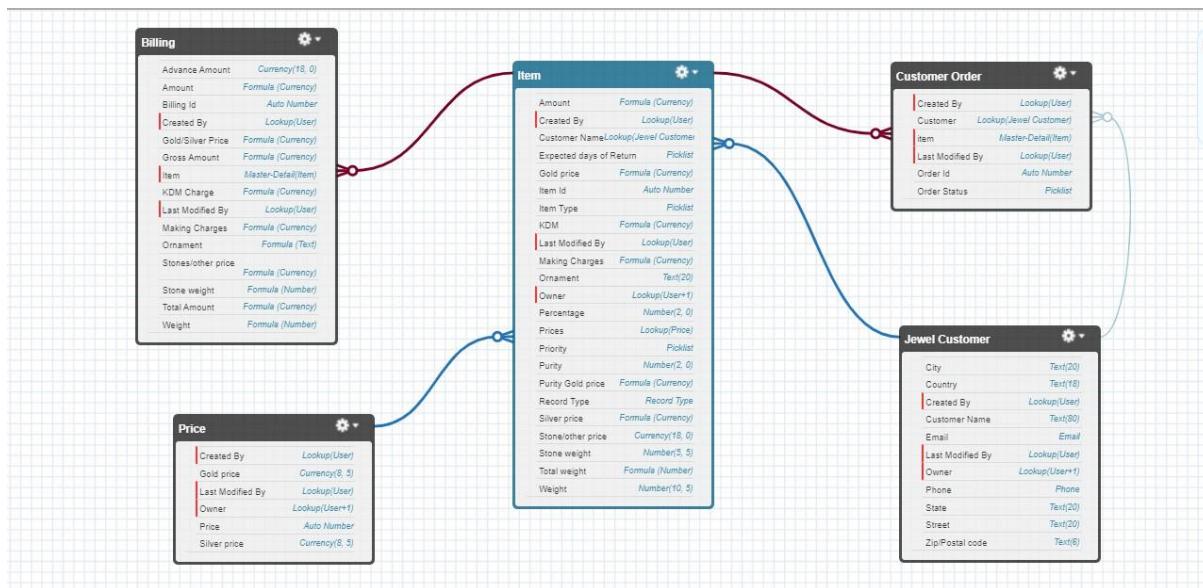
		Ornament	<p>Formula (Return Type:Text)</p> <p>Item_r.Ornament_c</p>
		Stone weight	<p>Formula (Return Type:Number) (Decimal=2)</p> <p>Item_r.Stone_weight_c</p>
		Weight	<p>Formula Return Type:Number (Decimal=2)</p> <p>Item_r.Total_weight_c</p>
		Amount	<p>Formula (Return Type:Currency) (Decimal=2)</p> <p>Item_r.Amount_c</p>

Gold/Silver Price	a (Return Type:Currency) (Decimal=2)	Formul
		IF(ISPICKVAL(Item_r.Item_Type_c , "Gold"), Item_r.Gold_price_c , Item_r.Silver_price_c)

		KDM Charge	Formula (Return Type:Currency) (Decimal=0) <div style="border: 1px solid black; padding: 2px; display: inline-block;">Item_r.KDM_c</div>
		Making Charges	Formula (Return Type:Currency) (Decimal=2) <div style="border: 1px solid black; padding: 2px; display: inline-block;">Item_r.Making_Charges_c</div>
		Stones/other price	Formula (Return Type:Currency) (Decimal=2) <div style="border: 1px solid black; padding: 2px; display: inline-block;">Item_r.Stone_other_price_c</div>
		Total Amount	Formula (Return Type:Currency) (Decimal=0) <div style="border: 1px solid black; padding: 2px; display: inline-block;">Amount_c + KDM_Charge_c + Stones_other_price_c + Making_Charges_c</div>
Billing			

Schema Builder

Schema Builder is a powerful tool within Salesforce that allows you to visualise, explore, and design the relationships between objects in your Salesforce organisation. It provides a graphical representation of the data model, making it easier to understand the structure and connections between different objects.

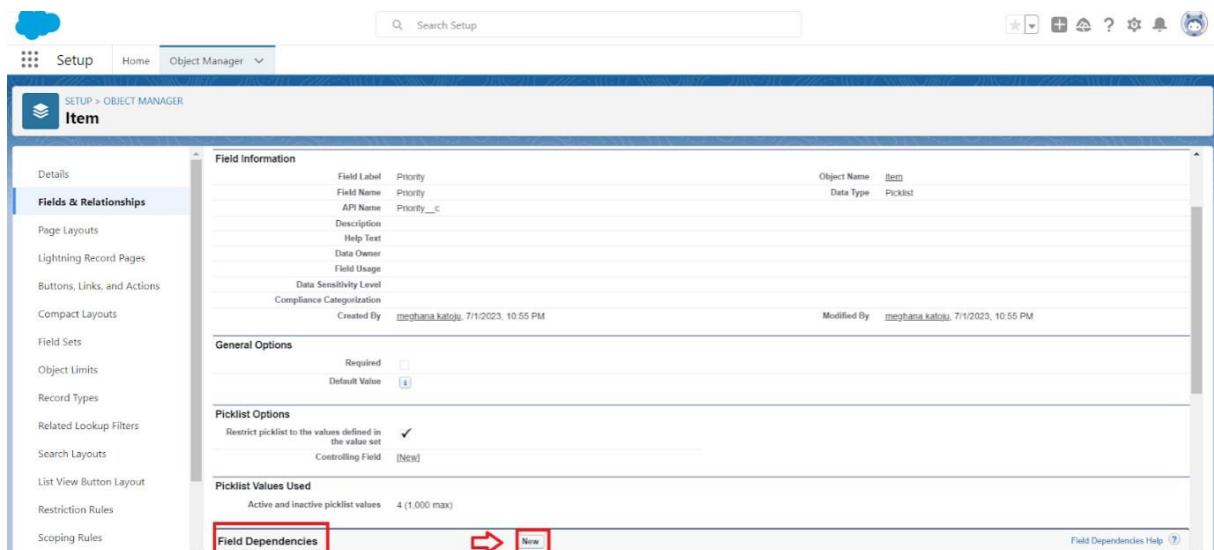


Creating the Field Dependencies

Use case:

Field Dependencies are used to create relationships between fields within an object. They allow you to control the visibility and availability of fields based on the values selected in other fields.

1. Go to setup >> click on Object Manager >> type object name(Item) in quick find bar >> click on the object.
2. Click on Fields & Relationships and click on the Priority field.



3. Search for Field Dependencies and click on New.

Select Controlling Field as “Priority” and Depending field as “Expected Days of Return” >> Continue.

New Field Dependency

Create a dependent relationship that causes the values in a picklist or multi-select picklist to be dynamically filtered based on the value selected by the user in another field.

- The field that drives filtering is called the “controlling field.” Standard and custom checkboxes and picklists with at least one and less than 300 values can be controlling fields.
- The field that has its values filtered is called the “dependent field.” Custom picklists and multi-select picklists can be dependent fields.

Step 1. Select a controlling field and a dependent field. Click Continue when finished.

Step 2. On the following page, edit the filter rules that control the values that appear in the dependent field for each value in the controlling field.

Select the “Expected Days of Return” values of related Priority values and Click on Include Values >> Save.

Creating the validation rule

Creating the validation rule for Postal Code field in Jewel Customer object

Note : check whether the fields mentioned in the formula field are created or not , if not go to activity 10 and create those fields mentioned in Jewel Customer object.

1. Go to setup >> click on Object Manager >> type object name(Jewel Customer) in quick find bar>> click on the object.
2. Click on the validation rule >> click New.

- Enter the Rule name as “Postal Code “.
- Insert the Error Condition Formula as :
 - AND(
 - OR(
 - LEN(Zip_Postal_code_c) <> 6,
NOT(REGEX(Zip_Postal_code_c, "^[0-9]{6}\$"))),
 - NOT(ISBLANK(Zip_Postal_code_c))
 -)
 -)

SETUP > OBJECT MANAGER
Jewel Customer

Details Fields & Relationships Page Layouts Lightning Record Pages Buttons, Links, and Actions Compact Layouts Field Sets Object Limits Record Types Related Lookup Filters Restriction Rules Scoping Rules Triggers Flow Triggers Validation Rules

Validation Rule Edit

Rule Name: **Postal Code** Active: 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 ▾

```
AND(
  OR(
    LEN( Zip_Postal_code_c ) <> 6,
    NOT(REGEX(Zip_Postal_code_c, "^[0-9]{6}$"))
  ),
  NOT(ISBLANK(Zip_Postal_code_c))
)
```

Check Syntax No errors found

Error Message

Example: Discount percent cannot exceed 30%
This message will appear when Error Condition formula is true

Error Message: **Must contain 6 digits**

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 **Zip/Postal code**

Save Save & New Cancel

- Enter the Error Message as “Must contain 6 digits”, select the Error location as Field and select the field as “Zip/Postal code”, and click Save.

