



A CRM APPLICATION FOR LAPTOP RENTALS

NALLA PREETHI
prethinalla@gmail.com

Project Abstract

The CRM application for laptop rentals focuses on delivering rental items to customers. It harnesses the capabilities of customer relationship management (CRM) to enhance customer experiences, streamline store operations, and boost overall efficiency. Furthermore, effective CRM practices include communicating with potential customers through email, ensuring that we engage with those identified as prospects.

The key features of the CRM application for laptop rentals are given below:

1. User-Friendly Interface

Easy navigation for customers to browse and rent laptops.

2. Automated Operations

Streamlines inventory management and order processing to reduce manual errors.

3. Data Analytics

Collects and analyzes customer data to optimize marketing strategies and inventory.

4. Targeted Email Communication

Engages potential customers with personalized email campaigns to nurture leads.

5. Customer Support

Provides easy access to assistance and feedback channels for improved service.

6. Performance Tracking

Monitors rental trends and customer satisfaction to enhance service offerings.

TASK 1: CREATING DEVELOPER ACCOUNT	6
TASK 2: ACCOUNT ACTIVATION.....	7
OBJECT CREATION	9
TASK 3: CREATE TOTAL LAPTOPS OBJECT.....	10
TASK 4: CREATE CONSUMER OBJECT	11
TASK 5: CREATE LAPTOP BOOKINGS OBJECT	12
TASK 6: CREATE BILLING PROCESS OBJECT	13
TASK 7: CREATING A CUSTOM TAB	14
THE LIGHTNING APP	16
TASK 8: CREATE A LIGHTNING APP	16
FIELDS	19
TASK 9: CREATING THE FIELD IN CONSUMER OBJECT	19
TASK 10: CREATING THE FIELD IN LAPTOPS BOOKINGS OBJECT	22
TASK 11: TO CREATE A FIELDS & RELATIONSHIP TO AN LAPTOP BOOKING	24
TASK 12: TO CREATE A FIELDS & RELATIONSHIP TO AN LAPTOP BOOKING.....	24
TASK 13: CREATION OF FIELDS & RELATIONSHIP FOR BILLING PROCESS.....	30
TASK 14: CREATING THE FIELD IN TOTAL LAPTOPS OBJECT	33
VALIDATION RULE	34
TASK 15: CREATING THE VALIDATION RULE FOR PHONE NUMBER	35
PROFILES	37
TASK 16: OWNER PROFILE.....	37
TASK 17: AGENT PROFILE	38
ROLES AND HIERARCHY 40	
TASK 18: CREATING OWNER ROLE	40
TASK 19: CREATING AGENT ROLES.....	41
USERS.....	41
TASK 20: CREATE USER	42
FLOWS	44
TASK 21: CREATE A FLOW ON DELL LAPTOP.....	45
TASK 22: CREATING FLOW ON ACER LAPTOP.....	51
TASK 23: CREATING A FLOW ON HP LAPTOP.....	54
TASK 24: CREATING A FLOW ON MAC LAPTOP.....	57
APEX	61
TASK 25: CREATE AN APEX CLASS.....	63
TASK 26: CREATE AN APEX TRIGGER	64
REPORTS.....	66

TASK 27: CREATE REPORT	66
TASK 28: SHARING REPORT TO OWNER.....	69
DASHBOARDS.....	71
TASK 29: CREATE DASHBOARD FOLDER	71

Introduction

In the rapidly evolving world of technology, the demand for laptops has never been higher. Laptop rentals have emerged as a convenient and cost-effective solution for individuals and businesses seeking temporary access to powerful computing devices. To effectively manage this growing market, a robust Customer Relationship Management (CRM) application is essential. The CRM application for laptop rentals serves as a comprehensive platform that streamlines the rental process, enhances customer engagement, and optimizes business operations.

Revolutionizing the Laptop Rental Industry

The introduction of this CRM application revolutionizes the laptop rental industry by providing a centralized system that caters to the needs of both renters and rental providers. By integrating key CRM principles, such as customer data management, targeted communication, and performance tracking, the application ensures a seamless and efficient rental experience.

Simplifying the Rental Process for Customers

One of the primary objectives of the CRM application is to simplify the rental process for customers. Through an intuitive user interface, renters can easily:

- Browse available laptops
- Compare specifications
- Manage their bookings with ease

This convenience not only attracts new customers but also fosters loyalty

among existing ones, leading to increased rental volumes and revenue.

Empowering Rental Providers with Efficient Tools

Moreover, the CRM application empowers rental providers with powerful tools to manage their operations effectively. Automated inventory management, order processing, and payment tracking reduce manual errors and save valuable time. This efficiency allows rental providers to focus on delivering exceptional customer service and expanding their business.

Data-Driven Insights for Informed Decisions

By incorporating data-driven insights, the CRM application enables rental providers to make informed decisions based on customer preferences, rental trends, and market demands. This data-driven approach helps optimize inventory levels, tailor marketing strategies, and identify new revenue streams, ultimately driving growth and profitability in the laptop rental industry.

A New Standard for Excellence in Laptop Rentals

The introduction of the CRM application for laptop rentals marks a significant milestone in the evolution of the rental industry. By combining cutting-edge technology with customer-centric principles, this application sets a new standard for excellence in laptop rentals, paving the way for a more efficient, engaging, and profitable future.

Salesforce

Salesforce is a dynamic customer success platform that equips businesses with the tools necessary to excel in sales, service, marketing, analytics, and customer engagement. By providing a centralized solution for managing customer relationships, Salesforce enables organizations to navigate the entire customer journey effectively, from initial outreach to ongoing support. This capability allows for the delivery of personalized

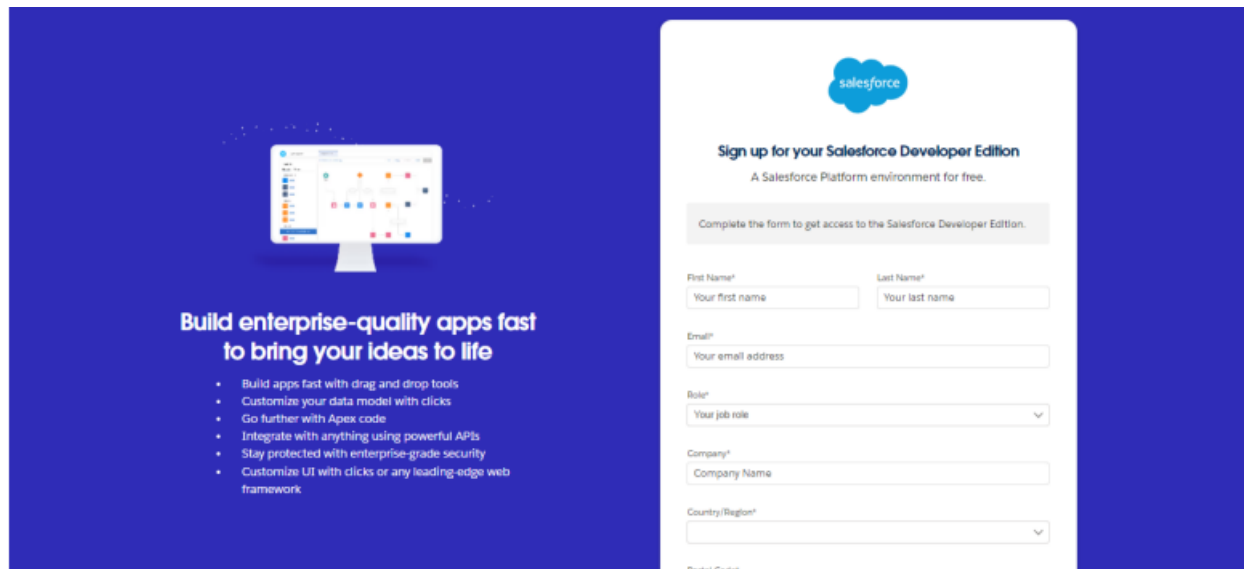
experiences that foster loyalty and satisfaction.

The platform is built on a secure cloud infrastructure, ensuring that your data is accessible from anywhere, while also offering the flexibility and scalability required to accommodate business growth and changing needs. Salesforce's extensive range of customizable tools allows organizations to tailor the platform to their unique requirements, enhancing operational efficiency and effectiveness. Additionally, Salesforce boasts a rich ecosystem of applications available through the Salesforce AppExchange, enabling businesses to integrate various functionalities and expand their capabilities seamlessly. This integration fosters collaboration among employees and partners, streamlining workflows and improving communication. By harnessing the power of data analytics, Salesforce provides valuable insights that inform strategic decision-making, helping businesses identify trends, optimize processes, and enhance customer interactions. Ultimately, Salesforce empowers organizations to thrive in a competitive landscape, build lasting relationships with customers, and achieve their strategic objectives, positioning them for long-term success in an ever-evolving market.

Task 1: Creating Developer Account

Creating a developer org in salesforce.

1. Go to <https://developer.salesforce.com/signup>
2. On the sign up form, enter the following details :
 - First name & Last name
 - Email
 - Role : Developer
 - Company : College Name
 - County : India
 - Postal Code : pin code
 - Username : should be a combination of your name and company

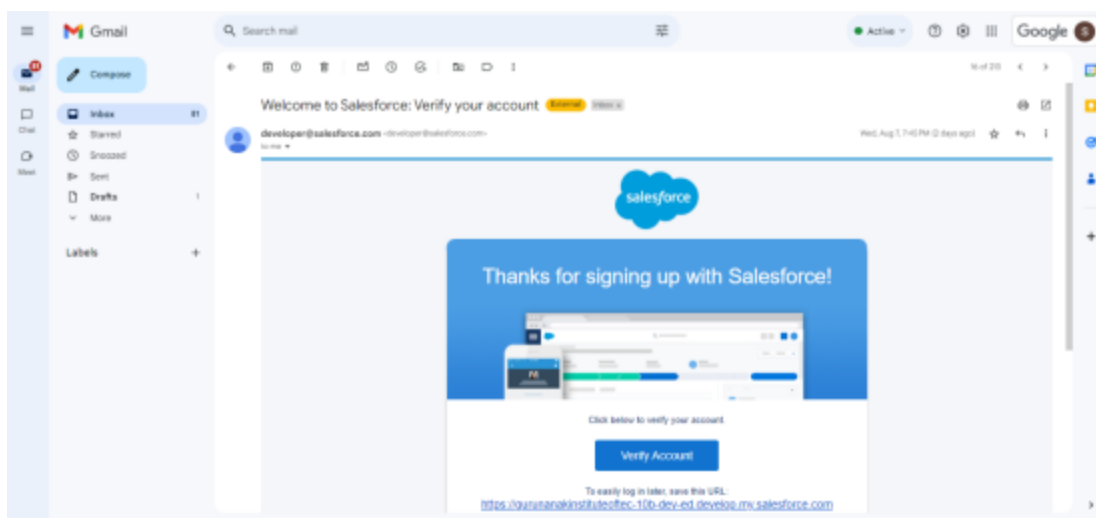


The image shows a sign-up form for Salesforce Developer Edition. On the left, there is a blue background with a white monitor icon displaying a Salesforce interface. Below the icon, the text reads: "Build enterprise-quality apps fast to bring your ideas to life". A bulleted list follows: "• Build apps fast with drag and drop tools", "• Customize your data model with clicks", "• Go further with Apex code", "• Integrate with anything using powerful APIs", "• Stay protected with enterprise-grade security", and "• Customize UI with clicks or any leading-edge web framework". On the right, a white form is set against a blue background. It features the Salesforce logo at the top, followed by the heading "Sign up for your Salesforce Developer Edition" and the subtext "A Salesforce Platform environment for free." Below this is a grey box with the instruction "Complete the form to get access to the Salesforce Developer Edition." The form fields include: "First Name*" (with placeholder "Your first name"), "Last Name*" (with placeholder "Your last name"), "Email*" (with placeholder "Your email address"), "Role*" (a dropdown menu with "Your job role" selected), "Company*" (with placeholder "Company Name"), "Country/Region*" (a dropdown menu), and "Referral Code*" (a text field).

This need not be an actual email id, you can give anything in the format :
 username@organization.com
 Click on sign me up after filling these.

