

# A CRM APPLICATION FOR LAPTOP RENTALS

CRM Application on Laptop rentals is about delivering the items to the customers by rental purpose. It leverages the power of customer relationship management (CRM) to enhance customer experiences, optimize store operations, and improve overall efficiency. Additionally to these, we also need to do an effective CRM i.e via communicating through email with the potential customers identified.

## 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 organized 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 verify account to activate your account. The email may take 5-10mins.
2. Click on Verify Account.
3. Give a password and answer a security question and click on change password.
4. Then you will redirect to your salesforce setup page.

## Object Creation

### 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:

1. Standard Objects: Standard objects are the kind of objects that are provided by salesforce.com such as users, contracts, reports, dashboards, etc.
2. 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.

**To Navigate to Setup page:**

Click on gear icon >> click setup.

**To create an object:**

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
2. On Custom object defining page:
3. Enter the label name, plural label name, click on Allow reports, Allow search.
4. Click on Save.

## Object Creation

**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:

1. Standard Objects: Standard objects are the kind of objects that are provided by salesforce.com such as users, contracts, reports, dashboards, etc.
2. 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.

**To Navigate to Setup page:**

Click on gear icon >> click setup.

**To create an object:**

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
2. On Custom object defining page:
3. Enter the label name, plural label name, click on Allow reports, Allow search.
4. Click on Save.

## Create Total Laptops Object

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

1) Enter the label name>> Total Laptops

2) Plural label name>> Total Laptops

3) Enter Record Name Label and Format

Record Name >>Total Laptops

Data Type >> Text

2. Click on Allow reports,Allow search and Track Field History,

3. Allow search >> Save.

## Create consumer Object

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

1) Enter the label name >> consumer

2) Plural label name >> consumer

3) Enter Record Name Label and Format

Record Name >> consumer\_name

Data Type >> Name

2. Click on Allow reports,Allow search and Track Field History,

3. Allow search >> Save.

## Create Laptop Bookings Object

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

1) Enter the label name >> Laptop Bookings

2) Plural label name >> Laptop Bookings

3) Enter Record Name Label and Format

Record Name >> Laptop Bookings

Data Type >> Name

2. Click on Allow reports,Allow search and Track Field History,
3. Allow search >> Save.

## Create Billing Process Object

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

1) Enter the label name >> Billing Process

2) Plural label name >> Billing Process

3) Enter Record Name Label and Format

Record Name >> Billing ProcessName

Data Type >> Name

2. Click on Allow reports,Allow search and Track Field History,
3. Allow search >> Save.

## 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:

#### 1. 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.

2. **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.

### **1. 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.

### **2. Lightning Component Tabs**

Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app.

### **3. 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.

## **Creating a Custom Tab**

### **To create a Tab:**

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

### **Activity 2: Creating Remaining Tabs**

Now create the Tabs for the remaining Objects, they are "consumer,Laptop Booking,Billing process".

Follow the same steps as mentioned in Activity -1 .

## **The Lightning App**

An app is a collection of items that work together to serve a particular function. In Lightning Experience, Lightning apps give your 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.

# Create a Lightning App

To create a lightning app page:

1. Go to setup page >> search “app manager” in quick find >> select “app manager” >> click on New lightning App.
2. Fill the app name in app details as LAPTOP RENTALS >>Next >> (App option page) keep it as default >> Next >> (Utility Items) keep it as default >> Next.
3. Upload a photo that is related to your app.
4. To Add Navigation Items:
5. Select the items (Total Laptops, consumer, Laptop Booking, Billing Process) from the search bar and move it using the arrow button >> Next.
6. 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 also 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

1. Standard Fields
2. 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 organizer 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.

## Creating the field in consumer object

### To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(consumer) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data Type as a “Phone”
4. Click on next
  - Fill the Above as following:
  - Field Label: Phone number
  - Field Name : gets auto generated
  - Click the required option checkbox.
  - Click on Next >> Next >> Save and new.

### To create another fields in an object:

1. Go to setup >> click on Object Manager >> type object name(consumer) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Email” and Click on Next
4. Fill the Above as following:
  - Field Label: Email
  - Field Name :It's gets auto generated
  - Click on Next >> Next >> Save and new.

The screenshot shows the 'consumer' object setup page. On the left, a sidebar lists various configuration options under 'Fields & Relationships'. The main area is titled 'New Custom Field' and 'Step 2. Enter the details'. The 'Field Label' is set to 'Email'. The 'Field Name' is also 'Email'. There are four text input fields: 'Description', 'Help Text', 'Required' (unchecked), 'Unique' (unchecked), 'External ID' (unchecked), and 'Auto add to custom report type' (checked). A checkbox for 'Default Value' is present but empty. At the bottom right, there are 'Previous', 'Next', and 'Cancel' buttons.

### To create another fields in an object:

1. Go to setup >> click on Object Manager >> type object name(consumer) in search bar >> click on the object.
2. Now click on "Fields & Relationships" >> New
3. Select Data type as a "Text Area" and Click on Next
4. Fill the Above as following:
  - Field Label: Address
  - Field Name : It's gets auto generated
  - Select Required field.
  - Click on Next >> Next >> Save and new.

### To create another fields in an object:

1. Go to setup >> click on Object Manager >> type object name(consumer) in search bar >> click on the object.
2. Now click on "Fields & Relationships" >> New
3. Select Data type as a "Picklist" and Click on Next
4. Fill the Above as following:
  - Field Label: consumer Status
  - Value - Select enter values with each value separated by a new line
  - Student
  - Employee
  - Others
  - Select required
  - Field Name :It's gets auto generated
  - Click on Next >> Next >> Save and new.

Field Label: consumer status

Values:  Use global picklist value set  Enter values, with each value separated by a new line

student  
employee  
others

Field Name: consumer\_status

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

## Creating the field in Laptops Bookings object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Laptop Booking) in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data Type as a “Picklist”
4. Picklist values are:-1.Dell 2. Acer 3.Hp 4.Mac

Field Label: Laptop names

Values:  Use global picklist value set  Enter values, with each value separated by a new line

Dell  
Acer  
Hp  
Mac

Field Name: Laptop\_type

Description:

Help Text:

Required:

Auto add to custom report type:

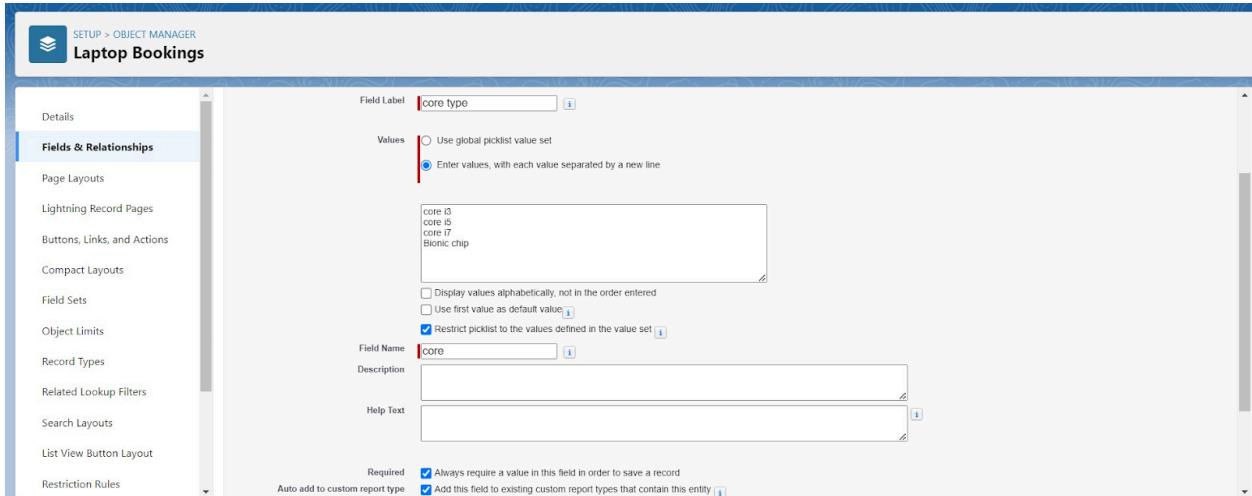
5. Select required
6. Click on Next >> Next >> Save and new

## 2. To Create a Fields & Relationship to an Laptop Booking Object

To create fields & relationship to an object:

1. Go to setup >> click on Object Manager >> type object name(Laptop Booking) in the search bar >> click on the object.

2. Now click on "Fields & Relationships" >> New
3. Select Data Type as a "Picklist"
4. Picklist values are:-1.core i3 2.Core i5 3.Core i7 .



5. Select required
6. Click on Next >> Next >> Save and new

#### **NOTE:-**

#### **Field Dependency:**

1. A field dependency refers to a relationship between two fields on an object where the values of one field determine the available values for another field. Field dependencies are commonly used to create picklist field relationships, where the available options in a dependent picklist are determined by the value selected in a controlling picklist.

#### **Need to use Field Dependency:**

1. By using the field dependency we can get the different Values by selecting the different Picklist.

## To Create a Fields & Relationship to an Laptop Booking Object

To create fields & relationship to an object:

- 1.Go to setup >> click on Object Manager >> type object name(Laptop Booking) in the search bar >> click on the object.
- 2.click field dependency and next
3. Click the include value for dell-core i3,i5,i7 and for acer i3,i4,i5 and for hp i3,i4,i5 and also for mac bionic chip include the values for it.

The screenshot shows the 'Object Manager' interface for the 'Laptop Bookings' object. On the left sidebar, under 'Fields & Relationships', the 'New' button is highlighted. In the main area, a new relationship is being defined with 'Controlling Field' set to 'Laptop names' and 'Dependent Field' set to 'core type'. The 'Instructions' section provides guidance on selecting cells in the picklist grid. A legend indicates that yellow cells represent 'Included Value' and white cells represent 'Excluded Value'. The picklist grid shows various laptop models (Dell, Acer, HP, Mac) and core types (core i3, core i5, core i7, Bionic chip), with some cells highlighted in yellow to show the selection process.

Click save.

NOTE:- fill the records which you have created in consumer and laptop bookings and give relations also. After saving the records go to the laptop bookings object and edit lookup to master the detailed relationship.

The screenshot shows the 'Object Manager' interface for the 'Laptop Bookings' object. The 'Fields & Relationships' tab is selected, displaying a list of 8 items. The table lists fields such as Amount, core type, Created By, Laptop Bookings Name, Laptop names, Last Modified By, Name, and Total no of laptops, along with their field labels, data types, controlling fields, and indexing status. The 'core type' field is shown as a picklist controlled by 'Laptop names'.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Amount	Amount__c	Currency(18, 0)		
core type	core__c	Picklist	Laptop names	
Created By	CreatedById	Lookup(User)		
Laptop Bookings Name	Name	Text(80)		✓
Laptop names	Laptop_type__c	Picklist		
Last Modified By	LastModifiedById	Lookup(User)		
Name	Name__c	Master-Detail(consumer)		✓
Total no of laptops	Total_no_of_laptops_c	Master-Detail(Total laptops)		✓

To Create a Rollup Summary Field in “Total Laptops Object”

1. After Creating the Lookup Relationship Than Only you can create the Rollup Summary
2. Go to setup >> click on Object Manager >> type object name(Total Laptops) in the search bar >> click on the object.
3. Now click on “Fields & Relationships” >> New

The screenshot shows the 'Total laptops' object setup page. On the left, a sidebar lists various configuration options under 'Fields & Relationships'. The main area is titled 'New Custom Field' and 'Step 1. Choose the field type'. It asks to specify the type of information. A 'Data Type' section contains several options: 'None Selected' (selected), 'Auto Number', 'Formula', 'Roll-Up Summary' (selected), 'Lookup Relationship', and 'Master-Detail Relationship'. Each option has a detailed description below it.

