

# **Project Report**

**Project Title: Laptop Rental System**

**Platform: Salesforce**

**Name :Bandi Yamuna**

**College: Lakkireddy BaliReddy College Of Engineering**

**Mail: 19400cm004@gmail.com**

# A CRM APPLICATION FOR LAPTOP RENTALS

## Project Overview

### Abstract:

The **Laptop Rentals** project aims to create a Salesforce-based solution to manage laptop rentals efficiently. By leveraging custom objects, automation, and dynamic reporting, the project addresses challenges such as real-time inventory tracking, automated notifications, and rental scheduling. This system is designed to enhance operational efficiency, ensure accurate data handling, and provide a seamless experience for administrators and users. The solution supports scalability, making it adaptable to future enhancements.

### Detailed Overview:

The **Laptop Rentals** project focuses on automating and streamlining laptop rental processes within an organization. It leverages Salesforce features such as custom objects, validation rules, flows, and dashboards to improve efficiency and accuracy. The project aims to:

- Maintain accurate inventory records.
- Automate booking and notification processes.
- Provide real-time insights into laptop availability and usage.

This solution not only enhances resource management but also aligns with organizational goals of improving service quality and scalability.

## Objectives

### Business Goals:

- Improve resource management for laptop rentals.
- Enhance customer satisfaction by automating notifications and ensuring timely responses.

### Specific Outcomes:

- A scalable solution for managing rental records.
- Automated notifications for booking confirmations and inventory updates.
- Real-time tracking of laptop availability and usage.

## Salesforce Key Features and Concepts Utilized

### 1. Objects:

- Created a custom object named **LaptopBookings** to manage rental records.
- Key fields include:
  - **Laptop Name** (Text)
  - **Core Type** (Picklist: e.g., i3, i5, i7)
  - **Amount** (Currency)
  - **Total No. of Laptops** (Number)
  - **Laptops Available** (Formula Field).

### 2. Tabs:

- Added a custom tab for **LaptopBookings** to make it accessible in the **Laptop Rentals Lightning App**.
- Enables users to view, create, and manage booking records.

### 3. The Lightning App:

- Built a **Laptop Rentals App** using Salesforce Lightning Experience.
- Integrated the **LaptopBookings** tab, dashboards, and reports into the app for streamlined navigation and usability.

### 4. Fields:

- Configured fields on the **LaptopBookings** object for tracking rental data:
  - Example: The **Laptops Available** formula dynamically updates as bookings are made.

### 5. Validation Rules:

- Implemented validation rules to ensure data integrity, such as:
  - Preventing bookings when **Laptops Available** is 0.
  - Validating email address formats for notification purposes.

## **6. Profiles and Permissions:**

- Configured profiles to control access to the **LaptopBookings** object and other resources.
- Example: Admins can create and edit all records, while regular users can only view their own bookings.

## **7. Roles and Hierarchy:**

- Defined roles to manage data visibility:
  - Admins: Full access to all records.
  - Managers: View records of users in their hierarchy.
  - Users: View and edit their bookings.

## **8. Users:**

- Created users with appropriate roles and profiles to match project requirements.

## **9. Flows:**

- Automated processes using Salesforce Flows:
  - Example 1: **Laptop Booking Notification Flow**—Sends email notifications to users and admins.
  - Example 2: **Inventory Update Flow**—Reduces **Laptops Available** after a booking is confirmed.

## **10. Reports:**

- Built reports for real-time insights into rentals and inventory:
  - Examples:
    - **Rental Summary Report**: Total bookings by laptop type.
    - **Revenue Report**: Total earnings from rentals over time.

## **11. Dashboards:**

- Created dashboards to visualize key metrics:
  - Widgets like pie charts for laptop type distribution and bar graphs for monthly

rental trends.

## Detailed Steps to Solution Design:

### 1. Object Creation

#### Create Total Laptops Object:

- 1) From the setup page click on Object Manager > Click on Create > Click on Custom Object

1) Under the Label name > Total Laptops

2) Plural label name > Total Laptops

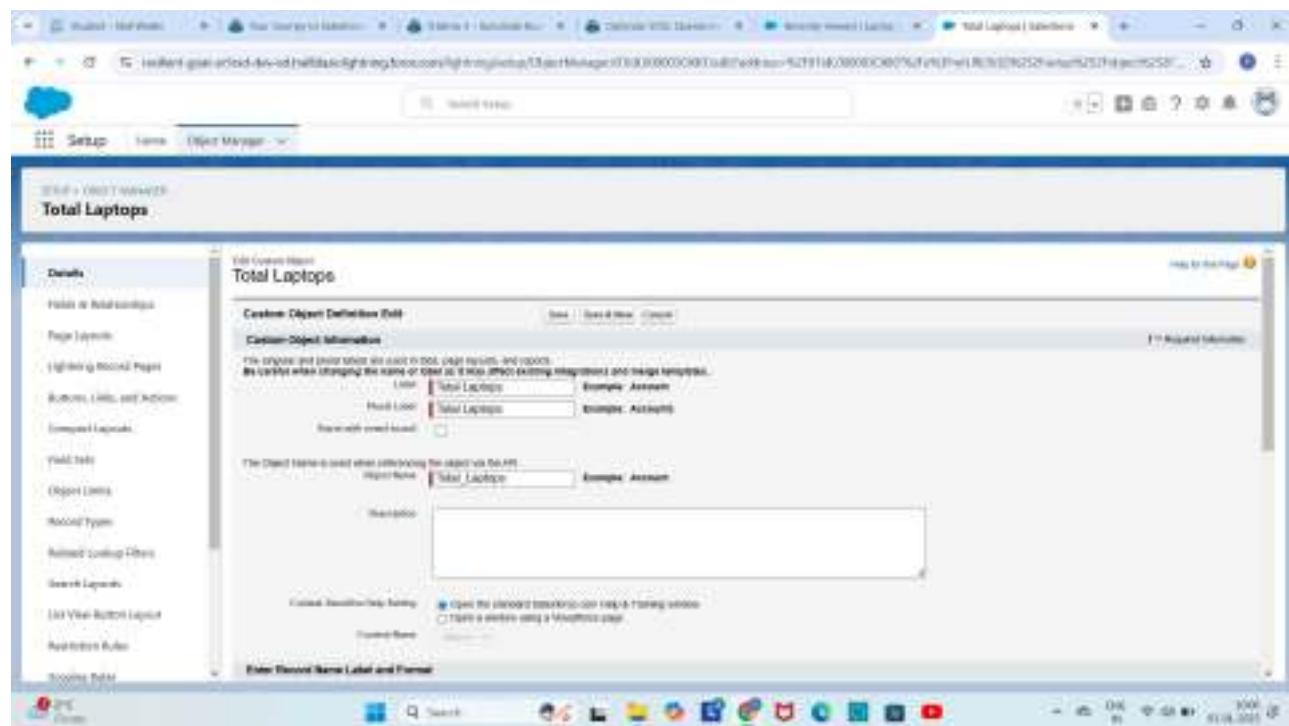
3) Under Record Name Label and Format

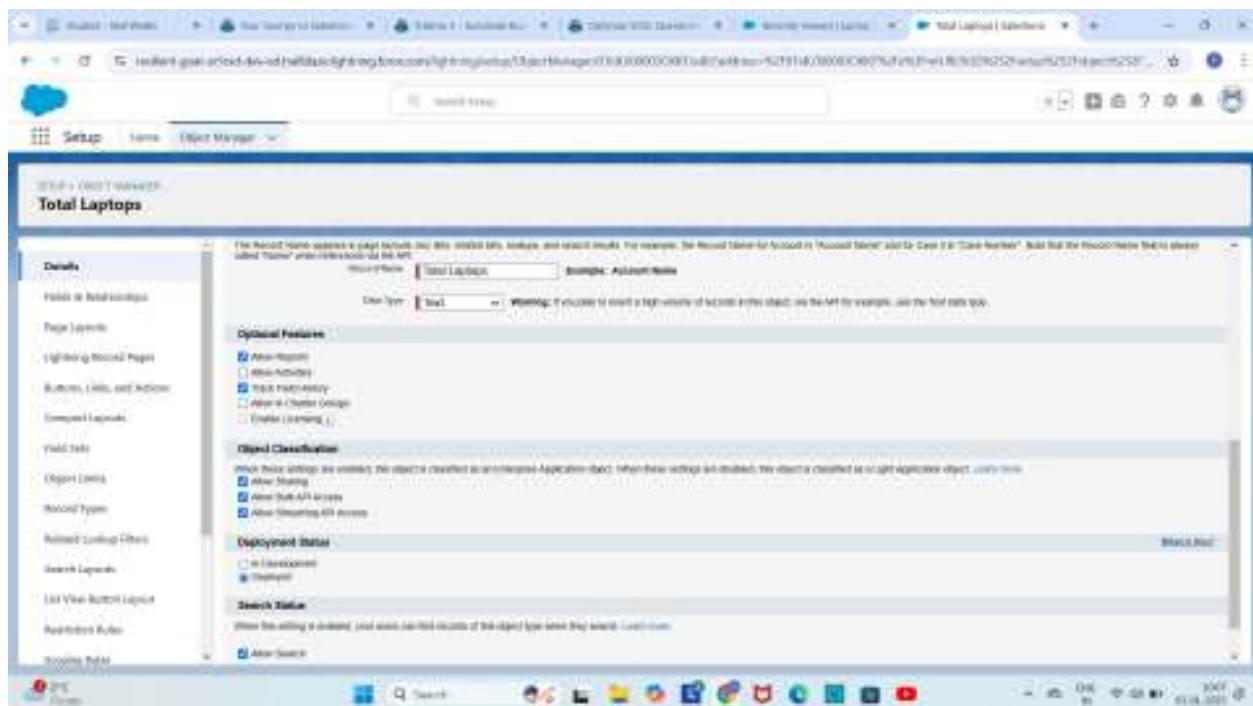
Record Name > Total Laptop

Date Type > Text

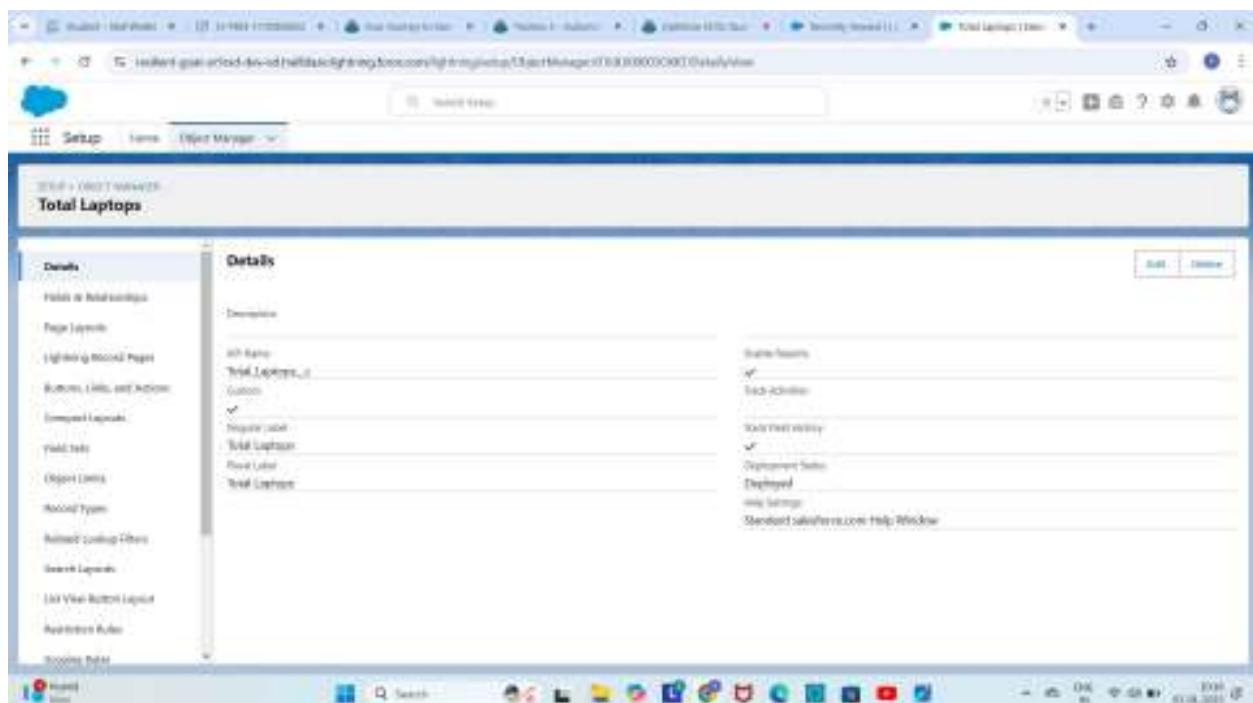
Click on Allow reports, Allow search and Track Field History

Allow search > Search





Total Laptops object created:



## Create consumer Object:

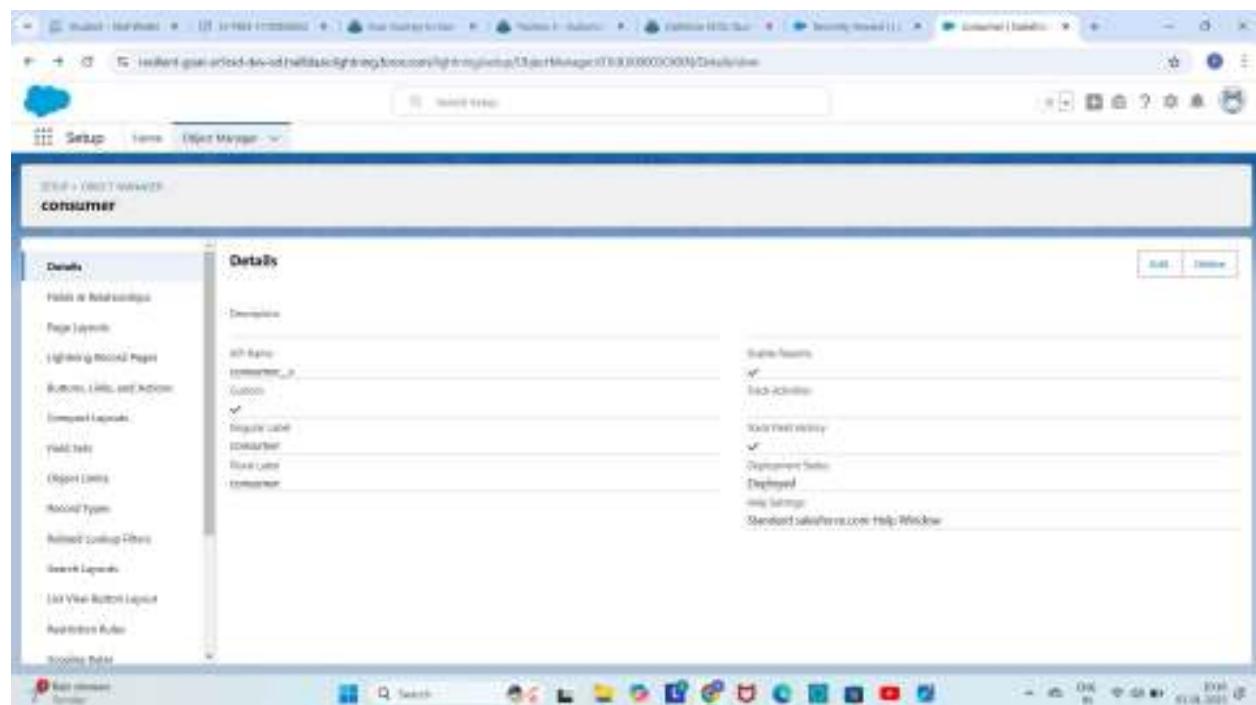
From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.