NOTE:

Create One more Validation rule for Jewel Customer object.

0. Enter Rule name as “ValidationRuleForJewelCustomerObject “.
1. Insert the Error Condition Formula as : -
OR(ISBLANK(City_c), ISBLANK(Country_c),ISBLANK(Phone_c),ISBLANK(State_c),ISBLANK(Street_c))
2. Enter the Error Message as “Please fill Required fields”, select the Error location as Top of Page and click Save.

Create Validation rule for Item object.

1. Enter Rule name as “ValidationRuleFor Item“.
2. Insert the Error Condition Formula as :-
OR(ISBLANK(Amount_c), ISBLANK(Customer_Name_c) ,ISBLANK(Gold_price_c),ISBLANK(KDM_c),ISBLANK(Ornament_c),ISBLANK(Percentage_c),ISBLANK(Making_Charges_c),ISBLANK(Prices_c),ISBLANK(Stone_weight_c),ISBLANK(Silver_price_c),ISBLANK(Stone_other_price_c),ISBLANK(Stone_weight_c),ISBLANK(Weight_c))
3. Enter the Error Message as “Please fill Required fields”, select the Error location as Top of Page and click Save.

Profiles

A profile is a group/collection of settings and permissions that define what a user can do in salesforce. Profile controls “Object permissions, Field permissions, User permissions, Tab settings, App settings, Apex class access, Visualforce page access, Page layouts, Record Types, Login hours & Login IP ranges. You can define profiles by the user's job function. For example System Administrator, Developer, Sales Representative.

Types of profiles in salesforce

1. Standard profiles:
By default salesforce provides below standard profiles.
 - Contract Manager
 - Read Only
 - Marketing User
 - Solutions Manager
 - Standard User
 - System Administrator.

We cannot deleted standard ones

Each of these standard ones includes a default set of permissions for all of the standard objects available on the platform.

2. Custom Profiles:
Custom ones defined by us.
They can be deleted if there are no users assigned with that particular one.

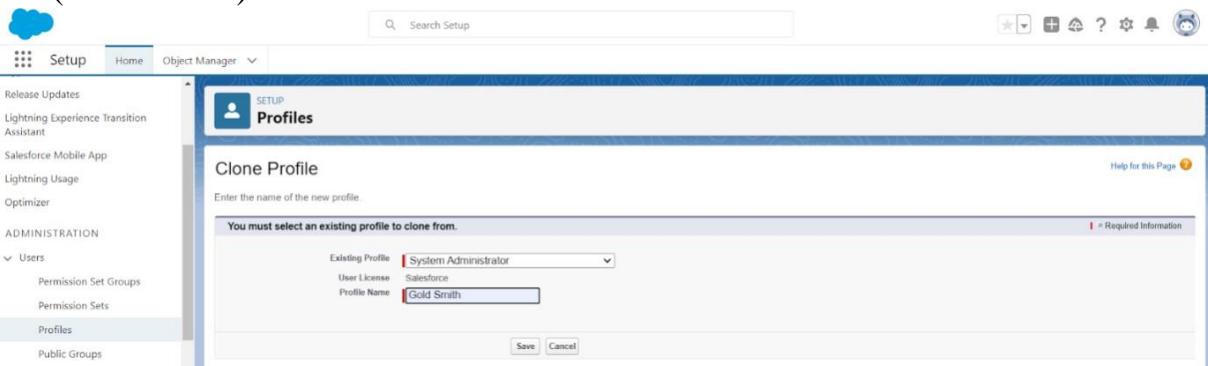
Use Case:

Great work Admin, you have done so good till now. The GoldSmith wants to differentiate the users based on their functionalities, position and based on this those users need to have the minimum access to the database object in the organisation. Now it's time to use your Admin skills to focus on the users, their functionality and position in the organisation in order to achieve the Goldsmith Smith requirements.

Gold Smith Profile

To create a new profile:

- Go to setup >> type profiles in quick find box >>click on profiles
?clone the desired profile (System Administrator) >> enter profile name (Gold Smith) >> Save.



- While still on the profile page, then click Edit.
- Scroll down to Custom Object Permissions and Give access permissions

	Basic Access	Data Administration					
		Read	Create	Edit	Delete	View All	Modify All
Assets	<input type="checkbox"/>						
Asset Services	<input type="checkbox"/>						
Billings	<input checked="" type="checkbox"/>						
Book1	<input type="checkbox"/>						
Book2	<input type="checkbox"/>						
Bot Commands	<input type="checkbox"/>						
Brokers	<input type="checkbox"/>						
Buyers	<input type="checkbox"/>						
Candidates	<input type="checkbox"/>						
Customer Orders	<input checked="" type="checkbox"/>						
	Basic Access	Data Administration					
		Read	Create	Edit	Delete	View All	Modify All
Items	<input checked="" type="checkbox"/>						
Jewel Customers	<input checked="" type="checkbox"/>						
Job Applications	<input type="checkbox"/>						
Job Postings	<input type="checkbox"/>						
Job Posting Sites	<input type="checkbox"/>						
Positions	<input type="checkbox"/>						
Prices	<input checked="" type="checkbox"/>						
Projects	<input type="checkbox"/>						
ProjectTasks	<input type="checkbox"/>						
Properties	<input type="checkbox"/>						

for Jewel Customer,Item,CustomerOrder,Prices,Billings .

- Scroll down and Click on Save.

Worker Profile

- Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Salesforce Platform User) >> enter profile name () >> Save.
- While still on the profile page, then click Edit.
- Scroll down to Custom Object Permissions and Give access permissions for Items, Price and Customer Order objects.
- Scroll down and Click on Save.

Roles

A role in Salesforce defines a user's visibility access at the record level. Roles may be used to specify the types of access that people in your Salesforce organisation can have to data.

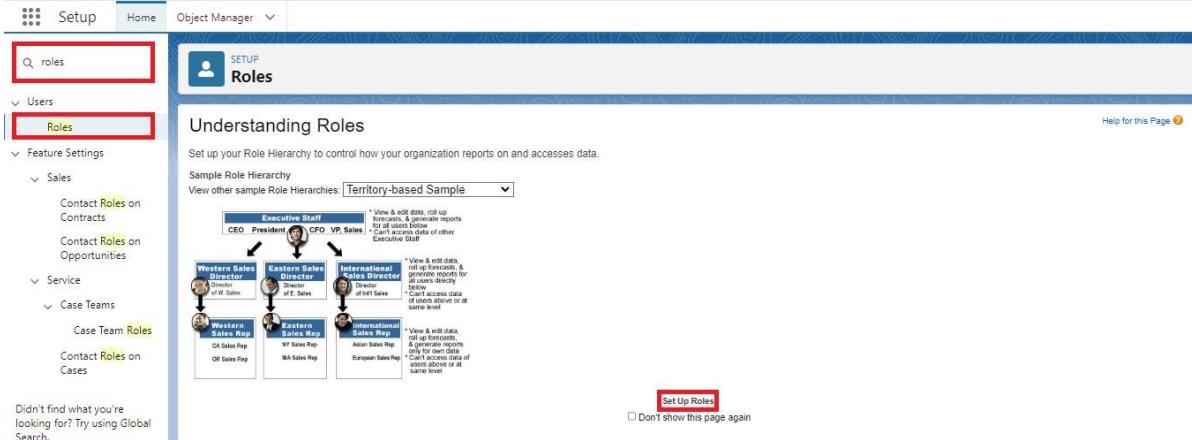
Simply put, it describes what a user could see within the Salesforce organisation.

Use Case:

You have successfully fulfilled the 1st requirement i.e., differentiating the users based on the functionality. Now comes the 2nd task of differentiating the users based on their position, using your excellent admin skills and expanding the custom roles for the organisation and assigning it to the users.