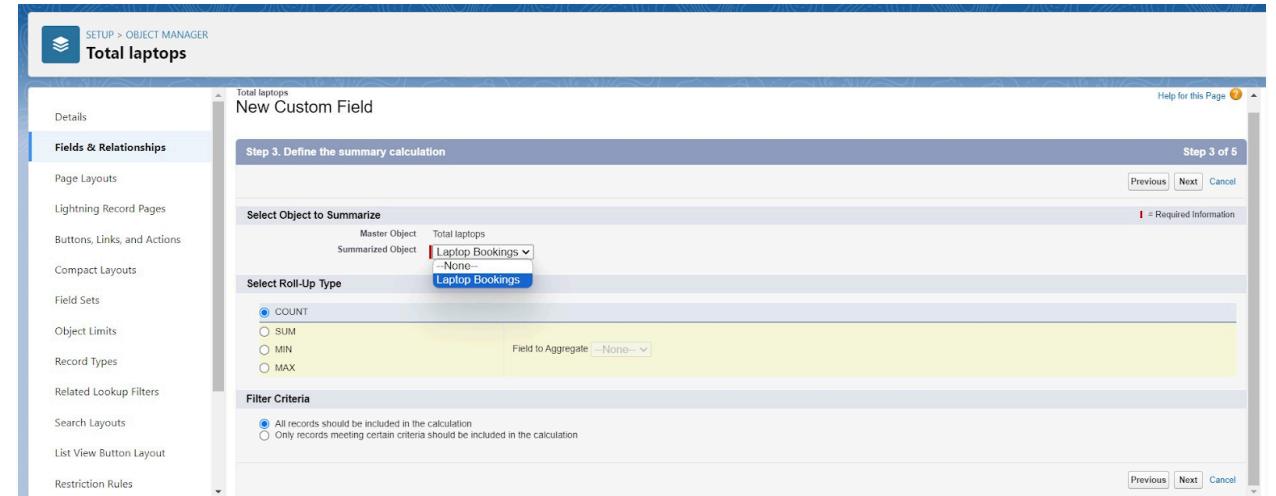
4.

- Select Data type as a “Roll-up Summary” and Click on Next

- Fill the Above as following:
- Field Label: Laptops delivered
- Field Name :It's gets auto generated

The screenshot shows the 'Total laptops' object setup page. On the left, a sidebar lists various configuration options under 'Fields & Relationships'. The main area is titled 'New Custom Field' and 'Step 2 of 5'. It asks to enter details. The 'Field Label' is set to 'Laptops delivered' and the 'Field Name' is also 'Laptops\_delivered'. There are fields for 'Description' and 'Help Text', both currently empty. A checkbox 'Auto add to custom report type' is checked, with a note 'Add this field to existing custom report types that contain this entity'. Navigation buttons 'Previous', 'Next', and 'Cancel' are at the bottom right.

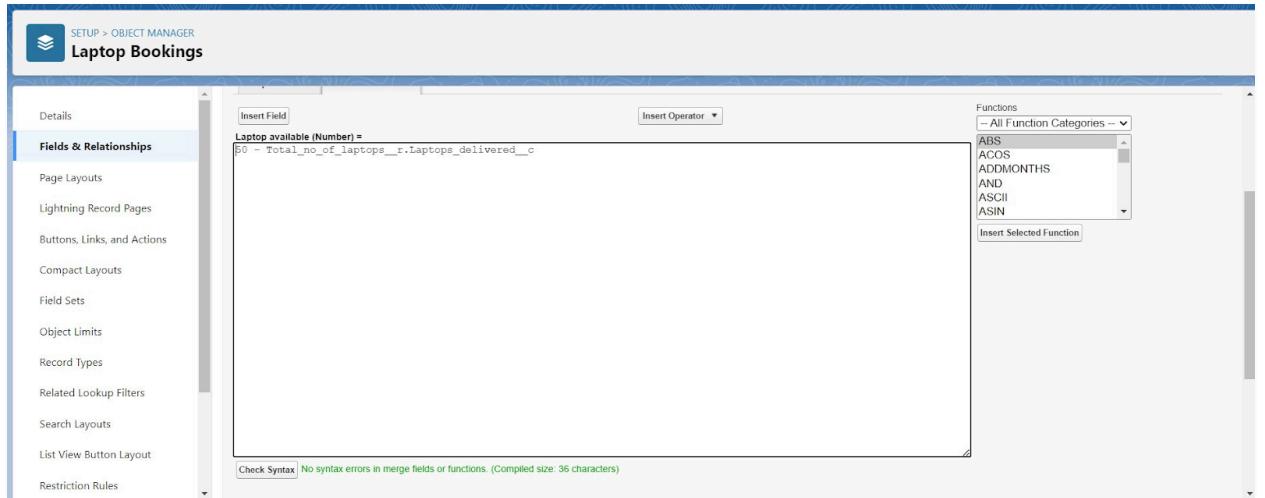
- Click on Next
- Select the Laptop Bookings in the Summarized Object
- Select the count Radio button in the select Roll-up Type



8.

8. To create fields in an object:

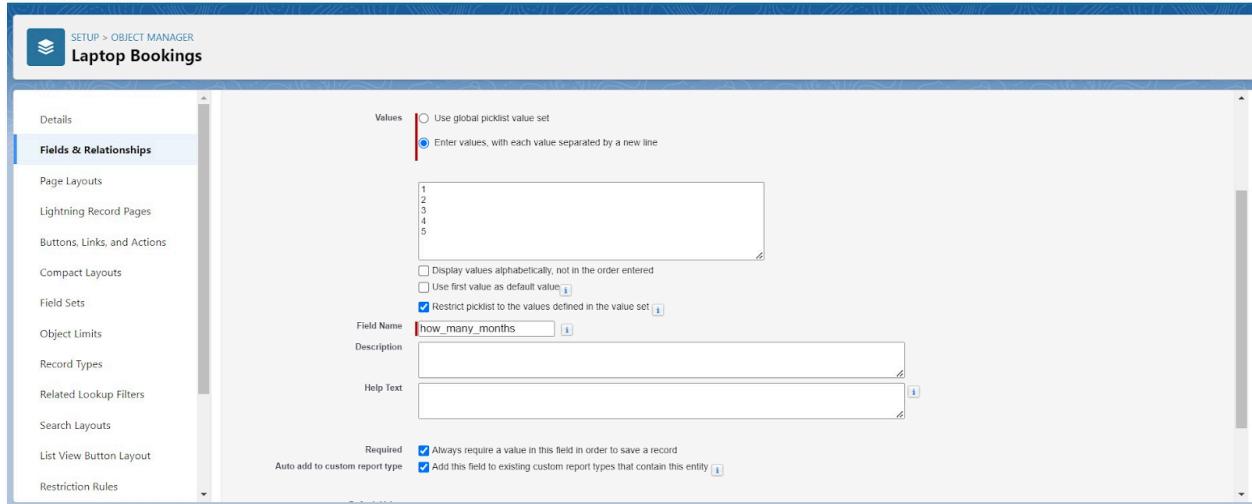
1. Go to setup >> click on Object Manager >> type object name(Laptop Booking) in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Formula” and Click on Next
4. Fill the Above as following:
  - Field Label: Laptops Available
  - Field Name : It's gets auto generated
  - Select the Formula Return Type as “Number”
  - Select the Decimal places as “0” and Click on Next
  - Click on the Advanced Formula and Enter the value in formula box “ 50 - ” and Click on insert field than you will find a pop window under the Laptop Booking select the Total No Of Laptops in the second Column and select the Laptops delivered in the third column and click on insert
  - “ 50 - Total\_no\_of\_laptops\_\_r.Laptops\_delivered\_\_c ” and Check Syntax



- Click on Next >> Next >> Save and new

To create fields in an object:

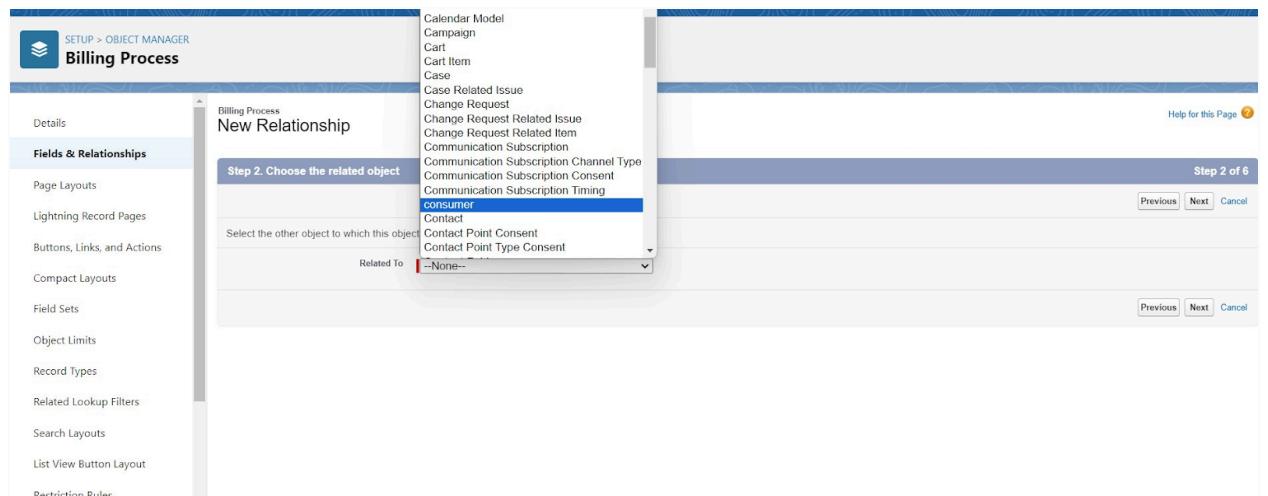
- 1.Go to setup >> click on Object Manager >> type object name(Laptop Booking) in the search bar >> click on the object.
- 3.Now click on “Fields & Relationships” >>New
- 4.Select Data Type as a “picklist”
5. Picklist values are 1.2.3.4.5
6. Click and save it.



## Creation of Fields & Relationship for Billing Process Object

1. To create fields & relationship to an object:

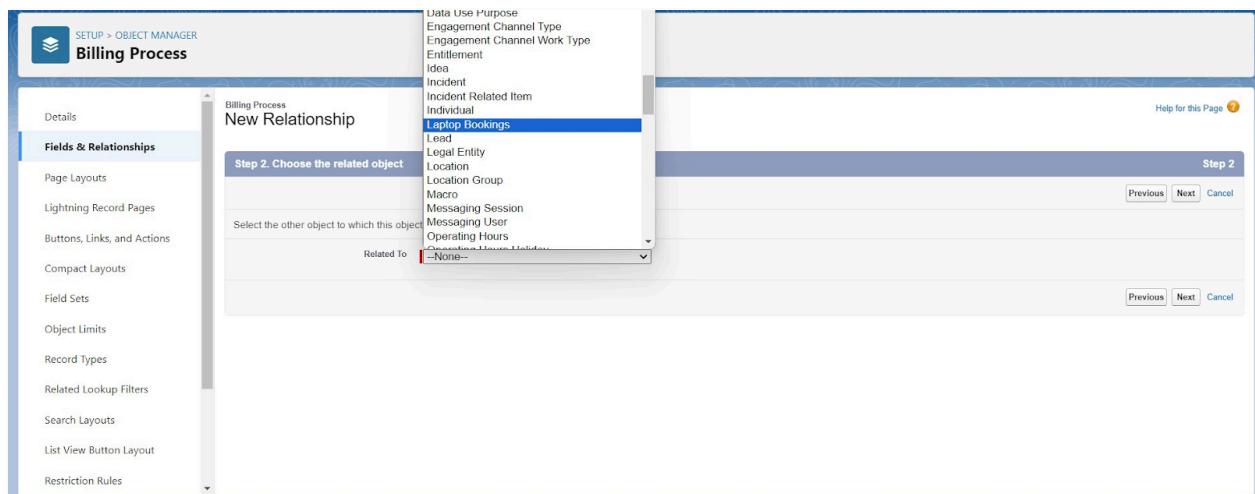
1. Go to setup >> click on Object Manager >> type object name(Billing Process) in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data Type as a “Master-detail Relationship”
4. Click on Next
5. Click on the Related to drop down and Select the consumer object and click on Next



- 6.
7. Fill the Above as following:
  - Change the Field Label: Name
  - Field Name :It's gets auto generated
  - Click on Next >> Next >> Save and new.