- 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

... Consumer object created successfully by following above steps:



## 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

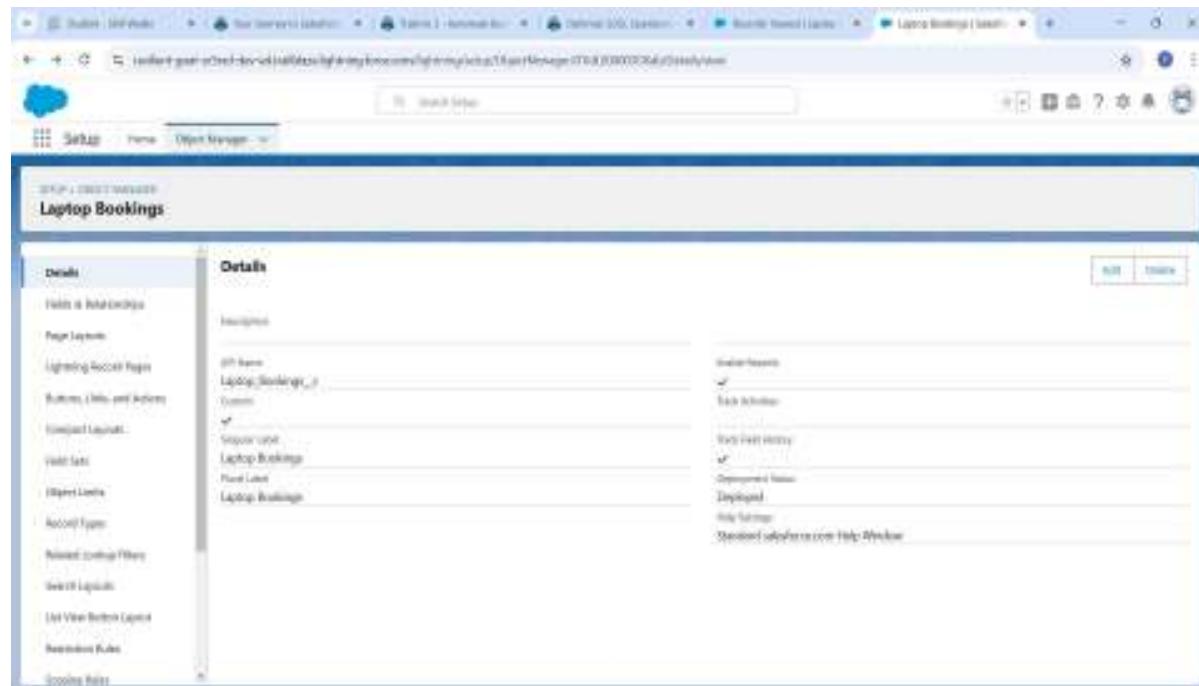
.... Record Name >> Laptop Bookings

.... Data Type >> Name

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

3. Allow search >> Save

Laptop Bookings object created successfully by following above steps:



## 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

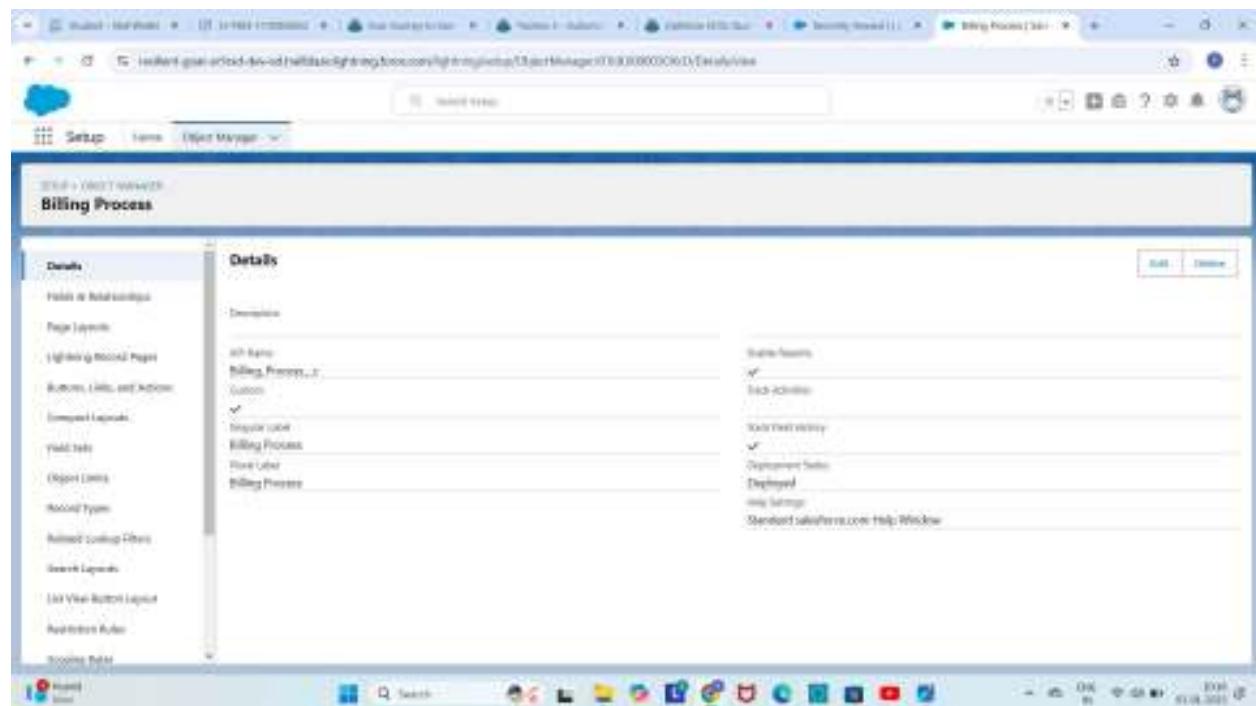
Record Name >> Billing Process Name

Data Type >> Name

4) Click on Allow reports, Allow search and Track Field History.

5) Allow search >> Save

Billing Process object created successfully by following above steps:



## Tabs

**What is 'Bulk' - A tool is like a user interface that is used to build records for objects and to view the records in the objects.**

# Creating a Custom Tab

1. Create a Rule(s)   
 Go to setup page >> Type Rule in Quick Find bar >> click on rule >> New (under custom object tab)

Select `ObjectList` Laptops >> Select the tab style >> Next {Add to profiles page} keep it as default >> Next {Add to Custom App}, uncheck the include tab.

Make sure that the Append tab to users' existing personal customizations is checked.

#### **flick wave**



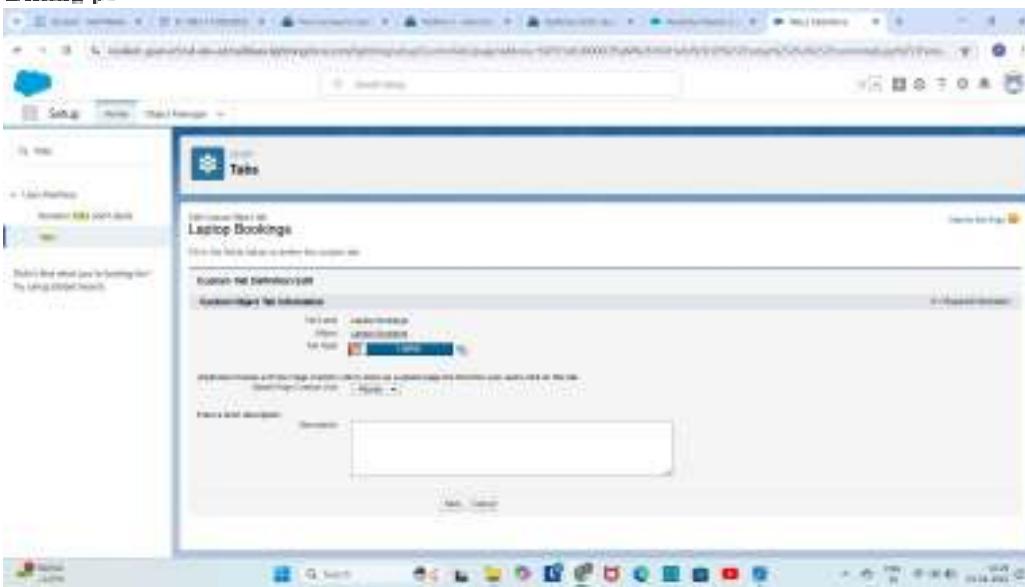


## Activity 2: Creating Remaining Tabs

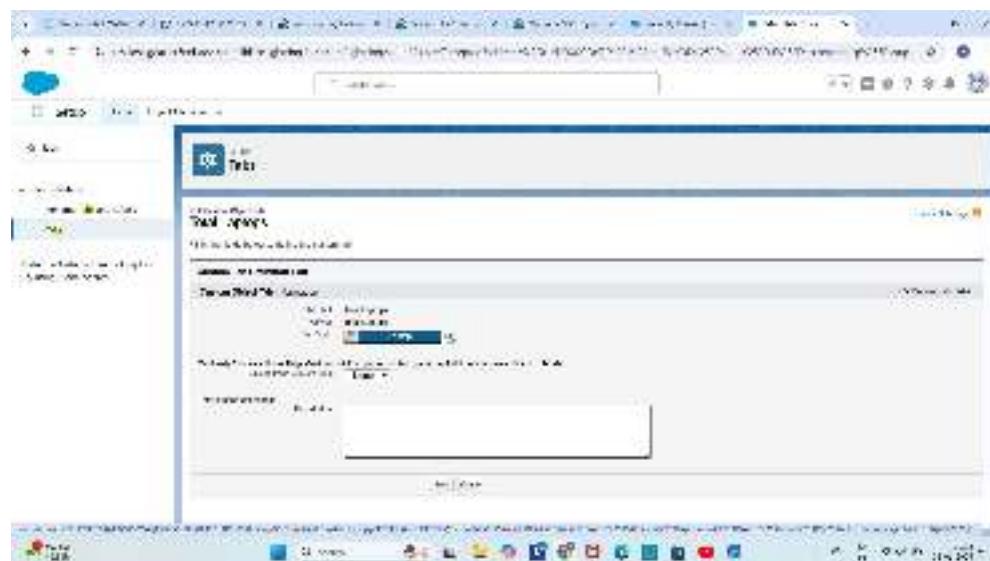
Now create the Tabs for the remaining Objects they are "Consumer,Laptop Booking,Billing process".

Follow the same steps as mentioned in Activity -1

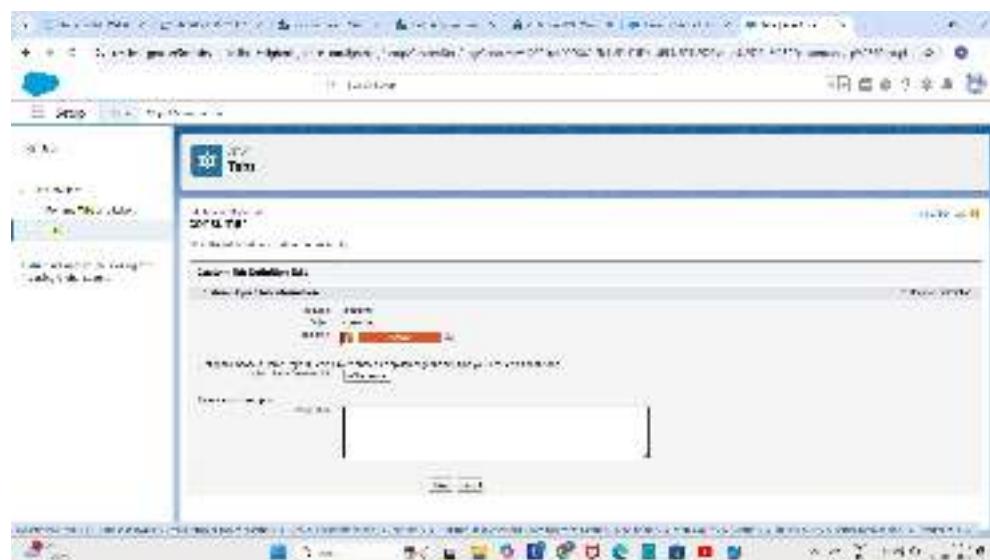
### Billing process Tab Created:



**Total Laptops Tab created:**



**Consumer Tab created:**



## Create a Lightning App

To create a lightning app page:

Go to setup page > search "app manager" in quick find > select "app manager" > click on New lightning App

The screenshot shows the Lightning Experience App Manager interface. On the left, there's a sidebar with navigation links like Setup, Home, Object Manager, and a search bar. The main area is titled "Lightning Experience App Manager" and displays a table of installed apps. The columns include App Name, Developer Name, Description, Last Modified, App Type, and Version. Some apps listed include "Data Loader", "Digital Experiences", "Dynamatrix", "Fileabinet IDE", "How We Sell Management", "LAPTOP RENTALS", "Marketing", "Playground", "Reporting", "Sales", "Service", and "Setup Console".

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.

This screenshot shows the "App Details" page for the "LAPTOP RENTALS" app. The left sidebar lists "App Settings", "App Details & Branding", "App Metrics", "Mobile", "Email", and "APIs". The main panel has tabs for "App Details" (selected), "App Options", and "App Utilities". Under "App Details", there are sections for "App Name" (set to "LAPTOP RENTALS"), "App Icon" (a yellow square icon with "Laptop Rent"), "App Description" (empty), "App URL" (empty), and "App Status" (set to "Enabled"). Below these are "App Details Form" and "App Utilities Form" sections, each containing a "Next Step" button.

Upload a photo that is related to your app

### To Add Navigation Items:

The screenshot shows the 'Add Navigation Items' dialog in Dynamics 365. The left pane displays a list of available items under 'Available Items', including 'Agent', 'Customer', 'Lead', 'Opportunity', 'Product', 'Sales Order', 'Task', and 'Ticket'. The right pane shows the 'Selected Items' list, which currently contains three items: 'Agent', 'Customer', and 'Lead'. At the bottom of the dialog, there is a toolbar with icons for 'Cancel', 'Save', 'Close', and 'Next'.

Select the items (Agent,Laptops,consumer,Laptop Booking,Billing Process) from the search bar and move it using the arrow button >> Next

### To Add User Profiles:

The screenshot shows the 'User Profiles' dialog in Dynamics 365. The left pane lists 'Available Profiles' such as 'Agent', 'Analytics Cloud Integration User', 'Analytics Cloud Security User', 'Authorized Vehicle', 'Authorized Vehicle', 'CRM Admin Portal Profile', 'Customer', 'Contact Manager', 'Data Dog Data Prof User', 'Customer Marketing Profile', and 'Customer Sales Profile'. The right pane shows the 'Selected Profiles' list, which currently contains one item: 'System Administrator'. At the bottom of the dialog, there is a toolbar with icons for 'Cancel', 'Save', 'Close', and 'Next'.

Select profile: 'System administrator' in the search bar > click on the arrow button > save & finish

# Fields:Creating the field in consumer object

## 1. To create fields in an object:

Go to setup >> click on Object Manager >> type object name(consumer) in search bar >> click on the object

Now click on "Fields & Relationships" >> New

Select Data Type as a "Phone"

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



## 2. To create another fields in an object:

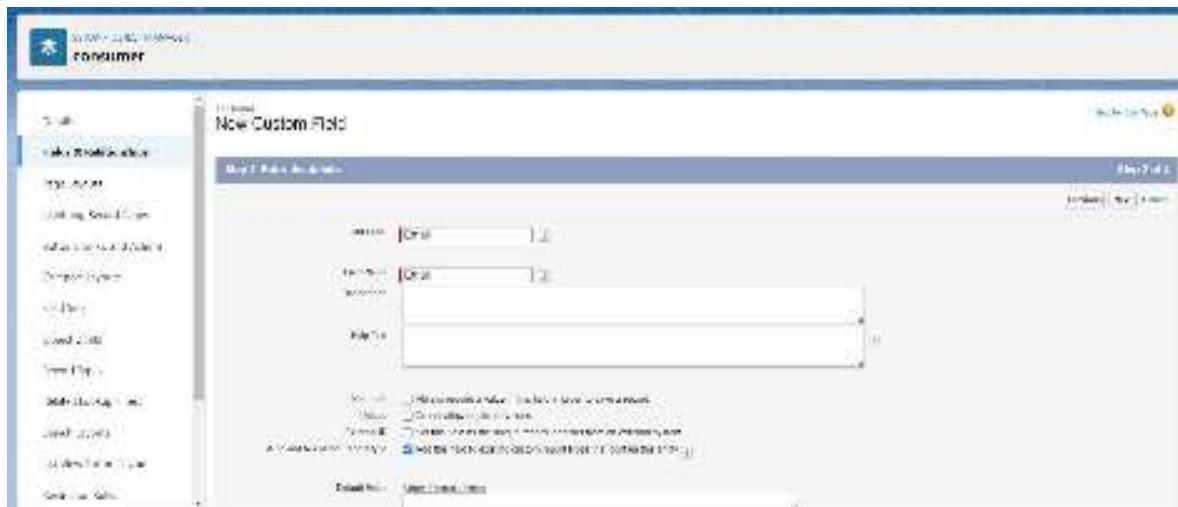
a. Go to setup >> click on Object Manager >> type object name(consumer) in search bar >> click on the object

b. Now click on "Fields & Relationships" >> New

c. Select Data type as a "Email" and Click on Next

d. Fill the Above as following:

- Field Label: Email
- Field Name: It's gets auto generated
- Click on Next >> Next >> Save and new



#### 3. To create another fields in an object:

- a. Go to setup > click on 'Object Manager' > type object name('consumer') in search bar > click on the object
- b. Now click on "Fields & Relationships" > New
- c. Select Data type as a "Text Area" and Click on Next
- d. 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



#### 4. To create another fields in an object:

Go to setup > click on 'Object Manager' > type object name('consumer') in search bar > click on the object