Creating Gold Smith Role

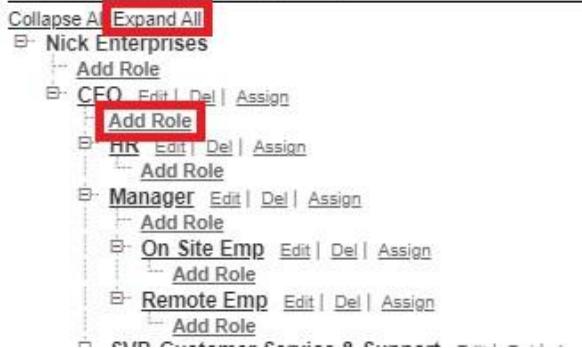
1. From setup ,Go to quick find >> Search for Roles >> click on set up



roles.

0. Click on Expand All and click on add role under whom this role works.

Your Organization's Role Hierarchy



- Give Label as “Gold Smith” and Role name gets auto populated. Check to whom this role (Gold Smith) reports. Then click on Save.

Role Edit
Gold Smith

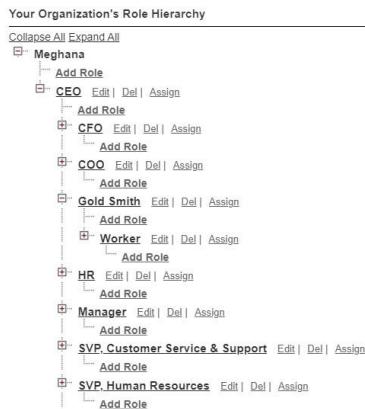
Label	Gold Smith
Role Name	Gold_Smith
This role reports to	CEO
Role Name as displayed on reports	Gold Smith

Save Save & New Cancel

Create one more role as Worker which reports to Gold Smith.

Creating the Role Hierarchy

You can build on the existing role hierarchy shown on this page. To insert a new role, click Add Role.



Users

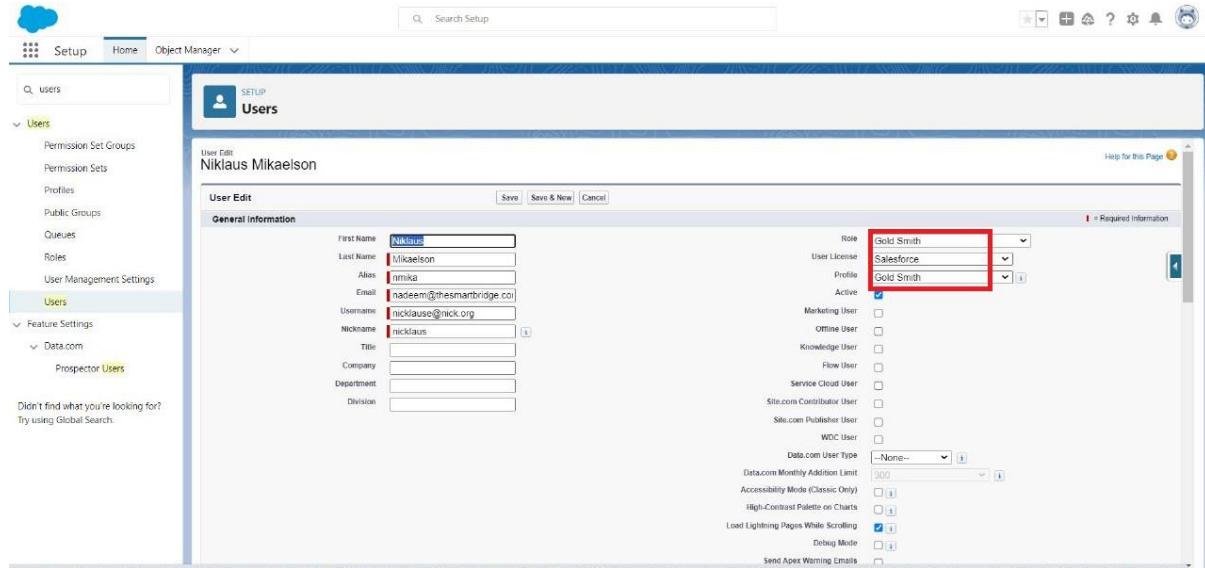
A user is anyone who logs in to Salesforce. Users are employees at your company, such as sales reps, managers, and IT specialists, who need access to the company's records. Every user in Salesforce has a user account. The user account identifies the user, and the user account settings determine what features and records the user can access.

Every user in Salesforce has a user account. The user account identifies the user, and the user account settings determine what features and records the user can access. Each user account contains at least the following:

- Username
- Email Address
- User's First Name (optional)
- User's Last Name
- Alias
- Nickname
- Licence
- Profile
- Role (optional)

Create User

1. Go to setup >> type users in quick find box >> select users >> click New user.
2. Fill in the fields
 1. First Name : Niklaus
 2. Last Name : Mikaelson
 3. Alias : Give a Alias Name
 4. Email id : Give your Personal Email id
 5. Username : Username should be in this form: text@text.text
 6. Nick Name : Give a Nickname
 7. Role : Gold Smith
 8. User licence : Salesforce



9. Profiles: Gold Smith

Save.

Create User

1. Go to setup >> type users in quick find box >> select users >> click New user.
2. Fill in the fields
 - First Name : Kol
 - Last Name : Mikaelson
 - Alias : Give a Alias Name
 - Email id : Give your Personal Email id
 - Username : Username should be in this form: text@text.text
 - Nick Name : Give a Nickname
 - Role : Worker
 - User licence : Salesforce Platform
 - Profiles : Worker
0. Save.

Note:

Create two more users as mentioned in activity 2 using the same profile.

Page layouts

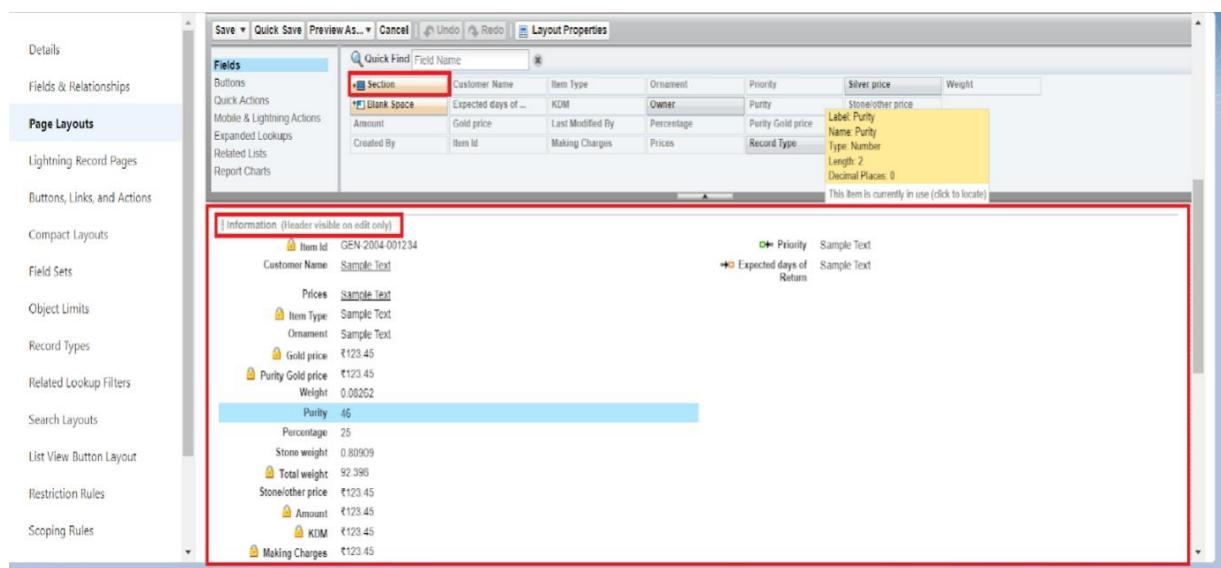
Page Layout in Salesforce allows us to customise the design and organise detail and edit pages of records in Salesforce. Page layouts can be used to control the appearance of fields, related lists, and custom links on standard and custom objects' detail and edit pages.