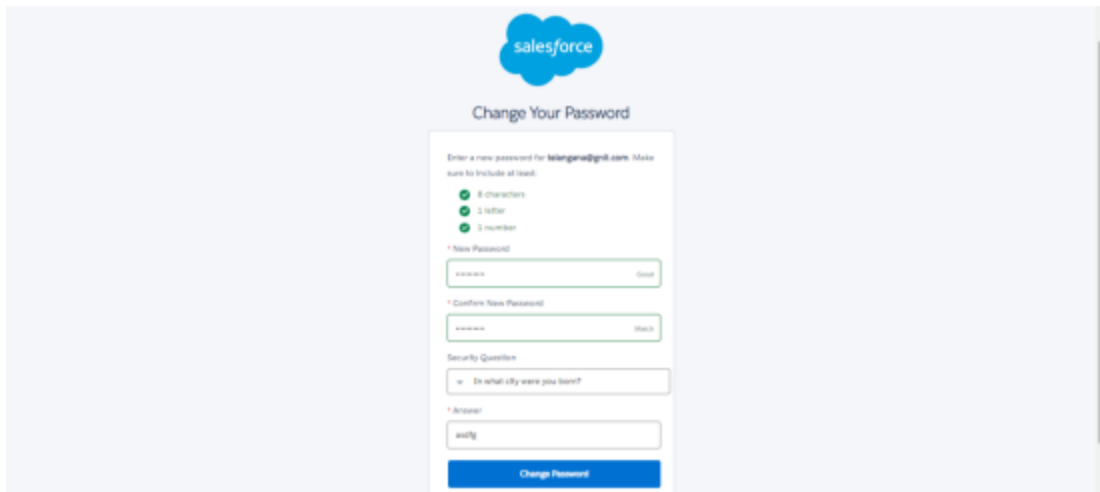
Task 2: Account Activation

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



2. Click on Verify Account.

3. Give a password and answer a security question and click on change password. Make sure to remember the password. In case you forget the password, you can simply click on 'Forgot password' option to create a new password. An email with the subject "Finish resetting your Salesforce password" will be sent for this process.



The image shows the 'Change Your Password' page in Salesforce. At the top is the Salesforce logo. Below it, the title 'Change Your Password' is centered. The form is titled 'Enter a new password for [belingrad@nit.com](#). Make sure to include all of these:

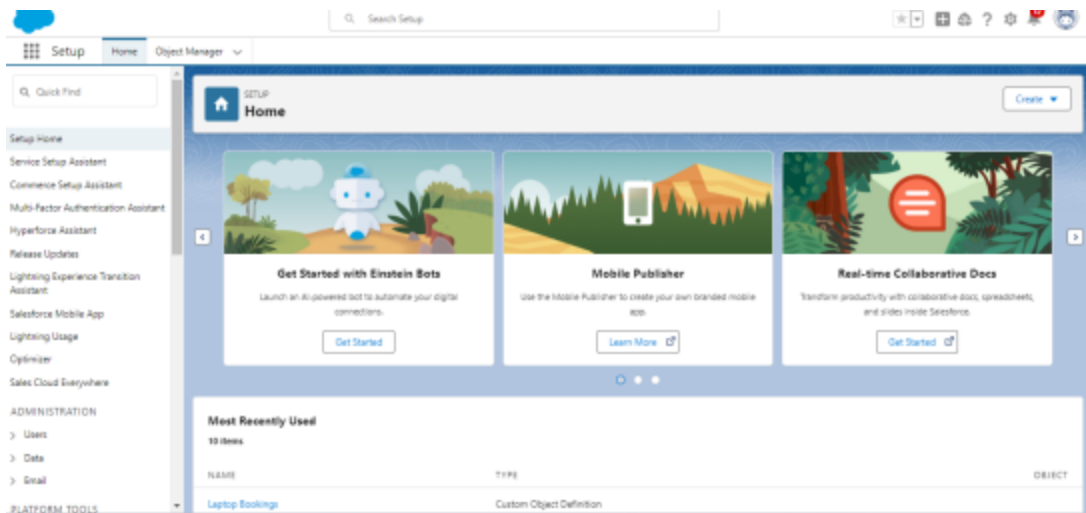
- 8 characters
- 5 letter
- 5 number

Below these requirements are three input fields:

- New Password:** A text input field with a 'Show' button to its right.
- Confirm New Password:** A text input field with a 'Show' button to its right.
- Security Question:** A dropdown menu with the selected option 'In what city were you born?'.
- Answer:** A text input field with the value 'indig'.

At the bottom of the form is a blue button labeled 'Change Password'.

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



Object Creation

What Is an Object?

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

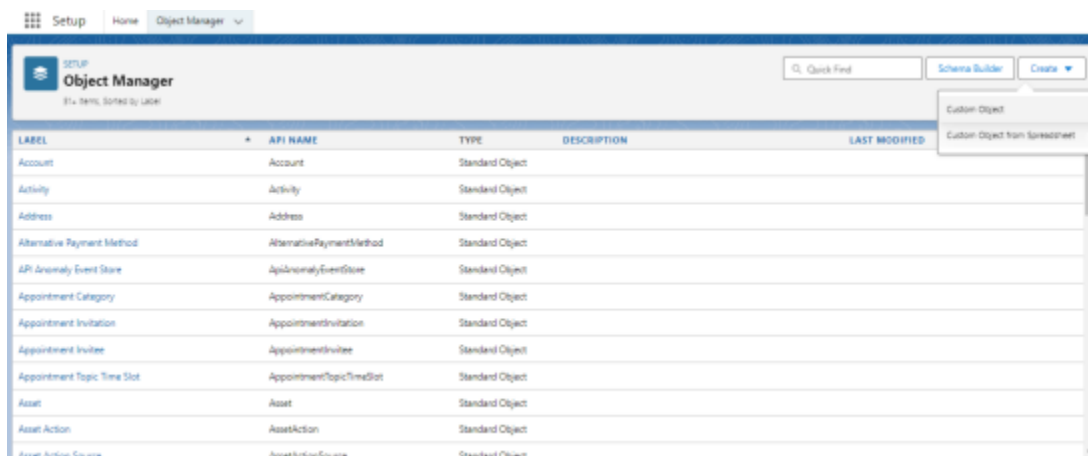
Salesforce objects are of two types:

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

2. Custom Objects: Custom objects are those objects that are created by users. They supply information that is unique and essential to their organization. They are the heart of any application and provide a structure for sharing data.

To 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 Home Object Manager

NEW New Custom Object

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles. [Get more info](#) [Get more info regarding profiles](#)

Custom Object Definition Edit Save Save & Next Cancel

Custom Object Information

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

Label Example: Account

Plural Label Example: Accounts

Starts with lower round ☐

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

Object Name Example: Account

Description

Context sensitive help setting ☒ Open the standard Salesforce.com help & Training window ☐ Open a window using a Visualforce page

Context Name

Enter Record Name Label and Format

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

Record Name Example: AccountName

Setup Home Object Manager

NEW New Custom Object

Object Type Warning: If you plan to insert a high volume of records into this object, via the API for example, use the Text data type.

Optional Features

☒ Allow Reports

☐ Allow Archives

☒ Track Field History

☐ Allow in Chatter Stream

☐ Enable Learning

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](#)

☒ Allow Search

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

☐ Add Name and Id

☐ Launch New Custom Tab Instant after saving the custom object

Save Save & Next Cancel

4. Click on Save.

Task 3: Create Total Laptops Object

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

1) Enter the label name>> Total Laptops

2) Plural label name>> Total Laptops

3) Enter Record Name Label and Format

Record Name >>Total Laptops

Data Type >> Text

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

3. Allow search >> Save.

Task 4: Create Consumer Object

1. 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.

Setup
New Custom Object

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles. [Go to Profiles](#) [Go to Permission Sets](#)

Custom Object Definition Edit Save Save & New Cancel

Custom Object Information Required Information

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

Label Example: Account

Plural Label Example: Accounts

Start with vowel sound ☐

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

Object Name Example: Account

Description

Custom derivative help setting ☒ Open the standard Salesforce.com Help & Training window ☐ Open a window using a Visualforce page

Custom Name

Enter Record Name Label and Format

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

Record Name Example: Account Name

Task 5: Create Laptop Bookings Object

1. 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

Setup
New Custom Object

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles. [Go to Profiles](#) [Go to Permission Sets](#)

Custom Object Definition Edit Save Save & New Cancel

Custom Object Information Required Information

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

Label Example: Account

Plural Label Example: Accounts

Start with vowel sound ☐

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

Object Name Example: Account

Description

Custom derivative help setting ☒ Open the standard Salesforce.com Help & Training window ☐ Open a window using a Visualforce page

Custom Name

Enter Record Name Label and Format

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

Record Name Example: Account Name

Task 6: Create Billing Process Object

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

Custom Object.

1) Enter the label name >> Billing Process

2) Plural label name >> Billing Process

3) Enter Record Name Label and Format

Record Name >> Billing ProcessName

Data Type >> Name

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

3. Allow search >> Save.

The screenshot shows the 'New Custom Object' setup page in Salesforce. The page title is 'New Custom Object' with a 'Setup' icon. Below the title, there is a yellow banner with a message: 'Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles. [Go to setup](#) [Visit about this message page](#)'. The main section is 'Custom Object Definition Edit' with buttons for 'Save', 'Save & New', and 'Cancel'. Below this is the 'Custom Object Information' section, which includes fields for 'Label' (Billing Process), 'Plural Label' (Billing Processes), 'Starts with lower bound' (unchecked), 'Object Name' (Billing Process), and 'Description'. There is also a section for 'Content: Interactive map: billing' with radio buttons for 'Open the standard Salesforce.com map & Training window' (selected) and 'Open a window using a Visualforce page'. At the bottom, there is a section for 'Enter Record Name Label and Format' with a 'Record Name' field (Billing ProcessName) and an 'Example' (Account Name).

Tabs

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

Types of Tabs:

1. Custom Tabs

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

2. Web Tabs:

Web Tabs are custom tabs that display web content or applications embedded in the salesforce.com window. Web tabs make it easier for your users to quickly access content and applications they frequently use without leaving the salesforce.com application.

1. Visualforce Tabs

Visualforce Tabs are custom tabs that display a Visualforce page. Visualforce tabs look and behave like standard salesforce.com tabs such as accounts, contacts, and opportunities.

2. Lightning Component Tabs

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

3. Lightning Page Tabs

Lightning Page Tabs let you add Lightning Pages to the mobile app navigation menu. Lightning Page tabs don't work like other custom tabs. Once created, they don't show up on the All Tabs page when you click the Plus icon that appears to the right of your current tabs. Lightning Page tabs also don't show up in the Available Tabs list when you customize the tabs for your apps.

Task 7: Creating A Custom Tab

To create a Tab:

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

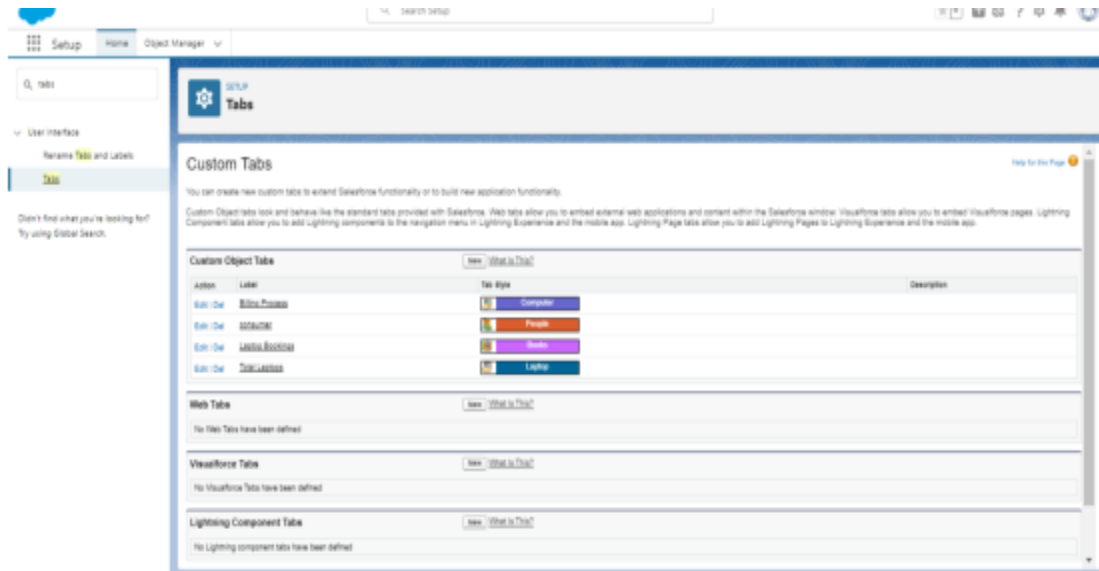
The screenshot shows the 'New Custom Object Tab' configuration page. The left sidebar has a search bar with 'Tabs' and a list of options including 'User Interface', 'Rename Tabs and Labels', and 'Tabs'. The main content area is titled 'New Custom Object Tab' and indicates 'Step 1 of 3: Enter the Details'. The instructions state: 'Choose the custom object for this new custom tab. FID in other details.' The configuration options include: 'Select an existing custom object or create a new custom object type' (set to 'Object'), 'Tab type' (set to 'Default'), an optional 'Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab' (set to 'None'), and a text area for 'Enter a short description' and 'Description'. 'Next' and 'Cancel' buttons are at the bottom right.

2. Select Object(Total Laptops) >> Select the tab style >> Next (Add to profiles page) keep it as default >> Next (Add to Custom App) uncheck the include tab .
3. Make sure that the Append tab to users' existing personal customizations is checked.
4. Click save.