2. To create another fields & relationship to an object:

1. Go to setup >> click on Object Manager >> type object name(Billing Process) in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data Type as a “Lookup Relationship”
4. Click on Next
5. Click on the Related to drop down and Select the Laptop Booking object and click on Next



- 6.
7. Fill the Above as following:
  - Change the Field Label: Laptop Booking
  - Field Name :It's gets auto generated
  - Click on Next >> Next >>Save and new.

3. Creation of another fields for the billing process object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Billing Process) in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data Type as a “Picklist”
4. Fill the Above as following:
  - Field Label: Payment Mode
  - Value >> Select enter values with each value separated by a new line
1. Cash

2. Check
3. Credit card
4. Debit card
5. UPI
6. Phonepe
7. Gpay
8. Paytm
- Select required
- Click on Next >> Next >> Save and new.

#### Cross Object Formula Field:

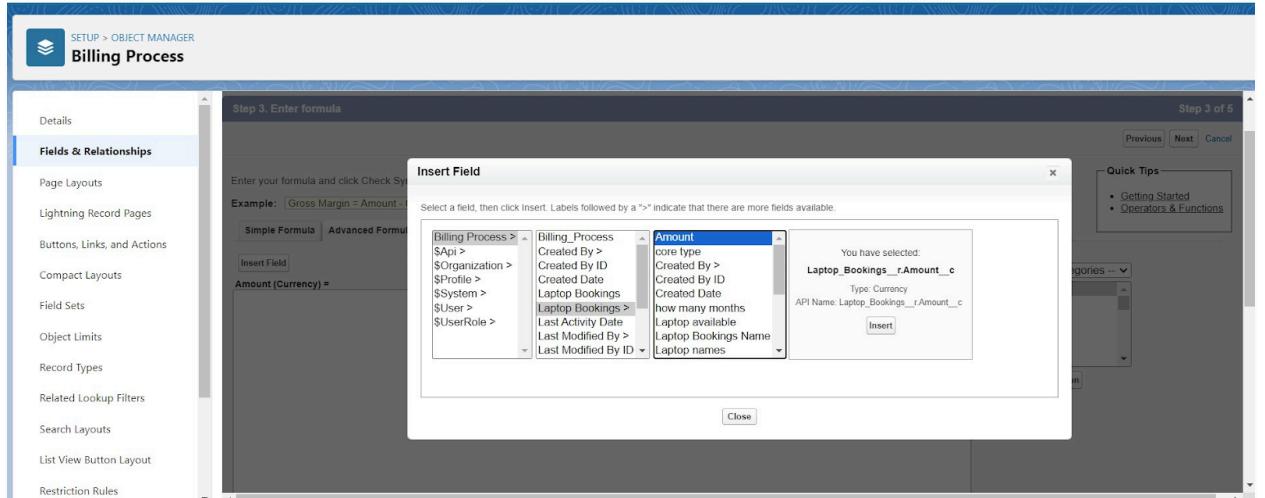
In Salesforce, a cross-object formula field allows you to create a formula that references fields from related objects. It enables you to perform calculations or display data from related records without the need for custom code or complex workflows.

#### Why do we need to create the Cross Object Formula Field:

If we want to get the Particular field from another object in that case we will use the Cross object Formula field. For that First we need to create the relationship b/w two objects and relate the field with formula data type.

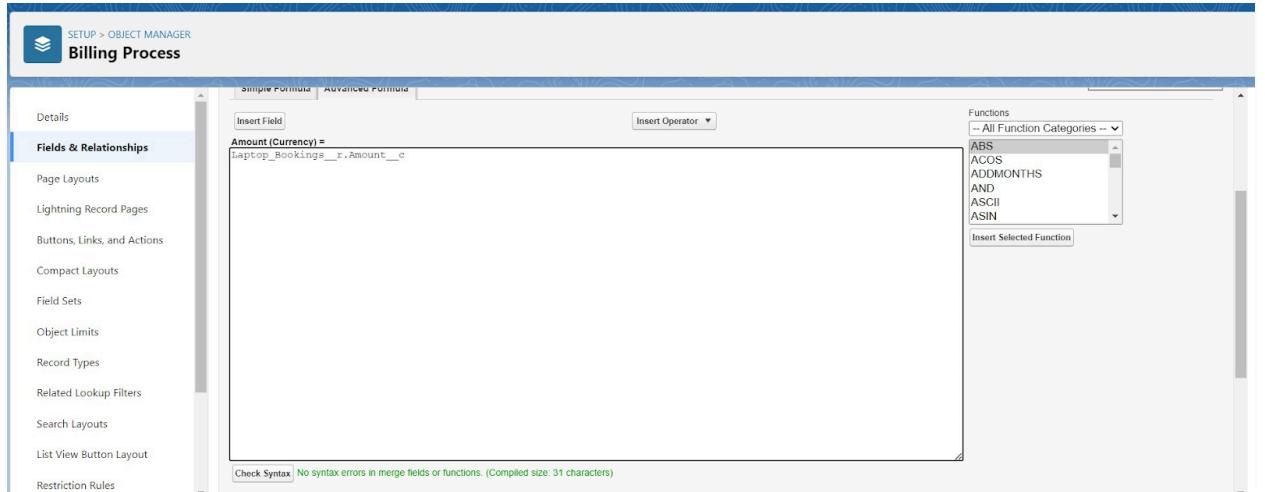
#### 4. Create a Cross object formula Field in billing process Object

1. Go to setup >> click on Object Manager >> type object name(Billing Process) in the search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data Type as a “Formula”
4. Click on Next
5. Enter the Field label: Amount, the Field name gets auto generated and click on Next.(Formula return type Number).
6. In the Advanced Formula Click on the Insert field in the popup Screen Select the Billing Process and in the second drop down select the Laptop Booking and in the three drop down select the Amount field and click on Insert
7. “ Laptop\_Booking\_\_r.Amount\_\_c ”.
8. Click on the Check syntax: No syntax errors in merge fields



9.

10. Click on Next >> Next >> Save and new.



11.

## Creating the field in Total Laptops object

1. To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Total Laptops) in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data type as a “Formula” and Click on Next
4. Fill the Above as following:
5. Field Label: Laptops Available
6. Field Name : It’s gets auto generated
7. Select the Formula Return Type as “Number”

Field Label: Laptops Available  
Field Name: Laptops\_Available

Formula Return Type: Number  
Options: Decimal Places 0 Example: 999

- 8.
9. Select the Decimal places as “0” and Click on Next

Note: I am Considering “Total No Of Laptops = 50” While creating a new record in Total Laptops Object.

1. Click on the Advanced
2. Formula “ 50 - Laptops\_delivered\_\_c ” and Check Syntax

Simple Formula | Advanced Formula

Laptops Available (Number) =  
50 - Laptops\_delivered\_\_c

Check Syntax No syntax errors in merge fields or functions. (Compiled size: 36 characters)

- 3.
4. Click on Next >>Next >>Save and new.

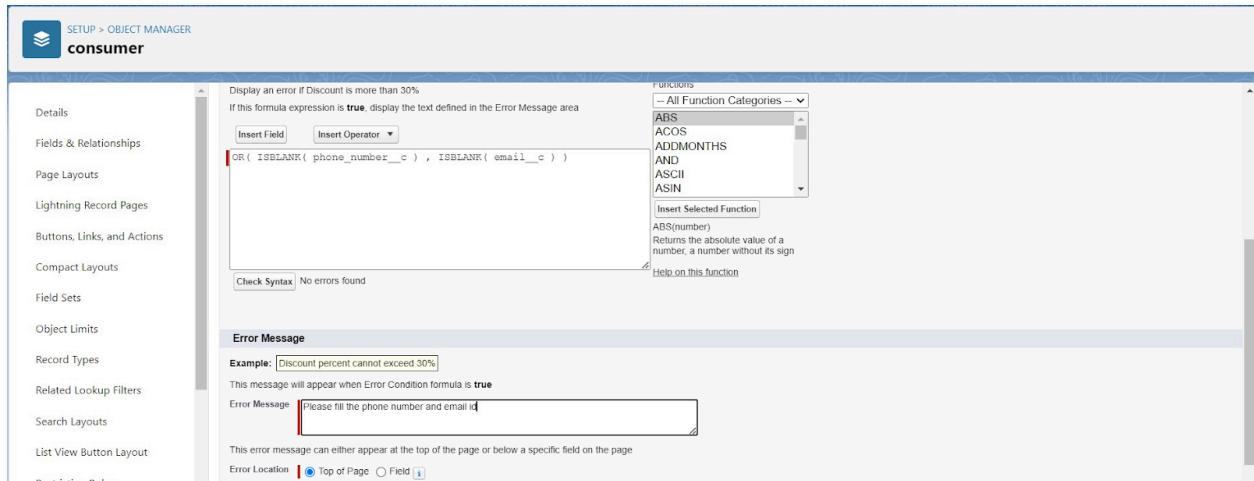
## Validation rule

Validation rules are applied when a user tries to save a record and are used to check if the data meets specified criteria. If the criteria are not met, the validation rule triggers an error message and prevents the user from saving the record until the issues are resolved.

Improve the quality of your data using validation rules. Validation rules verify that the data a user enters in a record meets the standards you specify before the user can save the record. A validation rule can contain a formula or expression that evaluates the data in one or more fields and returns a value of “True” or “False”. Validation rules also include an error message to display to the user when the rule returns a value of “True” due to an invalid value.

Creating the validation rule for phone number field in consumer object

1. Go to the setup page - click on object manager - From drop down click edit for consumer object.
2. Click on the validation rule - click New.
3. Enter the Rule name as “Phonenumberoremailblankrule ”.
4. Enter the description as “phone number and email number should not be blank”.
5. Enter the formula as “OR( ISBLANK( phone\_number\_\_c ) , ISBLANK( email\_\_c ) )” and check the syntax.



- 6.
7. Save the validation rule.

## Creating the validation rule for phone number field in consumer object

Creating the validation rule for phone number field in consumer object

1. Go to the setup page >> click on object manager >> From drop down click edit for consumer object.
2. Click on the validation rule >> click New.
3. Enter the Rule name as “Phonenumberoremailblankrule”.
4. Enter the description as “phone number and email number should not be blank”.
5. Enter the formula as “OR( ISBLANK( phone\_number\_c ), ISBLANK( email\_c ) )” and check the syntax.

The screenshot shows the Salesforce Object Manager interface for the 'consumer' object. In the formula editor, the formula `OR( ISBLANK( phone_number_c ), ISBLANK( email_c ) )` is entered. A tooltip for the `ABS` function is displayed, showing its description: `ABS(number)` Returns the absolute value of a number, a number without its sign. The error message field contains the text `Please fill the phone number and email id`. The sidebar on the left provides navigation links for various setup features.

- 6.
7. Save the validation rule.

## 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 delete 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.

# owner Profile

To create a new profile:

1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Standard User) >> enter profile name (owner) >> Save.

