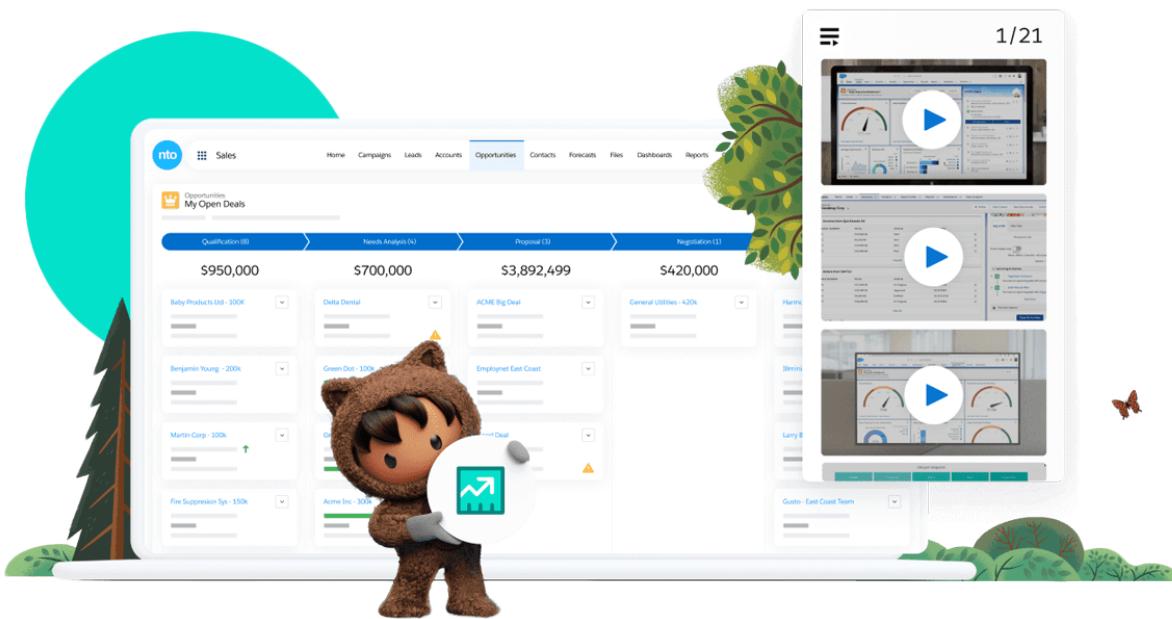


# Project: A CRM APPLICATION FOR LAPTOP RENTALS



By **Siddi Hima Meghana**  
[322103310214@gvpce.ac.in](mailto:322103310214@gvpce.ac.in)

## 1. Project Overview

This project, "**CRM Application for laptop rentals**" is focused on creating a Salesforce-based application to streamline laptop rental operations. By leveraging custom objects, validation rules, role-based access control, workflow automation, and reporting & dashboarding features within the Salesforce platform, the CRM provides a robust solution for managing the entire rental lifecycle. This comprehensive solution improves customer experience by ensuring timely communication and provides valuable insights for data-driven decision-making, ultimately driving business growth.

## 2. Objectives

### **Business Goals:**

- To optimize laptop rental operations by streamlining the entire rental process from booking to return.
- To increase revenue and profitability through effective inventory management.
- To build customer relationships by collecting valuable customer data and providing personalized service.

### **Specific Outcomes:**

- **Real-Time Tracking:** Implement a dashboard to monitor laptop availability, rental bookings, customer interactions, and revenue generation.
- **Automation of Processes:** Streamline rental bookings and inventory management using Salesforce Flows and Triggers.
- **User-Friendly Interface:** Develop tabs for easy customer onboarding, rental requests, and staff access to relevant information.

- **Comprehensive Reporting:** Generate detailed reports to analyze rental trends, identify areas for improvement, and track key performance indicators.
- **Scalability:** Create a flexible and scalable platform that can accommodate a growing number of laptops, customers, and rental requests.

### **3. Salesforce Key Features and Concepts Utilized**

**Custom Objects and Fields:** Created tailored objects to manage laptop inventory, rental bookings, customer information, and payment details.

**Lightning App Builder:** Designed an intuitive and visually appealing Lightning app interface for users to easily navigate and manage rental requests, track inventory, and access customer information.

**Flows and Automation:** Implemented automation for rental bookings, payment processing, and inventory updates to improve efficiency.

**Triggers:** Developed Apex triggers to automate backend processes, such as updating inventory availability upon rental bookings or sending automated email notifications to customers.

**Reports and Dashboards:** Built dynamic dashboards and reports to provide real-time insights into rental trends, rental revenue etc.

## 4. Detailed Steps to Solution Design

Topic no.	Topic Name
1	<b>Creating objects:</b> 1.1 Create Total Laptops Object 1.2 Create consumer object 1.3 Create Laptop Bookings object 1.4 Create Billing process object
2	<b>Create Custom Tabs</b>
3	<b>Create Lightning App</b>
4	<b>Fields:</b> 4.1 Creating The Field In Consumer Object 4.2 Creating The Field In Laptop Bookings Object 4.3 To Create A Fields and Relationship To Laptop Bookings And Total Laptops Object 4.4 Creation of Fields and Relationship For Billing Process Object 4.5 Creating The Field In Total Laptops Object
5	<b>Validation Rule:</b> Creating The Validation Rule For Phone Number Field In Consumer Object
6	<b>Profiles:</b> 6.1 Owner Profile 6.2 Agent Profile
7	<b>Roles And Hierarchy:</b> Creating Owner Role
8	<b>Users:</b> Create User

Topic no.	Topic Name
9	<p><b>Flows:</b></p> <ul style="list-style-type: none"> <li>9.1 Create A Flow On Dell Laptop</li> <li>9.2 Creating Flow On Acer Laptop</li> <li>9.3 Creating A Flow On Hp Laptop</li> <li>9.4 Creating A Flow On Mac Laptop</li> </ul>
10	<p><b>Apex:</b> Apex Trigger And Handler Class</p>
11	<p><b>Reports:</b></p> <ul style="list-style-type: none"> <li>11.1 Create Report</li> <li>11.2 Sharing Report To Owner</li> </ul>
12	<p><b>Dashboards:</b></p> <ul style="list-style-type: none"> <li>12.1 Create Dashboard Folder</li> <li>12.2 Create Dashboard</li> </ul>

## 1. Creating Objects

Objects in Salesforce are database tables that allow you to store data specific to your organization. Each object comprises records (rows) and fields (columns) that help organize and structure your data efficiently. We use objects to manage and relate various types of information, enabling seamless data tracking, reporting, and analysis within the Salesforce platform. Objects allow users to manage various types of information such as customer accounts, contacts, opportunities, and custom data specific to the organization. Salesforce provides standard objects like Account, Contact, and Opportunity, and users can also create **custom objects** to suit unique business needs.

### 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.

**SETUP** New Custom Object

Custom Object Definition Edit

Custom Object Information

The singular and plural labels are used in tabs, page layouts, and reports.

Label	Example: Account
Plural Label	Example: Accounts

Starts with vowel sound

The Object Name is used when referencing the object via the API.

Object Name	Example: 2 Account
-------------	--------------------

Description

Context-Sensitive Help Setting

- Open the standard Salesforce.com Help & Training window
- Open a window using a Visualforce page

Content Name

Enter Record Name Label and Format

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

Record Name	Example: 3 Account Name
-------------	-------------------------

Data Type

Optional Features

<input checked="" type="checkbox"/> Allow Reports	4
<input type="checkbox"/> Allow Activities	
<input type="checkbox"/> Track Field History	

Object Classification

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

- Allow Sharing
- Allow Bulk API Access
- Allow Streaming API Access

Deployment Status

- In Development
- Deployed

Search Status

When this setting is enabled, your users can find records of this object type when they search. [Learn more](#).

<input type="checkbox"/> Allow Search	1
---------------------------------------	---

Object Creation Options (Available only when custom object is first created)

- Add Notes and Attachments related list to default page layout
- Launch New Custom Tab Wizard after saving this custom object

**2**

4. Click on Save.

## 1.1 Create Total Laptops Object

To Create an object:

1. From the Setup page >> Click on Object Manager >> Click on create >> Click on Custom Object.
- 1.1 Enter the label name >> Laptops
- 1.2 Plural label name >> Laptops
- 1.3 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

The screenshot shows the Salesforce Setup interface. At the top, there's a navigation bar with icons for Home, Object Manager, and a search bar labeled 'Search Setup'. Below the navigation is a header bar with 'SETUP > OBJECT MANAGER' and the object name 'Total Laptops'. The main content area is divided into two columns: 'Details' on the left and 'Custom' on the right. The 'Details' column lists various configuration options like Fields & Relationships, Page Layouts, and Record Types. The 'Custom' column displays specific details for the 'Total Laptops' object, including its API Name ('Total\_Laptops\_\_c'), Singular Label ('Total Laptops'), and Plural Label ('Total Laptops'). On the far right, there are 'Edit' and 'Delete' buttons. A sidebar on the left provides links to other setup features such as Fields & Relationships, Page Layouts, and Lightning Record Pages.