Activity 2: Creating Remaining Tabs

Now create the Tabs for the remaining Objects, they are “consumer,Laptop Booking,Billing process”. Follow the same steps as mentioned above.

The final output will be shown as following:



The Lightning App

In the context of Lightning Experience, a Lightning app is a powerful tool that brings together a collection of related items, such as objects, tabs, and other components, to serve a specific function or purpose. These apps are designed to provide users with a convenient and streamlined way to access the necessary tools and information they need to perform their tasks effectively. By bundling these items into a single Lightning app, users can easily navigate between them using the navigation bar, improving their overall productivity and efficiency. One of the key advantages of Lightning apps is the ability to customize them with a unique color scheme and logo, allowing organizations to reinforce their brand identity and create a cohesive user experience. Additionally, Lightning apps can include a utility bar, which provides users with quick access to frequently used tools and features, further enhancing their workflow and productivity. Another notable feature of Lightning apps is the inclusion of Lightning page tabs, which enable users to switch between different pages or views within the app seamlessly. This functionality allows members of your organization to work more efficiently by easily transitioning between various aspects of their

work, such as sales, service, or marketing, without the need to navigate through multiple applications or interfaces. By leveraging the power of Lightning apps, organizations can empower their employees to work more effectively and efficiently, ultimately driving better business outcomes. The ability to customize and tailor these apps to specific organizational needs, while providing a consistent and intuitive user experience, makes Lightning apps a valuable asset in the Lightning Experience ecosystem.

Task 8: Create A Lightning App

To create a lightning app page:

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

Setup	Home	Object Manager
-------	------	----------------

Q, app manager

App

Don't find what you're looking for? Try using Global Search.

Lightning Experience App Manager

22 items • Sorted by App Name • Filtered by All experiences • Tablet Type

App Name	Developer Name	Description	Last Modified Date	App Type	Mobile L.
1 All Tabs	AppFaster		07/06/2024 7:08 pm	Classic	
2 Analytics Studio	Insights	Built CRM Analytics dashboards and apps	07/06/2024 7:08 pm	Classic	✓
3 App Launcher	AppLauncher	App Launcher tabs	07/06/2024 7:08 pm	Classic	✓
4 Automation	FlowApp	Automate business processes and repetitive tasks.	07/06/2024 7:08 pm	Lightning	✓
5 Bulk Solutions	LightningBolt	Discover and manage business solutions designed for your industry.	07/06/2024 7:08 pm	Lightning	✓
6 Community	Community	Salesforce CRM Communities	07/06/2024 7:08 pm	Classic	✓
7 Content	Content	Salesforce CRM Content	07/06/2024 7:08 pm	Classic	✓
8 Data Manager	DataManager	Use Data Manager to view trends, monitor usage, and manage recipes.	07/06/2024 7:08 pm	Lightning	✓
9 Digital Experiences	SalesforceCUI	Manage content and media for all of your sites.	07/06/2024 7:08 pm	Lightning	✓
10 LAPTOP RENTALS	Utrina		08/06/2024 1:08 pm	Lightning	✓
11 Lightning Usage App	LightningInstrumentation	View Adoption and Usage Metrics for Lightning Experience	07/06/2024 7:08 pm	Lightning	✓
12 Marketing CRM Classic	Marketing	Track sales and marketing efforts with CRM experts.	07/06/2024 7:08 pm	Classic	✓
13 Platform	Platform	The Fundamental Lightning Platform	07/06/2024 7:08 pm	Classic	✓
14 Queue Management	QueueManagement	Create and manage queues for your business.	07/06/2024 7:08 pm	Lightning	✓
15 Sales	Sales	The world's most popular sales force automation (SFA) solution	07/06/2024 7:08 pm	Classic	✓
16 Sales	UtrinaSales	Streamline your sales process with accounts, leads, opportunities, and more.	07/06/2024 7:08 pm	Lightning	✓

- 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.

New Lightning App

App Details & Branding

Give your Lightning app a name and description, upload an image and choose the highlight color for its navigation bar.

App Details

*App Name

LAPTOP RENTALS

*Developer Name

LAPTOP RENTALS

Description

Enter a description...

App Branding

Image

Primary Color Hex

Value

#0070C2

Drop Theme Options

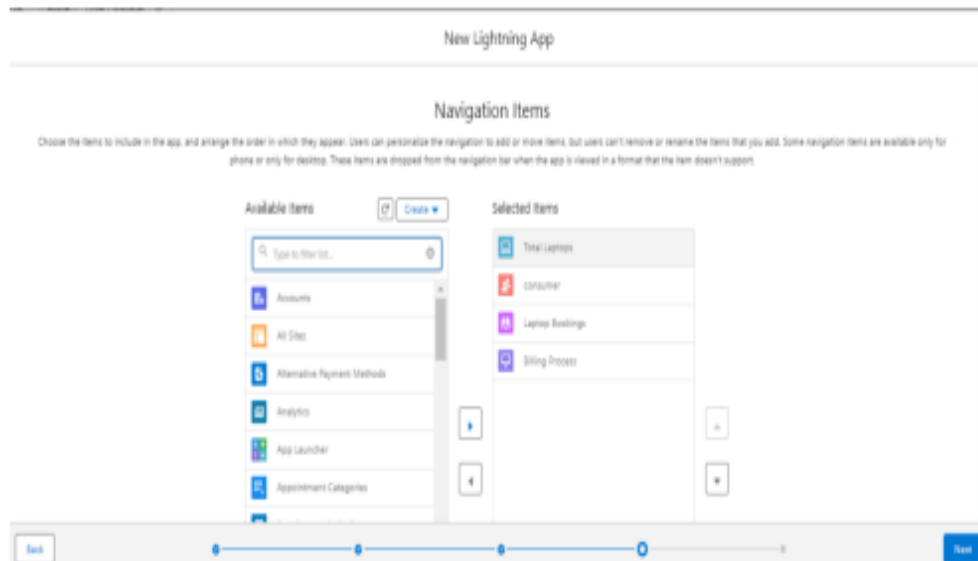
☐ Use the app's image and color instead of the org's custom theme

0

100%

Next

- Upload a photo that is related to your app.
- 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.

5. To Add User Profiles:



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

Fields

In the context of Salesforce, fields serve as the fundamental building blocks for storing data within objects, which are akin to tables in a relational database. These fields can hold a wide range of valuable information tailored to the specific requirements of each object, making the processes of searching, deleting, and editing records more efficient and user-friendly. There are two main types of fields in Salesforce:

Standard Fields

Standard fields are pre-defined by Salesforce and serve specific purposes. It's important to note that while users can freely delete non-required standard fields, they cannot delete required standard fields.

Some standard fields that are commonly found across Salesforce applications include:

- Created By: Identifies the user who created the record
- Owner: Specifies the user or queue responsible for the record
- Last Modified: Indicates the user who last modified the record
- Fields created during object creation

Custom Fields

Custom fields offer a high degree of flexibility, allowing users to modify them according to their unique requirements. Organizations can create custom fields as needed, and they are not always required to be included in records, unlike standard fields. The decision to include or exclude custom fields is at the discretion of the user, providing a level of customization that caters to specific business needs. By leveraging both standard and custom fields, Salesforce users can effectively manage and organize

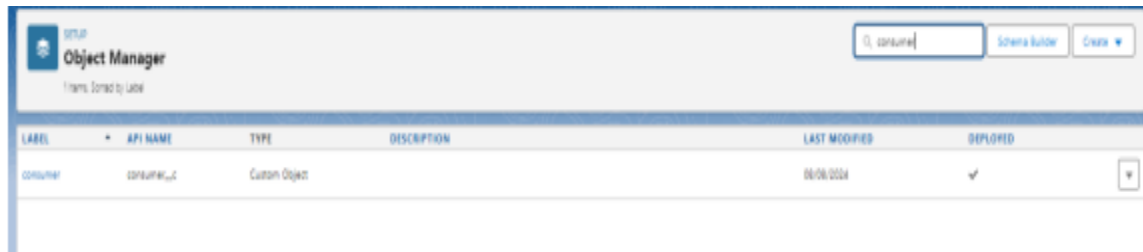
their data, streamlining processes and enhancing overall efficiency within their Salesforce applications.

Task 9: Creating The Field In Consumer Object

To create fields in an object:

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

on the object.



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

3. Select Data Type as a "Phone"

4. Click on next

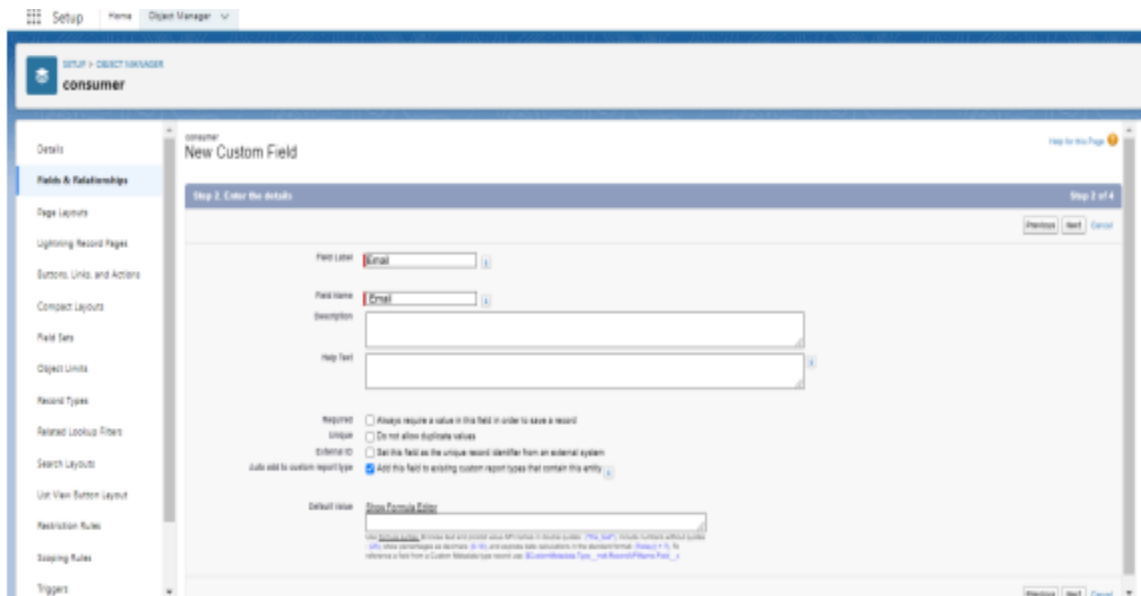
- Fill the Above as following:
- Field Label: Phone number

- Field Name : gets auto generated
- Click the required option checkbox.
- Click on Next >> Next >> Save and new.

The screenshot shows the 'New Custom Field' setup page for the 'consumer' object. The page is titled 'Step 2: Enter the details'. The 'Field Label' is 'Phone Number' and the 'Field Name' is 'Phone_number'. The 'Description' and 'Help Text' fields are empty. The 'Required' checkbox is checked, and the 'Always require a value in this field in order to save a record' option is selected. The 'Default Value' field is empty. The 'Previous', 'Next', and 'Cancel' buttons are visible at the bottom right.