The screenshot shows the Salesforce Setup interface for managing profiles. The top navigation bar has 'SETUP' and 'Profiles'. Below it, a message says: 'Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information.' It also notes: 'If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile.' A horizontal bar lists various system-wide permissions. The main section is titled 'Profile Detail' with a table:

	<a href="#">Edit</a>	<a href="#">Clone</a>	<a href="#">Delete</a>	<a href="#">View Users</a>
Name	owner			
User License	Salesforce			Custom Profile <input checked="" type="checkbox"/>
Description				
Created By	udayrishi.yelagandula, 10/07/2023, 10:56 am			Modified By <a href="#">udayrishi.yelagandula</a> , 10/07/2023, 10:56 am

Below this is a section titled 'Page Layouts' with a table:

Standard Object Layouts			
Global	<a href="#">Global Layout [View Assignment]</a>	Object Milestone	<a href="#">Object Milestone Layout [View Assignment]</a>
Email Application	Not Assigned <a href="#">[View Assignment]</a>	Operating Hours	<a href="#">Operating Hours Layout [View Assignment]</a>
Home Page Layout	DE Default <a href="#">[View Assignment]</a>	Opportunity	<a href="#">Opportunity Layout [View Assignment]</a>
Account	<a href="#">Account Layout [View Assignment]</a>	Opportunity Product	<a href="#">Opportunity Product Layout [View Assignment]</a>
Alternative Payment Method	<a href="#">Alternative Payment Method Layout [View Assignment]</a>	Order	<a href="#">Order Layout [View Assignment]</a>
Appointment Invitation	<a href="#">Appointment Invitation Layout [View Assignment]</a>	Order Product	<a href="#">Order Product Layout [View Assignment]</a>

2.

3. Scroll down to Custom Object Permissions and Give access permissions for Total Laptops, consumers , Laptop Booking and Billing Process objects as mentioned in the below diagram.

**Custom Object Permissions**

	Basic Access						Data Administration	
	Read	Create	Edit	Delete	View All	Modify All		
Leads	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Work Types	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Work Type Groups	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

**Session Settings**

Session Times Out After: 2 hours of inactivity

Session Security Level Required at Login: --None--

**Password Policies**

- 3.
4. Give Access and Save it.

## Agent Profile

1. Go to setup >> type profiles in quick find box >> click on profiles >> clone the desired profile (Standard Platform User) >> enter profile name (Agent) >> Save.
2. While still on the profile page, then click Edit.
3. Scroll down to Custom Object Permissions and Give access permissions for Total Laptops, consumer , Laptop Bookings and Billing Process objects as mentioned in the below diagram.

**Custom Object Permissions**

	Basic Access						Data Administration	
	Read	Create	Edit	Delete	View All	Modify All		
Billing Process	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
consumers	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
Laptop Bookings	<input checked="" type="checkbox"/>							
Total Laptops	<input checked="" type="checkbox"/>							

**Session Settings**

Session Times Out After: 2 hours of inactivity

Session Security Level Required at Login: --None--

**Password Policies**

- User passwords expire in: 90 days
- 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

- 4.
5. Give access and save it.

# Roles and Hierarchy

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 organization can have to data. Simply put, it describes what a user could see within the Salesforce organization.

## Creating owner Role

Creating owner 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 whom this role works.

Your Organization's Role Hierarchy

**Nick Enterprises**

- CEO
- HR
- Manager
- On Site Emp
- Remote Emp

**Add Role**

1. Give Label as “owner” and Role name gets auto populated. Then click on Save.

Role Edit

New Role

Role Edit

Label	owner
Role Name	owner
This role reports to	CEO
Role Name as displayed on reports	

Save Save & New Cancel

2.

3. Click and save it.

## Activity 2: Creating Agent roles

Creating another two roles under manager

1. Go to quick find - Search for Roles - click on set up roles.
2. Click plus on CEO role, and click add role under owner.

**Creating the Role Hierarchy**

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

**Your Organization's Role Hierarchy**

**smartbridge** **Add Role**

- CEO** **Edit | Del | Assign** **Add Role**
- CFO** **Edit | Del | Assign** **Add Role**
- COO** **Edit | Del | Assign** **Add Role**
- HR** **Edit | Del | Assign** **Add Role**
- owner** **Edit | Del |**  **Add Role**
- SVP.Customer Service & Support** **Edit | Del | Assign** **Add Role**
- SVP.Human Resources** **Edit | Del | Assign** **Add Role**
- SVP.Sales & Marketing** **Edit | Del | Assign** **Add Role**

**Help for this Page**

**Show in tree view**

3.

4. Give Label as “Agent” and Role name gets auto populated. Then click on Save.

## 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.

## Create User

1. Go to setup - type users in quick find box - select users -click New user.

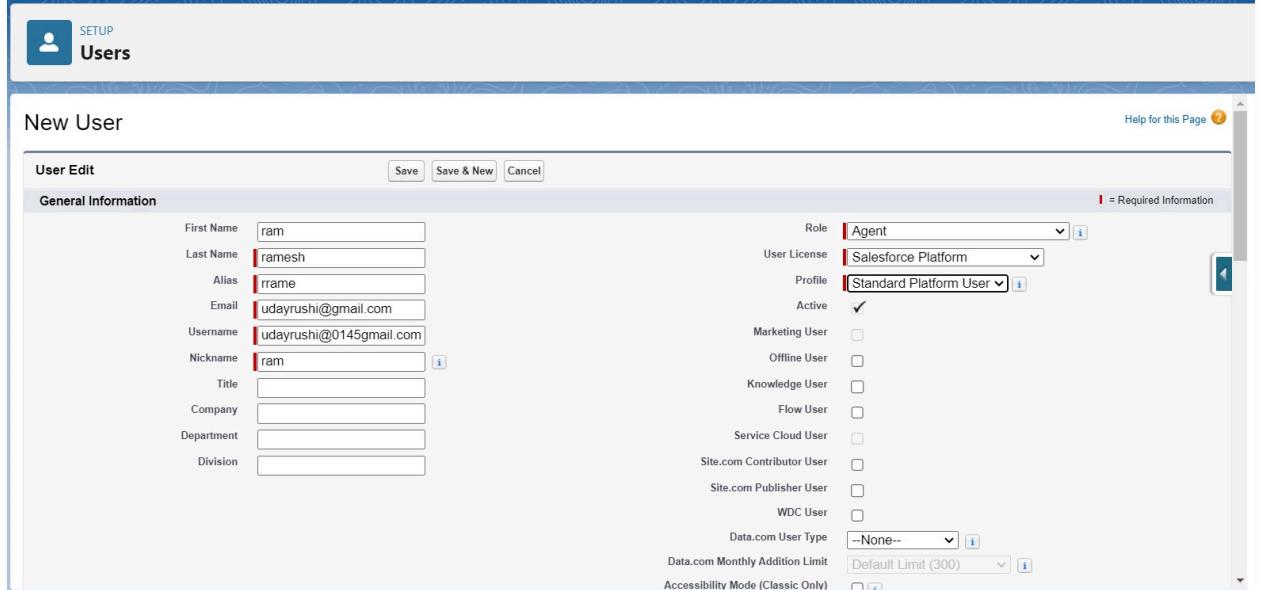
2. Fill in the fields
3. First Name : vicky
4. Last Name : y
5. Alias : Give a Alias Name
6. Email id : Give your Personal Email id
7. Username : Username should be in this form: text@text.text
8. Nick Name : Give a Nickname
9. Role : owner
10. User license : Salesforce
11. Profiles : owner.

Save it.

#### Activity 2: creating another users

1. Go to setup -type users in quick find box - select users -click New user.
2. Fill in the fields
3. First Name : ram
4. Last Name : ram
5. Alias : Give a Alias Name
6. Email id : Give your Personal Email id
7. Username : Username should be in this form: text@text.text
8. Nick Name : Give a Nickname
9. Role : Agent
10. User license : Salesforce platform

11. Profiles : standard platform user.



The screenshot shows the 'New User' setup page in Salesforce. The 'General Information' section contains the following data:

Field	Value
First Name	ram
Last Name	ramesh
Alias	rrame
Email	udayrushi@gmail.com
Username	udayrushi@0145gmail.com
Nickname	ram
Title	
Company	
Department	
Division	

The 'Profile' section shows 'Standard Platform User' selected. Other profile options like 'Marketing User', 'Offline User', etc., are shown with checkboxes. The 'Role' dropdown is set to 'Agent'. The 'User License' dropdown is set to 'Salesforce Platform'. The 'Active' checkbox is checked. The 'Data.com User Type' dropdown is set to '-None-'.

12.

13. Save it.

## 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.

In Salesforce, "flows" typically refer to Salesforce Flow, which is a powerful automation tool that allows you to create custom, automated processes in your Salesforce org without writing code. Salesforce Flow is a point-and-click tool that enables you to design and automate complex business processes, collect data, and interact with users in a visual interface. There are different types of flows in Salesforce, including:

**Screen Flows:** These are used to guide users through a series of screens to collect or display information.

Screen Flows are often used for data entry and updates.

**Autolaunched Flows:** These are flows that are triggered by events, such as when a record is created or updated. They don't require user interaction and can be used for background automation.

**Flow Builder:** Flow Builder is the visual interface used to create flows. It allows you to design flows by adding

elements, like screens, logic, and actions, using a drag-and-drop approach.

**Flow Templates:** Salesforce provides a library of pre-built flow templates that you can use as a starting point for your own flows. These templates cover a variety of use cases, from simple to complex.

**Scheduled Flows:** These are flows that you can schedule to run at specific times or intervals. They are often used for automating recurring tasks.

**Flow Elements:** Flow Builder offers various elements that you can use to create flows, such as variables, decisions, loops, and more. These elements allow you to build sophisticated logic into your flows.

**Subflows:** Subflows are reusable flow elements that you can incorporate into multiple flows, making it easier to manage and maintain complex processes.

**Record-Triggered Flows:** These are flows that are triggered when records meet specified criteria. They are often used for automating record updates and related actions.

Why do we need to create a flow:

To get the Amount Field automatic by the selection of laptop types the Amount is generated Automatically in the amount field.

## Create a Flow on dell laptop

Activity -

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

The screenshot shows the Salesforce Setup page. A red box labeled '1' highlights the 'Search Setup' bar at the top. Another red box labeled '2' highlights the 'Flows' link under the 'Process Automation' section. A third red box labeled '3' highlights the 'New Flow' button in the top right corner of the main content area.

Flow Label	Process Type	Ac...	Te...	Package State	Pa...	Last Modified By	Last Modified ...
Ac Amount update	Autolaunched Flow	<input type="checkbox"/>	<input type="checkbox"/>	Unmanaged		Veera Venkata Varaprasad Androthu	07/06/2023, 11:35 am
Book Appointment from Invitation	Salesforce Scheduler Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Cancel Item Flow	Screen Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Change Case Owner to Incident Owner	Screen Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Close Change Request & Related Issues	Screen Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			

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

New Flow

Core All + Templates

Screen Flow  
Guides users through a business process that's launched from Lightning pages, Experience Cloud sites, quick actions, and more.

Record-Triggered Flow  
Launches when a record is created, updated, or deleted. This autolaunched flow runs in the background.

Schedule-Triggered Flow  
Launches at a specified time and frequency for each record in a batch. This autolaunched flow runs in the background.

Platform Event—Triggered Flow  
Launches when a platform event message is received. This autolaunched flow runs in the background.

Autolaunched Flow (No Trigger)  
Launches when invoked by Apex, processes, REST API, and more. This autolaunched flow runs in the background.

Record-Triggered Orchestration  
Launches when a record is created or updated. An orchestration lets you create a multi-step, multi-user process.

**Create**

3. Select the Object as a Laptop Booking in the Drop down list.
4. Select the Trigger Flow when: “A record is Created or Updated”.
5. Select the Optimize 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  
Laptop Bookings

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.