## 1.2 Create Consumer Object

To Create an object:

1. From the Setup page >> Click on Object Manager >> Click on create >> Click on Custom Object.

1.1 Enter the label name >> consumer

1.2 Plural label name >> consumer

1.3 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

The screenshot shows the Salesforce Setup interface under the Object Manager section. A new object named 'consumer' is being created. The 'Details' tab is selected, showing the following configuration:

- Description:** consumer
- API Name:** consumer\_c
- Custom:** ✓
- Singular Label:** consumer
- Plural Label:** consumer

On the right side, under 'Enable Reports', 'Track Activities', and 'Track Field History', checkboxes are checked. Under 'Deployment Status', 'Deployed' is selected. The 'Help Settings' link points to the Standard salesforce.com Help Window.

## 1.3 Create Laptop Bookings Object

To Create an object:

1. From the Setup page >> Click on Object Manager >> Click on create >> Click on Custom Object.
  - 1.1 Enter the label name >> Laptop Bookings
  - 1.2 Plural label name >> Laptop Bookings
  - 1.3 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

The screenshot shows the Salesforce Setup interface under the Object Manager section. A new object named 'Laptop Bookings' is being created. The 'Details' tab is selected, showing the following configuration:

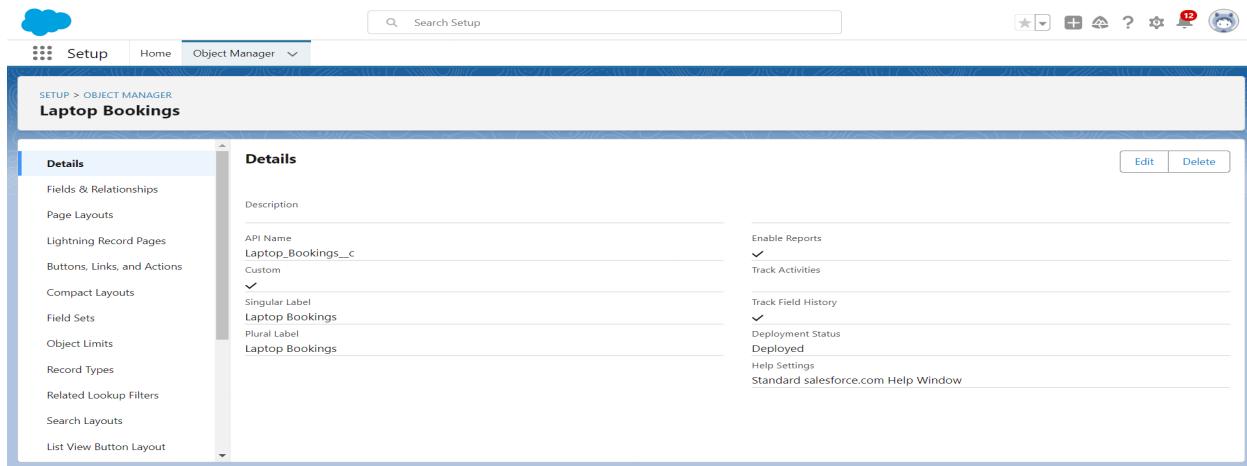
- Description:** Laptop Bookings
- API Name:** Laptop\_Bookings\_c
- Custom:** ✓
- Singular Label:** Laptop Bookings
- Plural Label:** Laptop Bookings

On the right side, under 'Enable Reports', 'Track Activities', and 'Track Field History', checkboxes are checked. Under 'Deployment Status', 'Deployed' is selected. The 'Help Settings' link points to the Standard salesforce.com Help Window.

## 1.4 Create Billing Process Object

To Create an object:

1. From the Setup page >> Click on Object Manager >> Click on create >> Click on Custom Object.
  - 1.1 Enter the label name >> Billing Process
  - 1.2 Plural label name >> Billing Process
  - 1.3 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



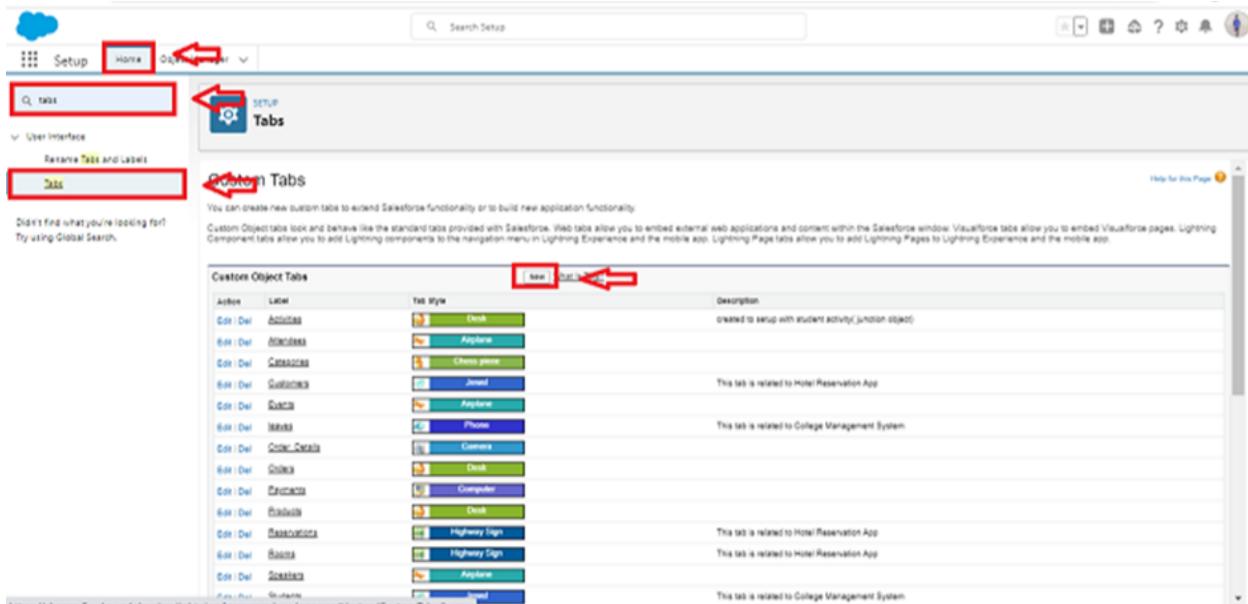
## 2. Creating Tabs

A **tab** is a user interface component that helps users organize, access, and manage records of different **objects** within the platform. Salesforce objects represent different types of data, such as Accounts, Contacts, Opportunities, or Custom Objects, and each object has its own tab. Here's a more detailed explanation of what a tab is and how it functions.

## 2.1 Creating Total Laptops 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.

## 2.2 Creating Remaining tabs:

1. Now create the Tabs for the remaining Objects, they are "consumer,Laptop Booking,Billing process".
2. Follow the same steps as 2.1 .

Custom Tabs

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

Custom Object Tabs

Action	Label	Tab Style	Description
Edit   Del	Billing.Process	Shopping Cart	
Edit   Del	consumer	People	
Edit   Del	Landoo Bookings	Presenter	
Edit   Del	Total Laptops	Laptop	

Web Tabs

No Web Tabs have been defined

Visualforce Tabs

New | What Is This?

### 3. Create Lightning App

#### To Create a Lightning app page:

1. Go to setup page >> search “app manager” in quick nd >> select “app manager” >> click on New lightning App.

App Manager | Salesforce

Search Setup

app manager

dev namespaces

Clone (Beta)

New Lightning App

Lightning Experience App Manager

Quickly create new Lightning apps by cloning existing apps. To use the beta feature, indicate that you've read all legal requirements and agree to participate by toggling Enable App Cloning. See additional details and terms in the Winter '23 release notes.

Enable App Cloning:  Enabled

35 items • Sorted by App Name • Filtered by All appmenagers • Table Type

App Name	Developer Name	Description	Last Modified	App Type	Vl...
All Tabs	AltTabSet		04/12/2022, 10:13 am	Classic	
Analytics Studio	Insights	Build CRM Analytics dashboards and apps	04/12/2022, 10:13 am	Classic	
App Launcher	AppLauncher	App Launcher tabs	04/12/2022, 10:13 am	Classic	
Boil Solutions	LightningBolt	Discover and manage business solutions designed for your industry.	04/12/2022, 10:16 am	Lightning	
Chatter Desktop	ChatterDesktop	Chatter Desktop is an Adobe AIR-based desktop application that lets Chatter users stay connecte...	23/12/2022, 4:04 pm	Connected (Managed)	
Chatter Mobile for BlackBerry	ChatterForBlackBerry	The Salesforce.com Chatter Mobile app lets you access Chatter data on the go. Use it to view fe...	23/12/2022, 4:05 pm	Connected (Managed)	
College Management System	Naadem	demo app	08/12/2022, 4:18 pm	Lightning	
Community	Community	Salesforce CRM Communities	04/12/2022, 10:13 am	Classic	
Content	Content	Salesforce CRM Content	04/12/2022, 10:13 am	Classic	
Data Manager	DataManager	Use Data Manager to view limits, monitor usage, and manage recipes.	04/12/2022, 10:13 am	Lightning	

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

**App Settings**

**Navigation Items**

**Available Items**

- Accounts
- All Sites
- Alternative Payment Methods
- Analytics
- App Launcher
- Appointment Categories
- Appointment Invitations
- Approval Requests
- Logout Action Source

**Selected Items**

- Total Laptops
- Laptop Bookings
- consumer
- Billing Process

5. To Add User Profiles:  
Search profiles (System administrator) in the search bar >> click on the arrow button >> save & finish.