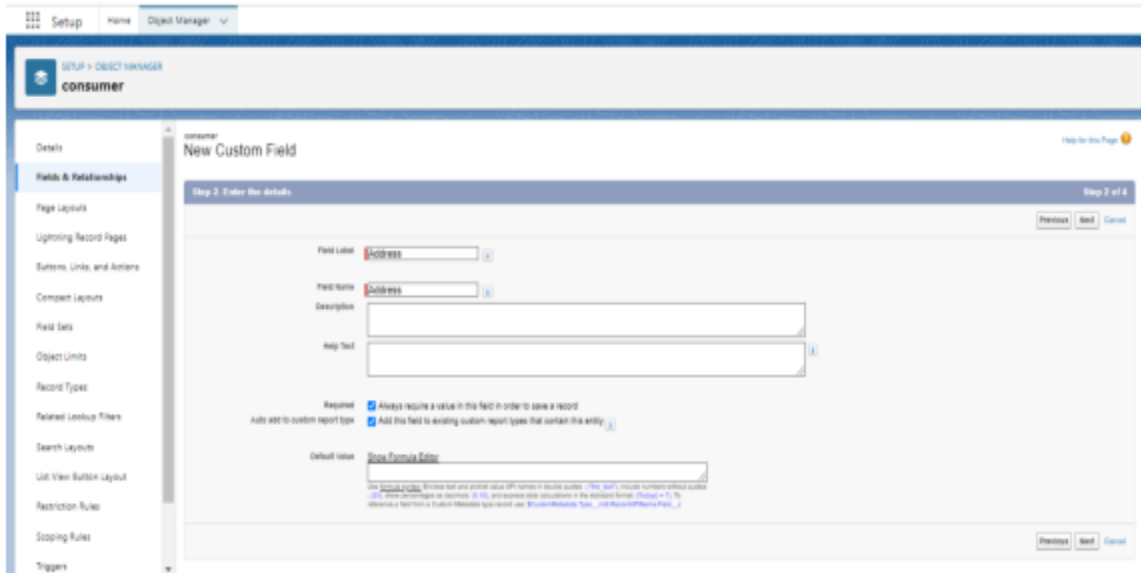
To create another fields in an object:

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



To create another fields in an object:

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



To create another fields in an object:

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

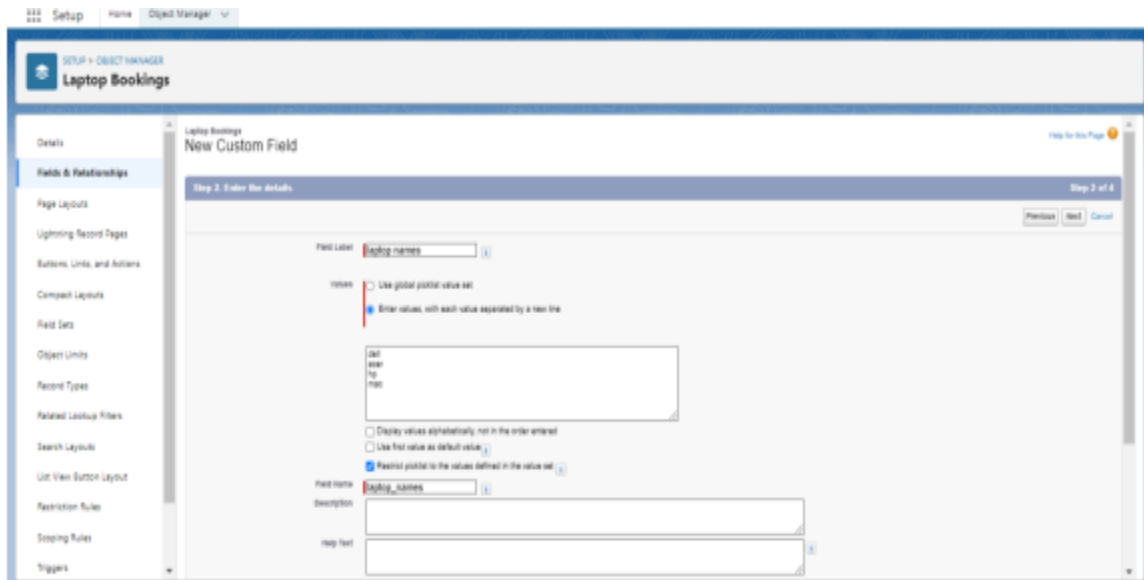
- Field Name :It's gets auto generated
- Click on Next >> Next >> Save and new.

The screenshot shows the Salesforce Setup - Object Manager interface for the 'consumer' object. The 'Fields & Relationships' tab is selected, and the 'Step 2: Enter the details' window is open. The 'Field Label' is 'consumer Status'. The 'Values' section shows a list of values: 'Student', 'Employee', and 'Other'. The 'Field Name' is 'Address'. The 'Description' and 'Help Text' fields are empty. The 'Required' checkbox is checked, and the 'Add this field to existing custom report types that contain this entity' checkbox is also checked.

Task 10: Creating The Field In Laptops Bookings Object

To create fields in an object:

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

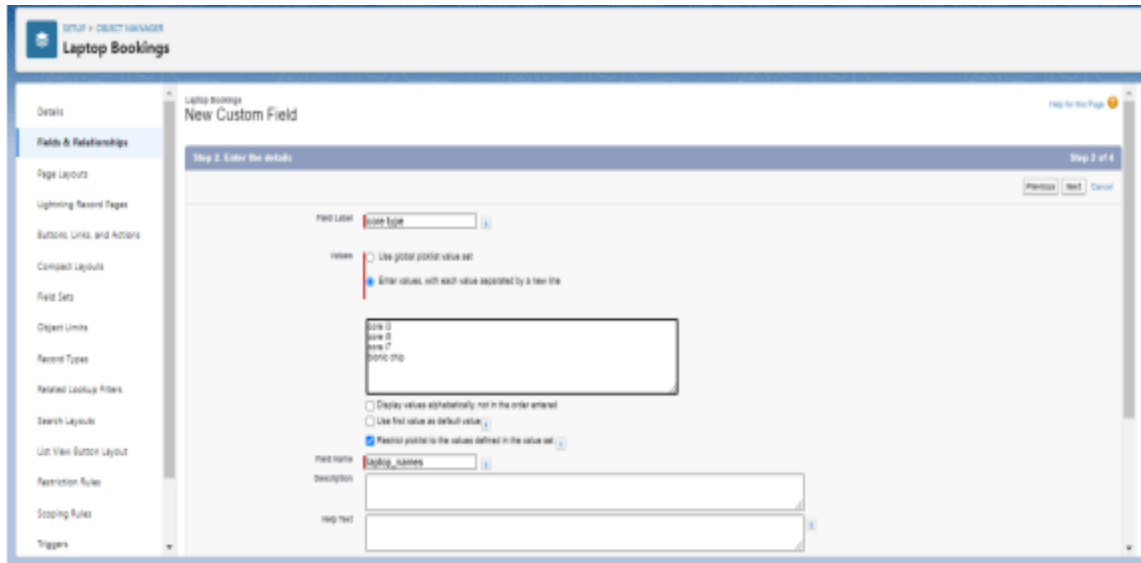


5. Select required
6. Click on Next >> Next >> Save and new

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

To create fields & relationship to an object:

1. Go to setup >> click on Object Manager >> type object name(Laptop Booking) in the search bar
>> click on the object.
2. Now click on “Fields & Relationships” >> New
3. Select Data Type as a “Picklist”
4. Picklist values are:-1.core i3 2. Core i5 3. Core i7 4.Bionic chip.



Field Dependency:

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

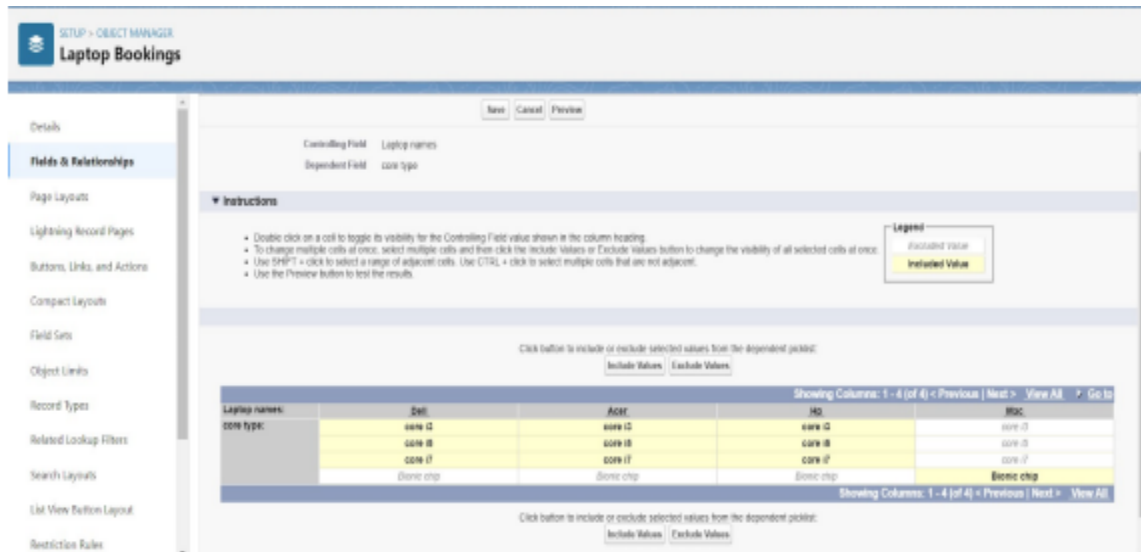
Need to use Field Dependency: By using the field dependency we can get the different Values by selecting the different Picklist.

Task 11: To Create A Fields & Relationship To An Laptop Booking Object

To create fields & relationship to an object:

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

for hp i3,i4,i5 and
also for mac bionic chip include the values for it



Task 12: To Create A Fields & Relationship To An Laptop Booking Object

1. 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 "Lookup 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: Name
 - Field Name :It's gets auto generated
7. Click on Next >> Next >> Save and new

The screenshot shows the 'New Relationship' form in the 'Laptop Bookings' setup. The form is titled 'Step 3. Enter the label and name for the lookup field'. It includes fields for 'Field Label' (set to 'Name'), 'Field Name' (set to 'Name'), 'Description', and 'Help Text'. Below these are options for 'Create Relationship Name' (set to 'Laptop_Bookings'), 'Required' (unchecked), 'Clear the value of this field' (checked), 'Don't allow deletion of the lookup record that's part of a lookup relationship' (unchecked), and 'Add this field to existing custom report types that contain this entity' (checked). At the bottom, there is a 'Lookup Filter' section with a link to 'Learn More'.

2. To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name (Laptop Booking) in the search bar >> click on the object.
3. Now click on "Fields & Relationships" >> New
4. Select Data Type as a "Currency"
5. Click on Next

Fill the Above as following:

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

SETUP > OBJECT MANAGER
Laptop Bookings

Details

Fields & Relationships

Page Layouts

Lightning Report Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

Triggers

Laptop Bookings
New Custom Field

Step 2: Enter the details

Field Label: Amount

Length: 15

Decimal Places: 0

Field Name: Amount

Description:

Help Text:

Required: ☐ Always require a value in this field in order to save a record
☒ Add this field to existing custom report types that contain this entity

Default value: Show Formula Editor

Previous Next Cancel

3. To Create a 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 “Lookup Relationship”
4. Click on Next

SETUP > OBJECT MANAGER
Laptop Bookings

Details

Fields & Relationships

Page Layouts

Lightning Report Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Record Types

Related Lookup Filters

Search Layouts

List View Button Layout

Restriction Rules

Scoping Rules

Triggers

Laptop Bookings
New Relationship

Step 3: Choose the related object

Select the other object to which this object is related.

Related To: New Lookup

Previous Next Cancel

5. Click on the Related to drop down and Select the “Total Laptops”

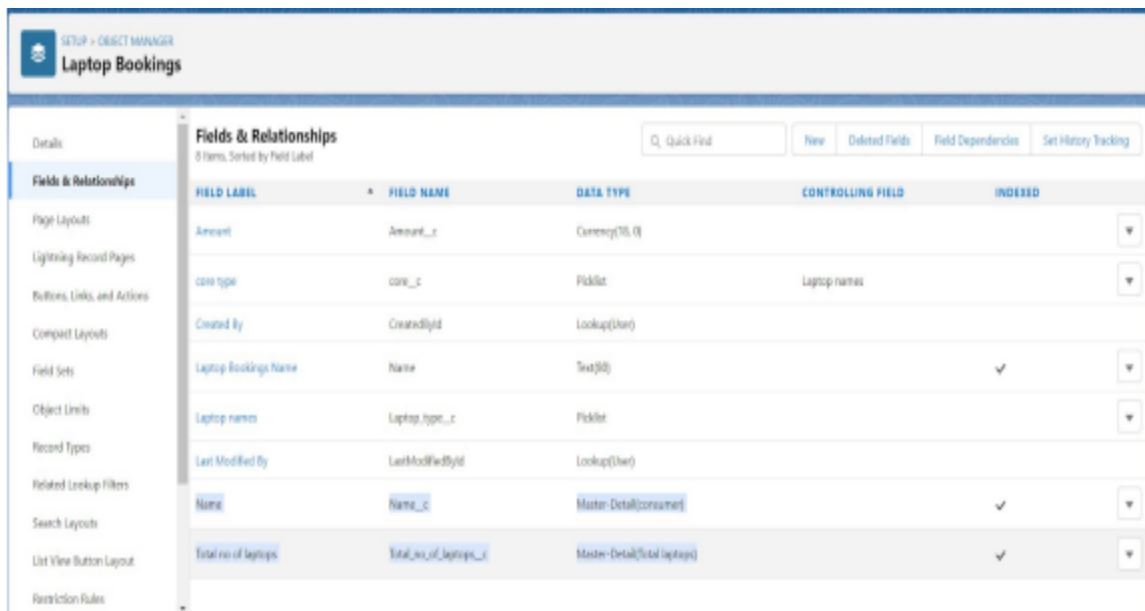
object and click on Next

Fill the Above as following:

- Change the Field Label: Total No Of Laptops
- Field Name :It's gets auto generated
- Click on Next >> Next >> Save and new.