**Cancel** **Done**

**Configure Start**

### 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

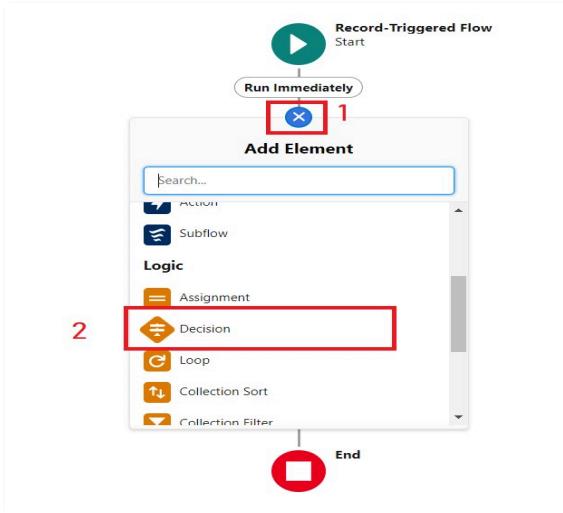
**\*Optimize the Flow for:**

**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. Under the Record-triggered Flow Click on “+” Symbol and In the Drop down List select the “Decision Element”.



7. Enter the Details Label: Field should be Update, API name: Gets Automatically Generated.
8. Enter the Outcome Details Label: dell , Outcome API name: Gets Automatically Generated.
- Resource: Select Record.Laptop booking\_c.
- Operator: Select Equals.
- Value: Select dell
- Add the same outcome order to acer , hp,mac.
- Click done.

**Edit Decision**

* Label field should updated	* API Name field_should_updated							
Description the field should be automatically updated								
<b>Outcomes</b> For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.								
<b>OUTCOME ORDER</b> ① + <ul style="list-style-type: none"> <li><input type="checkbox"/> dell</li> <li><input type="checkbox"/> acer</li> <li><input type="checkbox"/> hp</li> <li><input type="checkbox"/> mac</li> <li><input type="checkbox"/> false</li> </ul>	<b>OUTCOME DETAILS</b> <table border="1"> <tr> <td>* Label dell</td> <td>* Outcome API Name dell</td> </tr> <tr> <td colspan="2">Condition Requirements to Execute Outcome All Conditions Are Met (AND)</td> </tr> <tr> <td>Resource \$Record &gt; Laptop names X</td> <td>Operator Equals</td> <td>Value Dell</td> </tr> </table>	* Label dell	* Outcome API Name dell	Condition Requirements to Execute Outcome All Conditions Are Met (AND)		Resource \$Record > Laptop names X	Operator Equals	Value Dell
* Label dell	* Outcome API Name dell							
Condition Requirements to Execute Outcome All Conditions Are Met (AND)								
Resource \$Record > Laptop names X	Operator Equals	Value Dell						
<a href="#">Delete Outcome</a>								
<a href="#">Cancel</a> <a href="#">Done</a>								

9. Go to flow page

11. Beside dell there is a symbol ‘+’ click on that.

12. Again select decision

13. Enter the Details Label: Field should Update(any one u want), API name: Gets Automatically Generated.

14. select the Outcome Details Label: dell core i3 , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.core type.
- Operator: Select Equals.
- Value: Select core i3.
- Then again click the symbol ‘+’ outcome details

15. select the Outcome ‘+’ Details Label: dell core i5 , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.core type.
- Operator: Select Equals.
- Value: Select core i5.
- Then again click the symbol’+’ outcome details

16. Enter the Outcome Details Label: dell core i7 , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.core type.
- Operator: Select Equals.
- Value: Select core i7.

17. Click done.

**Edit Decision**

*Label <input type="text" value="field updated"/>	*API Name <input type="text" value="field_updated"/>
Description <div style="border: 1px solid #ccc; height: 40px; margin-top: 5px;"></div>	
<b>Outcomes</b> For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.	
<b>OUTCOME ORDER</b> <span style="font-size: small;">(i) +</span> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">           dell core i3         </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">           dell core i5         </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">           dell core i7         </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">           Default Outcome         </div>	<b>OUTCOME DETAILS</b> *Label <input type="text" value="dell core i3"/> *Outcome API Name <input type="text" value="dellcore_i3"/> Condition Requirements to Execute Outcome All Conditions Are Met (AND) Resource <input type="text" value="\$Record &gt; core type"/> Operator Equals Value core i3
<a href="#" style="margin-right: 10px;">Delete Outcome</a> <a href="#">Cancel</a> <a href="#">Done</a>	

18.

19. So go to the flow page select ‘+’ after core i3 then again select the decision.
20. Enter the Details Label: months selected , API name: Gets Automatically Generated.
21. Enter the Outcome Details Label: dell 1(i3) , Outcome API name: Gets Automatically Generated.
22.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: 1.
23. Enter the Outcome Details Label: dell 2(i3) , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: Select 2..
24. Click ‘+’ outcome details
25. Enter the Outcome Details Label: dell 3(i3) , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: Select 3..
26. Click ‘+’ outcome details
27. Enter the Outcome Details Label: dell 4(i3) , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: Select 4..
28. Click ‘+’ outcome details

29. Enter the Outcome Details Label: dell 5(i3) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 4.

The screenshot shows the 'Edit Decision' screen. At the top, there are fields for 'Label' (months selected) and 'API Name' (months\_selected). Below these are sections for 'Description' and 'Outcomes'. The 'Outcomes' section has a table for 'OUTCOME ORDER' with rows numbered 1 to 5. Row 1 is configured with 'Label' 1 and 'Outcome API Name' X1. A dropdown 'Condition Requirements to Execute Outcome' is set to 'All Conditions Are Met (AND)'. Row 5 contains a condition row with 'Resource' \$Record > how many months, 'Operator' Equals, and 'Value' 1. At the bottom right are 'Cancel' and 'Done' buttons.

OUTCOME ORDER	OUTCOME DETAILS
1	*Label: 1 *Outcome API Name: X1
2	
3	
4	
5	Resource: \$Record > how many months Operator: Equals Value: 1

30.

31. Follow the above picture you will understand.

32. After dell 1(i3) there is ‘+’ symbol like dell 2(i3),dell 3(i3),dell 4(i3),dell 5(i3).

33. Click on ‘+’ then select update records

34. Enter the Details Label: one month of dell i3 rate , API name: Gets Automatically Generated.

35. Field:- Amount\_c , value:- for dell 1(i3)-1000, dell 2(i3)-2000, dell 3(i3)-3000, dell 4(i3)-4000, dell 5(i3)-5000. Follow for all these finally

36. Click done.

37. Enter the Details Label: months selected , API name: Gets Automatically Generated.

38. Enter the Outcome Details Label: dell 1(i7) , Outcome API name: Gets Automatically Generated.

39.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: 1.

40. Enter the Outcome Details Label: dell 2(i7) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.

- Operator: Select Equals.

- Value: Select 2..

41. Click ‘+’ outcome details

42. Enter the Outcome Details Label: dell 3(i7) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 3..

43. Click ‘+’ outcome details

44. Enter the Outcome Details Label: dell 4(i7) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 4.

45. Click ‘+’ outcome details

46. Enter the Outcome Details Label: dell 5(i7) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 5.

The screenshot shows the 'Edit Decision' interface. At the top, there are fields for 'Label' (months selected) and 'API Name' (months\_selected). Below this is a 'Description' field. The main area is titled 'Outcomes' with the sub-instruction: 'For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.' There are five outcome details listed:

OUTCOME ORDER	OUTCOME DETAILS	Actions
1	*Label: 1 *Outcome API Name: X1 Condition Requirements to Execute Outcome: All Conditions Are Met (AND)	Delete Outcome
2		
3		
4		
5		

At the bottom right are 'Cancel' and 'Done' buttons.

47.

48. Follow the above picture you will understand.

49. After dell 1(i7) there is ‘+’ symbol like dell 2(i7),dell 3(i7),dell 4(i7),dell 5(i7).

50. Click on ‘+’ then select update records

51. Enter the Details Label: one month of dell i5 rate , API name: Gets Automatically Generated.

52. Field:- Amount\_c , value:- for dell 1(i7)-2000, dell 2(i7)-4000, dell 3(i7)-6000, dell 4(i7)-8000, dell 5(i7)-10000. Follow for all these finally

53. Click done.

## creating flow on acer laptop

1. Go to flow page
2. Beside acer there is a symbol ‘+’ click on that.
3. Again select decision
4. Enter the Details Label: Field is Update, API name: Gets Automatically Generated.
5. select the Outcome Details Label: acer core i3 , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.core type.
  - Operator: Select Equals.
  - Value: Select core i3.

Click done.

6. Go to flow page
7. Beside dell there is a symbol ‘+’ click on that.
8. Again select decision
9. Enter the Details Label: months selected , API name: Gets Automatically Generated.
10. Enter the Outcome Details Label: acer 1(i3) , Outcome API name: Gets Automatically Generated.
- 11.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: 1.

12. Enter the Outcome Details Label: acer 2(i3) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 2..

13. Click ‘+’ outcome details

14. Enter the Outcome Details Label: acer 3(i3) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 3..

15. Click ‘+’ outcome details

16. Enter the Outcome Details Label: acer 4(i3) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 4..

17. Click ‘+’ outcome details

18. Enter the Outcome Details Label: acer 5(i3) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 5..

**Edit Decision**

* Label acer months selected	* API Name acer_months_selected																								
Description																									
<p><b>Outcomes</b> For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.</p> <table border="1"> <thead> <tr> <th>OUTCOME ORDER</th> <th>OUTCOME DETAILS</th> <th>Condition Requirements to Execute Outcome</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>acer 1(i3)</td> <td>* Label acer 1(i3) * Outcome API Name acer_1_i3</td> <td>All Conditions Are Met (AND)</td> <td>Delete Outcome</td> </tr> <tr> <td>acer 2(i3)</td> <td></td> <td></td> <td></td> </tr> <tr> <td>acer 3(i3)</td> <td></td> <td></td> <td></td> </tr> <tr> <td>acer 4(i3)</td> <td></td> <td></td> <td></td> </tr> <tr> <td>acer 5(i3)</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		OUTCOME ORDER	OUTCOME DETAILS	Condition Requirements to Execute Outcome	Action	acer 1(i3)	* Label acer 1(i3) * Outcome API Name acer_1_i3	All Conditions Are Met (AND)	Delete Outcome	acer 2(i3)				acer 3(i3)				acer 4(i3)				acer 5(i3)			
OUTCOME ORDER	OUTCOME DETAILS	Condition Requirements to Execute Outcome	Action																						
acer 1(i3)	* Label acer 1(i3) * Outcome API Name acer_1_i3	All Conditions Are Met (AND)	Delete Outcome																						
acer 2(i3)																									
acer 3(i3)																									
acer 4(i3)																									
acer 5(i3)																									
Resource \$Record > how many months Operator Equals Value 1																									
<input type="button" value="Cancel"/> <input type="button" value="Done"/>																									

Click done.

19. After acer 1(i3) there is ‘+’ symbol like acer 2(i3),acer 3(i3),acer 4(i3),acer 5(i3).
20. Click on ‘+’ then select update records
21. Enter the Details Label: one month of acer i3 rate , API name: Gets Automatically Generated.
22. Field:- Amount\_c , value:- for acer 1(i3)-900, acer 2(i3)-1800, acer 3(i3)-2700, acer 4(i3)-3600, acer 5(i3)-4800. Follow for all these finally

Edit Update Records

**one month of acer i3 rate (one\_month\_of\_acer\_i3\_rate)**

\* How to Find Records to Update and Set Their Values

- Use the laptop bookings record that triggered the flow
- Update records related to the laptop bookings 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

Set Filter Conditions

Condition Requirements to Update Record

None—Always Update Record

Set Field Values for the Laptop Bookings Record

Field	Value
Amount_c	900

**+ Add Field**

Cancel Done

23.