**User Profiles**

**Available Profiles**

- Agent
- Analytics Cloud Integration User
- Analytics Cloud Security User
- Authenticated Website
- Authenticated Website
- B2B Reordering Portal Buyer Profile
- Contract Manager
- Custom: Marketing Profile
- Custom: Sales Profile
- Customer Contact Profile

**Selected Profiles**

- System Administrator

## 4. Fields

Fields are fundamental building blocks of objects. They store individual pieces of data in a record and define the data type, behavior, and purpose of that data. Fields are associated with both **standard** and **custom objects**. Fields in Salesforce are data containers within objects that store specific pieces of information, such as text, numbers, or dates.

### Steps to Create Field For a Object:

1. Go to setup >> click on Object Manager >> type object name in search bar >> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data Type and fill details like label, name, etc.
4. Click on Next >> Next >> Save and new.

### 4.1 Creating The Fields In Consumer Object

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The 'Object Manager' dropdown is open, showing 'consumer' as the selected object. The main content area displays the 'Fields & Relationships' section for the 'consumer' object. On the left, a sidebar lists various object settings: Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, and List View Button Layout. The 'Fields & Relationships' table has the following data:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address_c	Text Area(255)		
consumer Status	consumer_Status_c	Picklist		
consumer_name	Name	Text(80)		✓
Created By	CreatedBy	Lookup(User)		
Email	Email_c	Email		
Last Modified By	LastModifiedBy	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone number	Phone_number_c	Phone		

1. Create Phone number field:
  - Select Data Type as a “Phone” and click on next.
  - Field Label: Phone number
  - Field Name : gets auto generated
  - Click the required option checkbox.
2. Create Email field:
  - Select Data Type as a “Email” and click on next.
  - Field Label: Email
  - Field Name : gets auto generated
3. Create Address field:
  - Select Data Type as a “Text Area” and click on next.
  - Field Label: Address
  - Field Name : gets auto generated
4. Create Consumer status field:
  - Select Data Type as a “Picklist” and click on next.
  - Field Label: consumer Status
  - Value - Select enter values with each value separated by a new line
    - 1.Student
    - 2.Employee
    - 3.Others
  - Select required Field Name :It's gets auto generated

## **4.2 Creating The Fields In Laptop Bookings Object**

1. Create Laptop Names field:
  - Select Data Type as a “Picklist” and click on next.
  - Field Label: Laptop Names
  - Picklist values are: 1.Dell 2.Acer 3.Hp 4.Mac
  - Field Name : gets auto generated

2. To create Core Type field:

- Select Data Type as a “Picklist” and click on next.
- Field Label: Core Type
- Picklist values are: core i3, core i5, core i7, Bionic chip
- Select Required

3. To create field dependency in the Laptop Booking Object:

- Click field dependency and next.
- Select Controlling Field as Laptop Names and Dependent Field as Core Type
- 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 mac bionic chip include the values for it.

Laptop names:	Dell	Acer	Hp	Mac
core type:	core i3 core i5 core i7	core i3 core i5	core i3 core i5 core i7	Bionic chip

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

1. Create "Master-detail Relationship" in Laptop Booking 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 “Master-Detail Relationship” and 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: Consumer  
Field Name :It's gets auto generated
- Click on Next >> Next >> Save and new.

## 2. Create "Currency" field

- Field Label: Amount
- Length: (18,0)
- Field Name :It's gets auto generated
- Click on Next >> Next >> Save and new

## 3. Create relationship to "Total Laptops" object

- Select Laptop booking object in the object manager.
- Now click on “Fields & Relationships” >> New
- Select Data Type as a “Lookup Relationship” and Click on Next
- Click on the Related to drop down and Select the “Total Laptops” object and click on Next
- Change the Field Label: Total No Of Laptops
- Field Name :It's gets auto generated
- Click on Next >> Next >> Save and new.

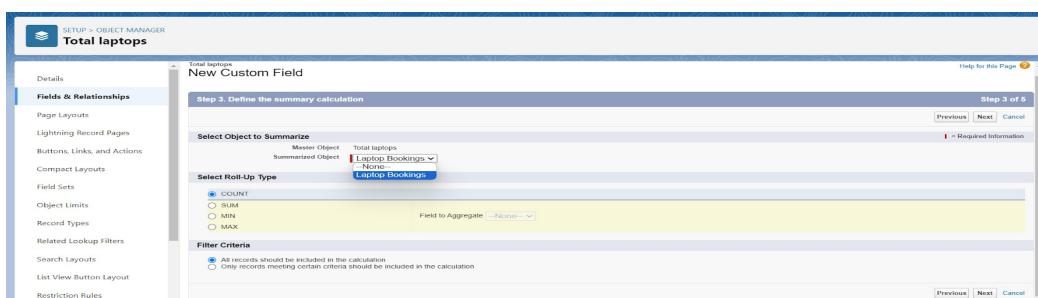
## 4. Create Email field in Laptop Bookings Object

## 5. Create "Rollup Summary" in Total Laptops Object

- Field Label: Laptops delivered
- Field Name :It's gets auto generated

## 6. Select the Laptop Bookings in the Summarized Object

## 7. Select the count Radio button in the select Roll-up Type



## 8. Create "Laptops Available" field:

- 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 - Total\_no\_of\_laptops\_\_r.Laptops\_delivered\_\_c"

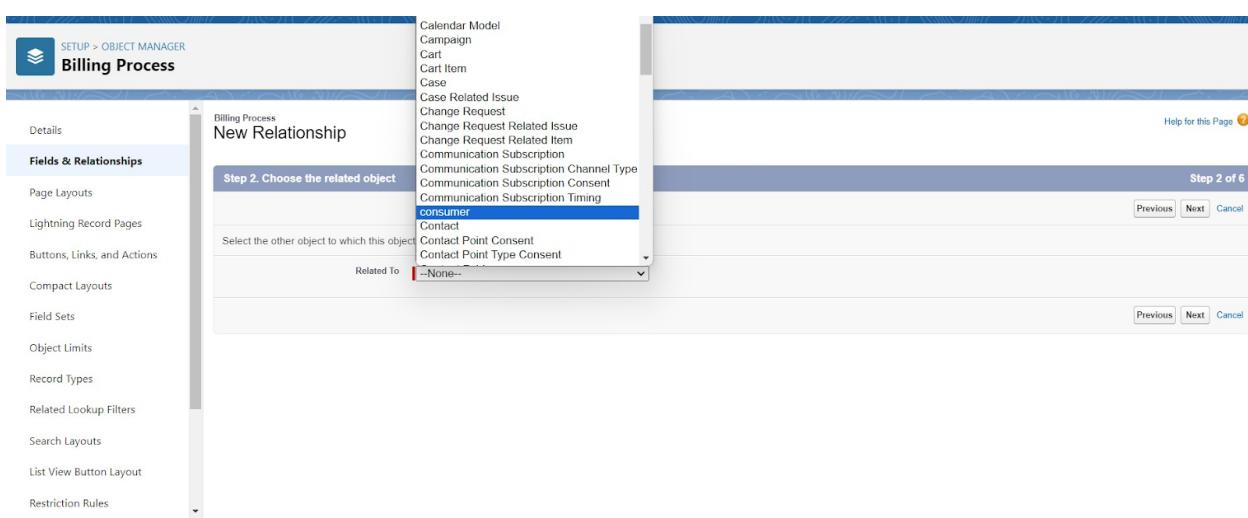
## 9. Create "how many months" field in Laptop Bookings object:

- Label: how many months
- Picklist values are: 1, 2, 3, 4, 5

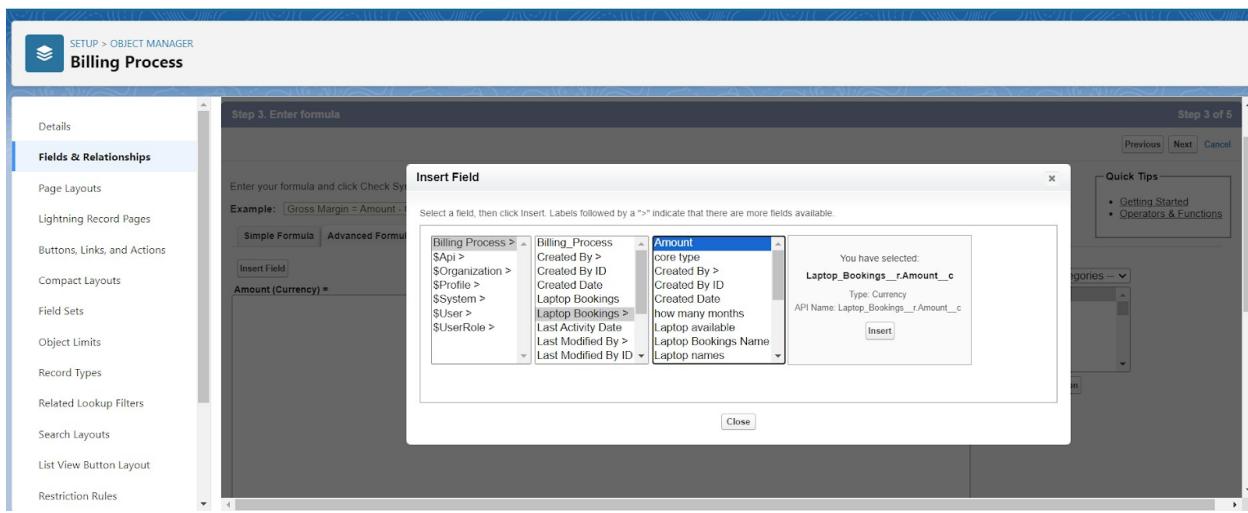
## 4.4 Creation of Fields & Relationship for Billing Process Object