Now click on "Fields & Relationships" > New

Select Data type as a "Picklist" and Click on Next

Fill the 4 hour as following

Field Label: consumer\_Status

Value - Select enter values with each value separated by a new line

Student

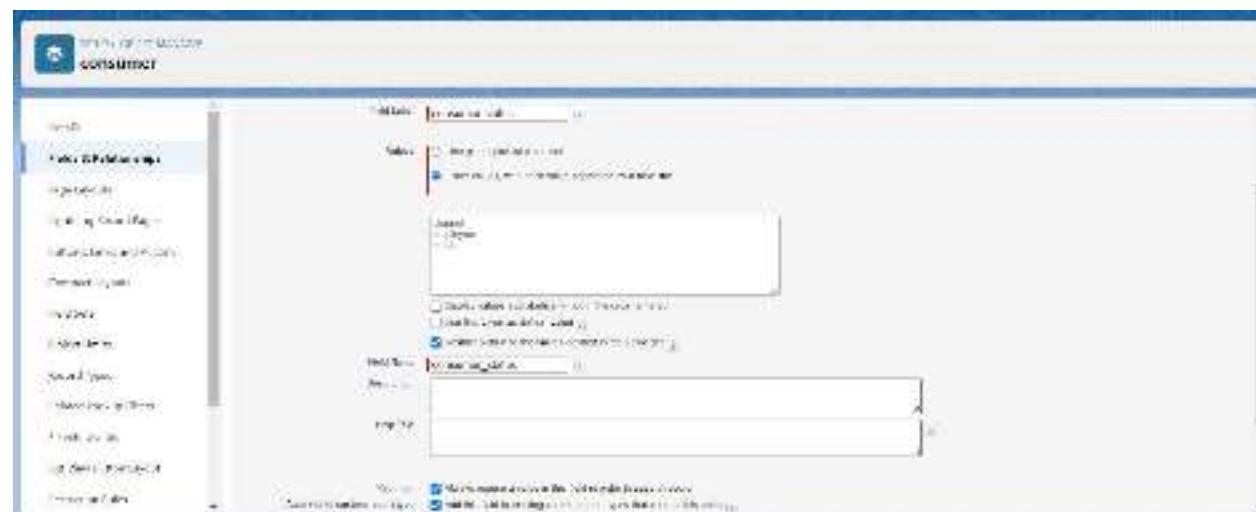
Employee

Others

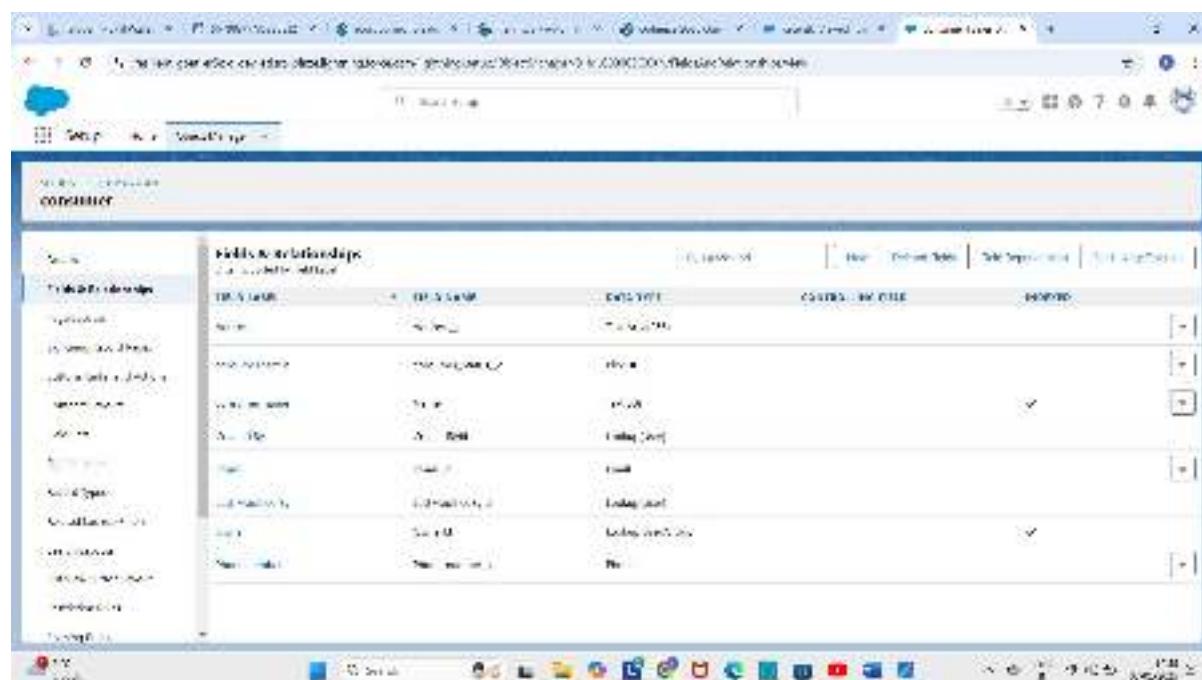
Select required

Field Name : It's auto generated

Click on Next >> Next >> Save and new



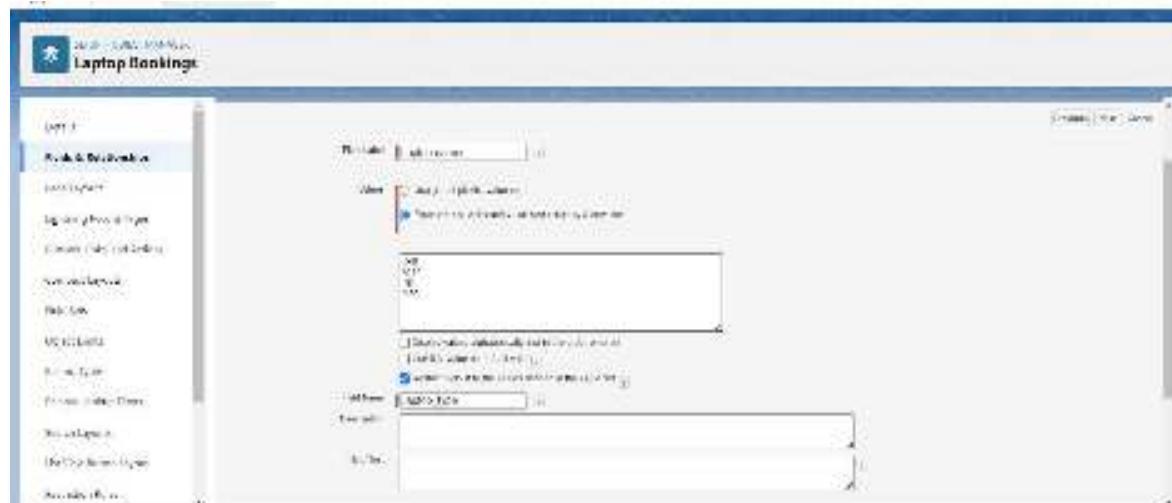
## Fields created in Consumer:



# Creating the field in Laptops Bookings object

## 1. In 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. Label Laptop Names
5. Picklist values are -1 Dell > Acer 3,11p > 4 Mac



6. Select required
7. Click on Next >> Next >> Save and now

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

### 1. 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" and Label Core Type
4. Picklist values are -1 Core i3 > Core i5 > Core i7 > Biomic Chip



#### 5. Select required

6. Click on Next >> Next >> Save and view

**NOTICE:**

#### **Field Dependency:**

A field dependency refers to a relationship between two fields in an object where the values of one field determine the available values for another field. Field dependencies are commonly used to ensure 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:**

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

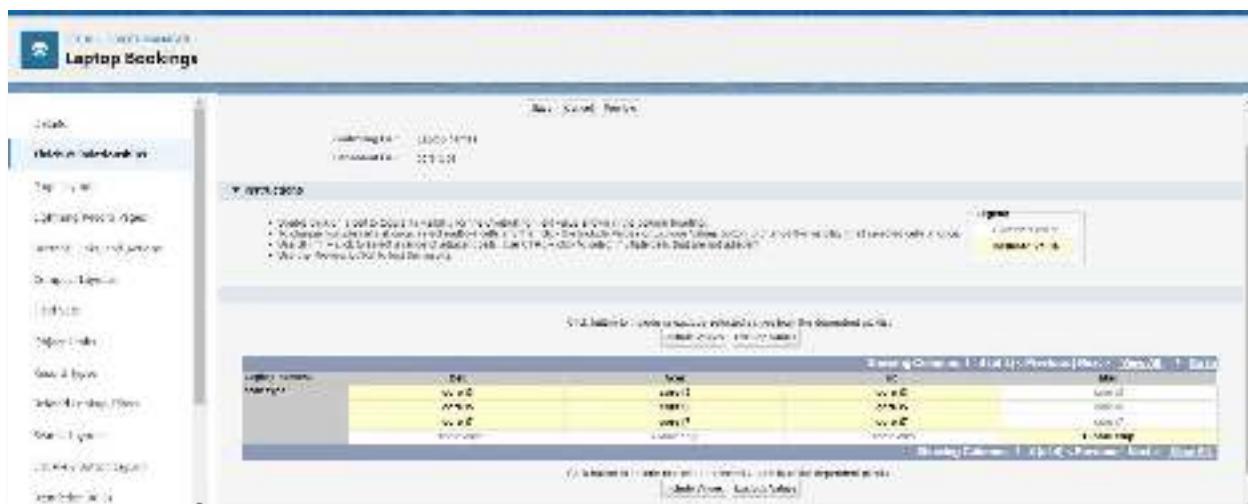
#### **3. To Create a Field Dependency in the Laptop Booking Object**

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

.... 2. click field dependency and next

.... 3. Select Controlling Field as Laptop Names and Dependent Field as Core i Type

.... 4. Click the include value for dell core i3,i5,i7 and for acer i3,i5,i7 and for hp i3,i5,i7 and also for macbook chip include the values for it



Click save

## To Create a Fields & Relationship to an Laptop Booking and Total Laptops Objects

### 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 "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. Fill the Above as following
  - Change the Field Label of consumer
  - Field Name will get auto generated

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
2. Now click on "Fields & Relationships" >> New
3. Select Data Type as a "Currency"
4. Click on Next

**Fill the & below as following**

- Field Label: Amount
  - Length: {18,0}
  - Field Name: H15 gets auto-generated
  - Click on Next > Next > Save and new...



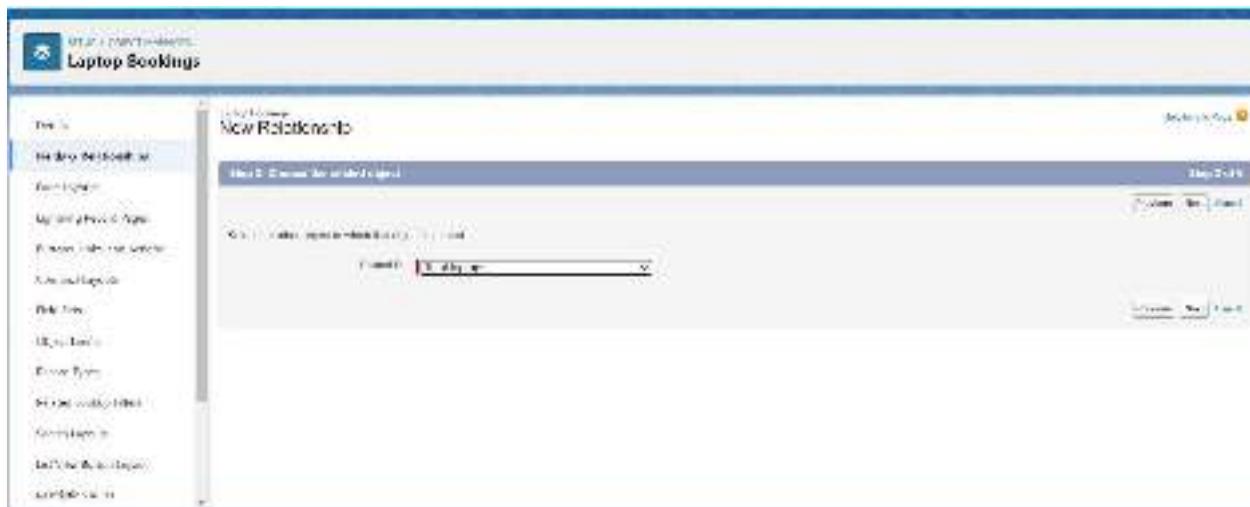
#### **To Create a Fields & Relationship to an Object**

Click setup > click on Object Manager > type object name (Lector Bookings) in the search bar > click on the object

**New** click on "Fields & Relationships" >> **New**

Select Data Type as a "Lookup Relationship"

Click on Next



Click on the Related to drop-down and Select the "Intel Laptops" object and click on Next.

- Fill the Above as following
- Change the Field Label "Serial No Of Laptop"
- Field Name will gets auto generated
- Click on Next > Next > Save and new

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

To create fields & relationship to an object

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

Now click on "Fields & Relationships" > New

Select Data Type as a "Email"

Click on Next and save it

#### Fields created in Laptop bookings

Fields & Relationships				
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD
Page Layouts	Record	RecordId	Character(40)	
Lightning Record Pages	Comments	Comments__r	Master-Detail[Comments]	
Buttons, Links, and Actions	Case Type	Case__c	Text	Case
Custom Labels	Created By	CreatedBy	Lookup	
Field Sets	Email Address	email__c	Email	
Object Limits	Last Activity	last_activity__c	Text	
Record Types	Last Activity Month	last_activity_month__c	Text	
Related Lookup Filters	Listing Bookings	Name	Text	
Search Layouts	Listing Name	LaptopName__c	Text	
List View Button Layout	Listing Number	LaptopNumber__c	Text	
Relationship Rules	Listing Status	Laptop_Status__c	Text	
Reporting Rules	Total No Of Listings	TotalNoOfListings__c	Master-Detail[Total Listings]	
Record Types	Laptop Number	Laptop_Number__c	Text	
Related Lookup Filters	Last Modified By	LastModifiedBy	Lookup	
Search Layouts	Name	Name__c	Text	
Custom Buttons Layout	Record Type	RecordType	Text	
Relationship Rules	Total No Of Listings	TotalNoOfListings__c	Master-Detail[Total Listings]	
Reporting Rules				

## "In Create a Rollup Summary Field in "Total Laptops Object"

After Creating the Master-Detail Relationship Than Only you can create the Rollup Summary

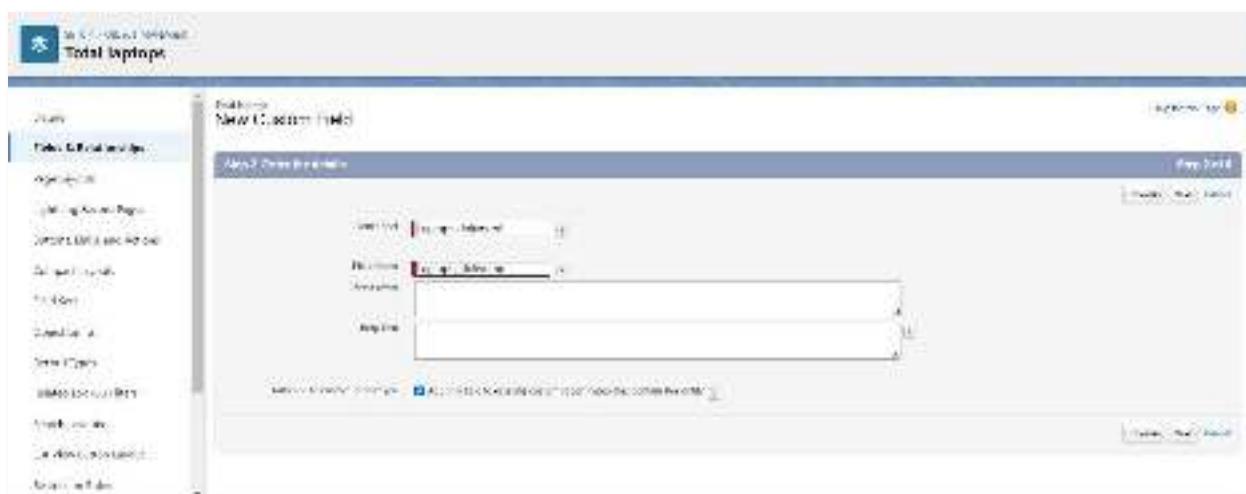
Go to setup >> click on Object Manager >> type object name('Total Laptops') in the search bar >> click on the object

Now click on "Fields & Relationships" >> New



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

- Fill the Above as following
- Field Label: Laptops deliveries
- Field Name: IT's gets with generated



- Click on Next

Select the Laptop Bookings in the Summarized Object

Select the count Radio button in the select Rollup Type



### Fields created in 'Total Laptops':

The screenshot shows the 'Fields & Relationships' list in the Dynamics 365 Object Manager for the 'Total Laptops' entity. The list includes the following fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	OPTIONS
Counted By	CountedBy	Lookup (Any)		
Laptops Available	Laptops_Available__c	Formula (Number)		
Owner Modified By	Laptops_DerivedFrom__c	Rolling Summary (COUNT Laptop Category)		
Name	Name	Text (250)		
Total Laptops	Name	Text (250)		

#### To create fields in an object:

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

Now click on "Fields & Relationships" >> New.

Select Data type as a "Formula" and Click on Next.

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 bar: '=SI - ' and Click on insert field then you will find a pop up 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
- '=SI - Total no of laptops - Laptops delivered' & " 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)

2. search bar >> click on the object.

3. Now click on "Fields & Relationships" >>New

4. Select Data type as a "picklist" and Label: how many months

..... Picklist values are 1, 2, 3, 4, 5

..... Click and save it

# Creation of Fields & Relationship for Billing Process Object

## 1. To create fields & relationship to an object

Go to setup >> click on Object Manager >> type object name(Billing Process) in the search bar >> click on the object.