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

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



FIELD LABEL	* FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Amount	Amount__c	Currency(15, 0)		
case type	case__c	Picklist	Laptop names	
Created By	CreatedById	Lookup(User)		
Laptop Bookings Name	Name	Text(50)		✓
Laptop names	Laptop_type__c	Picklist		
Last Modified By	LastModifiedById	Lookup(User)		
Name	Name__c	Master-Detail(consumer)		✓
Total no of laptops	Total_no_of_laptops__c	Master-Detail(total laptops)		✓

6. To create fields in an object:

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

2. Now click on “Fields & Relationships” >> New

3. Select Data type as a “Formula” and Click on Next

4. Fill the Above as following:

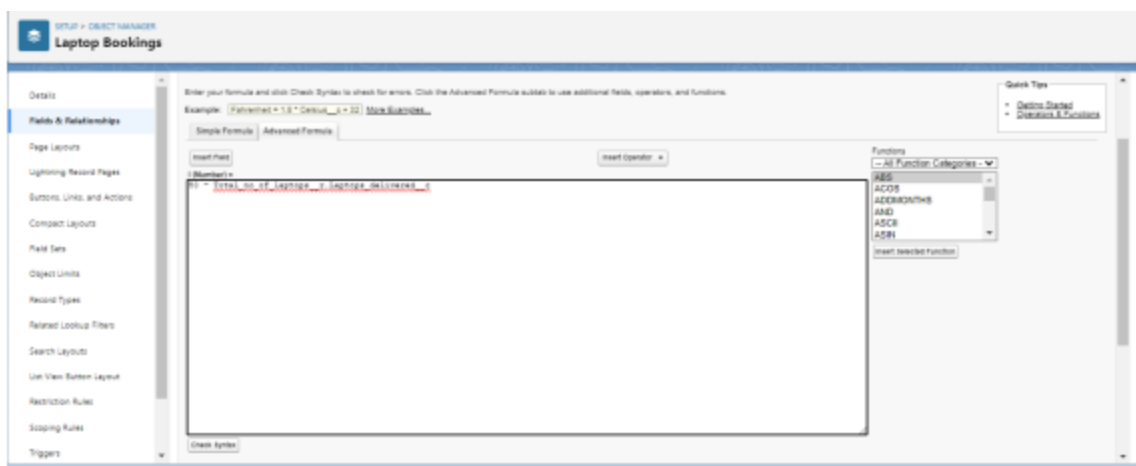
- Field Label: Laptops Available
- Field Name : It's gets auto generated
- Select the Formula Return Type as “Number”
- Select the Decimal places as “0” and Click on Next
- Click on the Advanced Formula and Enter the value in formula box “ 50 - ” and Click

on insert field than you will find a pop window under the Laptop Booking select the

Total No Of Laptops in the second Column and select the Laptops delivered in the

third column and click on insert

- “ 50 - Total_no_of_laptops__r.Laptops_delivered__c ” and Check Syntax



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

7. To create fields in an object:

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

The screenshot shows the Salesforce 'New Custom Field' setup page for the 'Laptop Bookings' object. The page is titled 'Step 2: Enter the details' and shows the configuration for a new picklist field. The 'Field Label' is set to 'Picklist'. Under the 'Values' section, there are options to 'Use global picklist value set' (unchecked) or 'Enter values, with each value separated by a new line' (checked). A text area for entering values is visible. Below this, there are checkboxes for 'Display values alphabetically, not in the order entered' (unchecked) and 'Use first value as default value.' (unchecked). The 'Restrict picklist to the values defined in the value set' checkbox is checked. At the bottom, there are input fields for 'Field Name', 'Description', and 'Help Text'.

Task 13: Creation Of Fields & Relationship For Billing Process Object

1. To create fields & relationship to an object:
1. Go to setup >> click on Object Manager >> type object name(Billing

Process) in the

search bar >> click on the object.

2. Now click on “Fields & Relationships” >> New

3. Select Data Type as a “Master-detail Relationship”

4. Click on Next

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

Setup > OBJECT MANAGER
Billing Process

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Triggers

Setup Process
New Custom Field

Step 1. Choose the field type

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

Data Type

☐ None Selected Select one of the data types below.

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

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

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

☐ Lookup Relationship Create a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other object is the source of the values in the list.

☒ Master Detail Relationship Create a special type of parent-child relationship between this object (the child, or "lead") and another object (the parent, or "leader"), where:

- The relationship field is required on all detail records.
- The ownership and sharing of a detail record are determined by the master record.
- When a user deletes the master record, all detail records are deleted.
- You can create roll-up summary fields on the master record to summarize the detail records.

The relationship field allows users to click on a lookup icon to select a value from a popup list. The master object is the source of the values in the list.

☐ External Lookup Relationship Create a relationship that links this object to an external object whose data is stored outside the Salesforce org.

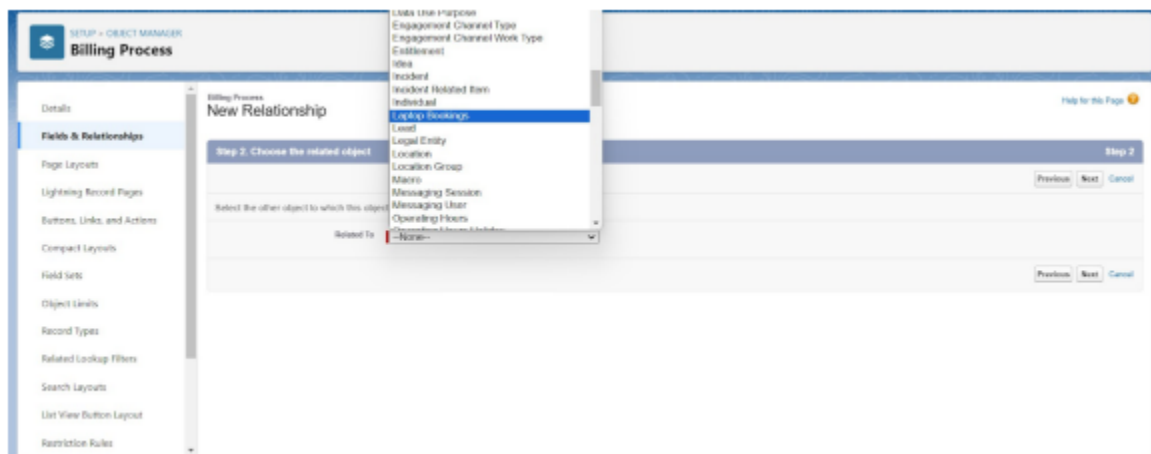
☐ Checkbox Allow users to select a True (checked) or False (unchecked) value.

6. Fill the Above as following:

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

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

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



3. Creation of another fields for the billing process object:

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

2. Now click on “Fields & Relationships” >> New

3. Select Data Type as a “Picklist”

4. Fill the Above as following:

- Field Label: Payment Mode
- Value >> Select enter values with each value separated by a new line

1. Cash
2. Check
3. Credit card
4. Debit card
5. UPI
6. Phonepe
7. Gpay

The screenshot displays the Salesforce Object Manager interface for the 'Billing Process' object. The 'Fields & Relationships' tab is selected, and the 'New Custom Field' wizard is open. The 'Field Label' is set to 'Payment Mode'. The 'Value' section is configured as a picklist with the following values: Cash, Check, Credit card, Debit card, UPI, Phonepe, and Gpay. The 'Field Name' is 'Payment_Mode'. The 'Help Text' field is empty.

Cross Object Formula Field:

In Salesforce, a cross-object formula field allows you to create a formula that references

fields from related objects. It enables you to perform calculations or display data from

related records without the need for custom code or complex workflows.

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

If we want to get the Particular field from another object in that case we will use the Cross

object Formula field. For that First we need to create the relationship b/w two objects and

relate the field with formula data type.

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

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

search bar >> click on the object.

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

3. Select Data Type as a "Formula"

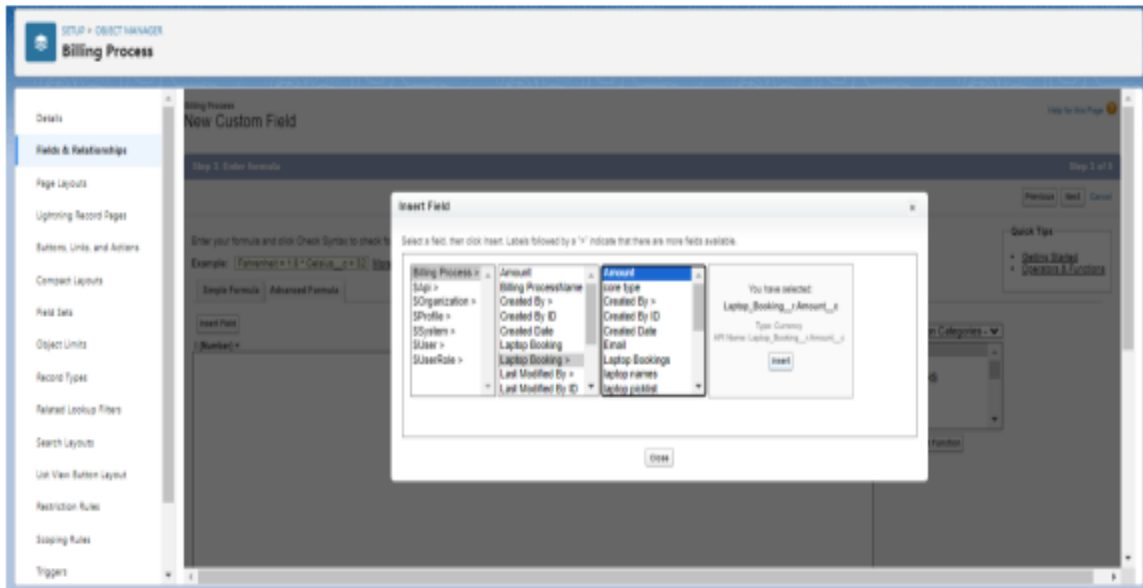
4. Click on Next

5. Enter the Field label: Amount, the Field name gets auto generated and click on

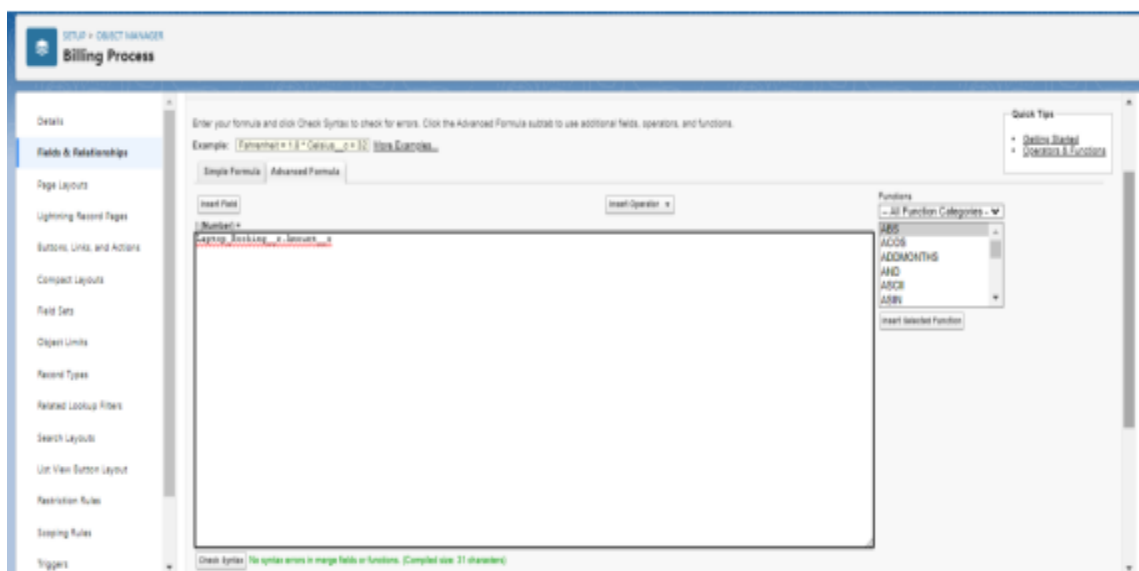
Next.(Formula return type Number).

6. In the Advanced Formula Click on the Insert field in the popup Screen Select the

Billing Process and in the second drop down select the Laptop Booking and in the



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



Task 14: Creating The Field In Total Laptops Object

1. To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Total Laptops) in search

bar >> click on the object.