### 1. Create relationship for consumer object:

- Now click on "Fields & Relationships" >> New in Billing Process object
- 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
- Click on the Related to drop down and Select the consumer object and click on Next
- Change the Field Label: Name and click on save and new



2. Create "Lookup Relationship" in Laptop Booking object
3. Create "Payment Mode" as picklist values (Cash, Check, Credit card, Debit card, UPI, Phonepe, Gpay, Paytm)
4. Create a Cross object formula Field in billing process Object:
  - Select Data Type as a "Formula" and Click on Next
  - Enter the Field label: Amount



- formula: " Laptop\_Booking\_\_r.Amount\_\_c "

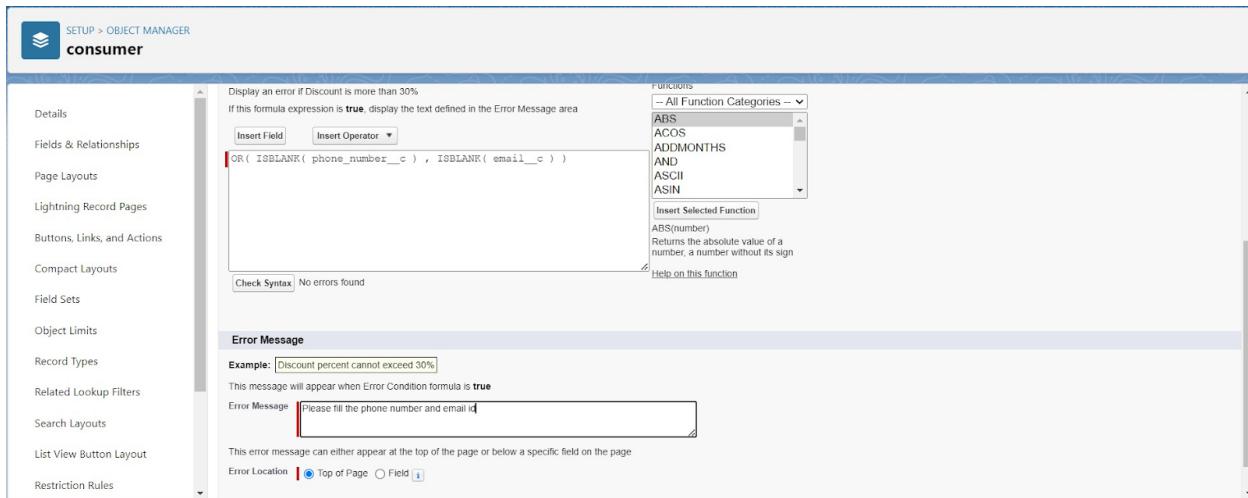
#### 4.5 Creating the field in Total Laptops object

- Go to setup >> click on Object Manager >> type object name(Total Laptops) in search bar >> click on the object.
- Now click on "Fields & Relationships" >> New Select Data type as a "Formula" and Click on Next
- Field Label: Laptops Available Field Name : It's gets auto generated Select the Formula Return Type as "Number"

## 5. 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.

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. 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.

### 6.1 Create Owner 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. In the top navigation bar, 'Setup' is selected. Below it, the 'Profiles' tab is active. On the left sidebar, 'Profiles' is also highlighted. A search bar at the top says 'Search Setup'. The main content area displays a 'Profile Detail' for 'owner'. The 'Profile' section shows the name 'owner' and a note about record types. The 'Profile Detail' section includes fields for 'Name' (owner), 'User License' (Salesforce), 'Description', 'Created By' (Hima Meghana Siddi), and 'Modified By' (Hima Meghana Siddi). Buttons for 'Edit', 'Clone', 'Delete', and 'View Users' are present. The 'Page Layouts' section lists standard object layouts for various objects like Global, Invoice, Email Application, Invoice Line, Home Page Layout, Lead, Account, Legal Entity, Alternative Payment Method, and Location. Each layout has a 'View Assignment' link next to it. A 'Custom Profile' checkbox is checked. A 'Help for this Page' button is in the top right corner.

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

The screenshot shows the Salesforce Setup interface with the following details:

- Header:** Search bar with "Search Setup" and a toolbar with icons for star, plus, cloud, question mark, gear, and user.
- Left Sidebar:** "Setup" tab selected, "Home" and "Object Manager" tabs, and a search bar with "Q profiles". Under "Users", "Profiles" is selected.
- Content Area:**
  - Profiles:** A table showing permissions for various objects like Fulfillment Orders, Work Orders, etc.
  - Custom Object Permissions:** Tables for Billing Process, consumer, Laptop Bookings, and Total Laptops.
  - 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.

## 6.2 Create 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. Scroll down to Custom Object Permissions and Given access permissions for Total Laptops, consumer , Laptop Bookings and Billing Process objects as mentioned in the below diagram.

The screenshot shows the Salesforce Setup interface with the following details:

- Header:** Search bar with "Search Setup" and a toolbar with icons for star, plus, cloud, question mark, gear, and user.
- Left Sidebar:** "Setup" tab selected, "Home" and "Object Manager" tabs, and a search bar with "Q profiles". Under "Users", "Profiles" is selected.
- Content Area:**
  - Communication Subscriptions:** A table showing permissions for various communication objects.
  - Custom Object Permissions:** Tables for Billing Process, consumer, Laptop Bookings, and Total Laptops.
  - Session Settings:** Session Times Out After: 2 hours of inactivity, Session Security Level Required at Login: None.

## 7. 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.

### Steps:

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.



### 7.1 Create Owner role

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

2. Click and save it.

### 7.2 Create Agent Role

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

The screenshot shows the Salesforce Setup Roles page. On the left, there's a sidebar with categories like Users, Feature Settings, Sales, Service, and Case Teams. Under Users, 'Roles' is selected. The main area is titled 'Creating the Role Hierarchy' and shows a tree view of roles. The hierarchy is as follows:

- GVPCE
  - CEO
  - CFO
  - COO
  - owner
- owner
  - Agent
  - SVP\_Customer\_Service & Support
    - Customer\_Support\_International
    - Customer\_Support\_North\_America
- Customer\_Support\_North\_America
  - Installation & Repair Services

## 8. 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
  - First Name : vicky
  - Last Name : y
  - Alias : Give a Alias Name

- Email id : Give your Personal Email id
- Username : Username should be in this form: text@text.text
- Nick Name : Give a Nickname
- User license : Salesforce

### 3. If the user is

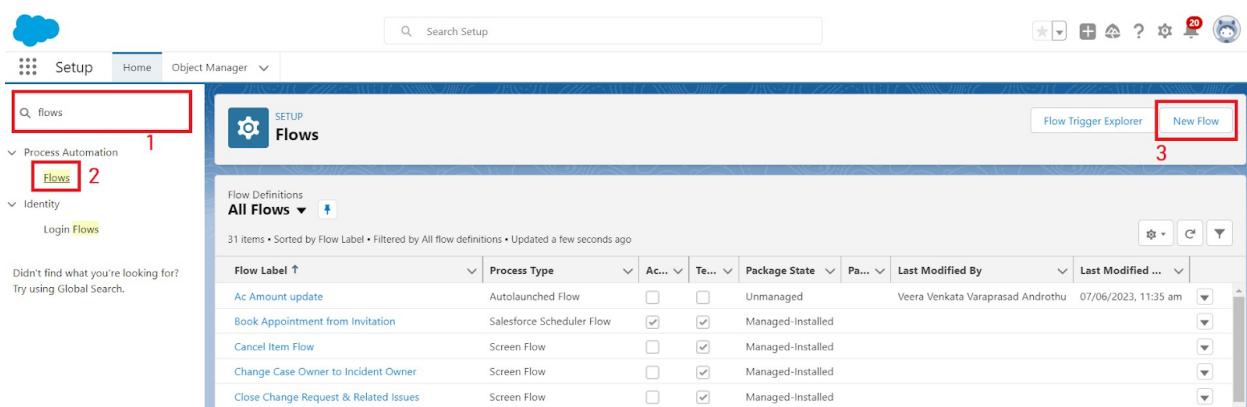
- "owner": Set the profiles and role field to owner
- "Agent": Set the profiles and role field to Agent

## 9. Flows

"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. 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.

### 9.1 Create a Flow on dell laptop

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 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](#)

5. Under the Record-triggered Flow Click on “+” Symbol and In the Drop down List select the “Decision Element”.
6. Enter the Details Label: Field should be Update, API name: Gets Automatically Generated.
7. Enter the Outcome Details Label: dell , Outcome API name: Gets Automatically Generated.
  - Resource: Select \$Record.Laptop\_name\_\_c.
  - Operator: Select Equals.
  - Value: Select dell Add the same outcome order to acer , hp, mac.
  - Rename Default outcome as False
  - Click done.
8. Beside dell there is a symbol ‘+’ click on that. Again select decision
9. Enter the Details Label: Field should Update(any one u want), API name: Gets Automatically Generated.
10. Select the Outcome Details Label: dell core i3 , Outcome API name:

Gets Automatically Generated. Resource: Select  
`{!$Record.core_type__c}`. Operator: Select Equals. Value: Select core i3.