Use Case:

Hurray!! you have completed the data model structure for your organisation but while looking at the detailed and edit pages it seems to be so clumsy, so decide to organise the page in a pleasant way for the sake of good and pleasant appearance and assemble all different kinds of information in different sections in order.

To Create a Gold Page layout

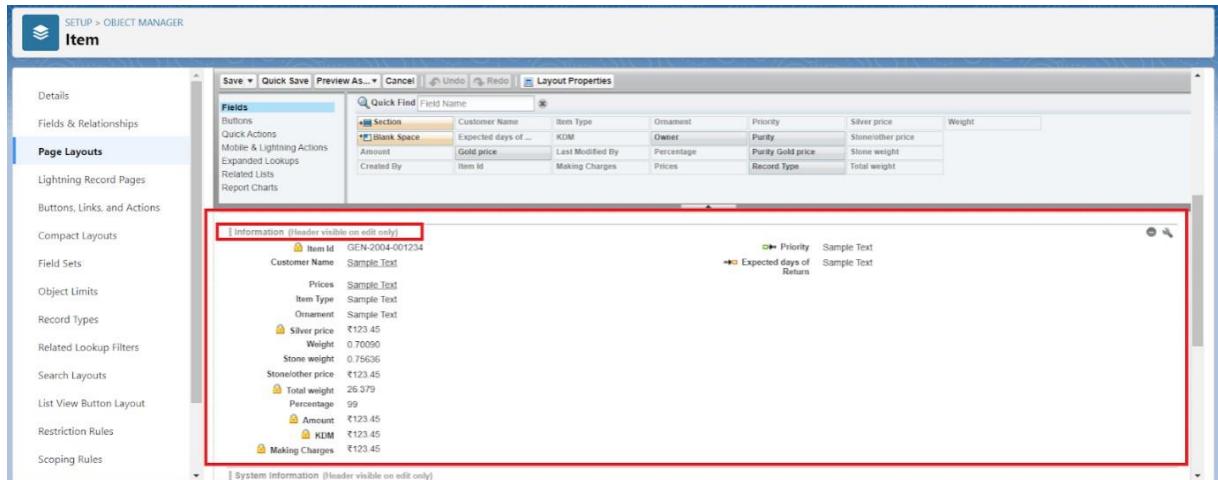
1. Go to Setup >> Click on Object Manager >> Search for the object (Item) >>From drop down click on Edit.
2. Click on Page layout >> Click on New.
3. Give Page layout Name as “Page Layout for Gold” and click on Save and New.
4. Arrange the field as shown in the Information Section ,remove fields which are related to Silver and click Ok.



5. Click Save.
6. Make sure your page layout looks like the picture above.

To Create a Silver Page layout

1. Go to Setup >> Click on Object Manager >> Search for the object (Item) >>From drop down click on Edit.
2. Click on Page layout >> Click on New.
3. Give Page layout Name as “Page Layout for Silver” and click on Save.
4. Arrange the field as shown in the Information Section ,remove fields which



are related to Gold and click Ok.

Record Types

Record Types are a way of grouping many records of one type for that object. These can be applied to any standard or custom object, and allow you to have a different page layout, fields, required fields, and picklist values. Record types allow administrators to create a different page layout with custom picklist fields and values for the same business process and various business processes.

Use Case:

All things done for the organisation. But some of the organisations feel it difficult to fill up all the details while creating a record, so GoldSmith assigned you a task to create different forms for Gold and Silver records based on their mode of work. As an Admin, you know how to achieve this.

1. Go to setup >> click on Object Manager >> type object name(Item) in quick find bar? click on the object.
2. Click on the Record Types >> click New.
3. Select Existing Record as “Master”,Record type Label as “Gold”,Description as “Gold items information”.
4. Uncheck for “Make Available”.
5. Scroll down and check for the Gold Smith,Worker JW & System Administrator profile and click on Next.

Customer Portal Manager Standard		<input type="checkbox"/>
External Apps Login User		<input type="checkbox"/>
External Identity User		<input type="checkbox"/>
Force.com - App Subscription User	Gold (Default) Silver	<input type="checkbox"/>
Force.com - Free User	Gold (Default) Silver	<input type="checkbox"/>
Gold Partner User	Gold (Default) Silver	<input type="checkbox"/>
Gold smith	Gold (Default) Silver	<input checked="" type="checkbox"/>
High Volume Customer Portal		<input type="checkbox"/>
High Volume Customer Portal User		<input type="checkbox"/>
HR	Gold (Default) Silver	<input type="checkbox"/>
HR Recruiter	Gold (Default) Silver	<input type="checkbox"/>
Identity User	Gold (Default) Silver	<input type="checkbox"/>
J Worker1	Gold (Default) Silver	<input checked="" type="checkbox"/>
J Worker2	Gold (Default) Silver	<input checked="" type="checkbox"/>
J WORKER3	Gold (Default) Silver	<input checked="" type="checkbox"/>
Manager	Gold (Default) Silver	<input type="checkbox"/>
Marketing User	Gold (Default) Silver	<input type="checkbox"/>
Minimum Access - Salesforce	Gold (Default) Silver	<input type="checkbox"/>
Partner App Subscription User	Gold (Default) Silver	<input type="checkbox"/>
Partner Community Login User		<input type="checkbox"/>
Partner Community User		<input type="checkbox"/>
Read Only		<input type="checkbox"/>
s1		<input type="checkbox"/>
Salesforce API Only System Integrations		<input type="checkbox"/>
Sales User		<input type="checkbox"/>
Sales User.		<input type="checkbox"/>
Silver Partner User		<input type="checkbox"/>
Solution Manager		<input type="checkbox"/>
Standard Platform User		<input type="checkbox"/>
Standard User		<input type="checkbox"/>

6. Select “Apply a different layout for each profile”, and change page layout to “Page Layout for Gold” for Gold Smith, Worker and System Administrator ?

Force.com - Free User	Item Layout
Gold Partner User	Item Layout
Gold smith	Page layout for Gold
High Volume Customer Portal	Item Layout
High Volume Customer Portal User	Item Layout
HR	Item Layout
HR Recruiter	Item Layout
Identity User	Item Layout
Manager	Item Layout
Marketing User	Item Layout
Minimum Access - Salesforce	Item Layout
Partner App Subscription User	Item Layout
Partner Community Login User	Item Layout
Partner Community User	Item Layout
Read Only	Item Layout
s1	Item Layout
Salesforce API Only System Integrations	Item Layout
Sales User	Item Layout
Sales User.	Item Layout
Silver Partner User	Item Layout
Solution Manager	Item Layout
Standard Platform User	Item Layout
Standard User	Item Layout

Save & New

HR	Item Layout ▾
HR Recruiter	Item Layout ▾
Identity User	Item Layout ▾
Manager	Item Layout ▾
Marketing User	Item Layout ▾
Minimum Access - Salesforce	Item Layout ▾
Partner App Subscription User	Item Layout ▾
Partner Community Login User	Item Layout ▾
Partner Community User	Item Layout ▾
Read Only	Item Layout ▾
s1	Item Layout ▾
Salesforce API Only System Integrations	Item Layout ▾
Sales User	Item Layout ▾
Sales User.	Item Layout ▾
Silver Partner User	Item Layout ▾
Solution Manager	Item Layout ▾
Standard Platform User	Item Layout ▾
Standard User	Item Layout ▾
Support User	Item Layout ▾
Support User.	Item Layout ▾
System Administrator	Item Layout ▾
Work.com Only User	Item Layout ▾
Worker	Page layout for Gold ▾

Activity 2: Create another Record Type with name “Silver” following the steps from Activity1.

Note: Use page layout for Silver.

Permission sets

A standard permission set consists of a group of common permissions for a particular feature associated with a permission set licence. Using a standard permission set saves you time and facilitates administration because you don't need to create the custom permission set.

Creating permission set

A permission set is a collection of settings and permissions that give users access to various tools and functions. Permission sets extend users' functional access without changing their profiles. Users can have only one profile but, depending on the Salesforce edition, they can have multiple permission sets.

The screenshot shows the Salesforce Setup interface. The top navigation bar includes 'Setup', 'Home', 'Object Manager', and a search bar. The left sidebar has sections for 'Q. Permission sets', 'Users', and 'Permission Sets'. The main content area is titled 'Permission Sets' and contains a table of existing permission sets like 'Buyer', 'Buyer Manager', etc. A red box highlights the 'New' button at the top left of the table.