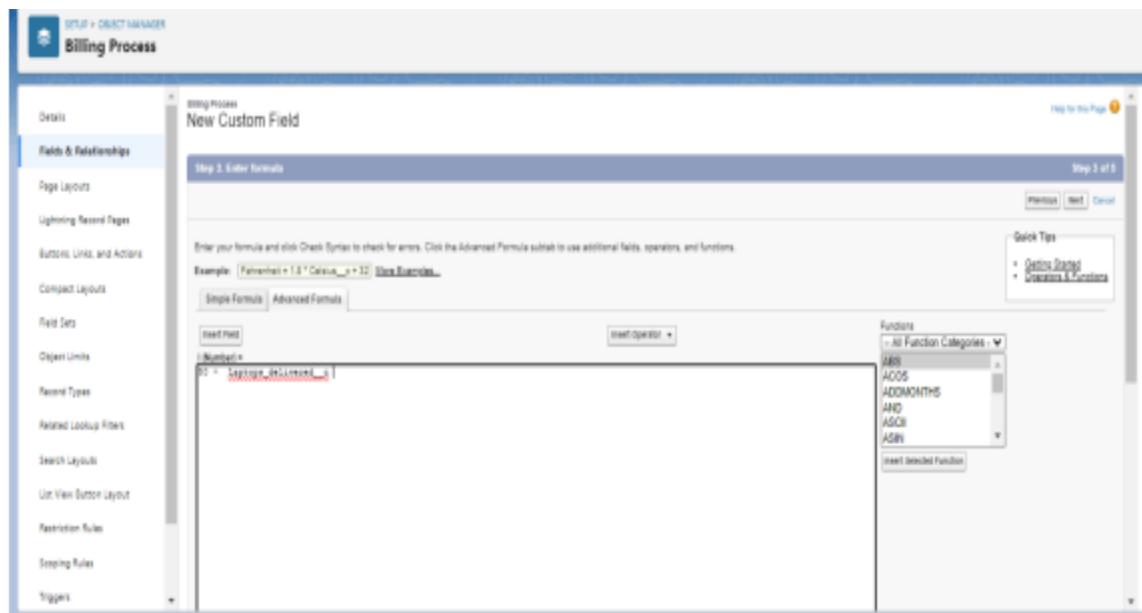
2. Now click on “Fields & Relationships” >> New

3. Select Data type as a “Formula” and Click on Next

4. Fill the Above as following:

5. Field Label: Laptops Available

6. Field Name : It's gets auto generated



Validation Rule

Validation rules in Salesforce are mechanisms that ensure data entered into records meets predefined criteria. When a user attempts to save a record, the validation rule evaluates the data and triggers an error message if it does not satisfy the specified conditions, preventing the record from being saved until the issues are resolved.

The main purpose of validation rules is to maintain high data quality by enforcing standards that align with business requirements. They consist of a formula or expression that

assesses the data entered in

one or more fields, returning a Boolean value of "True" if the data is invalid and "False" if it is valid. Each validation rule includes an error message that provides users with clear guidance on what went

wrong, along with the option to specify where the error message appears.

By implementing validation rules, organizations can improve data quality, provide user guidance, ensure consistency across records,

and reduce errors, ultimately optimizing their data management practices.

In summary, validation rules are a powerful feature in Salesforce that play a critical role in maintaining data integrity and quality. By applying specific criteria to the data entered by users, these rules help

organizations ensure that their information is accurate, consistent, and

reliable. The combination of formulas, error messages, and user guidance makes validation rules an invaluable asset for any

organization looking to optimize its data management practices.

Task 15: Creating The Validation Rule For Phone Number Field In Consumer Object

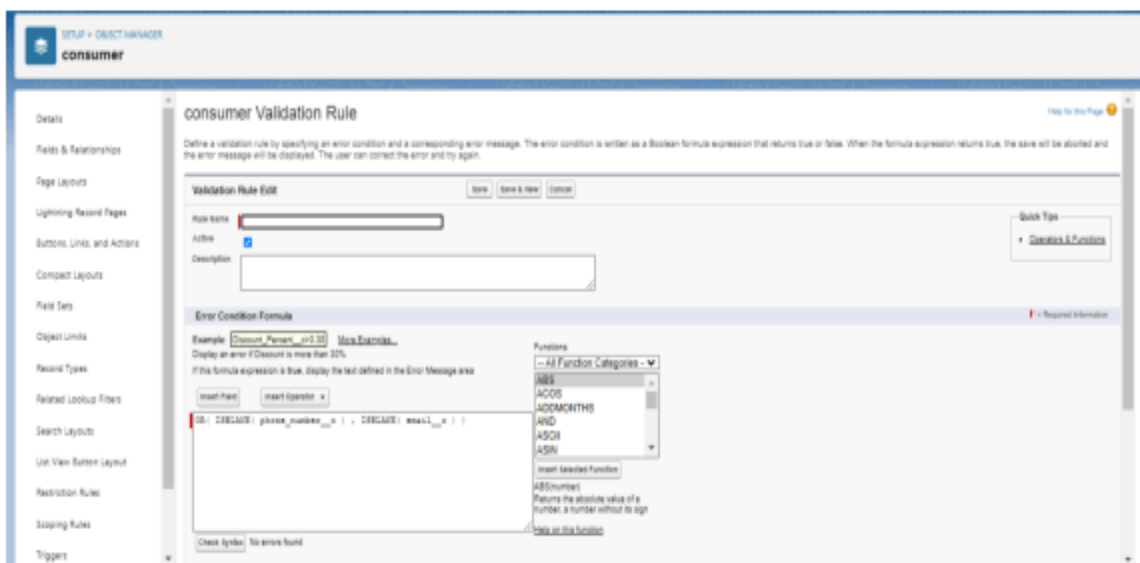
Creating the validation rule for phone number field in consumer object

1. Go to the setup page >> click on object manager >> From drop down click edit for consumer object.

2. Click on the validation rule >> click New.

- Enter the Rule name as "Phonenumberoremailblankrule".

- Enter the description as “phone number and email number should not be blank”.
- Enter the formula as “OR(ISBLANK(phone_number__c) , ISBLANK(email__c))” and check the syntax.



3. Save the validation rule.

Profiles

A profile in Salesforce is a collection of settings and permissions that define what a user can do within

the platform. Profiles control various aspects, including:

- Object permissions: Determine which objects a user can access and what actions they can perform (create, read, edit, delete)
- Field permissions: Specify which fields a user can view and edit within an object
- User permissions: Define additional permissions for users, such as the ability to manage other users or reset passwords
- Tab settings: Control which tabs are visible and accessible to users
- App settings: Determine which apps a user can access and use
- Apex class access: Specify which Apex classes a user can execute
- Visualforce page access: Define which Visualforce pages a user can access
- Page layouts: Determine the layout and sections displayed on pages for specific objects
- Record Types: Specify which record types are available to users
- Login hours & Login IP ranges: Restrict user access based on specific hours or IP addresses

Profiles are typically defined based on a user's job function, such as System Administrator, Developer, or

Sales Representative. This allows for the creation of tailored permissions that align with the responsibilities and needs of different roles within an organization.

Types of Profiles in Salesforce

1. Standard Profiles: Salesforce provides a set of predefined standard profiles by default

2. Custom Profiles: Custom profiles are created by the organization to suit specific needs

By leveraging profiles, Salesforce administrators can ensure that users have access to the necessary

objects, fields, and functionality required for their roles while maintaining security and data integrity

within the platform.

Task 16: Owner Profile

To create a new profile:

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

Q: profiles

Users

Profiles

Didn't find what you're looking for? Try using Global Search.

Profiles

Profile: OWN16T

Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information.

If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more Record Types available to users with this profile.

Link IT Chances to: [Custom Item Class Access to](#) | [Enabled Workflow Rule Access to](#) | [Enabled External Deep Source Access to](#) | [Enabled Object Contained Personal Access to](#) | [Enabled Custom Resizable Type Access to](#) | [Custom Custom Setting/Field Access to](#) | [Custom Case Access to](#) | [Custom Service Resource Status Access to](#) | [Custom Custom Permissions to](#)

Profile Detail

Name: owner

User License: Salesforce

Description: Custom Profile

Created By: [Lilitha.Sa](#) 08/08/2024, 12:07 pm

Modified By: [Lilitha.Sa](#) 08/08/2024, 12:40 pm

Page Layouts

Standard Object Layouts

Object	Layout	Location Group Assignment	Location Group Assignment Layout
Account	Custom Layout View Assignments		Custom Layout View Assignments
Email application	Not Assigned View Assignments		None View Assignments
Home Page Layout	OC Default View Assignments		Custom Main Menu Layout View Assignments
Account	Custom Layout View Assignments		Custom Main Menu Layout View Assignments
Alternative Payment Method	Alternative Payment Method Layout View Assignments		Opportunity View Assignments
Appointment Invitation	Appointment Invitation Layout View Assignments		Opportunity Product Layout View Assignments
Asset	Asset Layout View Assignments		Order View Assignments
Asset Content	Asset Content Layout View Assignments		Order Product View Assignments

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

Setup Home Object Manager

Q: profiles

Users

Profiles

Didn't find what you're looking for? Try using Global Search.

Profiles

Custom Object Permissions

Object	Read	Create	Edit	Delete	View All	Modify All
Billing Process	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
consumer	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Session Settings

Session Time-Out After: 12 hours of inactivity

Session Security Level Required at Login: None

Password Policies

User passwords expire in: 90 days

Enforce password history: 1 password remembered

Minimum password length: 8

Password complexity requirement: Must include alpha and numeric characters

Password question requirement: Cannot contain password

3. Give Access and Save it.

Task 17: Agent Profile

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

The screenshot shows the 'Profiles' setup page in a CRM system. The page is divided into several sections:

- Profiles:** A table with columns for 'Name', 'Status', 'Role', 'Permissions', and 'Actions'. It lists various profiles like 'Communication Subscription Consent', 'Communication Subscription Storage', 'Contact', 'Contact Point Addresses', 'Contact Point Consents', 'Contact Point Emails', 'Locations', 'Party Consents', 'Push Topics', 'Sellers', 'Streaming Channels', and 'User External Credentials'.
- Custom Object Permissions:** A table with columns for 'Object Name', 'Read', 'Create', 'Edit', 'Delete', 'View All', and 'Share All'. It lists permissions for 'Billing Process' and 'Laptop Bookings' objects.
- Session Settings:** A section for configuring session settings, including 'Session Times Out After' (set to 2 hours of inactivity) and 'Session Security Level Required at Login' (set to None).
- Password Policies:** A section for configuring password policies, including 'User passwords expire in' (set to 90 days), 'Enforce password history' (set to 3 passwords remembered), 'Minimum password length' (set to 8), 'Password complexity requirement' (set to Must include alpha and numeric characters), and 'Password question requirement' (set to Personal question recommended).

4. Give access and save it.

Roles And Hierarchy

In Salesforce, a role determines a user's visibility and access at the record level. Roles are utilized to define the types of data access that individuals within your Salesforce organization can have. Essentially, roles outline what information a user is able to see within the Salesforce environment.

Types of Roles in Salesforce

1. Standard Roles:

Salesforce provides predefined standard roles that come with a default set of permissions for accessing various records. These roles are integral to the platform's functionality and cannot be deleted.

2. Custom Roles:

Organizations can create custom roles tailored to their specific needs. Unlike standard roles, custom roles can be deleted if there are no users assigned to them. By establishing a clear role hierarchy, Salesforce ensures that users at higher levels have greater access to data than those at lower levels, facilitating effective data management and security within the organization.

Task 18: 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

roles

Users

Roles

Feature Settings

Sales

Connect **Roles** on Contacts

Connect **Roles** on Opportunities

Service

Case Teams

Case Team **Roles**

Connect **Roles** on Cases

Didn't find what you're looking for?
Try using Global Search.

Setup

Roles

Help for this Page

Creating the Role Hierarchy

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

Your Organization's Role Hierarchy

Customer Success

VP, Global Institute of Technology

CEO

CFO

COO

Operations

Support

VP, Customer Service & Support

Customer Support, International

Customer Support, North America

Installation & Remote Services

VP, Human Resources

VP, Sales & Marketing

Roles

Role Edit
New Role

Role Edit

Label: owner

Role Name: owner

This role reports to: CEO

Role Name as displayed on reports:

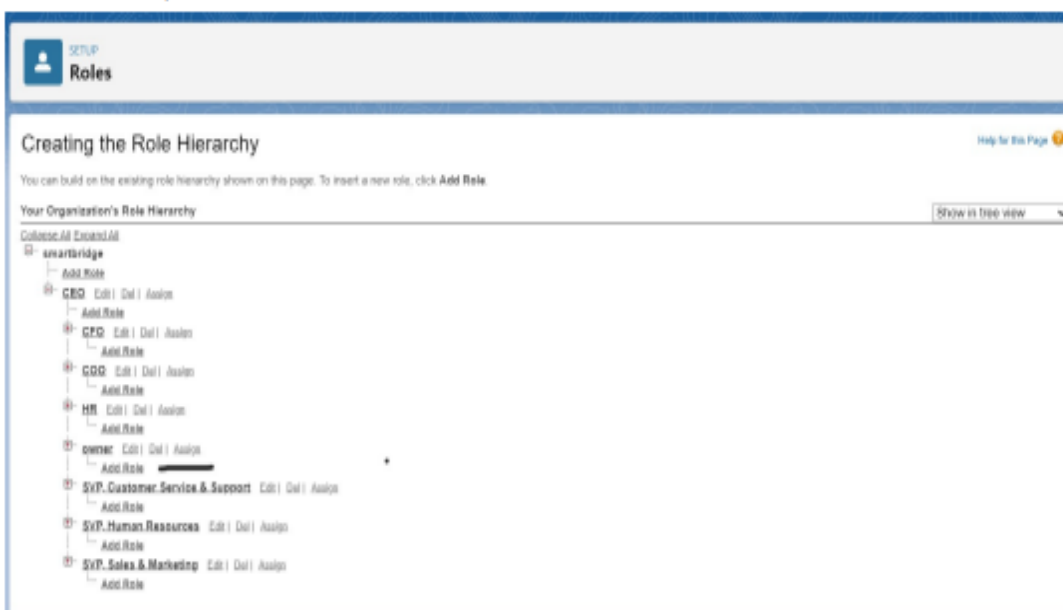
Save Save & New Cancel

4. Click and save it.

Task 19: Creating Agent roles

Creating another two roles under manager

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



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

Users

In the context of Salesforce, a user refers to any individual who logs in to the platform. These users are typically employees within your organization, such as sales representatives, managers, and IT specialists, who require access to the company's records stored in Salesforce. Each user has a unique user account that serves to identify them and defines the specific features and records they can access based on their assigned settings.