24. Click done.

## creating a flow on hp laptop

1. Go to flow page
2. Beside hp there is a symbol ‘+’ click on that.
3. Again select decision
4. Enter the Details Label: Field is Update, API name: Gets Automatically Generated.

5. select the Outcome Details Label: hp core i5 , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.core type.
  - Operator: Select Equals.
  - Value: Select hp i5.
6. Go to flow page
7. Beside hp there is a symbol ‘+’ click on that.
8. Again select decision
9. Enter the Details Label: hp field should be updated , API name: Gets Automatically Generated.
10. Enter the Outcome Details Label: hp 1(i5) , Outcome API name: Gets Automatically Generated.
11.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: 1.
12. Enter the Outcome Details Label: hp 2(i5) , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: Select 2..
13. Click ‘+’ outcome details
14. Enter the Outcome Details Label: hp 3(i5) , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: Select 3..
15. Click ‘+’ outcome details
16. Enter the Outcome Details Label: hp 4(i5) , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: Select 4.
17. Click ‘+’ outcome details
18. Enter the Outcome Details Label: hp 5(i5) , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.

- Value: Select 5.

**Edit Decision**

**Outcomes** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME ORDER	OUTCOME DETAILS	Actions
hp core i3	*Label: hp core i5 *Outcome API Name: hp_core_i5	<a href="#">Delete Outcome</a>
hp core i5		
hp core i7	Condition Requirements to Execute Outcome: All Conditions Are Met (AND)	
Default Outcome	Resource: \$Record > core type Operator: Equals Value: core i5 <a href="#">+ Add Condition</a>	<a href="#">Delete</a>
<b>When to Execute Outcome</b>		
<input checked="" type="radio"/> If the condition requirements are met <input type="radio"/> Only if the record that triggered the flow to run is updated to meet the condition requirements		
		<a href="#">Cancel</a> <a href="#">Done</a>

Click on done.

19. After hp 1(i5) there is '+' symbol like hp 2(i5), hp 3(i5), hp 4(i5), hp 5(i5).
20. Click on '+' then select update records
21. Enter the Details Label: one month of hp i5 rate , API name: Gets Automatically Generated.
22. Field:- Amount\_c , value:- for hp 1(i5)-1700, hp 2(i5)-3400, hp 3(i5)-5100, hp 4(i5)-6800, hp 5(i5)-8500. Follow for all these finally

**Edit Update Records**

**one month of hp i5 rate (one\_month\_of\_hp\_i5\_rate)** 

**\* How to Find Records to Update and Set Their Values**

Use the laptop bookings record that triggered the flow  
 Update records related to the laptop bookings 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

**Set Filter Conditions**

Condition Requirements to Update Record

None—Always Update Record ▾

**Set Field Values for the Laptop Bookings Record**

Field	Value
Amount_c	← 1700 

+ Add Field

Cancel
Done

Click done.

## creating a flow on mac laptop

1. Go to flow page
2. Beside mac there is a symbol ‘+’ click on that.
3. Again select decision
4. Enter the Details Label: mac should be Updated, API name: Gets Automatically Generated.
5. select the Outcome Details Label: mac laptop , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.core type.
  - Operator: Select Equals.
  - Value: Select Bionic Chip.

# APEX

## Apex OverView

Apex is a strongly typed, object-oriented programming language that allows developers to execute flow and transaction control statements on the Lightning platform server in conjunction with calls to the Lightning Platform API. Using syntax that looks like Java and acts like database stored procedures, Apex enables developers to add business logic to most system events, including button clicks, related record updates, and Visualforce pages. Apex code can be initiated by Web service requests and from triggers on objects.

It is as similar as java i.e, it also supports OOP( Object oriented programming) like Classes, objects, methods.

### Creating Classes :

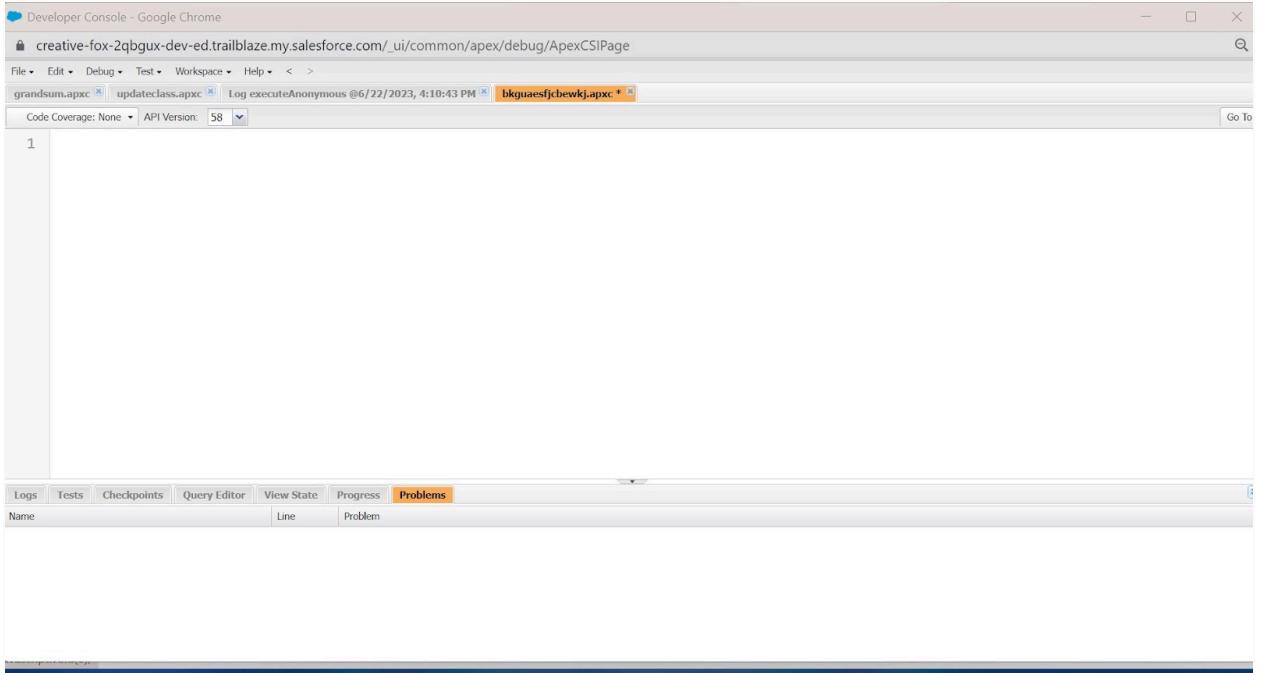
Apex classes are modeled on their counterparts in Java. You'll define, instantiate, and extend classes, and you'll work with interfaces, Apex class versions, properties, and other related class concepts.

- Class:
- As in Java, you can create classes in Apex. A class is a template or blueprint from which objects are created. An object is an instance of a class.
- Object

Object is an instance of a class, where it can access all the properties that are present in a class i.e, variables and methods.

### Steps to create a class in APEX:

1. Login to the trailhead account and navigate to the gear account in the top right corner.
2. Then we can see the Developer console. Click on the developer console and you will navigate to a new console window.



3. Then you can see many tools in the Toolbar of the new console window. Click on File, New and Apex Class.
4. Enter the name of the class to create a new class file.

#### Access specifiers in Apex :

Apex allows you to use the private, protected, public, and global access modifiers when defining methods and variables.

While triggers and anonymous blocks can also use these access modifiers, they aren't as useful in smaller portions of Apex. For example, declaring a method as global in an anonymous block doesn't enable you to call it from outside of that code.

#### Private:

This access modifier is the default, and means that the method or variable is accessible only within the Apex class in which it's defined. If you don't specify an access modifier, the method or variable is private.

#### Protected:

This means that the method or variable is visible to any inner classes in the defining Apex class, and to the classes that extend the defining Apex class. You can only use this access modifier for instance methods and

member variables. This setting is strictly more permissive than the default (private) setting, just like Java.

#### Public :

This means that the method or variable is accessible by all Apex within a specific package. For accessibility by all second-generation (2GP) managed packages that share a namespace, use public with the `@NamespaceAccessible` annotation. Using the public access modifier in no-namespace packages implicitly renders the Apex code as `@NamespaceAccessible`.

#### Global

This means the method or variable can be used by any Apex code that has access to the class, not just the Apex code in the same application. This access modifier must be used for any method that must be referenced outside of the application, either in SOAP API or by other Apex code. If you declare a method or variable as global, you must also declare the class that contains it as global. This is how a new class is created :

```
1 ▾ public class Student {
2
3 }
```

#### Triggers :

A trigger is a set of Apex code that runs before or after DML(Data Manipulation Language) events. A DML event could be a variety of data processing tasks that include the standard insert, update, and delete commands.

With Apex triggers, you can automate tasks that would otherwise be nearly impossible to accomplish using only the Salesforce user interface. Triggers enable you to create custom scripts that you can implement according to your needs, and the only limitation is your coding skills.

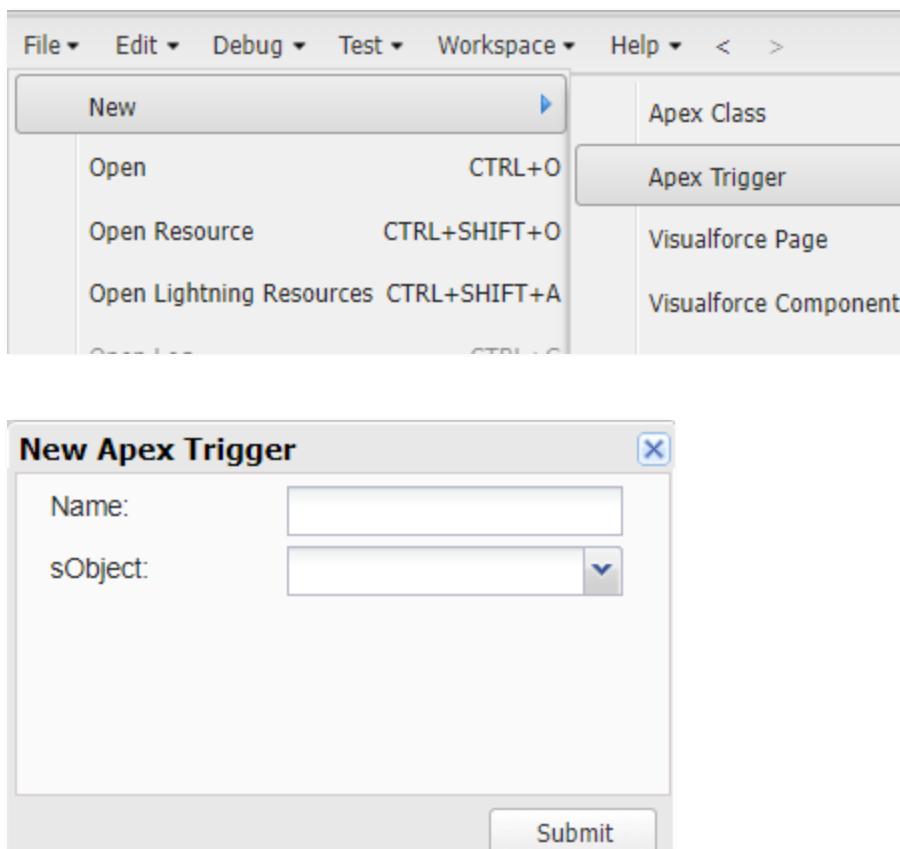
There are two Salesforce Apex trigger types:

Before triggers. These are helpful in cases that require a validation process before accepting a change. They run before any database changes. After triggers. These are helpful in cases where you need to modify your database records and when the necessary value is stored in other records. They run after any database changes. Both types will help you perform custom tasks and manage records effectively. They can help you perform

bulk actions as they can handle several records simultaneously.