11. Add the dell i5 and dell i7 outcome details in the similar way

**Edit Decision**

\* Label: field updated      \* API Name: field\_updated

Description:

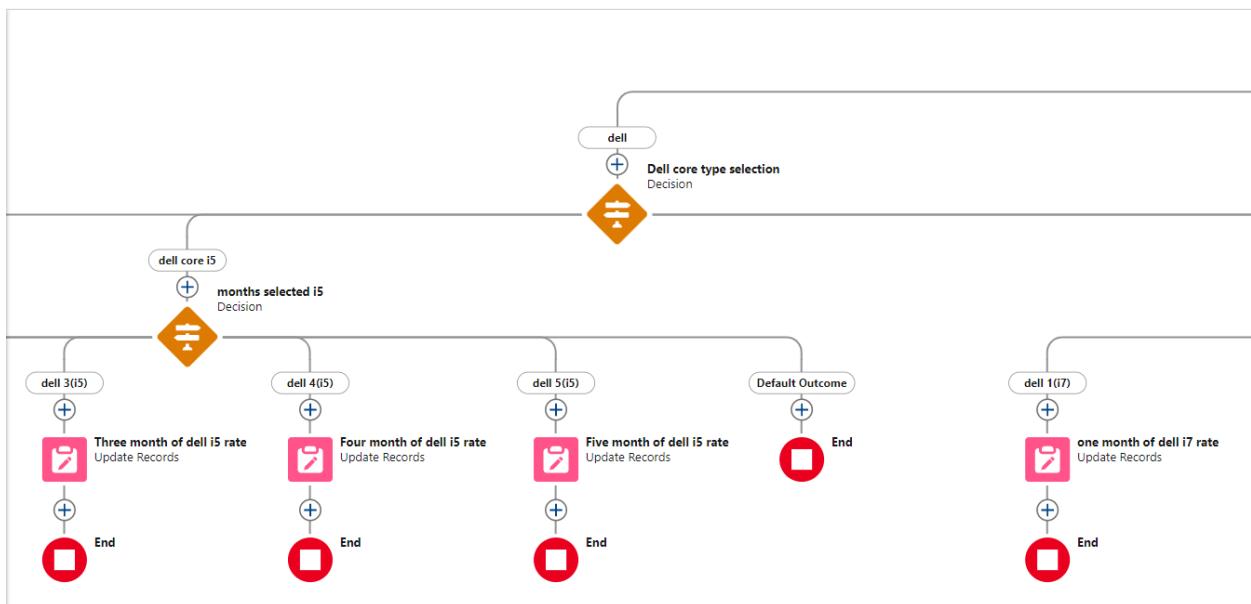
**Outcomes** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME ORDER	OUTCOME DETAILS	DELETE OUTCOME
1 dell core i3	* Label: dell core i3      * Outcome API Name: dellcore_i3 Condition Requirements to Execute Outcome: All Conditions Are Met (AND)	<a href="#">Delete Outcome</a>
2 dell core i5		
3 dell core i7		
Default Outcome	Resource: \$Record > core type      Operator: Equals      Value: core i3	<a href="#">Delete</a>

[Cancel](#) [Done](#)

12. So go to the flow page select '+' after core i3 then again select the decision.
13. Enter the Details Label: months selected , API name: Gets Automatically Generated.
14. Enter the Outcome Details Label: dell 1(i3) , Outcome API name: Gets Automatically Generated.
- Resource: Select Record.how many months
  - Operator: Select Equals
  - Value: 1
15. Similarly Enter the OutcomeDetails as
- dell 2(i3) -> resource: Record.how many months , value: 2
  - dell 3(i3) -> resource: Record.how many months , value: 3
  - dell 4(i3) -> resource: Record.how many months , value: 4
  - dell 5(i3) -> resource: Record.how many months , value: 5

16. Enter the Details Label: one month of dell i3 rate , API name: Gets Automatically Generated. Enter label name in a similar way for other update records.
17. 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 all these finally and update the Amount field value based on the number of months.
18. Perform these steps for core i5 and i7.
19. Click done.



## 9.2 Create A Flow on Acer Laptop

1. Enter the Details Label: Acer core type selection, API name: Gets Automatically Generated.
2. 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.
3. Similarly create outcomes for acer core i5 and acer core i7 also.

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	
acer core i3	*Label acer core i3	*Outcome API Name acer_core_i3
acer core i5		
acer core i7		
Default Outcome		

Condition Requirements to Execute Outcome  
All Conditions Are Met (AND)

Resource: \$Record > core type, Operator: Equals, Value: core i3

+ Add Condition

When to Execute Outcome  
 If the condition requirements are met  
 Only if the record that triggered the flow to run is updated to meet the condition requirements

Cancel Done

#### 4. Create Outcome Details Labels: acer 1(i3), acer 2(i3), acer 3(i3), acer 4(i3), acer 5(i3)

Edit Decision

\*Label  
acer months selected \*API Name  
acer\_months\_selected

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.

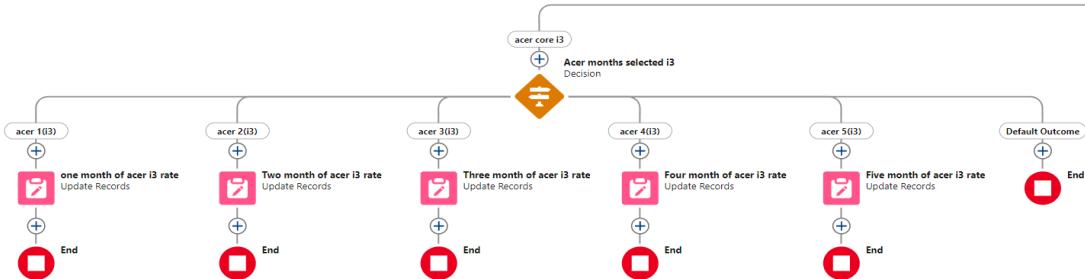
OUTCOME ORDER	OUTCOME DETAILS	
acer 1(i3)	*Label acer 1(i3)	*Outcome API Name acer_1_i3
acer 2(i3)		
acer 3(i3)		
acer 4(i3)		
acer 5(i3)		

Condition Requirements to Execute Outcome  
All Conditions Are Met (AND)

Resource: \$Record > how many months, Operator: Equals, Value: 1

Cancel Done

5. Enter the Details Label: one month of acer i3 rate , API name: Gets Automatically Generated.
6. 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.
7. Perform these steps for core i5 and i7.
8. Click done.



### 9.3 Create A Flow On Hp Laptop

1. Go to flow page
2. Beside hp there is a symbol '+' click on that.
3. Enter the Details Label: HP core selection, API name: Gets Automatically Generated.
4. Select the Outcome Details Label: hp core i3 , Outcome API name:
  - Resource: Select Record.core type.
  - Operator: Select Equals.
  - Value: Select hp i3.
5. Similarly create outcomes for hp core i5 and hp core i7 also.

#### 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.

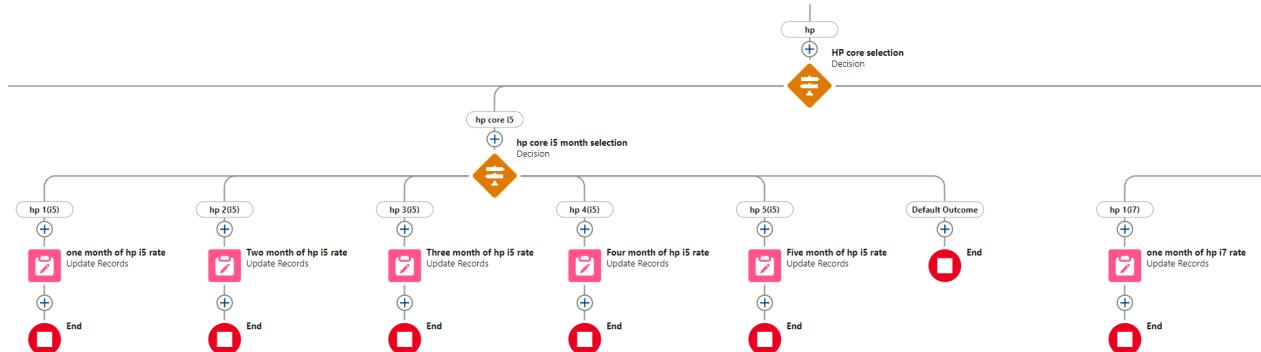
OUTCOME ORDER	OUTCOME DETAILS	Actions
hp core i3	*Label: hp core i3 *Outcome API Name: hp_core_i3 Condition Requirements to Execute Outcome: All Conditions Are Met (AND) Resource: \$Record > core type X Operator: Equals Value: core i3 + Add Condition	<a href="#">Delete Outcome</a>
hp core i5		
hp core i7		
Default Outcome		

**When to Execute Outcome**

If the condition requirements are met  
 Only if the record that triggered the flow to run is updated to meet the condition requirements

[Cancel](#) [Done](#)

6. Beside hp there is a symbol '+' click on that.
7. Enter the Details Label: hp core i5 month selection , API name: Gets Automatically Generated. Enter the Outcome Details Label: hp 1(i3) , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.how many months.
  - Operator: Select Equals.
  - Value: 1.
8. Similarly Create Outcome Details Labels: hp 2(i3), hp 3(i3), hp 4(i3), hp 5(i3).
9. 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.
10. Perform these steps for other core types.
11. Click done.