The user account is a crucial element in Salesforce, as it ensures that users are properly identified and granted the appropriate level of access to perform their job functions effectively. The account settings, which include factors like user licenses, profiles, and roles, determine the extent of a user's capabilities within the platform. This level of control ensures that sensitive data remains secure while enabling users to efficiently utilize Salesforce to meet their business objectives.

By establishing user accounts and defining their associated settings, Salesforce administrators can

effectively manage user access, maintain data integrity, and align user permissions with the organization's overall structure and requirements. This user management approach is essential for ensuring the smooth operation and optimal utilization of Salesforce within the company.

Task 20: Create User

Activity 1: Create user 1

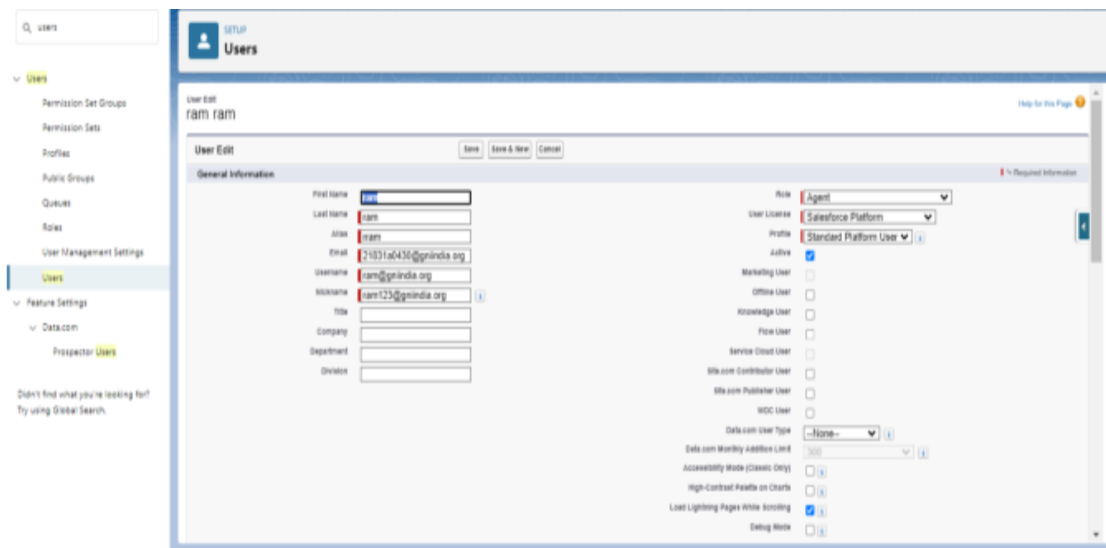
1. Go to setup - type users in quick find box - select users - click New user.
2. Fill in the fields
3. First Name : vicky
4. Last Name : y
5. Alias : Give a Alias Name
6. Email id : Give your Personal Email id
7. Username : Username should be in this form:
text@text.text
8. Nick Name : Give a Nickname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

The screenshot shows the Salesforce 'Users' setup page. On the left is a navigation menu with options like 'Users', 'Permission Set Groups', 'Permission Sets', 'Profiles', 'Public Groups', 'Queues', 'Roles', and 'User Management Settings'. The main area is titled 'User Edit' for user 'vicky y'. It contains two columns of fields. The left column includes 'First Name' (vicky), 'Last Name' (y), 'Alias' (y), 'Email' (21631a6436@ggindia.org), 'Username' (vicky@ggindia.org), 'Nickname' (vicky123@ggindia.org), 'Title', 'Company', 'Department', and 'Division'. The right column includes 'Role' (Agent), 'User License' (Salesforce), 'Profile' (Agent), 'Active' (checked), 'Marketing User' (unchecked), 'Offline User' (unchecked), 'Knowledge User' (unchecked), 'Flow User' (unchecked), 'Service Cloud User' (unchecked), 'B2B.com Contributor User' (unchecked), 'B2B.com Publisher User' (unchecked), 'WDC User' (unchecked), 'Data.com User Type' (None), 'Data.com Identity' (Add New), 'Accessibility Mode (Classic Only)' (unchecked), 'High-Contrast Palette on Charts' (unchecked), 'Load Lightning Pages While Browsing' (checked), and 'Debug Mode' (unchecked).

Save it.

Activity 2: creating another users

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



Flows

In Salesforce, a flow is a powerful tool that allows you to automate business processes,

collect and update data, and guide users through a series of screens steps. Flows are built using a visual interface and can be created without any coding knowledge. In Salesforce, "flows" typically refer to Salesforce Flow, which is a powerful automation tool

that allows you to create custom, automated processes in your Salesforce org without writing code. Salesforce Flow is a point-and-click tool that enables you to design and automate complex business processes, collect data, and interact with users in a visual

interface. There are different types of flows in Salesforce, including:

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

display information. Screen Flows are often used for data entry and updates.

2. Autolaunched Flows: These are flows that are triggered by events, such as when a

record is created or updated. They don't require user interaction and can be

used for

background automation.

3. Flow Builder: Flow Builder is the visual interface used to create flows. It allows you to

design flows by adding elements, like screens, logic, and actions, using a drag-anddrop approach.

4. Flow Templates: Salesforce provides a library of pre-built flow templates that you

can use as a starting point for your own flows. These templates cover a variety of use

cases, from simple to complex.

5. Scheduled Flows: These are flows that you can schedule to run at specific times or

intervals. They are often used for automating recurring tasks.

6. Flow Elements: Flow Builder offers various elements that you can use to create

flows, such as variables, decisions, loops, and more. These elements allow you to

build sophisticated logic into your flows.

7. Subflows: Subflows are reusable flow elements that you can incorporate into

multiple flows, making it easier to manage and maintain complex processes.

8. Record-Triggered Flows: These are flows that are triggered when records meet

specified criteria. They are often used for automating record updates and related

actions.

Why do we need to create a flow:

In summary, creating a flow in Salesforce is crucial for automating tasks,

streamlining processes, and enhancing user experiences. It empowers organizations to operate more efficiently while maintaining high standards of data quality and operational consistency

Select Object

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

* Object

Laptop Bookings

Configure Trigger

* Trigger the Flow When:

☐ A record is created

☐ A record is updated

☒ A record is created or updated

☐ A record is deleted

Set Entry Conditions

Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Cancel Done

Configure Start

Set Entry Conditions

Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Condition Requirements

None

* Optimize the Flow for:

Fast Field Updates

Update fields on the record that triggers the flow to run. This high-performance flow runs before the record is saved to the database.

Actions and Related Records

Update any record and perform actions, like send an email. This more flexible flow runs after the record is saved to the database.

☐ Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed

Cancel Done

8. Enter the Outcome Details Label: dell , Outcome API name: Gets Automatically Generated. • Resource: Select Record.Laptop booking__c. • Operator: Select Equals. • Value: Select dell • Add the same outcome order to acer , hp,mac. Click done

The screenshot shows the 'Edit Decision' interface. At the top, there are fields for '*Label' (containing 'field should be updated') and '*API Name' (containing 'field_should_updated'). Below these is a 'Description' field with the text 'the field should be automatically updated'. The 'Outcomes' section is expanded, showing a list of outcomes on the left: 'dell', 'acer', 'hp', 'mac', and 'false'. The 'dell' outcome is selected, and its details are shown on the right. The details include: '*Label' (dell), '*Outcome API Name' (dell), 'Condition Requirements to Execute Outcome' (All Conditions Are Met (AND)), 'Resource' (Record > Laptop names X), 'Operator' (Equals), and 'Value' (Dell). There are 'Cancel' and 'Done' buttons at the bottom right.

9. Go to flow page

10. Check the flow chart.

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

12. Again select decision

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

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

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

15.select the Outcome '+' Details Label: dell core i5 , Outcome API name: Gets

Automatically Generated.

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

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

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

17.Click done.

Edit Decision

* Label: 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 1 +

OUTCOME DETAILS Delete Outcome

* Label: dell core i3 * Outcome API Name: dellcore_i3

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

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

Cancel Done

18. So go to the flow page select '+' after core i3 then again select the decision.

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

20. 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.

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

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

22. Click '+' outcome details

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

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

24. Click '+' outcome details

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

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

26. Click '+' outcome details

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

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

The screenshot displays a configuration window for outcomes. At the top, there are fields for '* Label' (containing 'months selected') and '* API Name' (containing 'months_selected'). Below these is a 'Description' text area. A section titled 'Outcomes' includes a sub-header: '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.' On the left, an 'OUTCOME ORDER' list shows five items, with item 1 selected. The 'OUTCOME DETAILS' panel for item 1 shows: '* Label' as '1', '* Outcome API Name' as 'X1', 'Condition Requirements to Execute Outcome' as 'All Conditions Are Met (AND)', and a condition rule where 'Resource' is '\$Record > how many months X', 'Operator' is 'Equals', and 'Value' is '1'. A 'Delete Outcome' button is in the top right of the details panel. At the bottom right of the window are 'Cancel' and 'Done' buttons.

28. Follow the above picture you will understand.

29. After dell 1(i3) there is '+' symbol like dell 2(i3),dell3(i3),dell 4(i3),dell5(i3).

30. Click on '+' then select update records

31. Enter the Details Label: one month of dell i3 rate

32. API name: Gets Automatically Generated.

33. Field:- Amount__c

34. value:- for dell 1(i3)-1000, dell 2(i3)-2000, dell 3(i3)-3000, dell 4(i3)-4000, dell 5(i3)-5000.

35. Follow all these finally.

36. Click done.

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

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

39.

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

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

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

41. Click '+' outcome details

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

Automatically

Generated.

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

43.Click '+' outcome details

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

Generated.

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

45.Click '+' outcome details

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

Generated.

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

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 Label: one month of dell i5 rate , 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.

Task 22: Creating Flow On Acer Laptop

1. Go to flow page

2. Beside acer there is a symbol '+' click on that.

3. Again select decision

4. Enter the Details Label: Field is Update, API name: Gets Automatically Generated.

5. select the Outcome Details Label: acer core i3 , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.core type.
- Operator: Select Equals.
- Value: Select core i3.

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

Delete Outcome

acer core i3

*Label acer core i3

*Outcome API Name acer_core_i3

acer core i5

acer core i7

Condition Requirements to Execute Outcome

All Conditions Are Met (AND)

Resource Operator Value

\$Record > core type X Equals 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

6. Go to flow page

7. Beside dell there is a symbol '+' click on that.

8. Again select decision

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

10. Enter the Outcome Details Label: acer 1(i3) , Outcome API name: Gets Automatically Generated.

11.

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

- Value: 1.

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

Generated.

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

13.Click '+' outcome details

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

Generated.

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

15.Click '+' outcome details

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

Generated.

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

17.Click '+' outcome details

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

Generated.

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

Edit Decision

*Label: acer months selected *API Name: acer_months_selected

Description:

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 Condition Requirements to Execute Outcome: All Conditions Are Met (AND) Resource: \$Record > how many months Operator: Equals Value: 1
acer 2(i3)	
acer 3(i3)	
acer 4(i3)	
acer 5(i3)	

Delete Outcome Cancel Done

Click done.


19. After acer 1(i3) there is '+' symbol like acer 2(i3), acer 3(i3), acer 4(i3), acer 5(i3).

20. Click on '+' then select update records

21. Enter the Details Label: one month of acer i3 rate, API name: Gets Automatically

Generated.

22. Field:- Amount__c, value:- for acer 1(i3)-900, acer 2(i3)-1800, acer 3(i3)-2700, acer 4(i3)-3600, acer 5(i3)-4800. Follow for all these finally

one month of acer i3 rate (one_month_of_acer_i3_rate) 

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


- ☒ Use the laptop bookings record that triggered the flow
- ☐ Update records related to the laptop bookings record that triggered the flow
- ☐ Use the IDs and all field values from a record or record collection
- ☐ Specify conditions to identify records, and set fields individually

Set Filter Conditions

Condition Requirements to Update Record

None—Always Update Record ▼

Set Field Values for the Laptop Bookings Record

Field	Value	
Amount_c	900	

[+ Add Field](#)

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

Task 23: Creating A Flow On Hp Laptop

1. Go to flow page
2. Beside hp there is a symbol '+' click on that.
3. Again select decision
4. Enter the Details Label: Field is Update, API name: Gets Automatically Generated.
5. select the Outcome Details Label: hp core i5 , Outcome API name: Gets Automatically Generated.
 - Resource: Select Record.core type.
 - Operator: Select Equals.
 - Value: Select hp i5.
6. Go to flow page
7. Beside hp there is a symbol '+' click on that.