1. Go to setup >> type “permission sets” in quick search >> select permission sets >> New.

2. Enter the label name as “Per to Worker”, API will be auto populated ? save.

This screenshot shows the 'Enter permission set information' dialog. It has fields for 'Label' (set to 'Per to Worker') and 'API Name' (set to 'Per_to_Worker'). The 'Save' button is highlighted with a red box. The dialog also includes 'Description', 'Session Activation Required', and 'License' fields, along with 'Save' and 'Cancel' buttons at the bottom.

3. Under Apps Select object settings.

Apps

<u>Assigned Apps</u> Settings that specify which apps are visible in the app menu
<u>Assigned Connected Apps</u> Settings that specify which connected apps are visible in the app menu
<u>Object Settings</u> Permissions to access objects and fields, and settings such as tab availability
<u>App Permissions</u> Permissions to perform app-specific actions, such as "Manage Call Centers"
<u>Apex Class Access</u> Permissions to execute Apex classes
<u>Visualforce Page Access</u> Permissions to execute Visualforce pages
<u>External Data Source Access</u> Permissions to authenticate against external data sources
<u>Flow Access</u> Permissions to execute Flows
<u>Named Credential Access</u> Permissions to authenticate against named credentials
<u>Custom Permissions</u> Permissions to access custom processes and apps
<u>Custom Metadata Types</u> Permissions to access custom metadata types
<u>Custom Setting Definitions</u> Permissions to access custom settings

4. Click on Items object ? click on Edit ? underItem:Record Type

Permission Set Overview > Object Settings > Items

Items

Tab Settings

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

Item: Record Type Assignments

Record Types	Assigned Record Types
Gold	<input checked="" type="checkbox"/>
Silver	<input checked="" type="checkbox"/>

Object Permissions

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

Field Permissions

Assignments,enableGold,Silver ? Object permission check for read ,edit and create.

0. Click on Save.
0. After saving the permission click on the Manage assignment
0. Now click on the Add Assignment.

Select Users to Assign

All Users ▾

Full Name	Alias	Username	Role	Active	Profile
Chatter Expert	Chatter	chatty.00d5i000003ksyzea4.t4i5wtjeybt4@chatter.salesforce.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Chatter Free User
Integration User	integ	integration@00d5i000003ksyzea4.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Analytics Cloud Integration User
Mani deepak	mdeep	manideepak143@gmail.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Worker
Megha Katoju Site Guest User	guest	megha_katoju@00d5i000003ksyzea4.org.force.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Megha Katoju Profile
Meghana Katoj Site Guest User	guest	meghana_katoj@00d5i000003ksyzea4.org.force.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Meghana Katoj Profile

Cancel Next

8. Now select the users which you have created in user milestone, using Worker profile and click on Next ? Assign? Done.

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
Mani deepak	Worker	Worker	<input checked="" type="checkbox"/>	Salesforce Platform	Never Expires

Cancel Back Assign

Trigger

Use Case:

Trigger and Trigger handler is designed to handle scenarios where we used to update the "Paid Amount" field on a custom object called "Billing" based on the value in a field named "Paying Amount" during both record insertion and update operations. It Calculates and updates the "Paid Amount" field based on the existing "Paid Amount" and the new "Paying Amount" during record updates. This approach ensures that the "Paid Amount" accurately reflects the payments made by customers and provides a history of changes to the "Paid Amount" over time.

Trigger :

A trigger is a piece of Apex code that automatically runs before or after specific events, like record insertion, update, or deletion. Triggers are used to customise and automate actions in response to these events.

Create a Trigger Handler class

Trigger handler:

A trigger handler is a design pattern that organises trigger logic into separate classes. This helps in keeping code organised, reusable, and easier to maintain. The trigger handler class contains methods that handle the specific logic for different trigger events, improving code structure and readability. This approach is particularly useful for complex triggers or projects with multiple triggers, as it promotes modular coding practices and reduces the chances of code duplication.

CODE:

```
public class UpdatePaidAmountTriggerHandler {  
    public static void handleBeforeInsert(List<Billing__c> newBillings) { for (Billing__c billing : newBillings) {  
        billing.Paid_Amount__c = billing.Paying_Amount__c;  
    } }  
  
    public static void handleBeforeUpdate(Map<Id, Billing__c> oldBillingsMap, List<Billing__c> updatedBillings) {  
        for (Billing__c billing : updatedBillings) {  
            Billing__c oldBilling =  
                oldBillingsMap.get(billing.Id); Decimal  
                oldPaidAmount = oldBilling.Paid_Amount__c;  
                billing.Paid_Amount__c = oldPaidAmount + billing.Paying_Amount__c;  
        } }  
}
```

Create the trigger

CODE:

```
triggerUpdatePaidAmountTrigger on Billing__c (before insert,  
before update) { if (Trigger.isInsert) {  
    UpdatePaidAmountTriggerHandler.handleBeforeInsert(Trigger.new);  
} else if (Trigger.isUpdate) {  
    UpdatePaidAmountTriggerHandler.handleBeforeUpdate(Trigger.oldMap,  
    Trigger.new);  
}
```

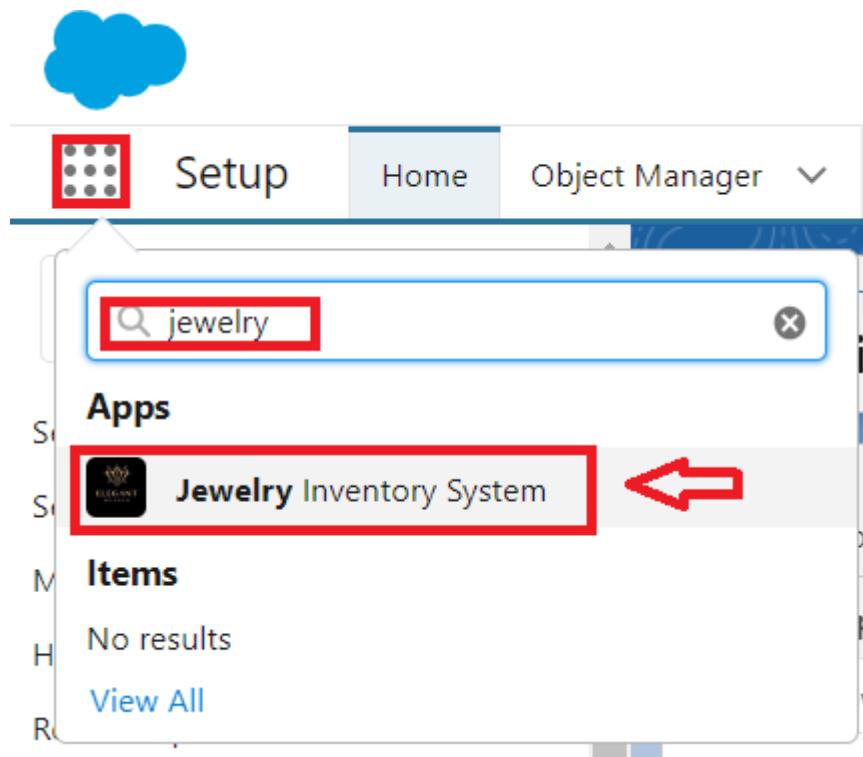
User Adoption

Use Case:

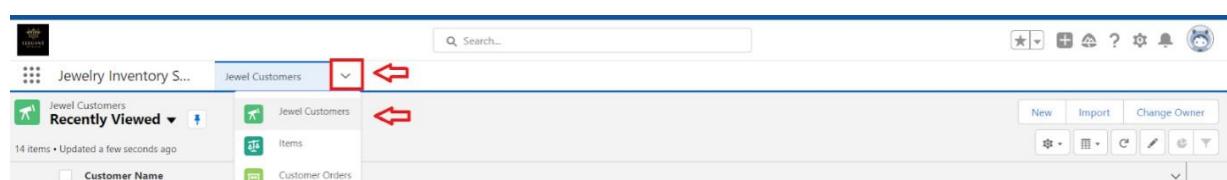
As a new Administrator, you perform user management tasks like creating and editing users, resetting passwords, granting permissions, configuring data access, and much more. In this unit, you will learn about users and how you add users to your Salesforce org.