## 9.4 Creating A Flow on Mac laptop

1. Beside mac there is a symbol '+' click on that.
2. Enter the Details Label: mac should be Updated, API name: Gets Automatically Generated.
3. Select the Outcome Details Label: mac laptop , Outcome API name: Gets Automatically Generated.
  - Resource: Select Record.core type.
  - Operator: Select Equals.

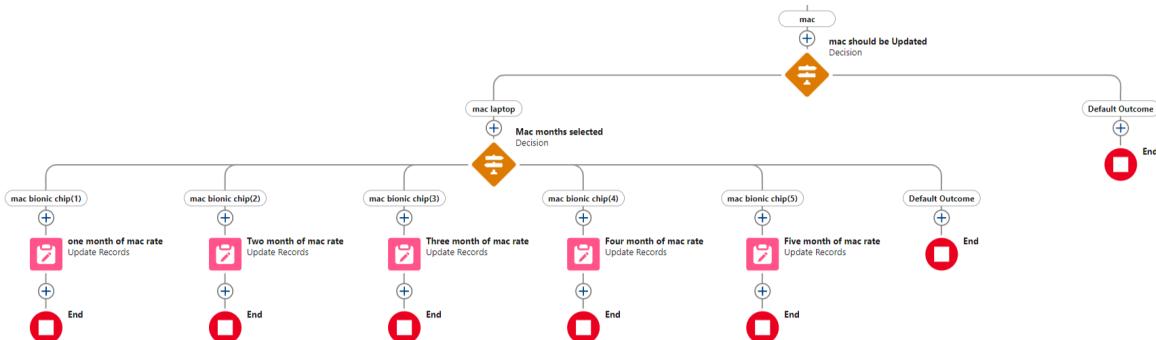
- Value: Select Bionic Chip.
- Enter the Details Label: Mac months selected , API name: Gets Automatically Generated.
  - Enter the Outcome Details Label: mac bionic chip(1) , Outcome API name: Gets Automatically Generated.
    - Resource: Select Record.how many months.
    - Operator: Select Equals.
    - Value: 1.
  - Similarly Create Outcome Details Labels: mac bionic chip(2),mac bionic chip(3), mac bionic chip(4), mac bionic chip(5)
  - Enter the Details Label: one month of mac rate , API name: Gets Automatically Generated.
  - 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 and click done.

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)		
mac bionic chip(4)		
mac bionic chip(5)		
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



## 10. Apex Trigger and Handler Class

### Apex

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

### Trigger

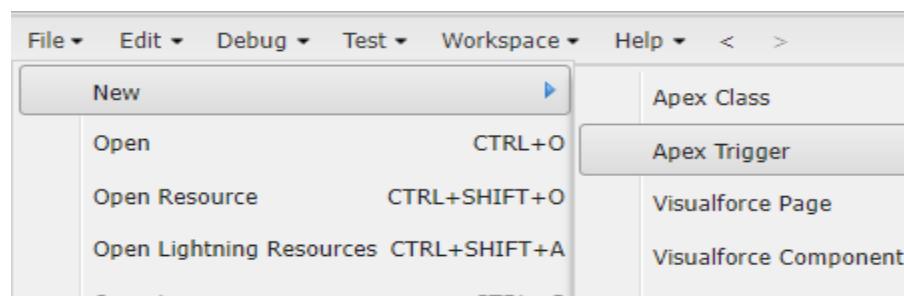
A trigger is a set of Apex code that runs before or after DML(Data Manipulation Language) events. A DML event could be a variety of data processing tasks that include the standard insert, update, and delete commands. With Apex triggers, you can automate tasks that would otherwise be nearly impossible to accomplish using only the Salesforce user interface.

There are two Salesforce Apex trigger types:

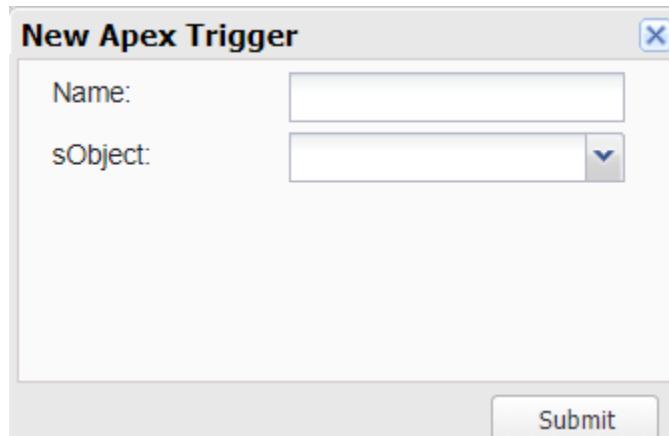
1. Before Triggers
2. After triggers

#### Create a new trigger:

1. Click on developer console and you will be navigated to a new console window.
2. Click on the File menu in the toolbar, and click on new Trigger.



3. Enter the trigger name and the object to be triggered.



## 10.1 To Create Laptop Booking Trigger

**Purpose:** This 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.

### Trigger Code:

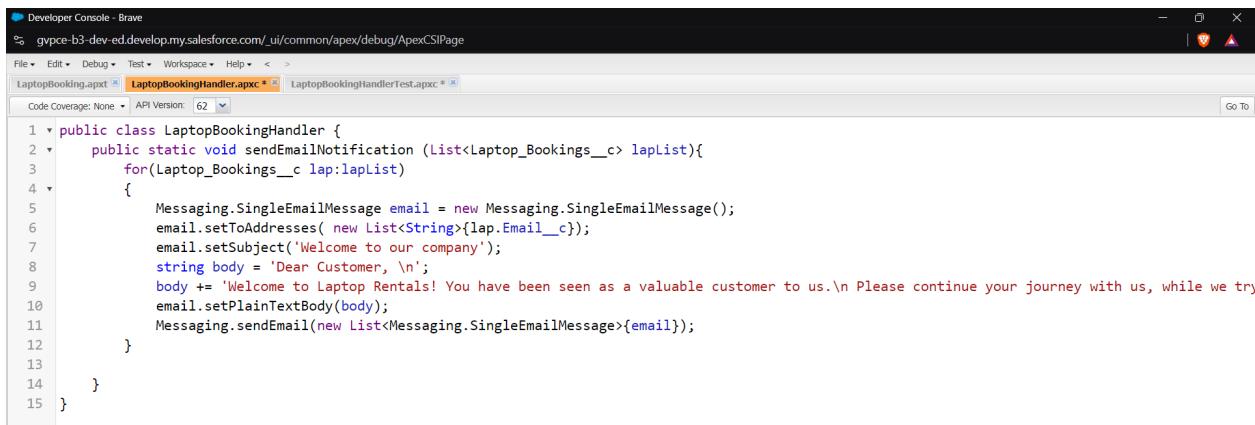
A screenshot of the Salesforce Apex code editor. The URL in the address bar is 'gvpce-b3-dev-ed.develop.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage'. The top menu includes File, Edit, Debug, Test, Workspace, Help, and navigation icons. The tabs show 'LaptopBooking.apxc' (selected), 'LaptopBookingHandler.apxc', and 'LaptopBookingHandlerTest.apxc'. The code coverage is set to 'None' and API Version is '62'. The code itself is:

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

## 10.2 Create Handler Class

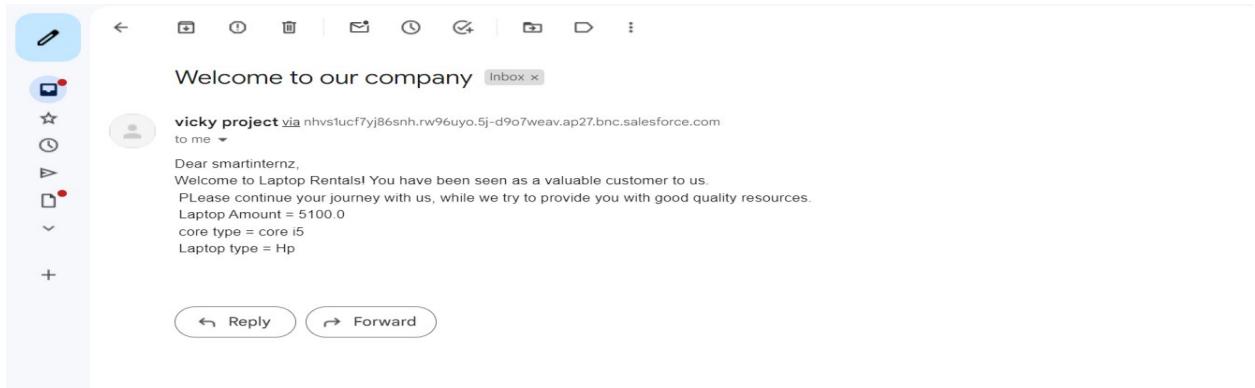
**Purpose:** In LaptopBookingHandler class, sendEmailNotification method is defined which automates sending email with the details like laptop type, core type, amount when a customer books laptops

## Code:



```
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 Customer, \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 we try
10             email.setPlainTextBody(body);
11             Messaging.sendEmail(new List<Messaging.SingleEmailMessage>{email});
12         }
13     }
14 }
15 }
```

## The output:



## 11. Reports

Reports give you access to your Salesforce data. It displays it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics.