8. Again select decision

9. Enter the Details Label: hp field should be updated , API name: Gets Automatically Generated.

10. Enter the Outcome Details Label: hp 1(i5) , Outcome API name: Gets Automatically Generated.

11. Here,

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

12. Enter the Outcome Details Label: hp 2(i5) , Outcome API name: Gets Automatically Generated.

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

13. Click '+' outcome details

14. Enter the Outcome Details Label: hp 3(i5) , Outcome API name: Gets Automatically Generated.

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

15. Click '+' outcome details

16. Enter the Outcome Details Label: hp 4(i5) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.

- Operator: Select Equals.

- Value: Select 4.

17.Click '+' outcome details

18.Enter the Outcome Details Label: hp 5(i5) , Outcome API name: Gets Automatically Generated.

- Resource: Select Record.how many months.

- Operator: Select Equals.

- Value: Select 5.


Click on done.

19.After hp 1(i5) there is '+' symbol like hp 2(i5), hp 3(i5), hp 4(i5),hp 5(i5).

20.Click on '+' then select update records

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

22.Field:- Amount__c , value:- for hp 1(i5)-1700, hp 2(i5)-3400, hp 3(i5)-5100, hp 4(i5)-6800, hp 5(i5)-8500. Follow for all these finally

one month of hp i5 rate (one_month_of_hp_i5_rate) 

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


- ☒ Use the laptop bookings record that triggered the flow
- ☐ Update records related to the laptop bookings record that triggered the flow
- ☐ Use the IDs and all field values from a record or record collection
- ☐ Specify conditions to identify records, and set fields individually

Set Filter Conditions

Condition Requirements to Update Record

None—Always Update Record ▼

Set Field Values for the Laptop Bookings Record

Field	Value	
Amount__c	1700	

[+ Add Field](#)

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

Task 24: Creating A Flow On Mac Laptop

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

EDIT DECISION

*Label: mac field should be updated

*API Name: mac_field_should_be_updated

Description:

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

OUTCOME ORDER: +

mac laptop

Default Outcome

OUTCOME DETAILS

*Label: mac laptop

*Outcome API Name: mac_laptop

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

Resource	Operator	Value
\$Record > core type X	Equals	Bionic chip

+ Add Condition

Cancel Done

Click done.

6. Go to flow page

7. Beside Mac there is a symbol '+' click on that.

8. Again select decision

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

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

11.

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

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

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

13.Click '+' outcome details

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

Automatically Generated.

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

15.Click '+' outcome details

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

Automatically Generated.

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

17.Click '+' outcome details

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

Automatically Generated.

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

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
mac bionic chip(1)	<div> <div>*Label</div> <div>mac bionic chip(1)</div> </div> <div> <div>*Outcome API Name</div> <div>mac_bionic_chip_1</div> </div> <div> <div>Condition Requirements to Execute Outcome</div> <div>All Conditions Are Met (AND)</div> </div> <div> <div>Resource</div> <div>\$Record > how many months X</div> </div> <div> <div>Operator</div> <div>Equals</div> </div> <div> <div>Value</div> <div>1</div> </div>
mac bionic chip(2)	
mac bionic chip(3)	
mac bionic chip(4)	
mac bionic chip(5)	
Default Outcome	<div> <div>+ Add Condition</div> </div> <div> <div>When to Execute Outcome</div> <div> <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 </div> </div>

Cancel Done

Click done.

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

20.Click on '+' then select update records

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

22.Field:- Amount__c , value:- for one month of mac bionic chip rate-1700, two month

of mac bionic chip rate-3400, three month of mac bionic chip rate-5100, four

month of mac bionic chip rate-6800, five month of mac bionic chip rate-8500.

Follow for all these finally

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

☒ Use the laptop bookings record that triggered the flow
☐ Update records related to the laptop bookings record that triggered the flow
☐ Use the IDs and all field values from a record or record collection
☐ Specify conditions to identify records, and set fields individually

Set Filter Conditions

Condition Requirements to Update Record

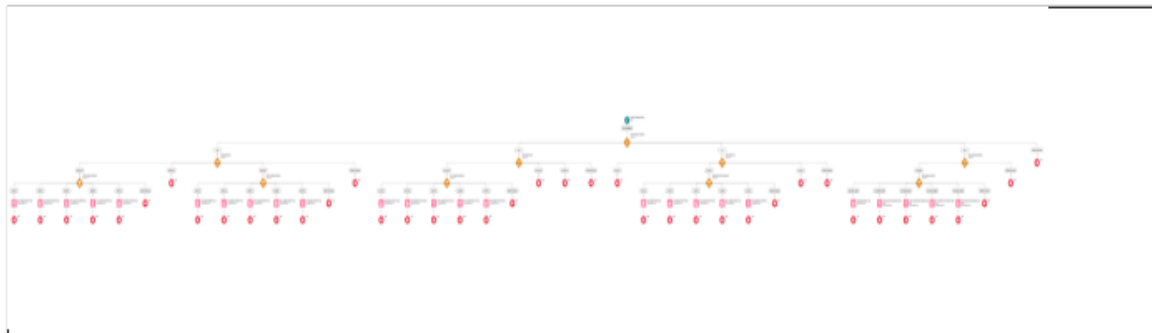
None—Always Update Record ▼

Set Field Values for the Laptop Bookings Record

Field	Value
Amount__c	2000

23. Click done.

FLOW:



- Click on save .
- Label:- Laptop distributions, api name:- automatically filled
- Save the flow and activate it.

APEX

Apex is a powerful, object-oriented programming language that allows developers to execute flow and transaction control statements on the Salesforce platform server, in conjunction with calls to the Salesforce API. Apex code resembles Java syntax and functions similar to database stored procedures, enabling developers to add business logic to various system events, such as button clicks, related record updates, and Visualforce pages. Additionally, Apex code can be initiated by web service requests and from triggers on objects.

Apex shares many similarities with Java, including support for object-oriented programming (OOP) concepts like classes, objects, and methods.

Creating Classes in Apex

Apex classes are modeled after their counterparts in Java. Developers can define, instantiate, and extend classes, as well as work with interfaces, Apex class versions, properties, and other related class concepts.

Class

Similar to Java, Apex allows the creation of classes. A class serves as a template or blueprint from which objects are created. An object is an instance of a class.

Object

An object is an instance of a class in Apex. When an object is created, it can access all the properties

present in the class, including variables and methods.

By leveraging Apex's object-oriented programming capabilities, developers can create modular,

reusable code that enhances the functionality and flexibility of Salesforce applications. The language's

syntax and behavior, which closely resemble Java, make it easier for developers familiar with Java to

transition to Apex development.

Access specifiers in Apex

Apex allows you to use the private, protected, public, and global access modifiers when

defining methods and variables.

While triggers and anonymous blocks can also use these access modifiers, they aren't as

useful in smaller portions of Apex. For example, declaring a method as global in an

anonymous block doesn't enable you to call it from outside of that code.

Private:

This access modifier is the default, and means that the method or variable is accessible

only within the Apex class in which it's defined. If you don't specify an access modifier, the

method or variable is private.

Protected:

This means that the method or variable is visible to any inner classes in the defining Apex

class, and to the classes that extend the defining Apex class. You can only use this access

modifier for instance methods and member variables. This setting is strictly more

permissive than the default (private) setting, just like Java.

Public :

This means that the method or variable is accessible by all Apex within a specific package.

For accessibility by all second-generation (2GP) managed packages that share a

namespace, use public with the `@NamespaceAccessible` annotation. Using the public

access modifier in no-namespace packages implicitly renders the Apex code as

`@NamespaceAccessible`.

Global:

This means the method or variable can be used by any Apex code that has access to the

class, not just the Apex code in the same application. This access modifier must be used for

any method that must be referenced outside of the application, either in SOAP API or by

other Apex code. If you declare a method or variable as global, you must also declare the

class that contains it as global. This is how a new class is created :

Triggers

A trigger is a set of Apex code that runs before or after DML(Data Manipulation Language)

events.

A DML event could be a variety of data processing tasks that include the standard insert,

update, and delete commands.

With Apex triggers, you can automate tasks that would otherwise be nearly impossible to

accomplish using only the Salesforce user interface. Triggers enable you to create custom scripts that you can implement according to your needs, and the only limitation is your coding skills.

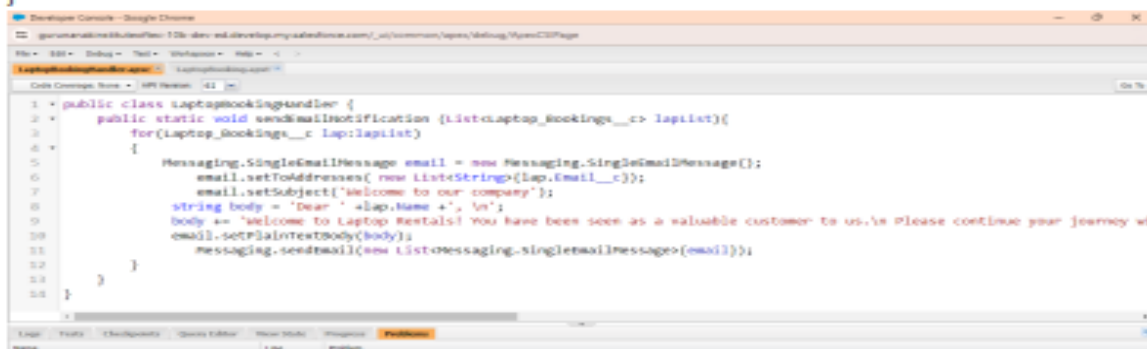
There are two Salesforce Apex trigger types:

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

The code snippet for the apex class is:

```
public class LaptopBookingHandler {  
    public static void sendEmailNotification (List<Laptop_Bookings__c> lapList){  
        for(Laptop_Bookings__c lap:lapList)  
        {  
            Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
```

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



Reports

Reports give you access to your Salesforce data. You can examine your Salesforce data in almost infinite combinations, display it in easy-to-understand formats, and share the resulting insights with others. Before building, reading, and sharing reports, review these reporting basics. In Salesforce.com we can easily generate reports in different styles. And can create reports in a very short time and also schedule the reports. Salesforce provides a powerful suit of analytic tools to help you organize, view and analyze your data. Types of Reports in Salesforce

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

1. Tabular Reports: Simple listing of data without any subtotals. This type of reports provide you most basically to look at your data. Use tabular reports when you want a

simple list or a list of items with a grand total. Example: This type of reports are used to list all accounts, List of contacts, List of opportunities.....etc.....

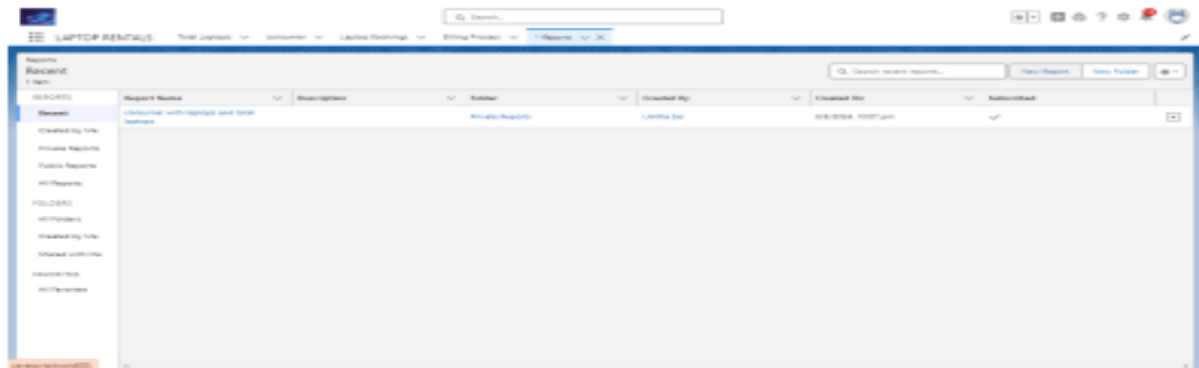
2. Summary Reports: This type of reports provide a listing of data with groupings and sub totals. Use summary reports when you want subtotals based on the value of a particular field or when you want to create a hierarchically grouped report, such as sales organized by year and then by quarter. Example: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

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

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

Task 27: Create Report

2. Click New Report.



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



4. Customize your report

5. Add fields from left pane as shown below

THANK YOU