Now click on "Fields & Relationships" >> New

Select Data Type as a "Master-detail Relationship"

Click on Next

Click on the Related to drop down and Select the consumer object and click on Next



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:

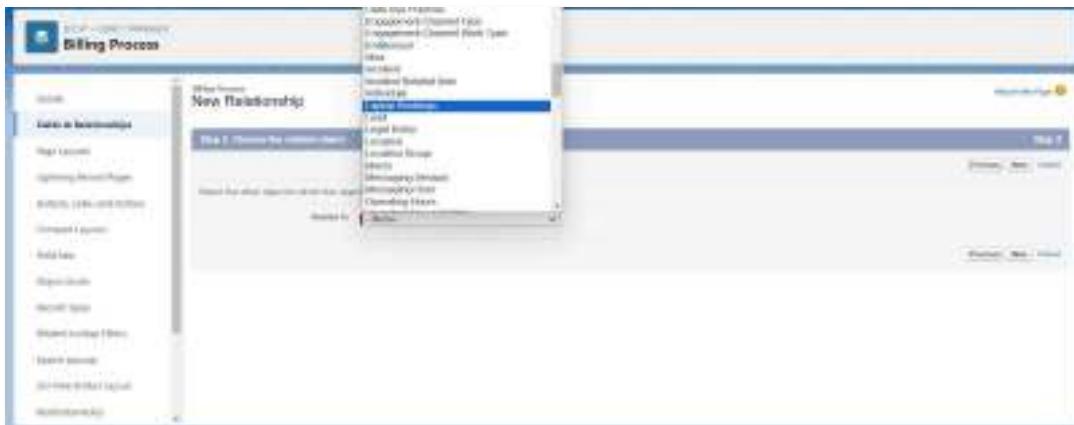
Go to setup >> click on Object Manager >> type object name(Billing Process) in the search bar >> click on the object

Now click on "Fields & Relationships" >> New

Select Data Type as a "Lookup Relationship"

Click on Next

Click on the Related to drop down and Select the Laptop Booking object and click on Next



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

**a. Creation of another fields for the billing process object**

**b. create fields in an object**

a. Go to setup >> click on ()bject Manager >> type object name(Billing Process) in the search bar >> click on the object

b. Now click on "Fields & Relationships" >> New

c. Select Data Type as a "Picklist"

d. Fill the Above as following

..... Field Label: Payment Mode

..... Value >> Select enter values with each value separated by a new line

..... Cash

..... Check

..... Credit card

..... Debit card

..... UPI

..... Phonepe

..... Gpay

..... Paytm

..... e... Select required

Click on Next >> Next >> Save and new

### Cross Object Formula Field:

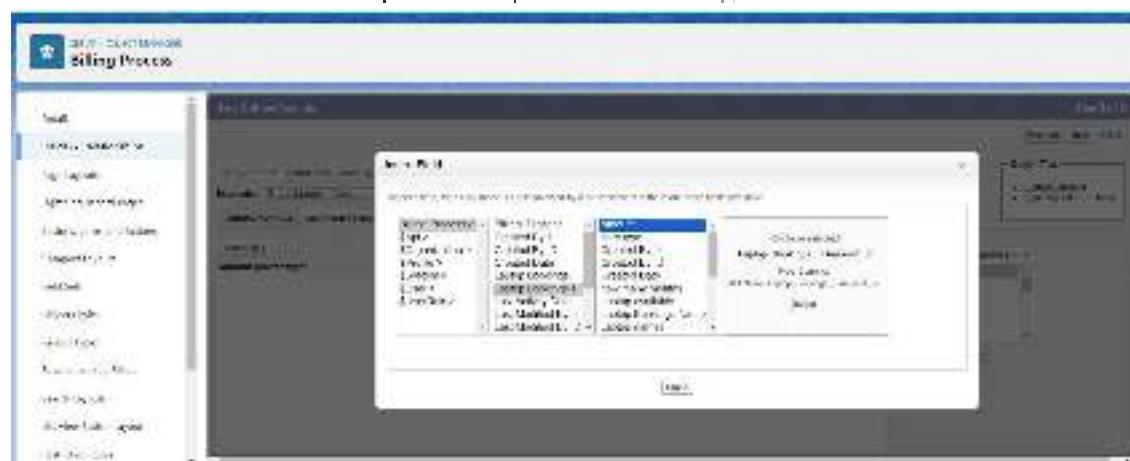
In Salesforce, a cross-object formula field allows you to create a formula that references fields from related objects. It enables you to perform calculations or display data from related records without the need for custom code or complex workflows.

### 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 ensure 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

- a. Go to setup >> click on ()bject Manager >> type object name(Billing Process) in the search bar >> click on the object
- b. Now click on "Fields & Relationships" >> New
- c. Select Data Type as a "Formula"
- d. Click on Next
- e. Under the Field label Amount, the Field name gets auto generated and click on Next.(Formula return type (Currency))
- f. 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
- g. "Laptop Booking .t.Amount - c"
- h. Click on the Check syntax: No syntax errors in merge fields



..... Click on Next >> New >> Saving new

## Fields Created in Billing Process:

The screenshot shows the Salesforce Object Manager interface for the 'Billing Process' object. The left sidebar lists various setup items like Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, and Visualforce Pages. The main area displays a table titled 'Fields & Relationships' with the following data:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	IMMEDIATE
Account	Account_c	Lookup (Parent)		
Billing Permissions	Name	Text		
Created By	CreatedBy	Lookups		
Laptop Rating	Laptop_Rating_c	Lookup (Parent)		
Last Modified By	LastModifiedBy	Lookups		
Name	Name_c	Text (Open-Ended)		
Payment Mode	Payment_Mode_c	Text		

## 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 pages >> 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 "Phonenumber/Emailnotblankrule"
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 )"
6. and check the SYNTAX.
7. Save the validation rule



Created validation rule in consumer



## Profiles:

### owner Profile

14. 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

Profile Details		CRM	Case	Opportunities	Activities
Name	CRM				
Email Address	CRM@CRM.com				
Mobile No.					
Office No.					
Created By	System Administrator - CRM (CRM)				
Modified By	System Administrator - CRM (CRM)				
Created Date	08/01/2012 10:50 AM				
Modified Date	08/01/2012 10:50 AM				
<b>Project Details</b>					
Open Projects (Count)	0	Open Projects (Count)	0	Open Projects (Count)	0
Completed Projects	0	Completed Projects	0	Completed Projects	0
On Hold Projects	0	On Hold Projects	0	On Hold Projects	0
Planned Projects	0	Planned Projects	0	Planned Projects	0
Active Projects	0	Active Projects	0	Active Projects	0
In Progress Projects	0	In Progress Projects	0	In Progress Projects	0
Delayed Projects	0	Delayed Projects	0	Delayed Projects	0
Approved Projects	0	Approved Projects	0	Approved Projects	0
Rejected Projects	0	Rejected Projects	0	Rejected Projects	0
Unassigned Projects	0	Unassigned Projects	0	Unassigned Projects	0
Archived Projects	0	Archived Projects	0	Archived Projects	0
Deleted Projects	0	Deleted Projects	0	Deleted Projects	0
Completed Tasks (Count)	0	Completed Tasks (Count)	0	Completed Tasks (Count)	0
Planned Tasks (Count)	0	Planned Tasks (Count)	0	Planned Tasks (Count)	0
Active Tasks (Count)	0	Active Tasks (Count)	0	Active Tasks (Count)	0
In Progress Tasks (Count)	0	In Progress Tasks (Count)	0	In Progress Tasks (Count)	0
Delayed Tasks (Count)	0	Delayed Tasks (Count)	0	Delayed Tasks (Count)	0
Approved Tasks (Count)	0	Approved Tasks (Count)	0	Approved Tasks (Count)	0
Rejected Tasks (Count)	0	Rejected Tasks (Count)	0	Rejected Tasks (Count)	0
Unassigned Tasks (Count)	0	Unassigned Tasks (Count)	0	Unassigned Tasks (Count)	0
Deleted Tasks (Count)	0	Deleted Tasks (Count)	0	Deleted Tasks (Count)	0
Completed Activities (Count)	0	Completed Activities (Count)	0	Completed Activities (Count)	0
Planned Activities (Count)	0	Planned Activities (Count)	0	Planned Activities (Count)	0
Active Activities (Count)	0	Active Activities (Count)	0	Active Activities (Count)	0
In Progress Activities (Count)	0	In Progress Activities (Count)	0	In Progress Activities (Count)	0
Delayed Activities (Count)	0	Delayed Activities (Count)	0	Delayed Activities (Count)	0
Approved Activities (Count)	0	Approved Activities (Count)	0	Approved Activities (Count)	0
Rejected Activities (Count)	0	Rejected Activities (Count)	0	Rejected Activities (Count)	0
Unassigned Activities (Count)	0	Unassigned Activities (Count)	0	Unassigned Activities (Count)	0
Deleted Activities (Count)	0	Deleted Activities (Count)	0	Deleted Activities (Count)	0

3. Scroll down to **Custom Object Permissions** and give access permissions for 'Intel Laptops...consumers - Laptop Booking and Billing Process' objects as mentioned in the below diagram.

### 3. Give Access and Share it

# Agent Profile

- 1 Go to setup >> type profiles in quick find box >> click on profiles >> select 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.

The screenshot shows the Oracle Database Profiles interface. At the top, there are two tabs: 'Profiles' and 'Restrictions'. Below the tabs, there are several sections:

- Custom Object Permissions:** Shows grants for 'EMPLOYEE' and 'DEPARTMENT' tables across users like 'HR', 'SCOTT', 'SH', 'SYSTEM', 'BKG', and 'FLO'.
- Service Settings:** Includes sections for 'Service Type', 'Service Name', and 'Service Description'.
- Permitted Profiles:** A table showing permitted profiles for users 'HR', 'SCOTT', 'SH', 'SYSTEM', 'BKG', and 'FLO'.

- 4 Click access and save it

## Roles and Hierarchy:

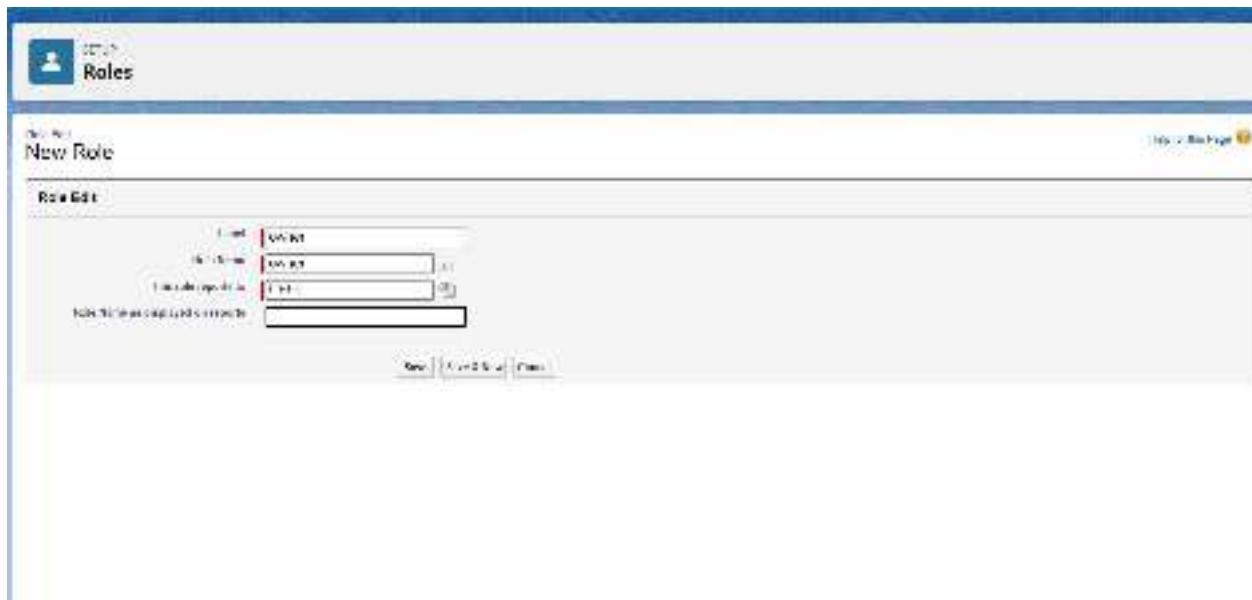
### Creating owner Role

- 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

The screenshot shows the Oracle Database Role Hierarchy interface. At the top, it says 'Your Organization's Role Hierarchy'. Below that is a tree structure of roles:

- Nick Enterprises**:
  - CEO**: [Edit](#) | [Del](#) | [Assign](#)
  - Add Role** (highlighted with a red box)
  - HR**: [Edit](#) | [Del](#) | [Assign](#)
  - Manager**: [Edit](#) | [Del](#) | [Assign](#)
  - On Site Emp**: [Edit](#) | [Del](#) | [Assign](#)
  - Remote Emp**: [Edit](#) | [Del](#) | [Assign](#)
  - Sales**: [Edit](#) | [Del](#) | [Assign](#)

- 1 Give 1 label as "owner" and Role name gets auto populated. Then click on Save



- 2 - Click and save it

## Activity 2: Creating Agent roles

Creating another two roles under Owner

- 1 - Click quick find - Search for Roles - click on set up roles
- 2 - Click plus on OTC roles and click add role under owner

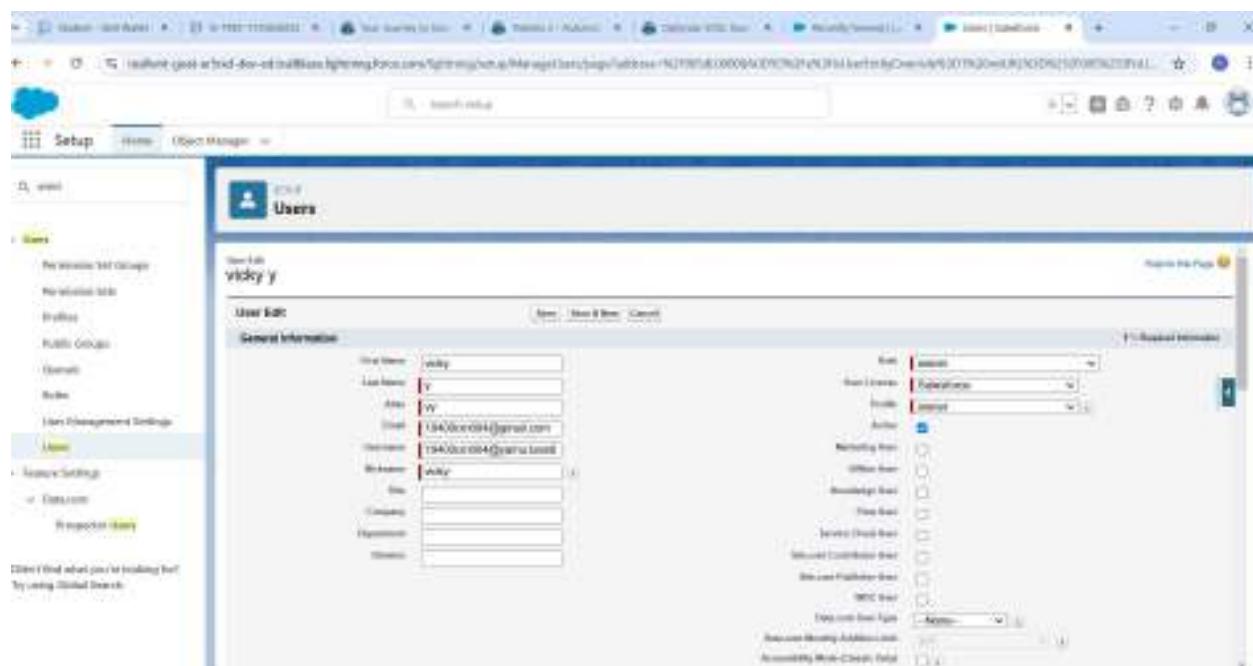


- 4 - Give Label as "Agent" and Role name gets auto populated. Then click on Save

# User

## Create User:

1. Go to setup - type users in quick find bar - 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 - Salesperson
11. Profiles - owner