### Types of Reports in Salesforce:

- Tabular
- Summary
- Matrix
- Joined Reports

## 11.1 Create Report

1. Go to the app "Laptop Rentals".
2. click on the reports tab and Click New Report.
3. Select report type from category and click on start report.
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 bucket list.

Edit Bucket Column

* Field	* Bucket Name	
Amount	x	types of versions

	Range	Bucket	
Add ►	<= 900	basic	x
Add ►	> 900 to 1500	intermediate	x
Add ►	> 1,500 to 10000	high	x
	> 10,000	very high	x

Treat empty Amount values in the report as zeros.

Cancel Apply

7. Select Types of version in Group By Rows to create a summary report.
8. Click on Save & run it.

## Report output:

LAPTOP RENTALS Total Laptops Laptop Bookings consumer Billing Process Reports Dashboards

Report: Total Laptops with Laptop Bookings and consumer  
Laptop Analytics

Total Records	Total Amount
7	₹85,350

types of versions	Total Laptops	Laptop Bookings	Consumer	Amount
basic (1)	1	distribution to poor	Meghana	₹50
Subtotal				₹50
intermediate (1)	3	interview purpose	likhit	₹1,200
Subtotal				₹1,200
high (4)	3	sales	Meghana	₹3,000
	2	agencies	Meghana	₹4,000
	50	GDPR_Auditor	likhit	₹2,000
	2	hawai	mac_user	₹5,100
Subtotal				₹14,100
very high (1)	50	sales	likhit	₹70,000
Subtotal				₹70,000
Total (7)				₹85,350

## 11.2 Sharing Report To Owner

1. Click edit drop down and select subscribe option

Total Records	Total Amount	Total Laptops Available			
6	₹19,900	94			
Subtotal					
Basic (1)	Dell i3	Shruthi	₹1,000	48	Dell core i3 50
Intermediate (1)	Acer	Swetha	₹1,500	46	50
High (4)	Acer	Swetha	₹4,800	46	50
	Acer	Swetha	₹3,800	46	50
	Acer	Swetha	₹3,800	46	50
	Dell i3 Booking	Shruthi	₹5,000	48	Dell core i3 50
Subtotal			₹17,400	94	
Total (6)			₹19,900	94	

2. Fill the details like frequency, time, mail to whom the report has to be sent.

Settings

Frequency

Daily  Weekly  Monthly

Time

8:00 am

Attachment

Attach File

Recipients

Send email to

Me

Edit Recipients

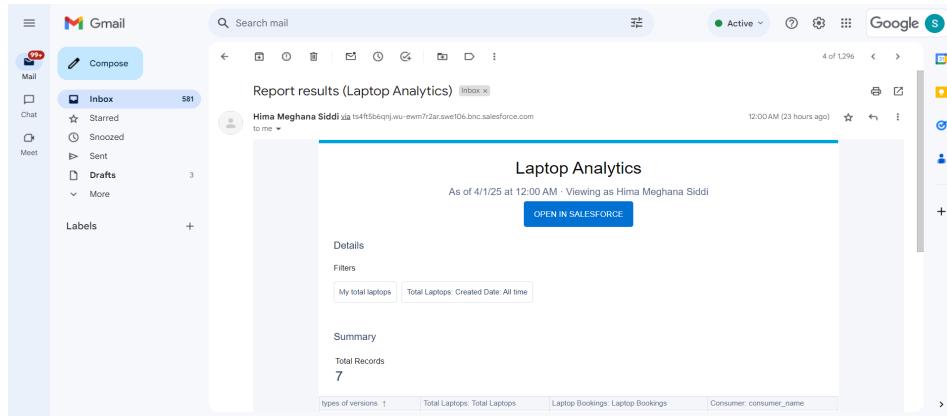
Run Report As

Me  Another Person

Cancel Save

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.

The report being sent to mail looks like:



## 12. Dashboards

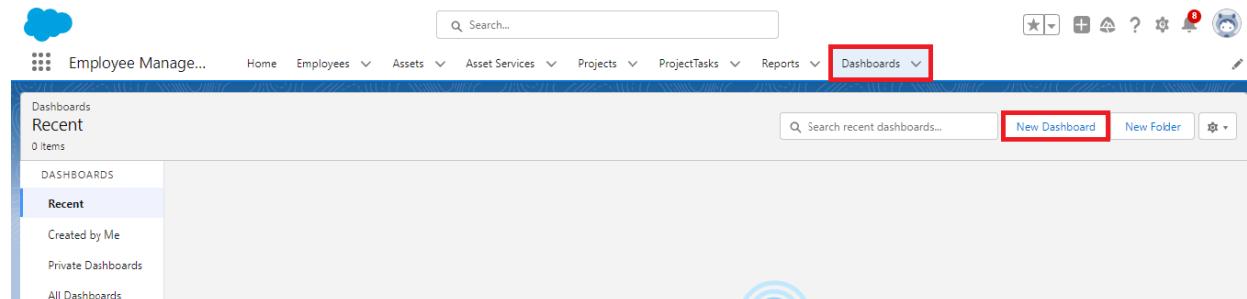
Dashboards help you visually understand changing business conditions so you can make decisions based on the real-time data you've gathered with reports. Use dashboards to help users **identify trends**, sort out quantities, and measure the impact of their activities.

### 12.1 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.

### 12.2 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 and Click done.

## Dashboard output:

LAPTOP RENTALS Total Laptops Laptop Bookings consumer Billing Process Reports Dashboards

Dashboard data analytics of laptops total amount of data in dashboards Last refreshed 3 days ago. Refresh this dashboard to see the latest data. As of 01-Jan-2025, 11:37 pm Viewing as Hima Meghana Siddi

Refresh Edit Subscribe

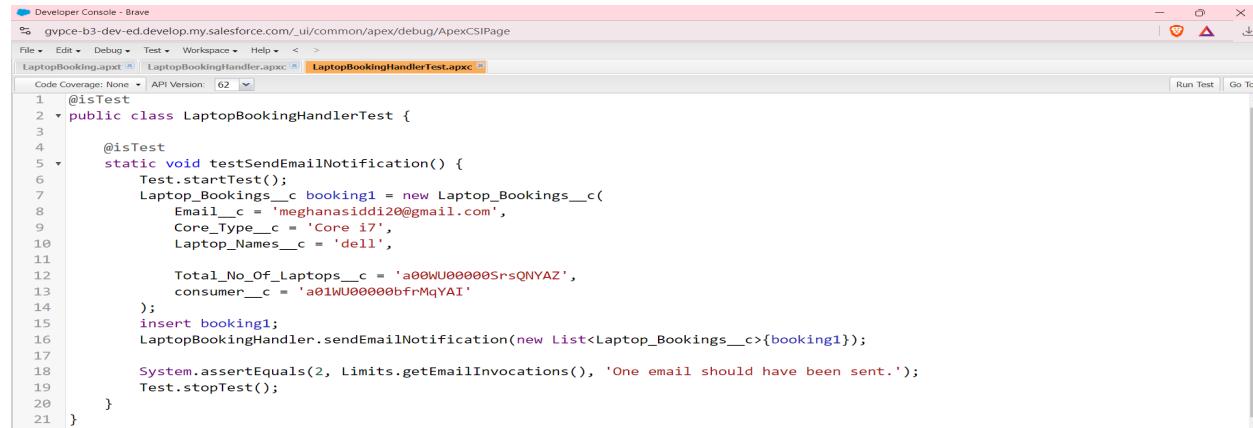
consumer: consumer_...	Laptop Bookings: Laptop ...	Total No Of Laptops: Total...
likhit	sales	50
likhit	GDPR_Auditor	50
likhit	interview purpose	3
mac_user	hawai	2
Meghana	sales	3
Meghana	agencies	2
Meghana	distribution to poor	1