Create a Record (Jewel Customer)

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



3. Click on Drop Down and Click on the Jewel Customer tab.



4. Click New.

5. Fill the Details and click on Save.

View a Record (Jewel Customer)

1. Click on App Launcher on the left side of the screen.

2. Search Jewelry Inventory System & click on it.
3. Click on the Jewel Customer Tab.
4. Click on any record name. you can see the details of the Jewel Customer.

Delete a Record(Jewel Customer)

1. Click on App Launcher on the left side of the screen.
2. Search Jewelry Inventory System & click on it.
3. Click on the Jewel Customer Tab.
4. Click on Arrow at right hand side on that Particular record.
5. Click delete.

Note:Create at least 10 records for each of the objects: Jewel Customer,Price,Item,Customer Order and Billing.

Reports

Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics.

Types of Reports in Salesforce

1. Tabular
2. Summary
3. Matrix
4. Joined Reports

Use Case:

The GoldSmith of an organisation wants to have a brief data on Gold Items,SilverItems,Customer Orders and Billings. So he can have a clear picture of his organisation and be able to make any decisions required based on this data. So he calls you on this task and wants you to represent the data in an appropriate way.

Let's create a Report.

Create Report

1. Go to the app >> click on the reports tab
2. Click New Report.

The screenshot shows the Jewelry Inventory Software interface. The top navigation bar includes a search bar and various icons. Below it, a sidebar on the left lists categories like 'Recent', 'Reports', 'Folders', and 'Favorites'. The main area displays a table of reports with columns for 'Report Name', 'Folder', 'Created By', 'Created On', and 'Subscribed'. A red box highlights the 'Reports' category in the sidebar, and another red box highlights the 'Reports' button in the top right of the main area.

3. Select report type from category or from report type panel or from search panel ? click on start report.

The screenshot shows the Report Builder interface with a 'Create Report' dialog open. The left sidebar shows categories like 'Recently Used' and 'All'. The main area has a search bar labeled 'Select a Report Type' with 'PRICE' typed in. Below it, a table lists report types with columns for 'Report Type Name', 'Category', and 'Standard'. A red box highlights the 'All' button in the sidebar, another red box highlights the 'PRICE' search term in the search bar, and a third red box highlights the 'Prices' report type in the list.

4. Customise your report

The screenshot shows the Report Builder interface with a 'New Prices Report' configuration. The left sidebar shows 'REPORT' and 'New Prices Report'. The main area has sections for 'Outline' and 'Filters'. Under 'Outline', there are 'Groups' (with 'GROUP ROWS' selected) and 'Columns' (with 'Add column...'). A red box highlights the 'Add column...' button in the 'Columns' section. The top right of the screen has buttons for 'Run' (highlighted with a red box), 'Save & Run', 'Save', 'Close', and 'Update Preview Automatically'.

- Add fields from the left pane as shown below.

5. Save or run it.

REPORT ▾

New Prices Report ▾ Prices

Fields

Groups

GROUP ROWS

Add group...

Columns

Add column...

Price: Price

Gold price

Silver price

Previews a limited number of records. Run the report to see everything.

	Price: Price	Gold price	Silver price
1	p-022	₹60,000.00000	₹71,000.00000
2	p-021	₹63,000.00000	₹72,000.00000
3	p-027	₹62,350.00000	₹70,200.00000
4	p-029	₹58,700.00000	₹69,000.00000
5	p-030	₹66,000.00000	₹78,000.00000
6	p-026	₹62,000.00000	₹70,000.00000
7	p-025	₹58,000.00000	₹69,000.00000
8	p-028	₹59,900.00000	₹73,000.00000
9	p-024	₹62,000.00000	₹73,000.00000
10	p-023	₹58,000.00000	₹69,000.00000
11		₹609,950.00000	₹714,200.00000

Update Preview Automatically

Note: Reports may get varied from the above pictures as the data might be different.

Reports

1. Create a report with report type: “Item with Billings”.
2. Create a report with report type: “Billings with item and Customer order”.

Dashboards

Dashboards help you visually understand changing business conditions so you can make decisions based on the real-time data you've gathered with reports. Use dashboards to help users identify trends, sort out quantities, and measure the impact of their activities. Before building, reading, and sharing dashboards, review these dashboard basics.

Use Case:

As an Admin for the organisation you keep pushing yourself to reach out the business requirements to take the organisation to peak heights and all your superiors are very much impressed with your efforts and work dedication. In addition with reports you make an ease for the GoldSmith in viewing the reports with data visualisation. So he doesn't have to search for the data he wants to check.

Create Dashboard

1. Go to the app >> click on the Dashboards tabs.

Employee Manage...

Home Employees Assets Asset Services Projects ProjectTasks Reports Dashboards

Dashboards

Recent

0 Items

DASHBOARDS

Recent

Created by Me

Private Dashboards

All Dashboards

Search recent dashboards... New Dashboard New Folder

2. Give a Name and click on Create.

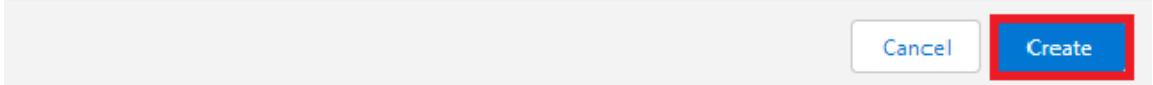
New Dashboard

* Name
Dashboard 1

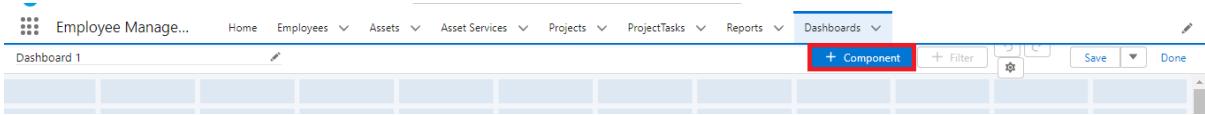
Description

Folder
Private Dashboards Select Folder

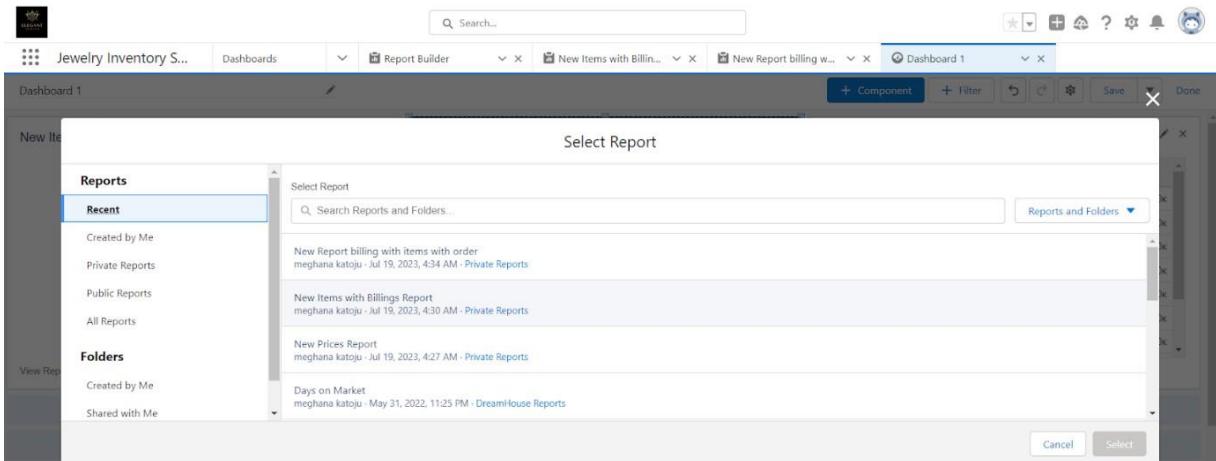
Cancel Create



3. Select add component.

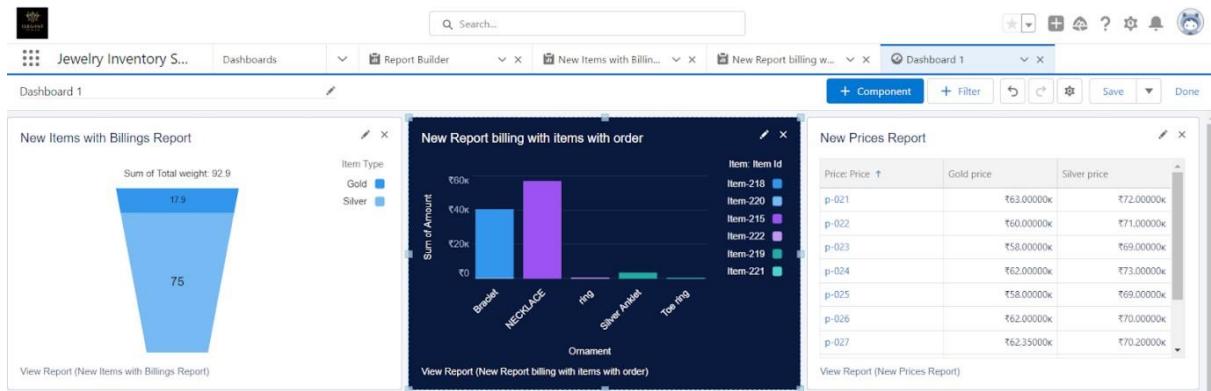


4. Select a Report and click on select.



5. Click Add then click on Save and then click on Done.

Activity 2: Create another Dashboard as we discussed in activity 1.