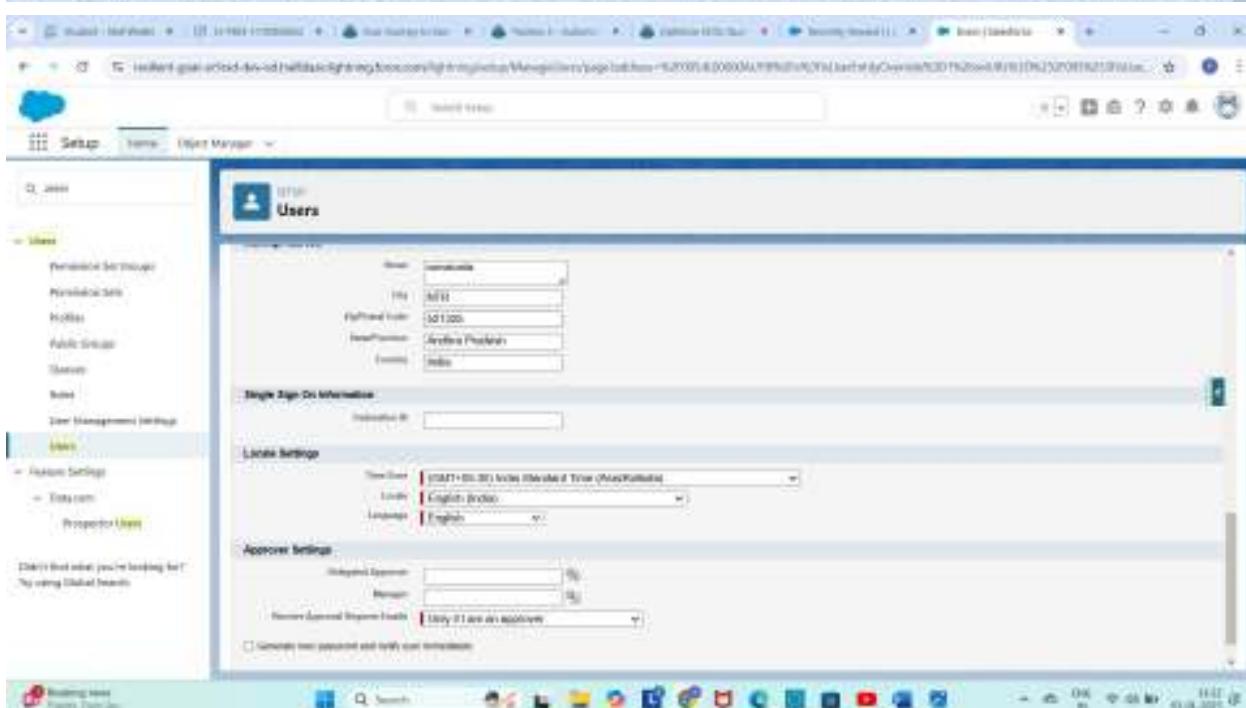
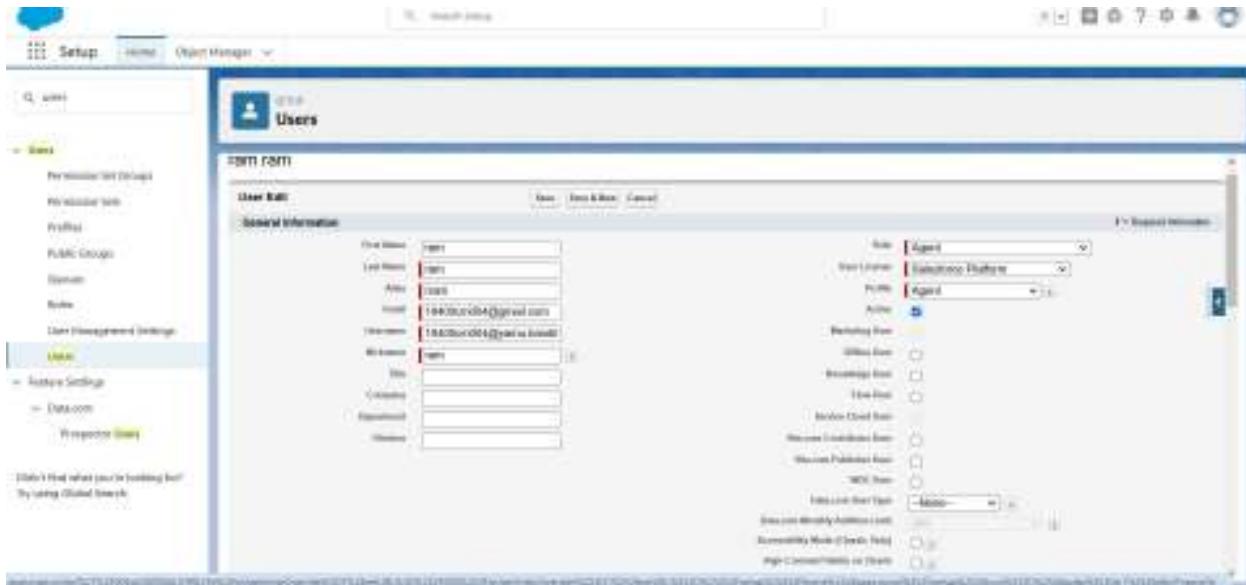


Save it

## Activity 2: creating another user-

1. Go to setup -type users in quick find bar - select users -click New user
2. Fill in the fields
3. First Name - arun
4. Last Name - arun

- 5. Alias : Give a Alias Name
  - 6. Email id : Give your Personal Email id
  - 7. Username : Username should be in this form test@text{text}
  - 8. Nick Name : Give a Nickname
  - 9. Role : Agent
  - 10. User license : Salesforce platform
  - 11. Profiles : Agent



13. *Never it*

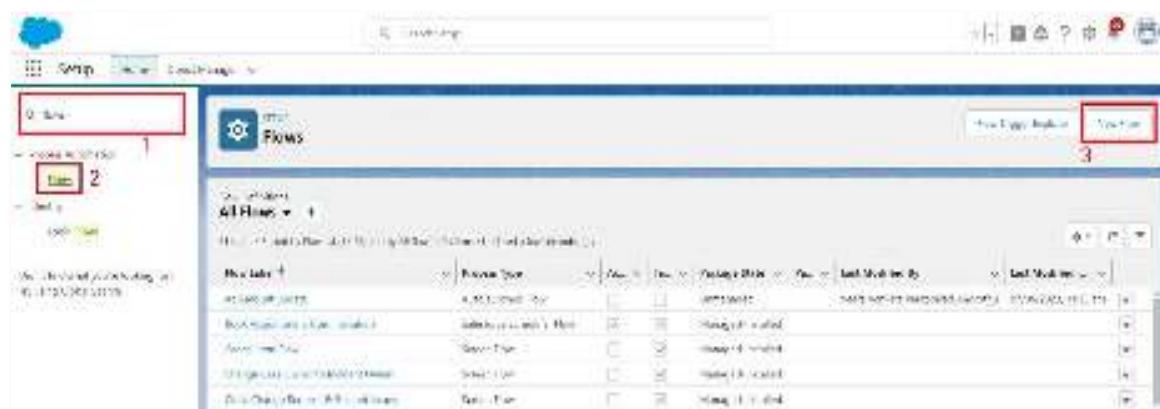
## Flows:

### Create a Flow on dell laptop:

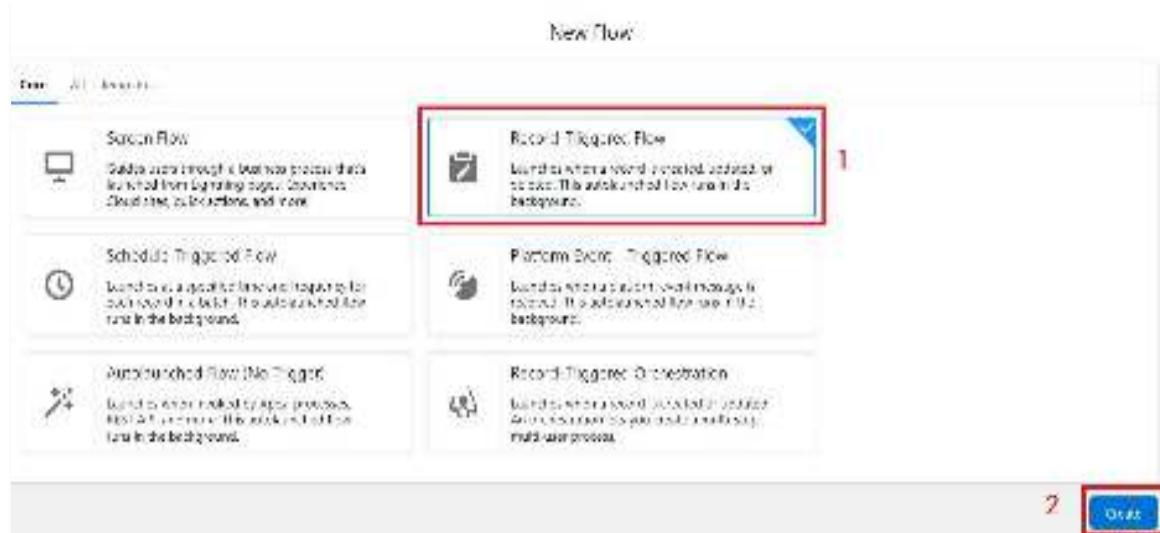
Activity -

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

Flow



- 2 Select the Record-triggered flow and Click on 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 you want to trigger the flow. In this example we will select Account.

**Object:**

**Configure Trigger**

**Trigger the Flow With:**

- A record inserted
- A record updated
- A record inserted or updated
- A record deleted

**Set Entry Conditions**

Specify entry conditions of the record to which the trigger applies and the record to which the flow executes. You can enter a condition or choose a condition from the dropdown menu.

For example, if the flow triggers off a record update, it can have a condition that says Only where record updated to meet the condition requirement is true. Then it will trigger the flow for that specific record.

**Cancel**    **Done**

**Configure Start**

### Set Entry Conditions

Select the entry conditions that determine which records the trigger can execute. You can enter a condition or choose a condition from the dropdown menu.

If no condition is selected, the trigger will run on all records that meet the entry condition. If you add a condition, the trigger will only run on records that meet the condition requirement.

**Condition Requirements:**

**None**

**\*Optional Decision Rule:**

**True False Logic:**

Applies to every record that meets the entry condition. If the logic is true, the flow will run for the record. If false, it will skip.

**Action and Related Records:**

To continue reading about the conditions, click here.

If you click here, you will see that the record is used in the action.

**Note:** This decision rule applies to every record that meets the entry condition. If the logic is true, the flow will run for the record. If false, it will skip.

**Cancel**    **Done**

Under the Record-triggered Flow (Click on “+” Symbol, and in the Drop-down List select the “Decision Element”)



- 7 Under the Details Label Field should be Update API name Gets Automatically Generated
- 8 Under the Outcome Details Label field Outcome API name Gets Automatically Generated
  - Resource: Select \$Request.Laptop.name = c
  - Operator: Select Equals
  - Value: Select dell
  - Add the same outcome option to acer - hpc man
  - Rename Default outcome as False
  - Click done

**Edit Decision**

Condition	Value	Outcome API Name
E.g.	c	c
E.g.	hpc	hpc
E.g.	All Conditions Are Satisfied	
E.g.	Else	
E.g.	Elsewhere	

Conditions: For each path the flow can take, choose a condition. For each outcome, specify the condition that must be met for the flow to take that path.

Done Outcome

10 Click to flow page

11 Beside dell there is a symbol '+' click on that

12 Again select decision

13. Under the Details Label: Field should (Update(any one you want), API name: lets Automatically Generated

14. select the Outcome Details Label: dell core i7 , Outcome API name: lets

Automatically Generated

- Resource: Select /ISRecord/core\_type/{c}

- Operator: Select Equals

- Value: Select core i7

- Then again click the symbol '+' outcome details

15. select the Outcome '-' Details Label: dell core i5 , Outcome API name: lets

Automatically Generated

- Resource: Select Reward core type

- Operator: Select Equals

- Value: Select core i5

- Then again click the symbol '+' outcome details

16. Under the Details Label: dell core i7 , Outcome API name: lets

Automatically Generated

- Resource: Select Reward core type

- Operator: Select Equals

- Value: Select core i7

17. Click done

edit Decision

Condition	Condition Type	Action
1. Dell i7	Equal	Outcome API Name: dell core i7
2. Dell i5	Equal	Outcome API Name: dell core i5
3. Dell i3	All Conditions Are Met (AND)	
else	Else	Outcome API Name: dell core i5

Conditions: For each path the following conditions or actions. For each outcome specify the conditions that must be met for the flow to take that path.

Condition 1: Dell i7

Condition 2: Dell i5

Condition 3: Dell i3

Action: Outcome API Name: dell core i7

Action: Outcome API Name: dell core i5

Action: Dell core i5

Cancel Done

18. Go to the flow page select '-' after core i7 then again select the decision

19. Under the Details Label: monthly selected.. API name: lets Automatically

Generated

20. Enter the Outcome Detail: Label: dell 1(i3), Outcome API name: Gets Automatically Generated

21.

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

23. Enter the Outcome Detail: Label: dell 2(i3), Outcome API name: Gets Automatically Generated

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

24. Click “+” outcome detail

25. Enter the Outcome Detail: Label: dell 3(i3), Outcome API name: Gets Automatically Generated

- Resource: Selector Record how many months
- Operator: Selector Equals
- Value: Selector 3

26. Click “+” outcome detail

27. Enter the Outcome Detail: Label: dell 4(i3), Outcome API name: Gets Automatically Generated

- Resource: Selector Record how many months
- Operator: Selector Equals
- Value: Selector 4

28. Click “+” outcome detail

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

- Resource: Selector Record how many months
- Operator: Selector Equals
- Value: Selector 5

**Edit Decision**

Decision ID:	10000000000000000000000000000000	Date:	2023-01-01
Description:	Test decision for outcome 1		
<b>Outcomes</b> : A decision can have multiple outcomes. Each outcome has a probability and a value. The outcome with the highest probability will be selected.			
Outcome Order	+ Outcome Details	0.9999999999999999	
1	Value	Outcome 1 Value	
2	Value	Outcome 2 Value	
<b>Decision Requirements to Execute Decision</b>			
All outcomes are met (0/0)			
Reason	Condition	Operator	Value
1	Blah Blah Blah	is equal to	Blah Blah Blah

31. Follow the above picture you will understand
  32. After dell 1(i3) then i+ '-' symbol like dell 2(i3),dell 3(i3),dell 4(i3),dell 5(i3)
  33. Click on '-' then select update records for each and every outcome
  34. Under the Details Label: one month of dell 1(i3) name : API name: Lets Automatically Generated. Under label name it a similar way for other update records
  - 35.
  36. Click done
  
  37. Under the Details Label: months selected : API name: Lets Automatically Generated
  38. Under the Outcome Details Label: dell 1(i7) : Outcome API name: Lets Automatically Generated
  39.
    - Resource: Select Record how many months
    - Operator: Select Equals
    - Value: 1
  40. Under the Outcome Details Label: dell 2(i7) : Outcome API name: Lets

Automatically Generated

- Resource: Select Reward how many months
- Operator: Select Equals
- Value: Select 2

41 Click “+” outcome details

42 Enter the Outcome Detail: Label: dell 3(17), Outcome API name: Lets

Automatically Generated

- Resource: Select Reward how many months
- Operator: Select Equals
- Value: Select 3

43 Click “+” outcome details

44 Enter the Outcome Detail: Label: dell 4(17), Outcome API name: Lets

Automatically Generated

- Resource: Select Reward how many months
- Operator: Select Equals
- Value: Select 4

45 Click “+” outcome details

46 Enter the Outcome Detail: Label: dell 5(17), Outcome API name: Lets

Automatically Generated

- Resource: Select Reward how many months
- Operator: Select Equals
- Value: Select 5

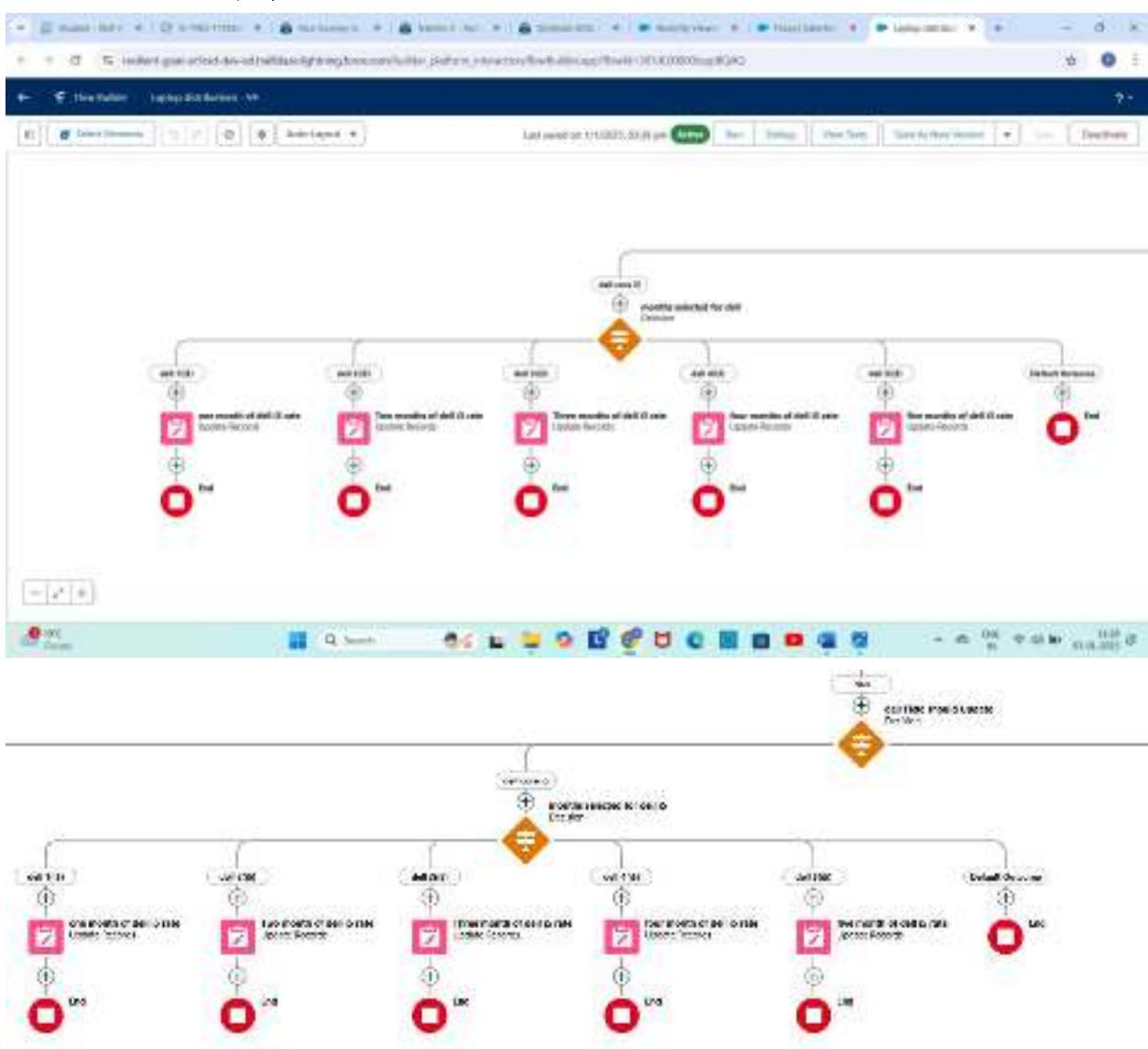
Edit Decision

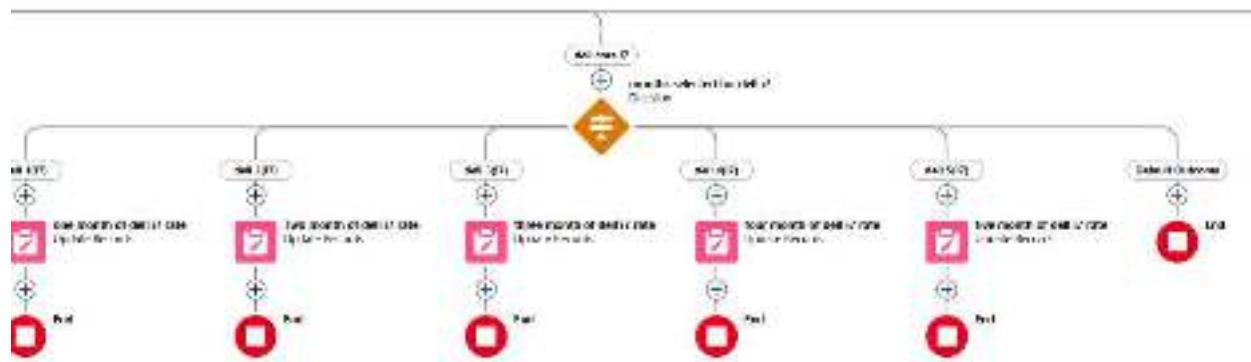
Decision ID:	10000000000000000000000000000000	Save New
Description:		
Outcomes: You can add multiple outcomes here. Each outcome has a condition and a value. The condition is evaluated first and if true, the value is used.		
Outcome Detail # +	Automatically Generated	View Details
P 1	Used	Outcome API Name: Lets
P 2	-	31
Condition Requirements for Outcome		
All Database API Has (60)		
Resource:	None	Value:
None		None
<input type="button" value="Cancel"/> <input type="button" value="Next"/>		