How to create a new trigger :

1. While still in the trailhead account, navigate to the gear icon in the top right corner.
2. Click on developer console and you will be navigated to a new console window.
3. Click on the File menu in the toolbar, and click on new- Trigger.
4. Enter the trigger name and the object to be triggered.



Syntax For creating trigger :

The syntax for creating trigger is :

Trigger [trigger name] on [object name]( Before/After event)

```
{  
}
```

```

1 trigger LaptopBooking on Laptop_Bookings__c (After insert,after update) {
2
3     if(trigger.isAfter && ( trigger.isInsert || trigger.isupdate))
4     {
5         LaptopBookingHandler.sendEmailNotification(trigger.new);
6     }
7
8 }
9

```

Trigger code:

```
trigger LaptopBooking on Laptop_Bookings__c (After insert,after update) {
```

```

if(trigger.isAfter && ( trigger.isInsert || trigger.isupdate))
{
    LaptopBookingHandler.sendEmailNotification(trigger.new);
}
```

Note:- copy the API names

- 1.LaptopBooking - trigger name
- 2.Laptop\_Bookings\_\_c - as per your org(go to laptop bookings object and copy from that object api name).

Handler Class:

```

1 public class LaptopBookingHandler {
2     public static void sendEmailNotification (List<Laptop_Bookings__c> lapList){
3         for(Laptop_Bookings__c lap:lapList)
4         {
5             Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
6             email.setToAddresses( new List<String>{lap.Email__c});
7             email.setSubject('Welcome to our company');
8             string body = 'Dear ' +lap.Name +', \n';
9             body += 'Welcome to Laptop Rentals! You have been seen as a valuable customer to us.\n Please continue your journey with us, while
10             email.setPlainTextBody(body);
11             Messaging.sendEmail(new List<Messaging.SingleEmailMessage>{email});
12         }
13     }
14 }
15
16

```

Logs Tests Checkpoints Query Editor View State Progress Problems

Name Line Problem

Code Snippet :

```

public class LaptopBookingHandler {
    public static void sendEmailNotification (List<Laptop_Bookings__c> lapList){
        for(Laptop_Bookings__c lap:lapList)
        {
            Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
            email.setToAddresses( new List<String>{lap.Email__c});
            email.setSubject('Welcome to our company');
            string body = 'Dear ' +lap.Name +', \n';
            body += 'Welcome to Laptop Rentals! You have been seen as a valuable customer to us.\n Please
continue your journey with us, while we try to provide you with good quality resources. \n Laptop Amount =
+ lap.Amount__c + '\n core type =' +lap.core__c +' \n Laptop type =' +lap.Laptop_type__c;
            email.setPlainTextBody(body);
            Messaging.sendEmail(new List<Messaging.SingleEmailMessage>{email});

        }
    }
}

```

Note:-

1. Class name:- LaptopBookingHandler
2. API Name:- Laptop\_Bookings\_\_c(as per your org go to laptop booking object and copy from that).

- 3.core\_\_c (as per your org go to laptop booking object and copy from that).
- 4.Laptop\_type\_\_c.(as per your org go to laptop booking object and copy from that).

In this project , trigger is called whenever the particular record's sum exceeds the threshold i.e minimum business requirement value. Then the code in the trigger will get executed.

Result:

The screenshot shows a Gmail inbox with the following details:

- Subject:** Welcome to our company [Inbox]
- From:** vicky project via nhvs1ucf7yj86snh.rw96uyo.5j-d9o7weav.ap27.bnc.salesforce.com to me ▾
- Message Content:**

Dear smartinternz,  
 Welcome to Laptop Rentals! You have been seen as a valuable customer to us.  
 Please continue your journey with us, while we try to provide you with good quality resources.  
 Laptop Amount = 5100.0  
 core type = core i5  
 Laptop type = Hp
- Buttons:** Reply, Forward

Note: Before creating reports just fill the 10-12 records in the Laptop Bookings object.

Create records for each one you have to create at least 2 different records i.e dell(i3), dell(i7),acer(i3),hp(i5),mac(bionic chip).

**Edit Decision**

*Label mac field should be updated	*API Name mac_field_should_be_updated
Description   	
Outcomes For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.	
<b>OUTCOME ORDER</b> <span style="font-size: small;">(i) +</span> <span style="font-size: small;">mac laptop</span>	<b>OUTCOME DETAILS</b> *Label mac laptop *Outcome API Name mac_laptop Condition Requirements to Execute Outcome All Conditions Are Met (AND) Resource: \$Record > core type Equals Value: Bionic chip + Add Condition
<span style="border: 1px solid #ccc; padding: 2px 5px; border-radius: 5px; color: #0072bc; text-decoration: none; font-weight: bold;">Cancel</span> <span style="border: 1px solid #0072bc; padding: 2px 5px; border-radius: 5px; color: white; background-color: #0072bc; text-decoration: none; font-weight: bold;">Done</span>	

Click done.

6. Go to flow page
7. Beside Mac there is a symbol ‘+’ click on that.
8. Again select decision
9. Enter the Details Label:Mac months selected , API name: Gets Automatically Generated.
10. Enter the Outcome Details Label: mac bionic chip(1) , Outcome API name: Gets Automatically Generated.
11.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: 1.
12. Enter the Outcome Details Label: mac bionic chip(1) , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: Select 2..
13. Click ‘+’ outcome details
14. Enter the Outcome Details Label: mac bionic chip(1) , Outcome API name: Gets Automatically

Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 3..

15. Click ‘+’ outcome details

16. Enter the Outcome Details Label: mac bionic chip(1) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 4.

17. Click ‘+’ outcome details

18. Enter the Outcome Details Label: mac bionic chip(1) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.
- Operator: Select Equals.
- Value: Select 5.

Edit Decision

**mac months selected** (mac\_months\_selected)

**Outcomes** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME ORDER	OUTCOME DETAILS	Delete Outcome
mac bionic chip(1)	*Label mac bionic chip(1) *Outcome API Name mac_bionic_chip_1	
mac bionic chip(2)		
mac bionic chip(3)	Condition Requirements to Execute Outcome All Conditions Are Met (AND)	
mac bionic chip(4)		
mac bionic chip(5)	Resource \$Record > how many months Operator Equals Value 1	
Default Outcome	+ Add Condition	
	When to Execute Outcome <input checked="" type="radio"/> If the condition requirements are met <input type="radio"/> Only if the record that triggered the flow to run is updated to meet the condition requirements	

**Cancel** **Done**

Click done.

19. After mac bionic chip(1) there is ‘+’ symbol like mac bionic chip(2), mac bionic chip(3), mac bionic chip(4),mac bionic chip(5).

20. Click on ‘+’ then select update records

21. Enter the Details Label: one month of mac rate , API name: Gets Automatically Generated.

22. Field:- Amount\_c , value:- for one month of mac bionic chip rate-1700, two month of mac bionic chip rate-3400, three month of mac bionic chip rate-5100, four month of mac bionic chip rate-6800, five month of mac bionic chip rate-8500. Follow for all these finally

**Edit Update Records**

---

**\* How to Find Records to Update and Set Their Values**

- Use the laptop bookings record that triggered the flow
- Update records related to the laptop bookings 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

---

**Set Filter Conditions**

Condition Requirements to Update Record

None—Always Update Record ▾

---

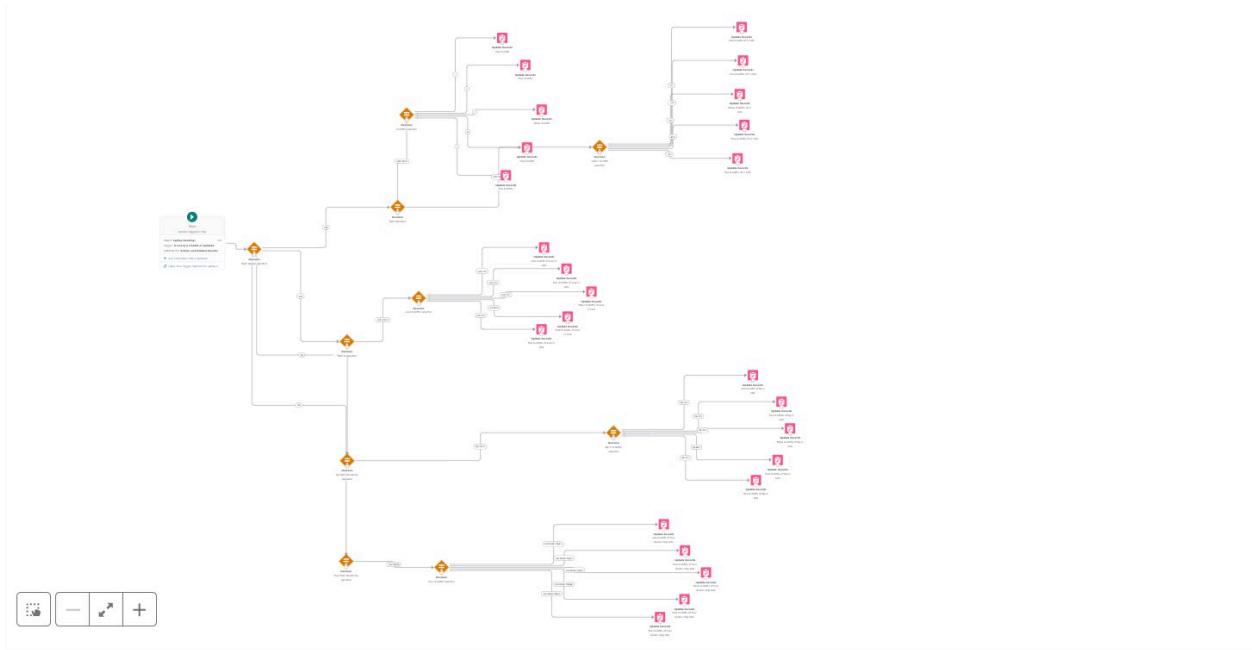
**Set Field Values for the Laptop Bookings Record**

Field	Value
Amount_c	← 2000 <span style="font-size: small;">Delete</span>

Cancel
Done

Click done.

FLOW:



Click on save .

Label:- Laptop distributions, api name:- automatically filled

Save the flow and activate it.

## 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.

In Salesforce.com we can easily generate reports in different styles. And can create reports in a very short time and also schedule the reports. Salesforce provides a powerful suit of analytic tools to help you organize, view and analyze your data.

### Types of Reports in Salesforce

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

1. Tabular Reports: Simple listing of data without any subtotals. This type of reports provide you most basically

to look at your data. Use tabular reports when you want a simple list or a list of items with a grand total.

Example: This type of reports are used to list all accounts, List of contacts, List of opportunities.....etc.....

2. Summary Reports: This type of reports provide a listing of data with groupings and sub totals. Use summary reports when you want subtotals based on the value of a particular field or when you want to create a hierarchically grouped report, such as sales organized by year and then by quarter.

Example: All opportunities for your team sub totaled by Sales Stage and Owner.

3. Matrix Reports: This type of reports allow you to group records both by row and by column. A comparison of related totals, with totals by both row and column. Use matrix reports when you want to see data by two different dimensions that aren't related, such as date and product.

Example: Summarize opportunities by month vertically and by account horizontally.

4. Joined Reports: Blocks of related information in a single report. This type of reports enable you to adopt five different blocks to display different types of related data. Each block can own unique columns, summary fields, formulas, filters and sort order. Use joined reports to group and show data from multiple report types in different views.

Example: You can build a report to show opportunity, case and activity data for your accounts.

## Create Report

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

3. Select report type from category or from report type panel or from search panel “consumer with Laptop Bookings and total laptops” >> click on start report.