Flows

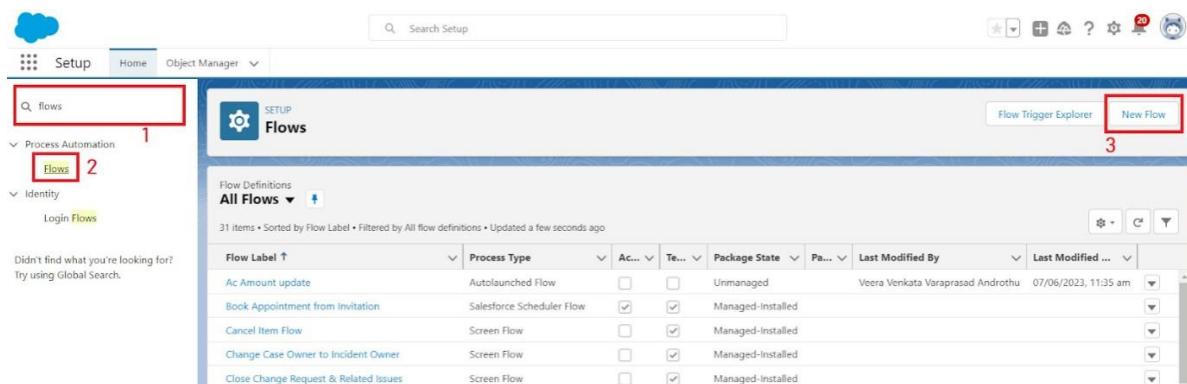
In Salesforce, a flow is a powerful tool that allows you to automate business processes, collect and update data, and guide users through a series of screens or steps. Flows are built using a visual interface and can be created without any coding knowledge.

Use Case:

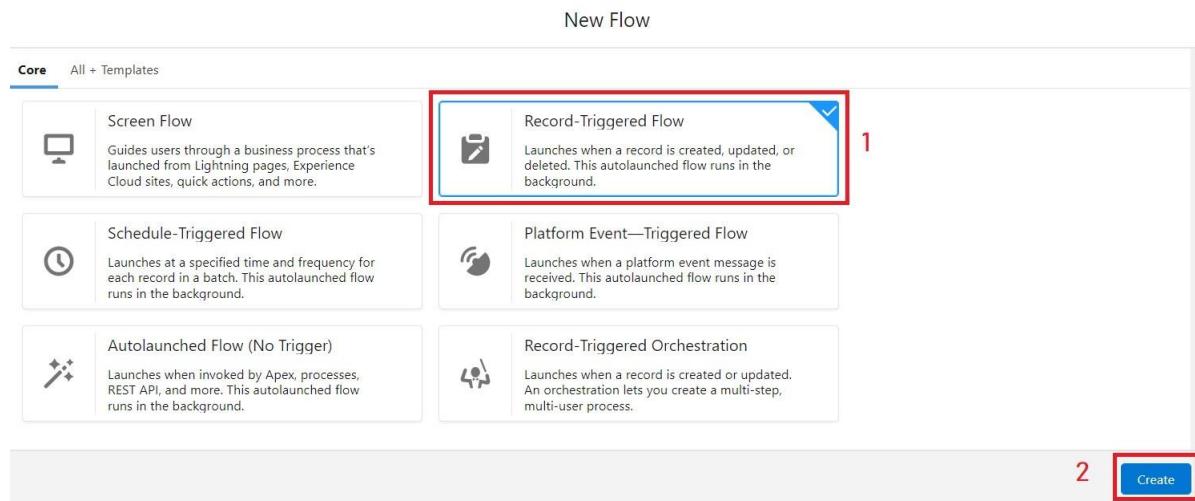
Flows, also known as Salesforce Flows or Visual Flows, are powerful declarative automation tools in Salesforce that allow users to create and manage complex business processes without the need for code. Flows are designed using a drag-and-drop interface, making them easy to use for both administrators and developers. They can be used for various automation tasks like email triggers including data entry, record updates, and guided user interactions.

Create a Flow

1. Go to setup >> type Flow in quick find box >> Click on the Flow and Select the New Flow.



2. Select the Record-triggered flow and Click on Create.



3. Select the Object as a “Billing” in the Drop down list.
4. Select the Trigger Flow when: “A record is Created or Updated”.
5. Select the Optimise the flow for: “Actions and Related Records” and Click on Done.

Configure Start

Select Object

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

* Object
Item

Configure Trigger

* Trigger the Flow When:

- A record is created
- A record is updated
- A record is created or updated
- A record is deleted

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

Cancel Done

6. Now change the mode from Auto-layout to free-form.
7. Now select the manger option in the toolbox, click New resource.
8. Select the resource type as text template.

The screenshot shows the Salesforce Flow Builder interface. The main window title is "Email - VS". On the left, there's a "Toolbox" with sections for "Elements" (selected) and "Manager". Under "ELEMENTS", there are "Text Templates (1)" and "Actions (1)". A "RESOURCES" section lists "Billing". The central area is titled "Configure Start" and contains three tabs: "Select Object", "Configure Trigger", and "Set Entry Conditions". The "Select Object" tab has a search bar and a dropdown for "Object" set to "Billing". The "Configure Trigger" tab shows "Trigger the Flow When:" with "A record is created or updated" selected (highlighted with a red box). The "Set Entry Conditions" tab has a note about minimizing unnecessary flow executions. At the bottom right of the dialog are "Cancel" and "Done" buttons, with "Done" highlighted with a red box. Below the dialog, the main workspace shows a "New Resource" card with a dropdown for "Resource Type". The dropdown menu is open, showing options: "Variable", "Constant", "Formula", "Text Template" (which is highlighted with a red box), and "Stage".

9. Enter the API name as “Email body”.

Edit Text Template

* API Name
EmailBody

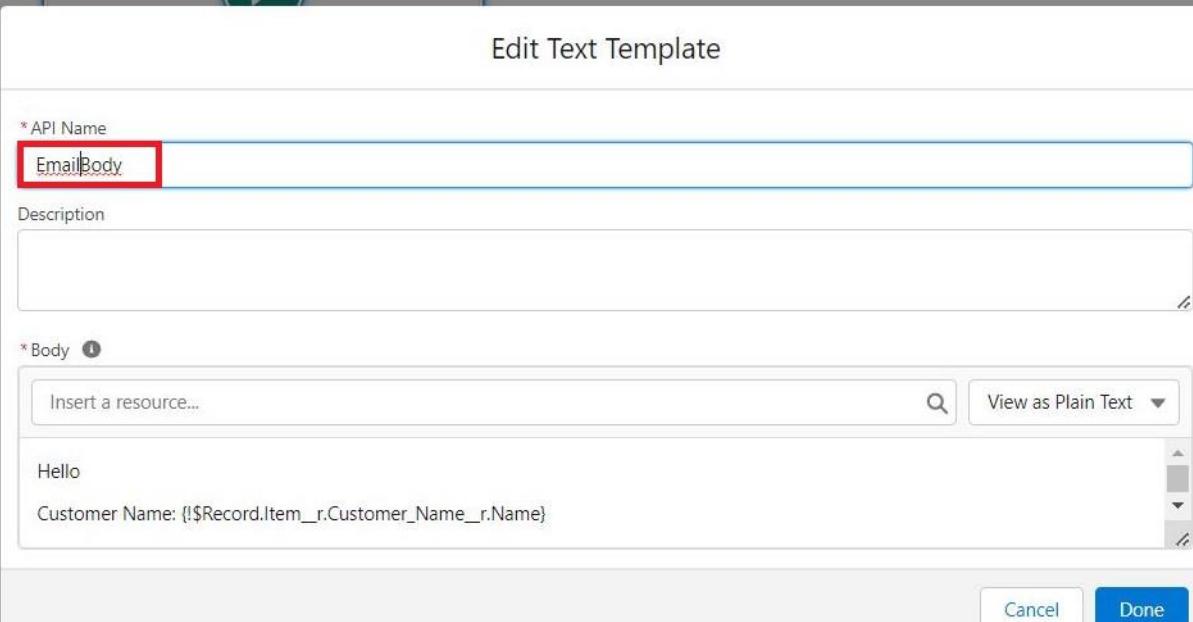
Description

* Body

Insert a resource...