## 5. Testing and Validation

### 5.1 Unit Testing

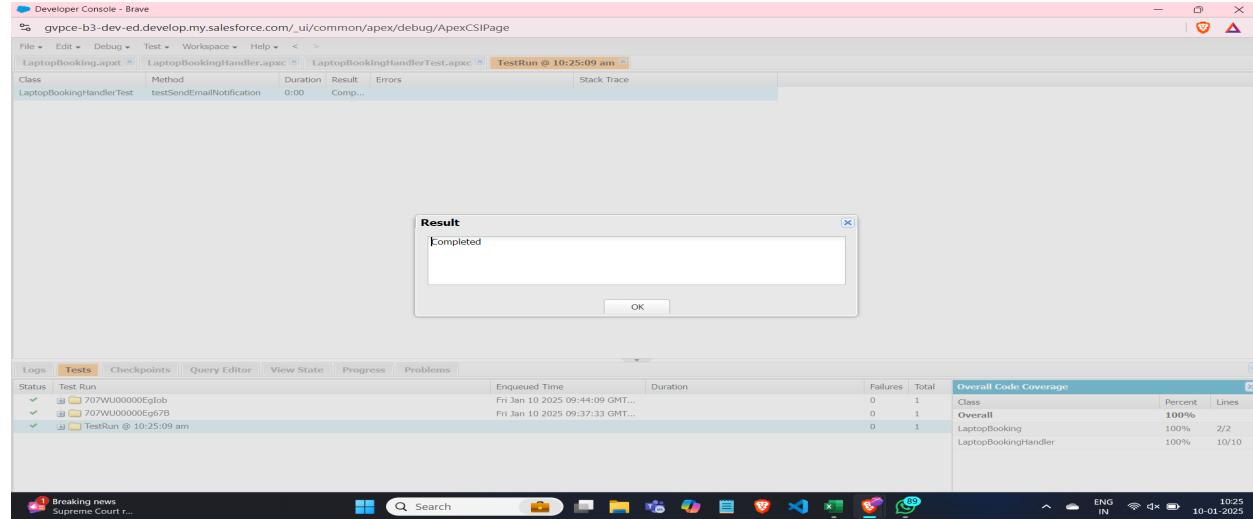
#### Testing the Laptop Booking Handler class

#### LaptopBookingHandlerTest.apxc code snippet:



```
Developer Console - Brave
gvpcce-b3-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >
LaptopBooking.apxc [ ] LaptopBookingHandler.apxc [ ] LaptopBookingHandlerTest.apxc [ ]
Code Coverage: None API Version: 62 Run Test Go To
1 @isTest
2 public class LaptopBookingHandlerTest {
3
4     @isTest
5     static void testSendEmailNotification() {
6         Test.startTest();
7         Laptop_Bookings__c booking1 = new Laptop_Bookings__c(
8             Email__c = 'meghanasiddi20@gmail.com',
9             Core_Type__c = 'Core i7',
10            Laptop_Names__c = 'dell',
11
12            Total_No_Of_Laptops__c = 'a00WU000005rsQNYAZ',
13            consumer__c = 'a01WU000000bfrMqYAI'
14        );
15        insert booking1;
16        LaptopBookingHandler.sendEmailNotification(new List<Laptop_Bookings__c>{booking1});
17
18        System.assertEquals(2, Limits.getEmailInvocations(), 'One email should have been sent.');
19        Test.stopTest();
20    }
21 }
```

#### Output:

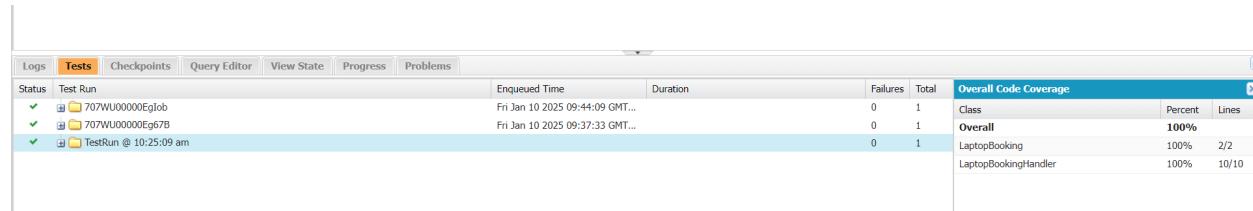


The screenshot shows the Salesforce Developer Console interface. At the top, there's a navigation bar with links like File, Edit, Debug, Test, Workspace, Help, and a search bar. Below the navigation is a tabs section with LaptopBooking.apxc, LaptopBookingHandler.apxc, and LaptopBookingHandlerTest.apxc. The LaptopBookingHandlerTest.apxc tab is active. On the right side of the tabs, there's a "Run Test" button and a "Go To" button.

Below the tabs, there's a code editor window containing the test class code. The code defines a class `LaptopBookingHandlerTest` with a single test method `testSendEmailNotification`. The test creates a new `Laptop\_Bookings\_\_c` record, inserts it, and then calls the `sendEmailNotification` method on the `LaptopBookingHandler` class. It then asserts that two email invocations were made.

At the bottom of the interface, there's a "Logs" tab and a "Tests" tab. The "Tests" tab is currently selected and shows a table with one row: "Class: LaptopBookingHandlerTest, Method: testSendEmailNotification, Duration: 0:00, Result: Comp...". To the right of the table is a "Result" dialog box with the message "Completed" and an "OK" button. The "Logs" tab shows a list of log entries, all of which are successful (green checkmarks). On the far right, there's a system status bar showing the date and time (10-01-2025, 10:25), language (ENG), and location (IN).

Overall code coverage is 100%



This screenshot shows the same Salesforce Developer Console interface as the previous one, but the "Tests" tab is not selected. Instead, the "Logs" tab is selected, showing the same list of successful log entries. To the right of the logs, there's a "Overall Code Coverage" table. This table shows the following data:

Class	Percent	Lines
Overall	100%	
LaptopBooking	100%	2/2
LaptopBookingHandler	100%	10/10

The "Overall" row indicates a 100% code coverage across all classes. The `LaptopBooking` and `LaptopBookingHandler` classes also show 100% coverage.

## Explanation:

"**System.assertEquals**" is a method in Salesforce's Apex programming language, used for unit testing. It helps verify that the expected value matches the actual value produced by your code

apex

 Copy code

```
System.assertEquals(Object expected, Object actual, String message);
```

This test is successfully completed indicating all functionalities working good.

## 5.2 Validation

I observed that

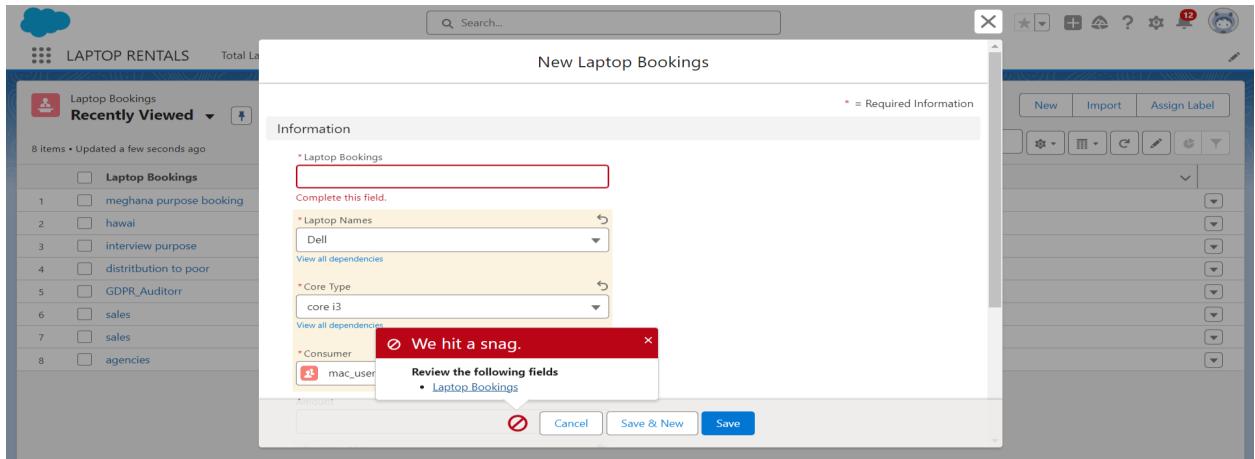
- In Consumer, Laptopbookings, Total Laptops objects, the required fields are set up and if when creating records, if any required field is not entered, then it throws an error like

javascript

 Copy code

```
Required fields are missing: <Field Name>
```

- It is verified when creating Laptop bookings object and missed a field, then the output is



- The validation rule "Phonenumberoremailblankrule" in consumer object ensures that at least one of the fields, phone\_number\_c or email\_c, is filled when creating or updating a record in the Consumer object.  
If both fields are left blank, the rule prevents the record from being saved and displays an error message.  
It displayed the message “phone number and email number should not be blank”

## 6. Key Scenarios Addressed by Salesforce in Implementation Of The Project

- **Customer Relationship Management (CRM):** Salesforce handles managing and analyzing customer interactions and data, improving customer service and engagement.
- **Sales Process Automation:** It automates various sales processes, like tracking leads, opportunities, and performance, helping sales teams close deals faster.
- **Marketing Campaigns and Engagement:** Salesforce enables personalized marketing campaigns and provides tools for tracking and analyzing their effectiveness.

- **Service Management:** It supports case management, service cloud, and provides tools for customer support teams to resolve issues efficiently.
- **Analytics and Reporting:** Salesforce helps generate real-time reports and dashboards, providing insights into business operations and customer behavior.
- **Collaboration and Integration:** The platform ensures streamlined workflows and improved productivity.

## 7. Conclusion

The **Laptop Rental CRM** has successfully created a digital platform to streamline the entire laptop rental process. By implementing real-time tracking, efficient coordination of rentals, and a user-friendly interface, the CRM has improved the efficiency and effectiveness of laptop rental operations. Key milestones include integrating online booking, automating payment processing, and developing tools for inventory management and customer relationship management. With this project, we have taken a step toward optimizing the rental process, enhancing customer satisfaction, and increasing the profitability of the laptop rental business.