47 Follow the above picture you will understand

48. After dell 1(i7) there is '+' symbol like dell 2(i7),dell 3(i7),dell 4(i7),dell 5(i7)
49. Click on '+' then select update records
50. Enter the Details 1 month of dell i5 name , API name: Gets Automatically Generated
51. 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
52. Click done
53. Follow the steps from 37 to 57 for Dell i5 and update the Amount for each month (1,2,3,4,5) as 1500, 2500, 3500, 4500, 5500 respectively

Flow created on Dell laptop:





Overall dell Laptop flow



## creating flow on acer laptop

- Go to flow page
- Right side, there is a symbol '+' click on that
  - Again select devision
  - Enter the Details Label: Acer core type selection, API name: Gets Automatically Generated
  - Select the Outcome Details Label: acer core i5 - Outcome API name: Gets Automatically Generated
    - Resource: Select Record core type.
    - Operator: Select Equals
    - Value: Select core i5
  - Similarly create outcomes for acer core i5 and acer core i7 also

**Edit Decision**

**OUTCOME**: For each rule, if the outcome is not selected, click 'Add' to add a new outcome. For each outcome, click 'Edit' to edit the outcome details.

<b>OUTCOME ID:</b> <input type="text" value="1"/>	<b>OUTCOME DETAILS:</b>	<b>OUTCOME API NAME:</b> <input type="text" value="Outcomes"/>
<b>1. Condition:</b> <input type="text" value="Age"/>	<b>Value:</b> <input type="text" value="18+"/>	<b>Outcome API Name:</b> <input type="text" value="Outcomes"/>
<b>2. Condition:</b> <input type="text" value="Gender"/>	<b>Value:</b> <input type="text" value="Male"/>	<b>Outcome API Name:</b> <input type="text" value="Outcomes"/>
<b>3. Condition:</b> <input type="text" value="Age"/>	<b>Value:</b> <input type="text" value="18+"/>	<b>Outcome API Name:</b> <input type="text" value="Outcomes"/>

**Default Outcomes:**

<b>Condition:</b> <input type="text" value="Age &gt;= 18"/>	<b>Outcome:</b> <input type="text" value="Outcomes"/>	<b>Value:</b> <input type="text" value="18+"/>
<b>+ Add condition</b>		

**When to Decide Outcome:**

If the user has no previous reward  
 Only if the user has not received the first reward in succession over the last three months

**Cancel** **Done**

Click done

- Click flow page
- Beside dell there is a symbol '+' click on that
- Again select decision
- Under the Details Label Acer month selected, API name: Gets Automatically Generated
- Under the Outcome Details Label acer 1(i3), Outcome API name: Gets Automatically Generated
- Resource: Seller Reward how many months
  - Operator: Seller Equals
  - Value: 1
- Under the Outcome Details Label acer 2(i3), Outcome API name: Gets Automatically Generated
  - Resource: Seller Reward how many months
  - Operator: Seller Equals
  - Value: Seller 2
- Click '-' outcome details
- Under the Outcome Details Label acer 3(i3), Outcome API name: Gets Automatically Generated

- Resource: Select Reward how many months
- Operator: Select Equals
- Value: Select 3
- a Click “+” outcome details
- b Under the Outcome Details Label: acer 4(i3) Outcome API name: Acer Automatically Generated
- Resource: Select Reward how many months
- Operator: Select Equals
- Value: Select 4
- a Click “+” outcome details
- b Under the Outcome Details Label: acer 5(i3) Outcome API name: Acer Automatically Generated
- Resource: Select Reward how many months
- Operator: Select Equals
- Value: Select 5

**Edit Decision**

Condition	Decision
Resource: acer 1(i3)	Label: acer 1(i3)
Operator: Equals	Outcome API Name: acer 1(i3)
Value: 100	
Condition Requirements to Decide Outcome	
All conditions must be true:	
Resource: acer 1(i3)	Operator: Equals
Value: 100	Outcome API Name: acer 1(i3)
<input type="button" value="Cancel"/> <input type="button" value="Done"/>	

Click done.

- a After acer 1(i3) there is “+” symbol like acer 2(i3),acer 3(i3),acer 4(i3),acer 5(i3)
- b Click on “+” then select update records
- c Under the Details Label: one month of acer i3 name: API name: Acer Automatically Generated
- d Field: Amount – c values 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
- c) Follow the steps from b to 22 and create decision and Update record elements for Acer core i5 and Acer core i7. But enter different Amount values.

### 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

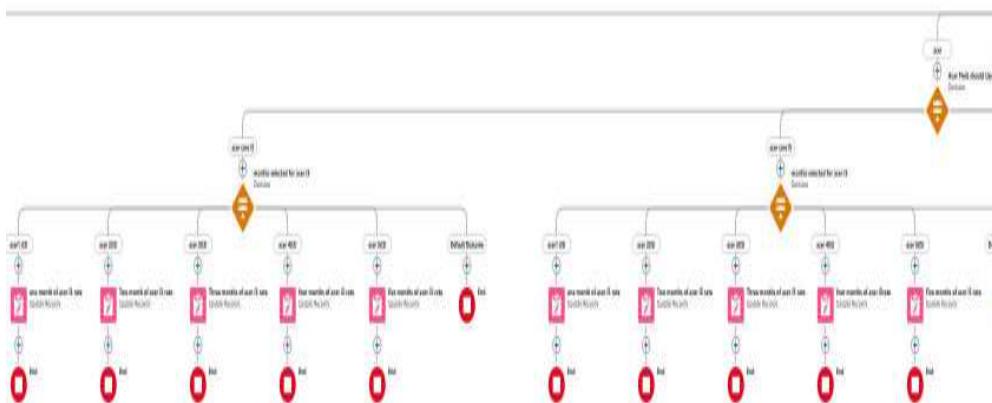
Field	Value
Amount_c	500

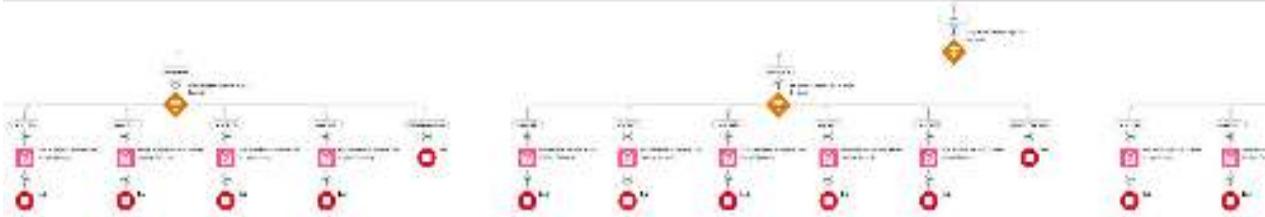
+ Add Field

Cancel Done

f) Click done

Flows Completed on acer Laptop





## creating a flow on hp laptop:

[Go to flow page](#)

1.
  - a. Beside 'hp' there is a symbol '...', click on that
  - b. Again select decision
  - c. Under the Details Label: HP core selection, API name: Gets Automatically Generated
  - d. select the Outcome Details Label: hp core i5 , Outcome API name: Gets Automatically Generated
    - Resource: Select Reward core type,
    - Operator: Select Equals
    - Value: Select hp i5
2.
  - a. Similarly create outcomes for hp core i5 and hp core i7 also
  - b. Go to flow page
3.
  - a. Beside 'hp' there is a symbol '...', click on that
  - b. Again select decision
4.
  - a. Under the Details Label: hp core i5 month selection , API name: Gets Automatically Generated
  - b. Under the Outcome Details Label: hp i5S , Outcome API name: Gets Automatically Generated
    - c.
      - Resource: Select Reward how many months
      - Operator: Select Equals

5.

  - Value: 1
  - a Under the Outcome Details Label [hp 2(i5)], Outcome API name: 4 lets Automatically Generated
  - Resource: Select Record how many months
  - Operator: Selector Equals
  - Value: Selector 2
  - a Click “+ outcome details”
  - b Under the Outcome Details Label [hp 3(i5)], Outcome API name: 4 lets Automatically Generated
  - Resource: Select Record how many months
  - Operator: Selector Equals
  - Value: Selector 3
  - a Click “+ outcome details”
  - b Under the Outcome Details Label [hp 4(i5)], Outcome API name: 4 lets Automatically Generated
  - Resource: Selector Record how many months
  - Operator: Selector Equals
  - Value: Selector 4
  - a Click “+ outcome details”
  - b Under the Outcome Details Label [hp 5(i5)], Outcome API name: 4 lets Automatically Generated
  - Resource: Selector Record how many months
  - Operator: Selector Equals
  - Value: Selector 5

6. Click on done

7.

- After hp 1(i5) there is '+' symbol like hp 2(i5), hp 3(i5), hp 4(i5), hp 5(i5)
- Click on '+' then select update records
- Enter the Details Label: one month of hp i5 rate , API name: tiers  
Automatically Generated
- Field: Amount - c. values 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
- Follow the steps from 5 to 21 and create decision and Update record elements for hp core i3 and hp core i7. But enter different Amount values

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 Do and Of 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

8. Click done

Flows created on hp laptop with core types:

HP i3

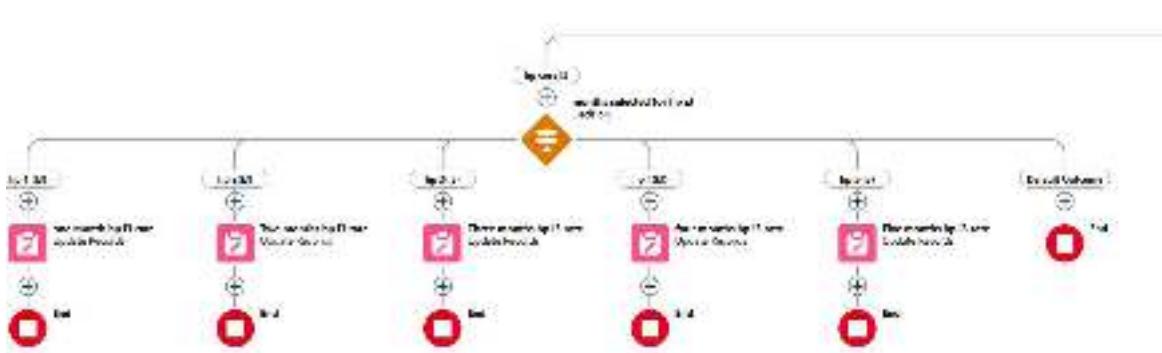


Fig 15

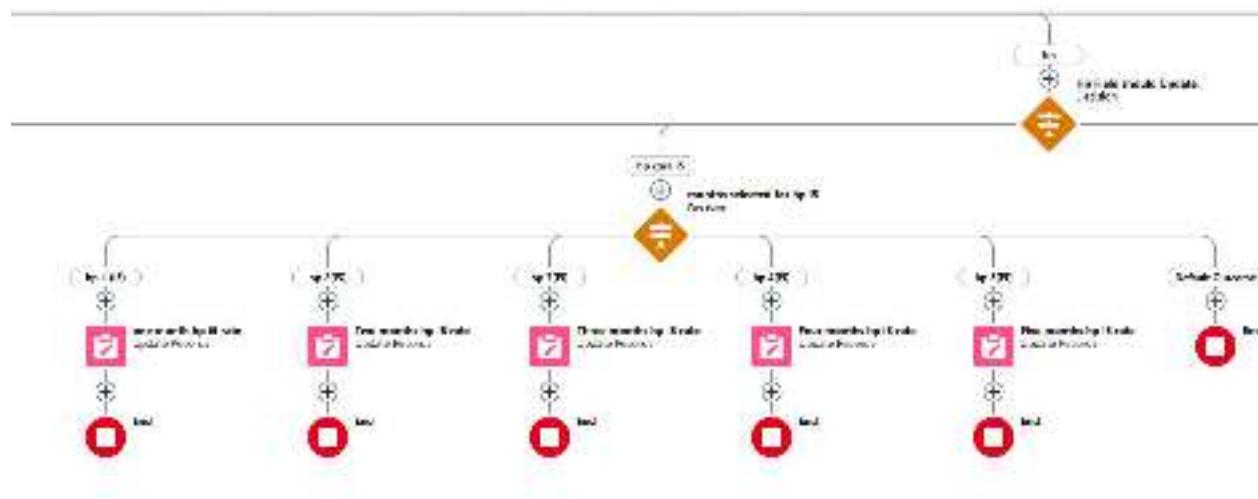
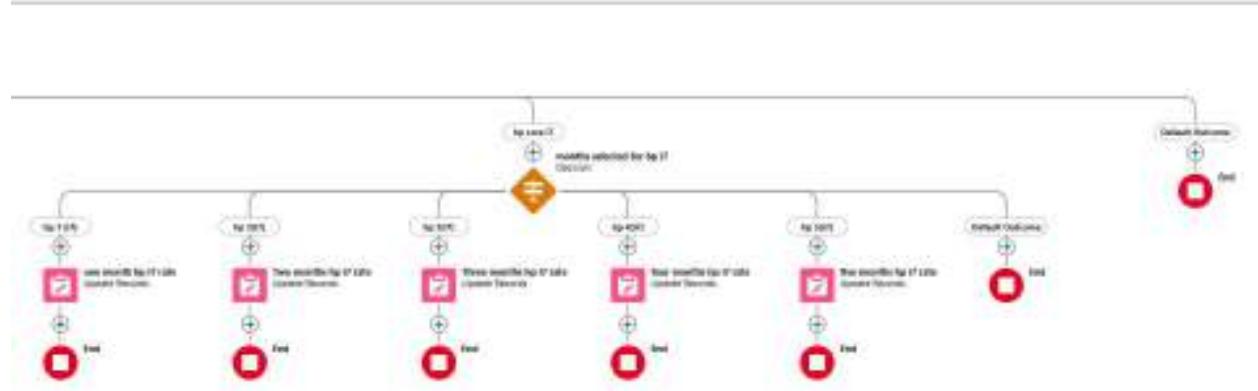
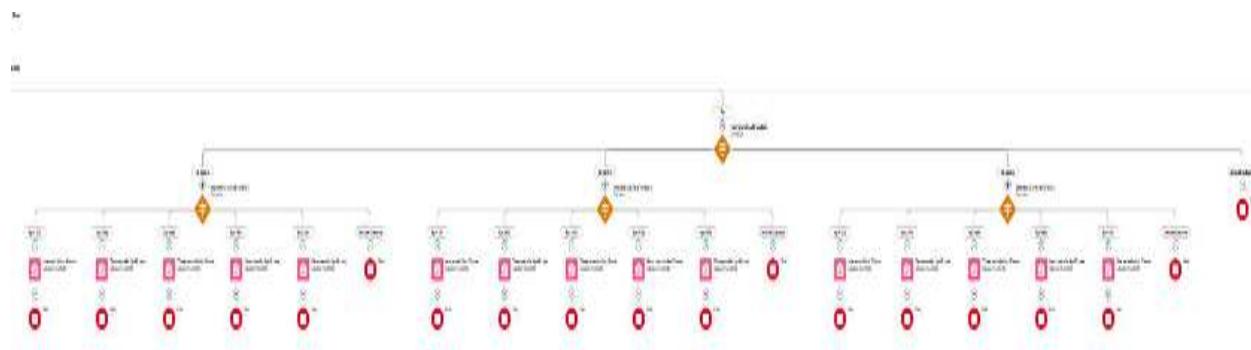