Hello
Customer Name: {!\$Record.Item__r.Customer_Name__r.Name}

Cancel Done



10. Change the view as Rich Text ? View to Plain Text.

11. In the body field paste the syntax that is given below.

Hello

Customer Name: {!\$Record.Item__r.Customer_Name__r.Name}

Here are the details for the item you purchased with Jewellery

Inventory System Item Type: {!\$Record.Item__r.Item_Type_c}

Ornament:

{ !\$Record.Ornament_____c }

Weight: { !\$Record.Weight

c}grams Amount:

{ !\$Record.Amount_c }

12. Click done.

13. Now click on elements, and drag the action element into the preview pane.

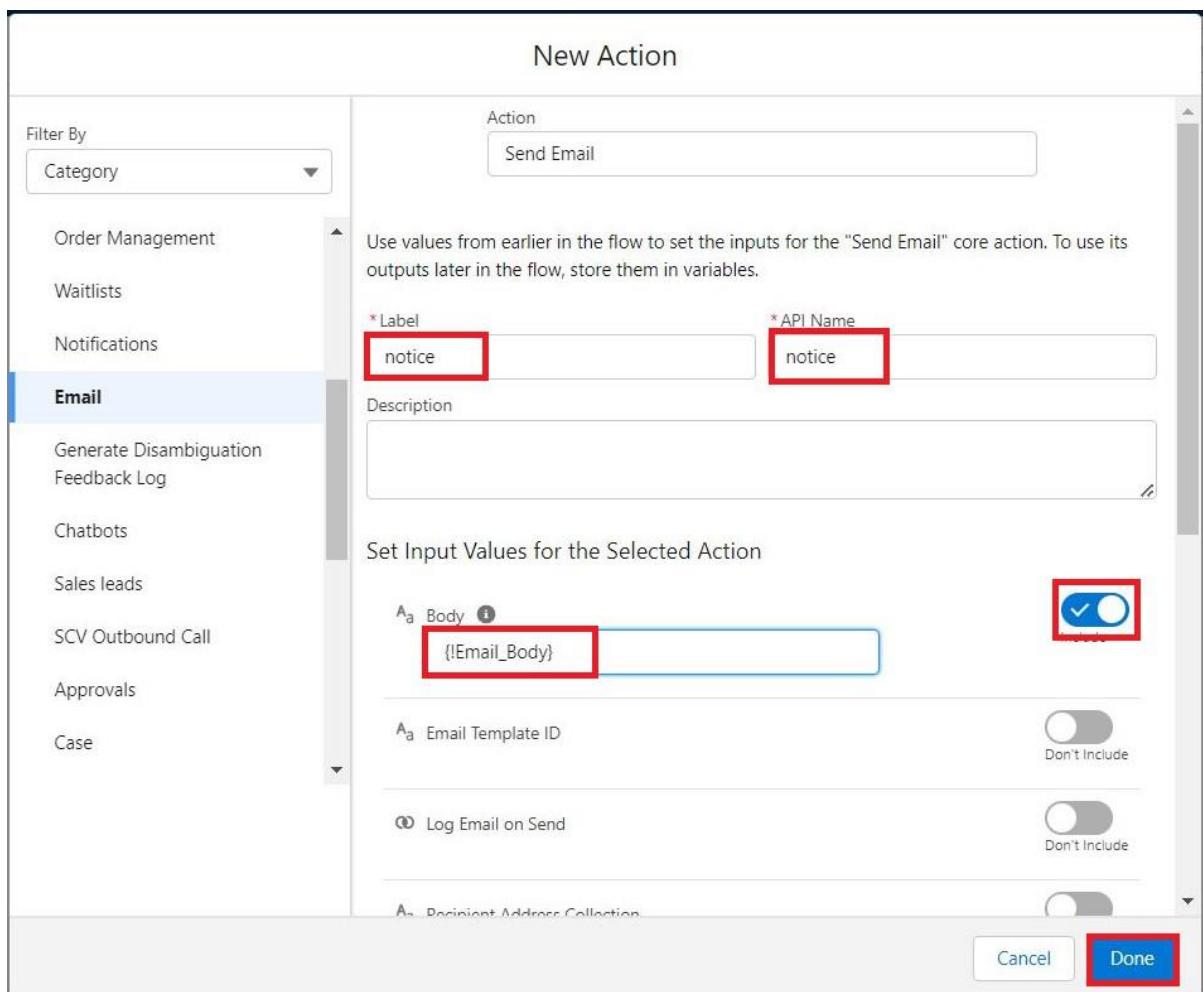
14. Their action bar will be opened in that search for “ send email ” and click on it.

15. Give the label name as “ notice”

16. API name will be auto populated.

17. Enable the body in set input values for the selected action.

18. Select the text template that was created.

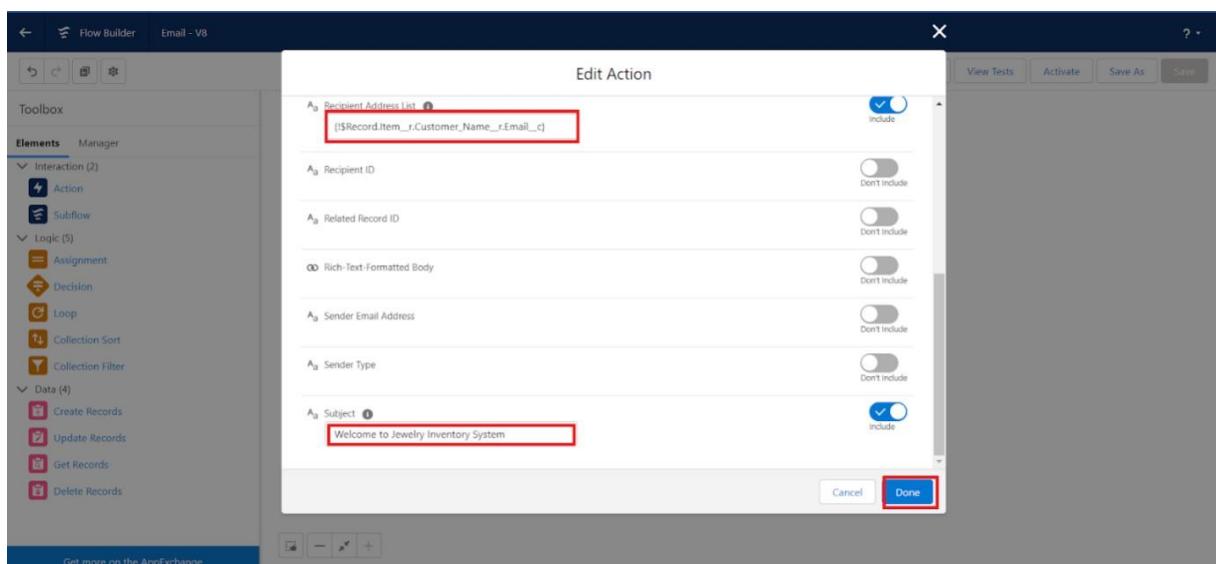


19. Include Recipient Address list, select the email form the record.

({ !\$Record.Item_r.Customer_Namer.Email_c })

20. Include the subject as “Welcome to Jewelry Inventory System ”.

21. Click done.



22. Now drag the path from the start to the action element.

23. Click on save. Given the Flow label , Flow Api name will be auto populated.

24. And click save, and click on activate.