Report Type Name	Category
Activities with consumers	Standard
Total Laptops with Laptop Bookings and consumers	Standard
consumers	Standard
consumers with Laptop Bookings and Total Laptops	Standard
consumers with Billing Process	Standard
consumers with Billing Process and Laptop Bookings	Standard
consumer History	Standard

4. Customize your report

5. Add fields from left pane as shown below

The screenshot shows a Report Builder interface for a dataset named 'consumer with laptops and total laptops' under 'consumers with Laptop Bookings and Total Laptops'. The report is grouped by 'types of versions'. The main table has columns: Laptop Bookings: Laptop Bookings Name, consumer: consumer Name, Total no of I..., Laptop names, core type, and Amount. A subtotal row is present, followed by another group labeled 'high' with 8 rows. The interface includes a sidebar for 'Groups' and 'Columns', and various reporting options like 'Add Chart', 'Save & Run', and 'Run'.

Follow the above image group rows and columns.

6. Click the column drop down and select bucket list.

The 'Edit Bucket Column' dialog box is open, showing a configuration for the 'Amount' field. The 'Bucket Name' is set to 'types of versions'. The table below defines four buckets based on ranges of 'Amount':

	Range	Bucket
<input type="button" value="Add ▶"/>	<= 900	basic
<input type="button" value="Add ▶"/>	> 900 to 1500	intermediate
<input type="button" value="Add ▶"/>	> 1,500 to 10000	high
<input type="button" value="Add ▶"/>	> 10,000	very high

At the bottom, there is a checked checkbox 'Treat empty Amount values in the report as zeros.' and two buttons: 'Cancel' and 'Apply'.

Click apply it.

Follow the picture and save or run it.

The screenshot shows a data reporting interface with the following details:

- Report Title:** Report: consumers with Laptop Bookings and Total Laptops  
consumer with laptops and total laptops
- Total Records:** 11
- Total Amount:** ₹81,600
- Table Headers:** consumer: consumer Name, Laptop Bookings: Laptop Bookings Name, Address, phone number, Laptop names, core type, types, Amount, types of versions.
- Data Rows:**
  - rakesh (1)**: smartinternz, warangal, 7894561235, Mac, Bionic chip, high, ₹8,000, high
  - Subtotal**
  - rushi (2)**: google, gachibowli, 7538964123, Dell, core i5, high, ₹6,000, high
  - Subtotal**
  - sunny (1)**: smartinternz, madhapur, 7534127896, Dell, core i3, high, ₹3,000, high
  - Subtotal**
  - sure (4)**: smartinternz, kphb, 7895621430, Acer, core i5, high, ₹6,500, high
  - pandora, kphb, 7895621430, Hp, core i7, high, ₹11,000, very high
  - stacknexus, kphb, 7895621430, Acer, core i7, high, ₹9,000, high
  - stacknexus, kphb, 7895621430, Acer, core i3, high, ₹3,600, high
  - Subtotal**
  - uday (3)**: pandora, uppal, 7894561230, Hp, core i5, high, ₹8,500, high
- Buttons:** Row Counts, Detail Rows, Subtotals, Grand Total.

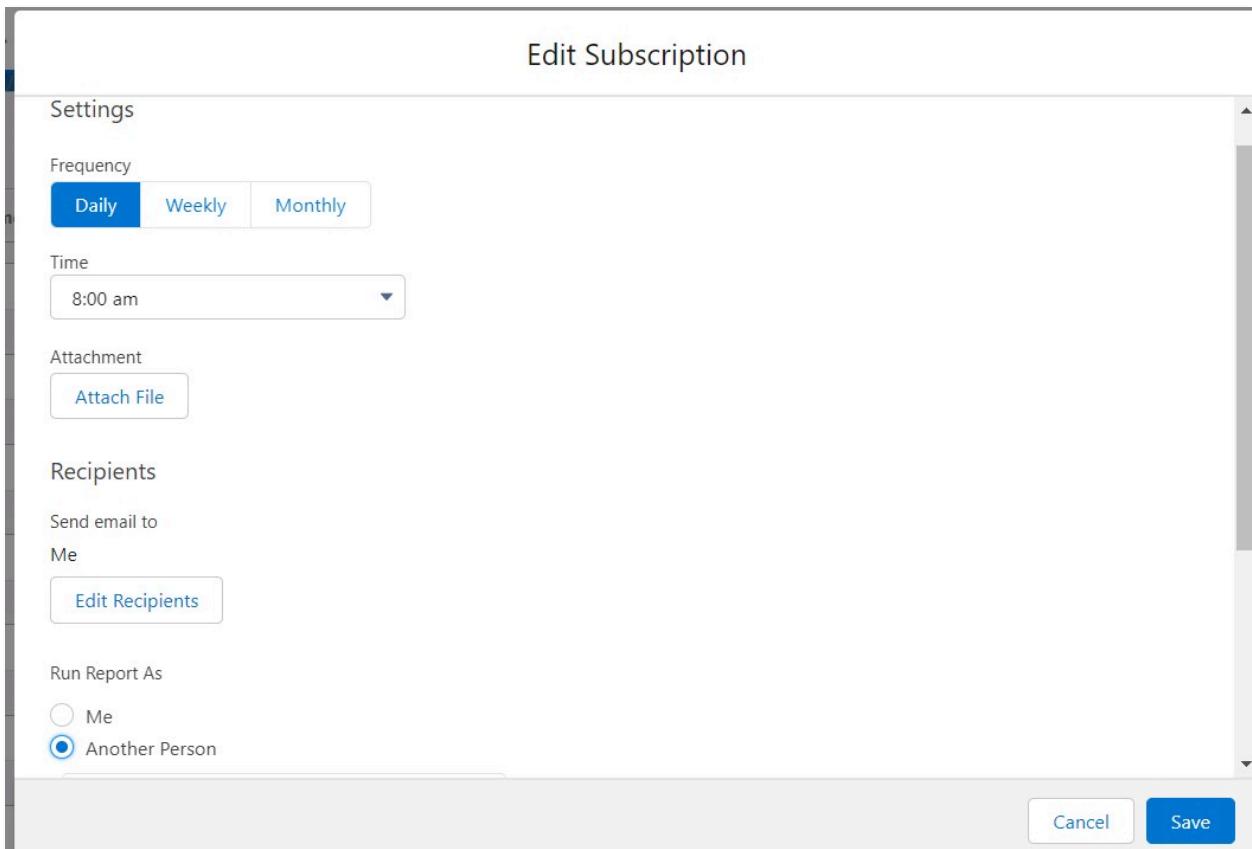
## Sharing report to owner

1. Click edit drop down and select subscribe option

The screenshot shows the same reporting interface with the following changes:

- Edit Drop-down:** A context menu is open at the top right, showing options: Save As, Save, Subscribe, Export, Delete, and Add to Dashboard. The "Subscribe" option is highlighted.
- Table Headers:** Laptop Bo..., con..., Total no of lapt..., Laptop available, Address, phone number, Email, Laptop names, core type, Amount.
- Data Rows:** The same 12 rows from the previous screenshot are listed.

2. Follow as per below image.



3. After selecting the run report as a “another person” select your personal account or whom you want to send that mail to.
4. Click save.

NOTE: The owner gets daily email notification of that laptop booking report.so that he can see all data remotely.

## 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.

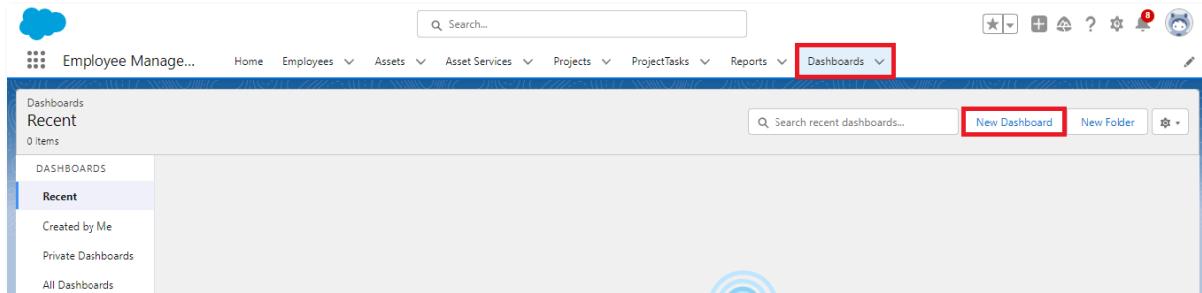
## Create Dashboard Folder

1. Click on the app launcher and search for the dashboard.
2. Click on the dashboard tab.

3. Click the new folder, give the folder label as “total rent amount”.
4. Folder unique names will be auto populated.
5. Click save.

## Create Dashboard

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



2. Give a Name and select the folder that was created, and click on create.

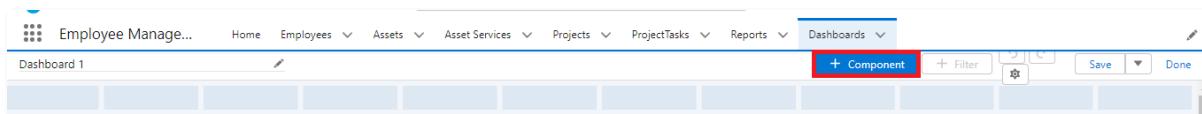
**New Dashboard**

\* Name  
data analytics of laptops

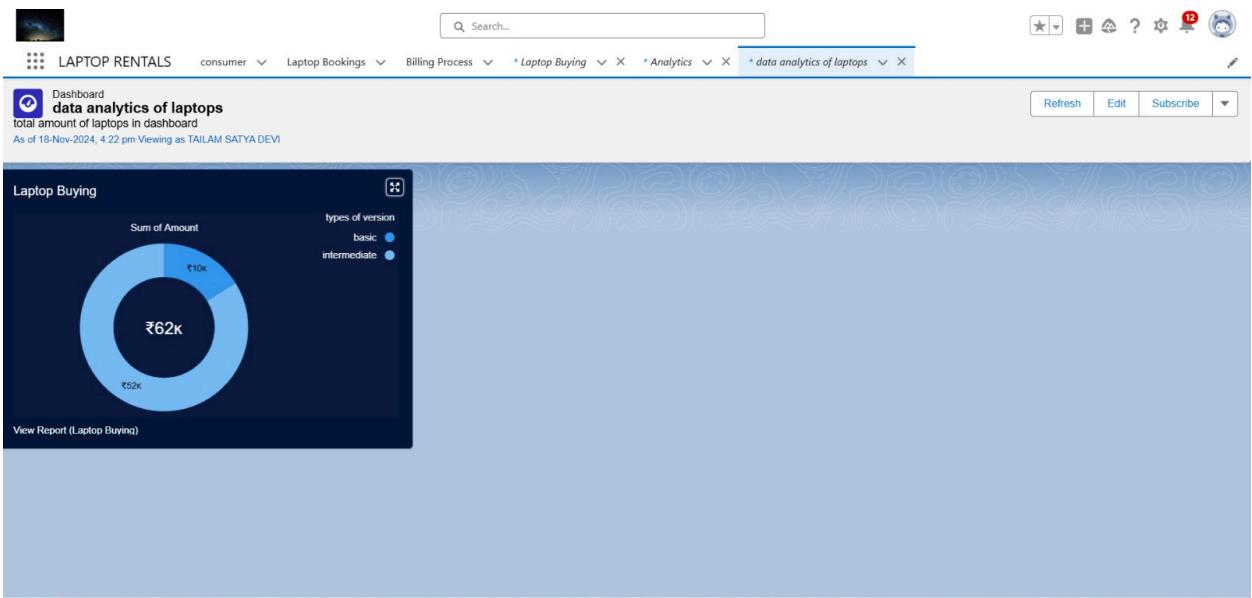
Description  
total amount of data in dashboards

Folder  
total rents amount

3. Select add component.



4. Select a Report and click on select.



5. Select the dark component and add to the dashboards.
6. Save it.
7. Click done.