Fig 17

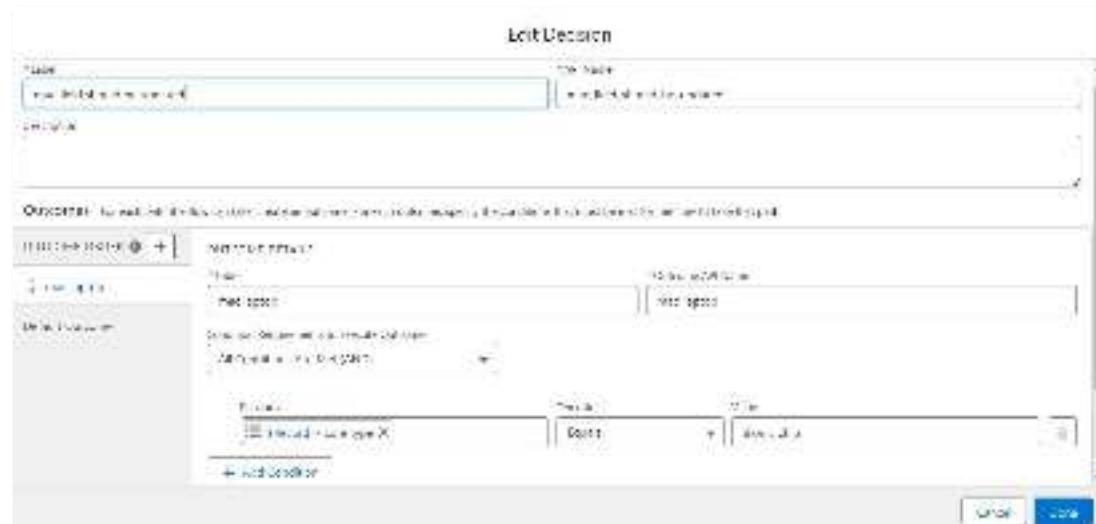


## Overall flow on IIP:



## creating a flow on mac laptop:

- a. Go to flow page
- b. Beside menu there is a symbol click on that
- c. Again select decision
- d. Under the Details Label user should be Updated API name: Gets Automatically Generated
- e. select the Outcome Details Label: mac laptop , Outcome API name: Gets Automatically Generated
- f. Resource: Silver Reward card type.
- g. Operator: Silver Equals
- h. Value: Silver Rionic Chip



Click done

Go to flow page

- a. Beside Mac, there is a symbol "...", click on that
- b. Again select division
- c.
  - a. Under the Details Label Mac month selected . API name: Gets Automatically Generated
  - b. Under the Outcome Details Label mac biome chip(1) . Outcome API name: Gets Automatically Generated
  - c.
    - Resource: Selector Reward how many months
    - Operator: Selector Equals
    - Value: 1
- a. Under the Outcome Details Label mac biome chip(2) . Outcome API name: Gets Automatically Generated
- b.
  - Resource: Selector Reward how many months
  - Operator: Selector Equals
  - Value: Selector 2
- a. Click "+ outcome details"
- b. Under the Outcome Details Label mac biome chip(3) . Outcome API name: Gets Automatically Generated
- b.
  - Resource: Selector Reward how many months
  - Operator: Selector Equals
  - Value: Selector 3
- a. Click "+ outcome details"
- b. Under the Outcome Details Label mac biome chip(4) . Outcome API name: Gets Automatically Generated
- b.
  - Resource: Selector Reward how many months
  - Operator: Selector Equals
  - Value: Selector 4
- a. Click "+ outcome details"
- b. Under the Outcome Details Label mac biome chip(5) . Outcome API name: Gets Automatically Generated
- b.
  - Resource: Selector Reward how many months

- Operator: Selector Equals
- Value: Selector S

Edit Decision

**mar\_months\_selected (mar\_months\_element)**

Conditions: For each row in the Source table, evaluate the condition for which the row is included in the target table. If no conditions are present, all rows are included.

**Decision Order** + **Decision Details** **Create Outcome**

**Condition** **Value** **Else value for Select**

1. **mar\_months\_element** **Value**: **mar\_hirec\_chip(1)** **Else value for Select**: **mar\_hirec\_chip(2)**

2. **mar\_hirec\_chip(1)** **Value**: **mar\_hirec\_chip(2)** **Else value for Select**: **mar\_hirec\_chip(3)**

3. **mar\_hirec\_chip(2)** **Value**: **mar\_hirec\_chip(3)** **Else value for Select**: **mar\_hirec\_chip(4)**

4. **mar\_hirec\_chip(3)** **Value**: **mar\_hirec\_chip(4)** **Else value for Select**: **mar\_hirec\_chip(5)**

5. **mar\_hirec\_chip(4)** **Value**: **mar\_hirec\_chip(5)** **Else value for Select**: **mar\_hirec\_chip(6)**

**Default Outcome** **+ Calculation**

**When to Execute Outcome** **On**

**On** **This condition always applies** **Only if the record triggered the key or it is part of the workflow instance**

**Save** **Done**

Click done

- After `mar_hirec_chip(1)` there is '-' symbol like `mar_hirec_chip(2), mar_hirec_chip(3), mar_hirec_chip(4), mar_hirec_chip(5)`.
- Click on '-' then select `update records`
- Enter the `Label`: one month of mar rate - API name: `lets Automatically generated`
- Field: `Amount` - c\_value: for one month of mar hirec chip rate-1700 .. two month of mar hirec chip rate-3400 .. three month of mar hirec chip rate- \$100 .. four month of mar hirec chip rate-6500 .. five month of mar hirec chip rate-\$500. Follow for all these finally

Click done

**Edit Update Records**

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

Use the source backlog record that triggered the flow

Update records related to the target backlog record that triggered the flow

Use the ID and/or field values from a record in record collection

Specify conditions on backlog records and set them individually

**Set Filter Conditions**

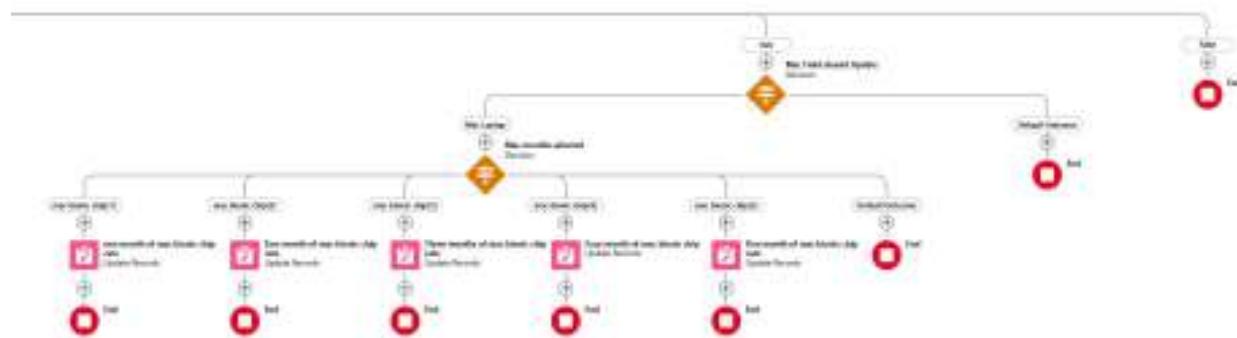
**Update Recordset (Source Record)** **None—Always Update All**

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

**Field** **Amount\_c** **Value** **Amount\_c**

**Cancel** **Save**

## VI. CW on Mac Laptop

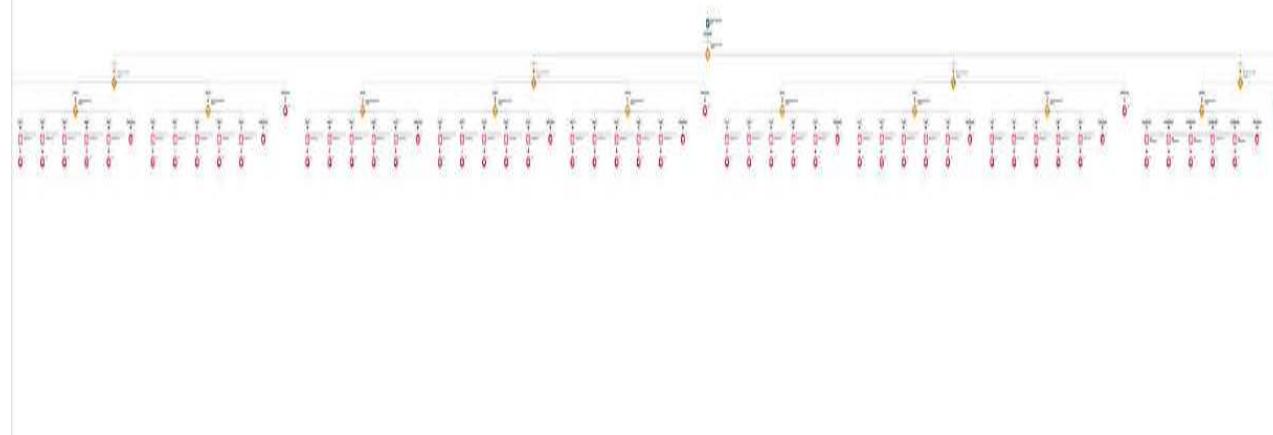


Click on save

I select Laptop distributions, goi names automatically filled

Save the flow and activate it

Laptop Distribution Flow:

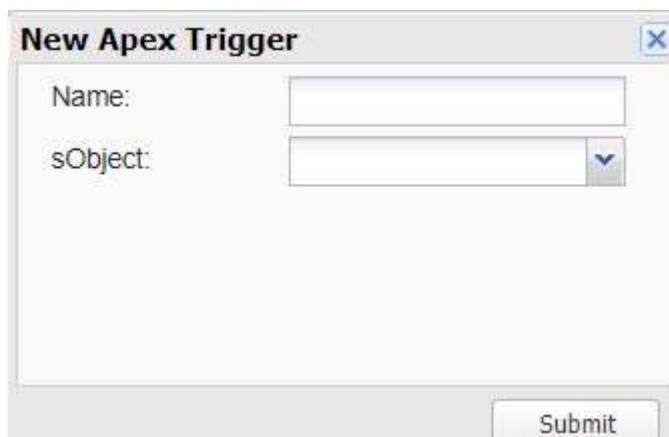
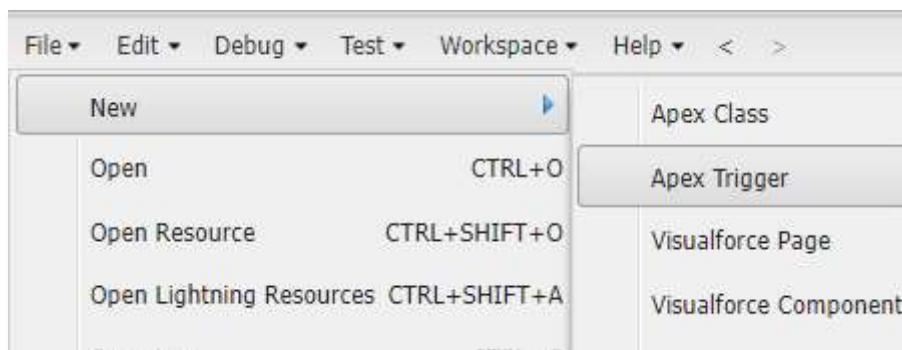


**Apex:**

## Apex Trigger and Handler Class

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);

{

The screenshot shows the Salesforce Developer Console with the title "Developer Console - Setup Home". The URL is "google-7da-dev-edddevwp.my.salesforce.com/sf/composite/apex/debug/AutoS3Steps". The tab bar shows "LaptopBooking.apfc" is selected. The status bar indicates "Code Coverage: None - API Version: 54.0". The code editor contains the following Apex trigger:

```
trigger LaptopBooking on Laptop_Bookings__c (After insert,after update) {
    if(trigger.isAfter && ( trigger.isInsert || trigger.isupdate))
    {
        LaptopBookingHandler.sendEmailNotification(trigger.new);
    }
}
```

Trigger code

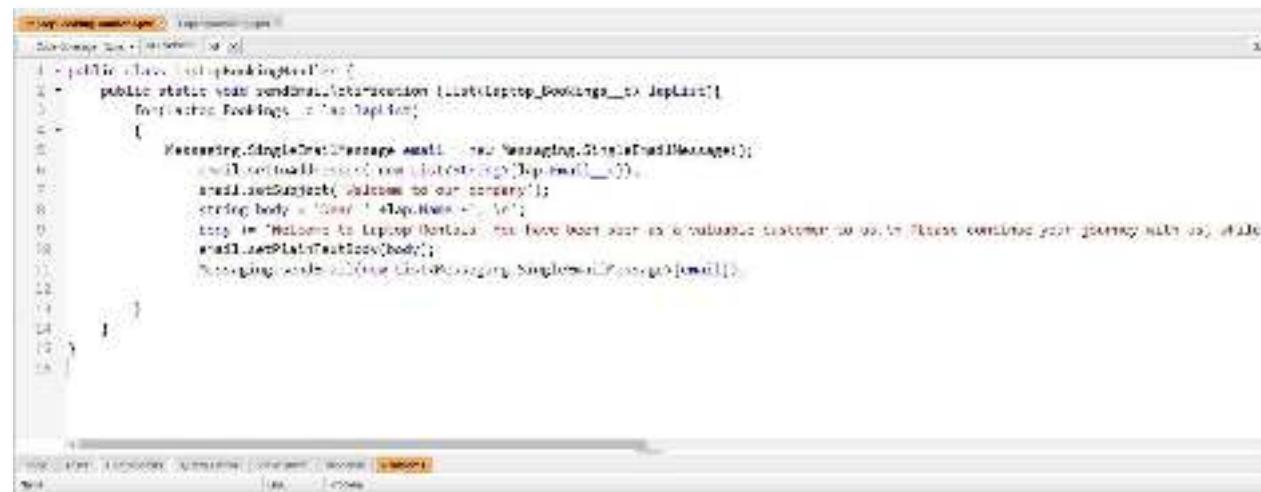
```
trigger LaptopBooking on Laptop_Bookings__c (After insert,after update) {
```

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

Notes- copy the API names

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

### Handler Class:



The screenshot shows a Java code editor with the following code:

```
1 package com.laptrn.core;
2
3 import java.util.List;
4
5 public class LaptopBookingHandler {
6
7     public static void sendEmailNotification (List<Laptop_Bookings> lapList) {
8         for(Laptop_Bookings lap : lapList) {
9             {
10                 Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
11                 email.setToAddresses(new List<String>(lap.Email));
12                 email.setSubject('Welcome to our company');
13                 String body = "Dear "+lap.Name+",\n";
14                 body += "Welcome to Laptop Rentals. You have been seen as a valuable customer to us. Please continue your journey with us, while";
15                 email.setPlainTextBody(body);
16                 Messaging.SingleEmailMessage[] emails = {email};
17             }
18         }
19     }
20 }
```

### Code Snippet :

```
public class LaptopBookingHandler {
    ...
    public static void sendEmailNotification (List<Laptop_Bookings> c: lapList){
        for(Laptop_Bookings c: lapList)
        {
            Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
            email.setToAddresses(new List<String>(lap.Email));
            email.setSubject('Welcome to our company');

            String body = "Dear Customer, \n";
            body += "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. An Laptop Amount : "+lap.Amount + " in con type "+lap.con_type + " in Laptop type "+lap.Laptop_name + " c:
            email.setPlainTextBody(body);

            Messaging.sendEmail(new List<Messaging.SingleEmailMessage>{email});
        }
    }
}
```

### Name:-

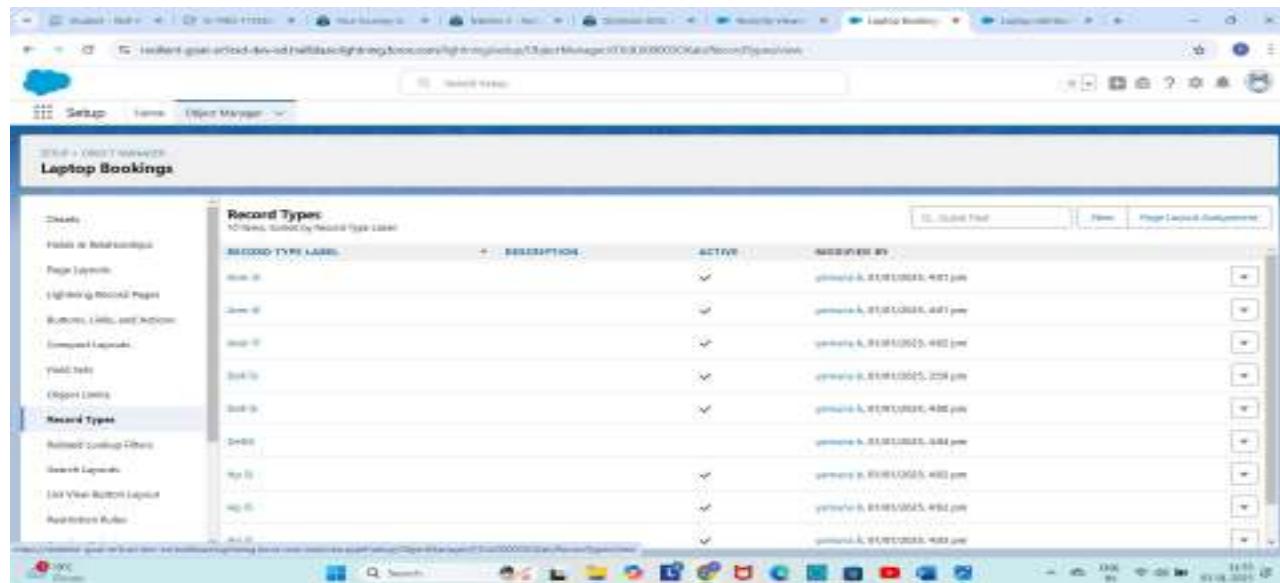
1. Class name: LaptopBookingHandle
  2. API Name: Laptop\_Bookings - e.(as per your logic to laptop booking object and copy from that)
  3. score - e.(as per your logic to laptop booking object and copy from that).
  4. Laptop type - e.(as per your logic to laptop booking object and copy from that).

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

## Reports:

## Create Report

- 1 Note: Before creating reports just fill the 10-12 records in the Laptop Bookings object.
  - 2 Create records for each one you have to create at least 2 different records i.e dell(i3), dell(i7), acer(i3), hp(i5), mae(bionic chip).



The screenshot displays the Salesforce Lightning Experience interface. At the top, there is a blue header bar with the Salesforce logo, a 'Setup' button, a 'Home' button, and an 'Object Manager' dropdown.

The main area is divided into sections:

- App Launcher:** A search bar containing the text 'Lap'. Below it is a list of items under the heading 'Items':
  - Laptop Bookings
  - Total Laptops
- Buttons, Links, and Actions:** A horizontal bar with several icons.
- Recently Viewed:** A list titled 'Recently Viewed' with 11 items:
  - 1 Laptop Bookings
  - 2 Pending
  - 3 Requests
  - 4 New IP
  - 5 Services
  - 6 Ap
  - 7 4
  - 8 2020
  - 9 2020
  - 10 2020
  - 11 2020

The bottom of the screen shows the Windows taskbar with various pinned application icons.

3

Go to the app > click on the reports tab

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.

A screenshot of a "Select a Report Type" dialog. On the left, there is a search bar containing the text "G\_CRM". To the right of the search bar is a list of report types, each with a status indicator (e.g., Draft, Published) and a dropdown arrow. The list includes:

- Draft
- Published

4 Create a simple tabular report

5 Add fields from left pane make sure that Amount field will be selected

6 Click the Amount column drop down and select Invoicer list

*Field	*Bucket Name	
Amount	X types of versions	X
Range	Bucket	X
≤ 100	basic	X
> 100 to ≤ 1500	intermediate	X
> 1500 to ≤ 10000	high	X
>	very high	X

Click apply it

4. Select 'Hyperlink version in Group By Rows' to create a summary report. Follow the image for other fields.

Customer with laptop and total laptop					
Customer		Customer with laptop		Customer with total laptop	
Customer ID	Name	Customer ID	Name	Customer ID	Name
1	John Doe	1	John Doe	1	John Doe
2	Jane Smith	2	Jane Smith	2	Jane Smith
3	Mike Johnson	3	Mike Johnson	3	Mike Johnson
4	Sarah Williams	4	Sarah Williams	4	Sarah Williams
5	David Lee	5	David Lee	5	David Lee
6	Emily Davis	6	Emily Davis	6	Emily Davis
7	Robert Green	7	Robert Green	7	Robert Green
8	Sarah Johnson	8	Sarah Johnson	8	Sarah Johnson
9	David Williams	9	David Williams	9	David Williams
10	Emily Lee	10	Emily Lee	10	Emily Lee

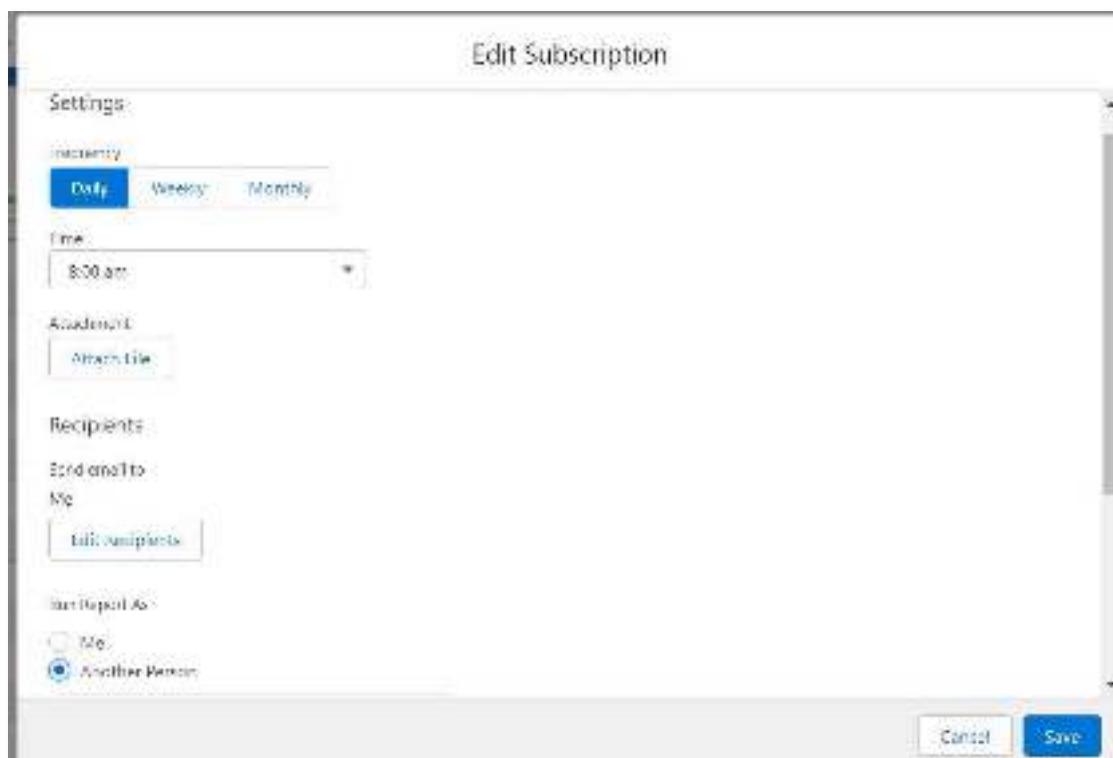
Click on Start to run it

Type of service	Leasing Bookings	Loan Bookings	personnel	consumer	Date	Amount	Loaned Available	Total No Of Laptops	Total Laptops
Leasing	None	None				\$0.00	-10	1	
Housing	1	None				\$1,000	-10	0	
work	None	None				\$1,000	-10	0	
Residency	None	None				\$1,000	-10	0	
test	None	None				\$1,000	-10	0	
Hosting	None	None				\$1,000	-10	0	
One	None	None				\$1,000	-10	0	
work P	None	None				\$1,000	-10	0	
hp	None	None				\$1,000	-10	0	
<b>Subtotal</b>						<b>\$0.00</b>	<b>-100</b>		
<b>Leasing</b>									
Leasing	1	None	None	None		\$1,000	-10	1	
<b>Subtotal</b>						<b>\$1,000</b>	<b>-100</b>		
<b>Total</b>						<b>\$0.00</b>	<b>-100</b>		

## Sharing report to owner

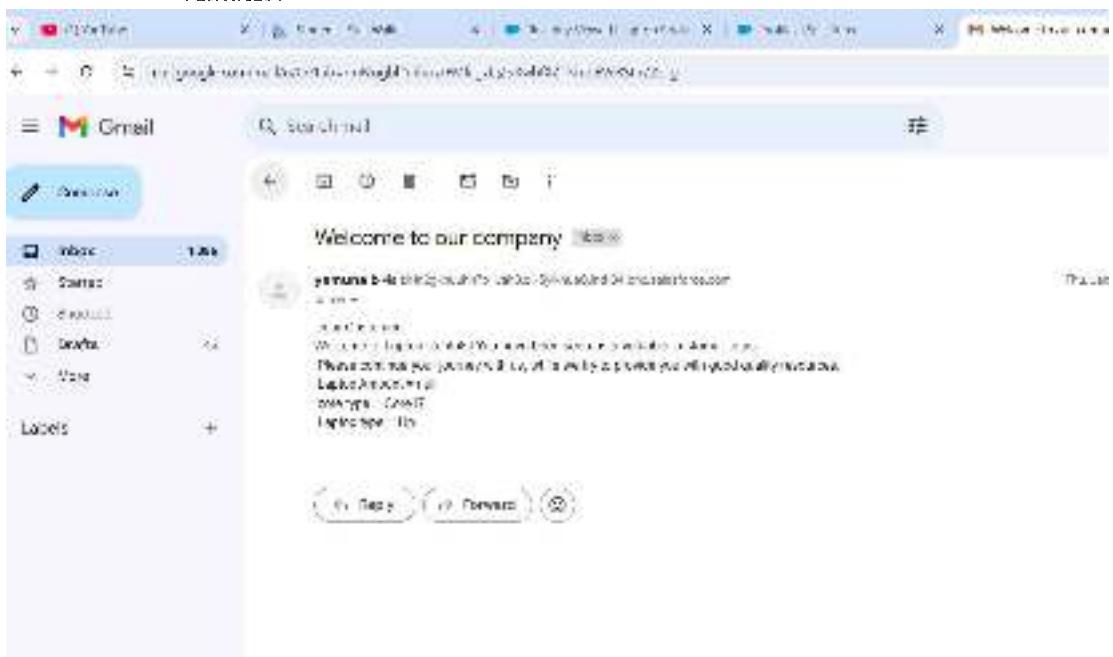
- a Click edit drop down and select subscribe option

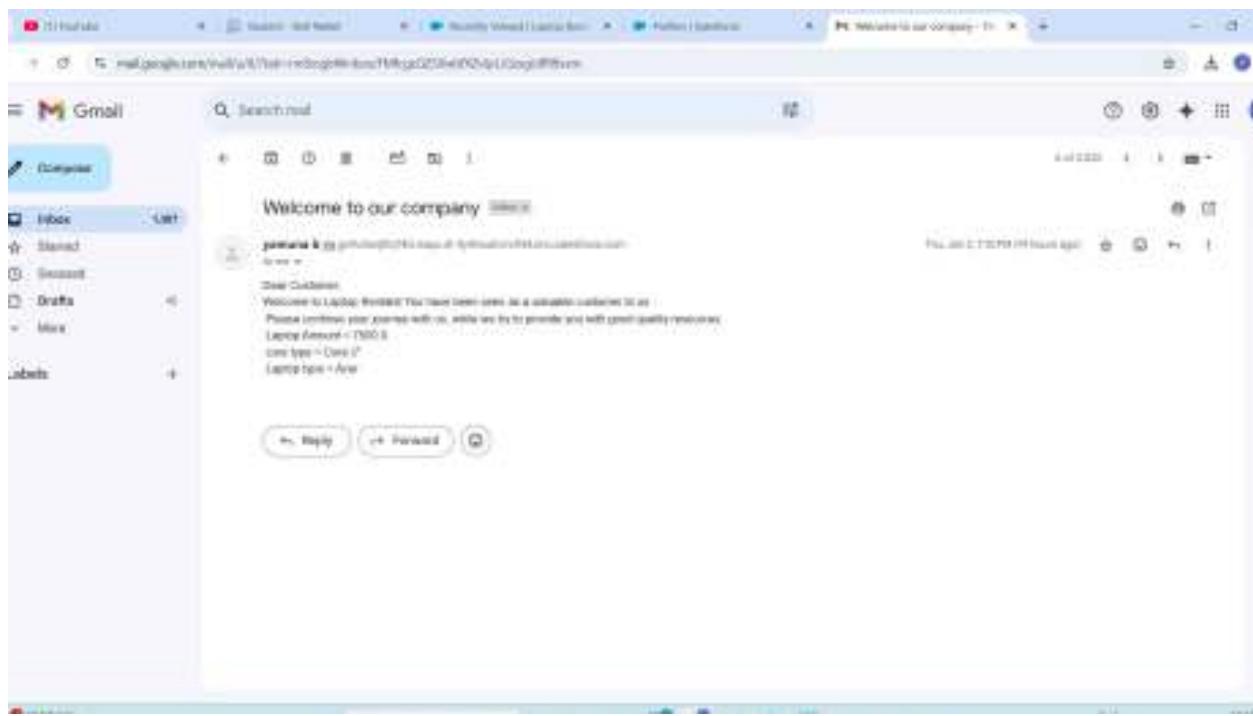
- o Follow as per below image.



- After selecting the run report as a "another person", select your personal account or whom you want to send that mail to
- Click save.

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





## 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 "test1" retain name
- 4 Folder unique names will be auto populated
- 5 Click save

## Create Dashboard

- 1 Go to the app to click on the Dashboards tab



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

**New Dashboard**

\* Name:

Description:

Folder:

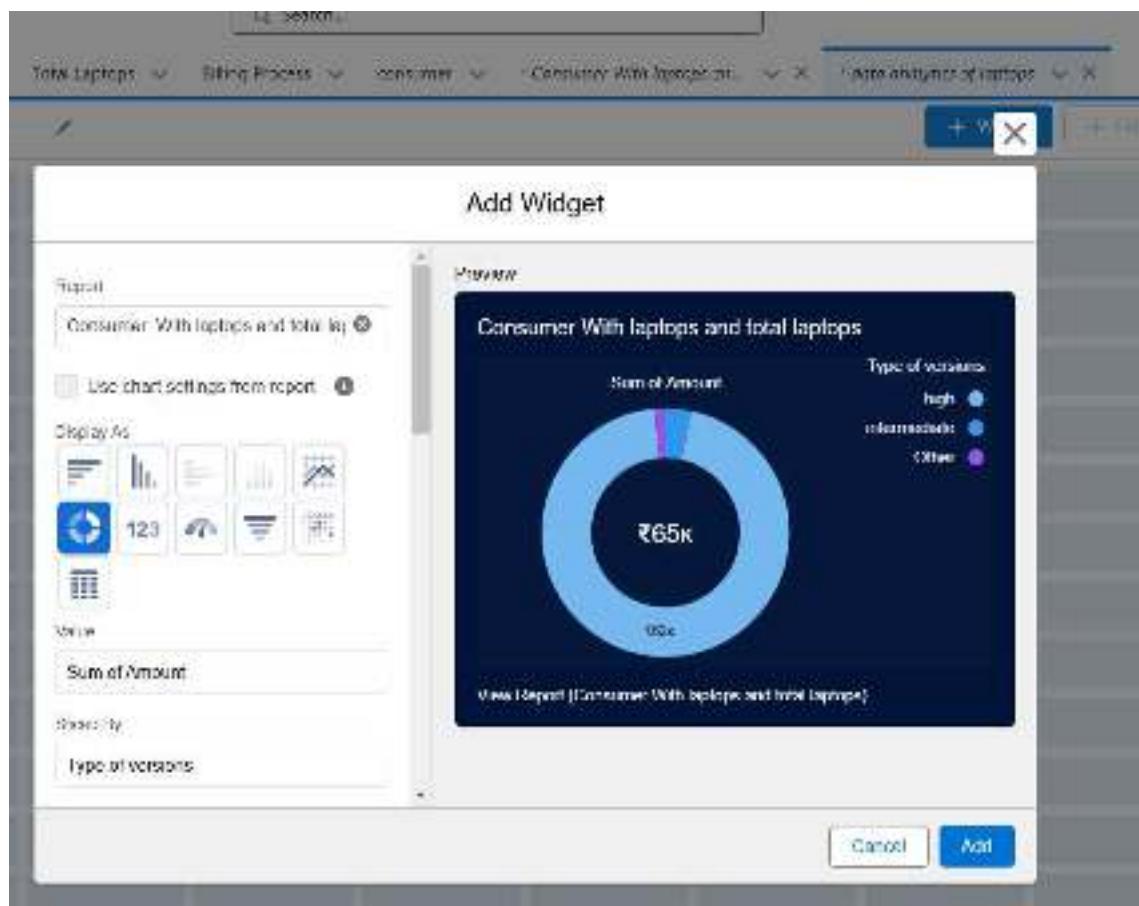
**3 Select a widget**

LAPTOP RENTALS   

**4 Select a Report and click on select**

Search Report

Report	Consumer With laptop amount	Consumer With laptop amount
Consumer With laptop amount	Consumer With laptop amount	Consumer With laptop amount
Consumer With laptop amount	Consumer With laptop amount	Consumer With laptop amount



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

## 5. Testing and Validation

- **Unit Testing:** Apex test classes for validating trigger logic and email notifications.
- **User Interface Testing:** Ensure correct data display and functionality for booking and inventory tracking.

## 6. Key Scenarios Addressed by Salesforce in the Implementation

- Real-time updates on laptop availability after each booking.
- Automated email notifications for booking confirmations.
- Preventing overbooking by dynamically validating available stock.
- Detailed reporting on laptop rentals and inventory status.

## **Conclusion:**

**Summary of Achievements:** The project successfully automated the management of laptop rentals, improved inventory tracking accuracy, and enhanced user experience with real-time notifications and dashboards. This solution is scalable and supports future enhancements like integration with external systems for broader functionality.

**THE END**



