

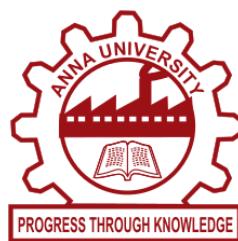
PROJECT REPORT

Naan Mudhalvan – Salesforce Developer

Project Title:
TripAdvisor E-Management

Team Members:

NAME	NM ID	32 Digits NM ID
Sriman Yaghav C (T.L)	aut710021104307	FABBB07CAD2ADA9A32E41A976BC78491
Ragavan K	aut710021104310	7E1DF4E4D01B65A8C7F45671CCD86A1D
Umar Ahmed Khan A	au710021104032	6A8BB118261CE23A5F6493C4549B657D
Nikhil M	aut710021104321	B42B95E24A916099F774D984E6E6403C



ANNA UNIVERSITY
REGIONAL CAMPUS COIMBATORE

Report: TripAdvisor E-Management

Project Overview

Project Title: *TripAdvisor E-Management*

This project is focused on developing the *TripAdvisor E-Management System* using Salesforce to streamline travel-related services, such as hotel, food option, and flight management, along with customer discount automation and notification services. The project aims to address inefficiencies in data handling and communication within TripAdvisor's ecosystem by implementing a comprehensive and automated solution. The primary challenge was to ensure seamless integration of various services to enhance operational efficiency, reduce manual errors, and improve the user experience.

By leveraging Salesforce's robust platform features, such as automation tools, Apex triggers, and schedulable classes, this project provides a scalable, reliable, and efficient solution to meet the business needs of TripAdvisor.

Project Description:

The TripAdvisor E-Management system, integrated with Salesforce, aims to provide an all-in-one travel companion app that empowers users to plan, book, and make the most of their trips. This system brings TripAdvisor's massive repository of user-generated reviews and insights into the Salesforce ecosystem, allowing users to access information on hotels, flights, food options, and customer deals seamlessly. This report outlines the system requirements, acceptance criteria, and solutions developed for this integration, with a focus on automation and streamlined customer experience.

Short Description:

TripAdvisor E-Management Solution streamlines travel itinerary management, centralizing booking data, trip tracking, and user feedback, making it easier for organizations to oversee and enhance travel experiences.

Objectives

Business Goals:

1. Automate the management of hotels, flights, and food options to reduce manual administrative tasks.
2. Provide a personalized customer experience through automated discount mechanisms.
3. Ensure timely communication with customers via email notifications for flight reminders.

Specific Outcomes:

1. Automated tracking and updating of hotel information based on food options.
2. Automated discount calculation and application based on customer purchase thresholds.
3. Flight reminder emails sent automatically 24 hours before flight departure, improving customer satisfaction.
4. Accurate, real-time reporting and data analytics for better business insights.

Salesforce Key Features and Concepts Utilized

The project leverages the following Salesforce features and functionalities:

1. Custom Objects:

- **Hotel Object:** Stores hotel-related data and calculates the total number of food options associated with each hotel.
- **Food Option Object:** Tracks food options linked to hotels.
- **Flight Object:** Manages flight bookings and departure schedules.
- **Customer Object:** Stores customer information, such as name and booking details.

2. Apex Triggers:

- Ensures automatic updating of hotel data whenever food options are added or modified.
- Example: Updating the total count of food options for each hotel.

3. Flows:

- Automates customer discount application based on purchase amounts.
- Example: Full discounts for purchases exceeding 3000 and partial discounts for purchases between 1500 and 3000.

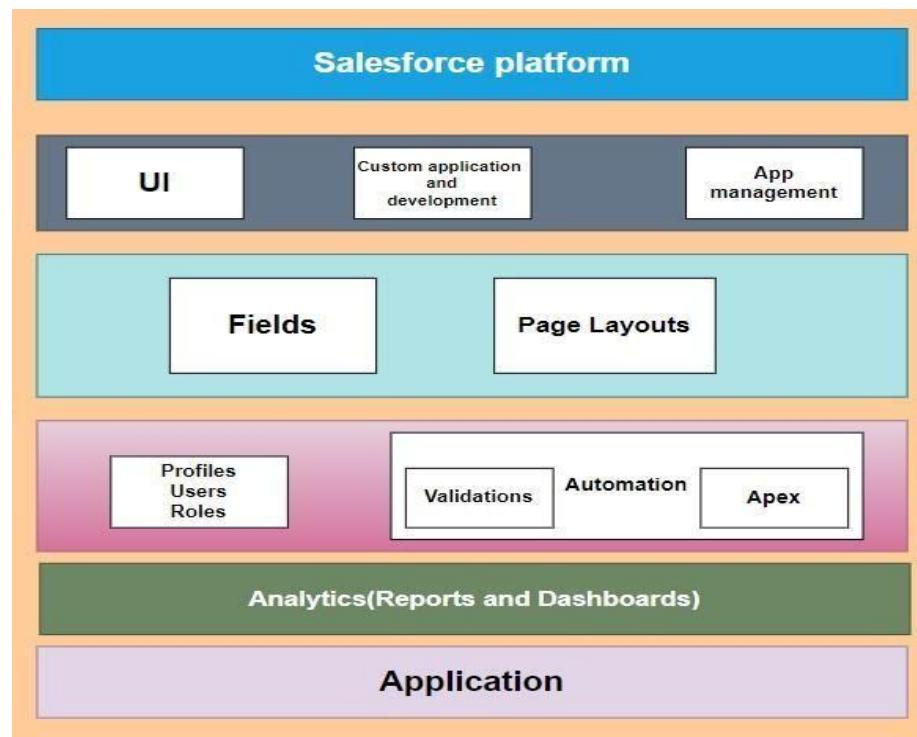
4. Schedulable Apex Classes:

- Handles scheduled email notifications for customers regarding upcoming flights.
- Example: Sends email reminders 24 hours before the scheduled departure.

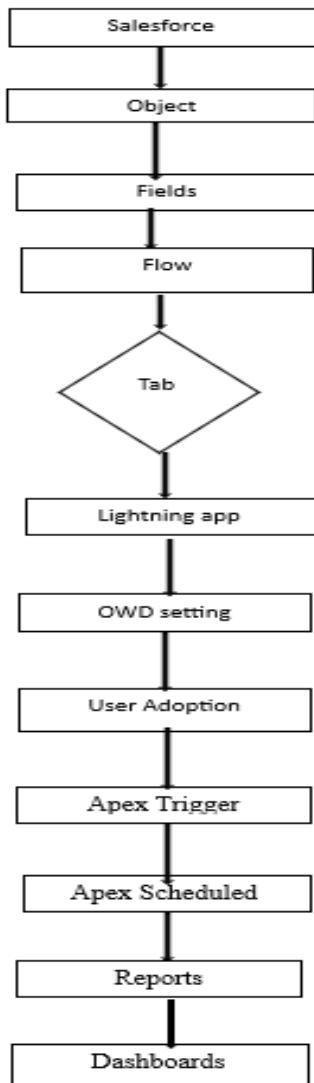
5. Reports and Dashboards:

- Tracks customer discounts, flight bookings, and hotel occupancy rates for operational insights.

Technical Architecture:



Project Flow:



Detailed Steps to Solution Design

Milestone 1- Salesforce:

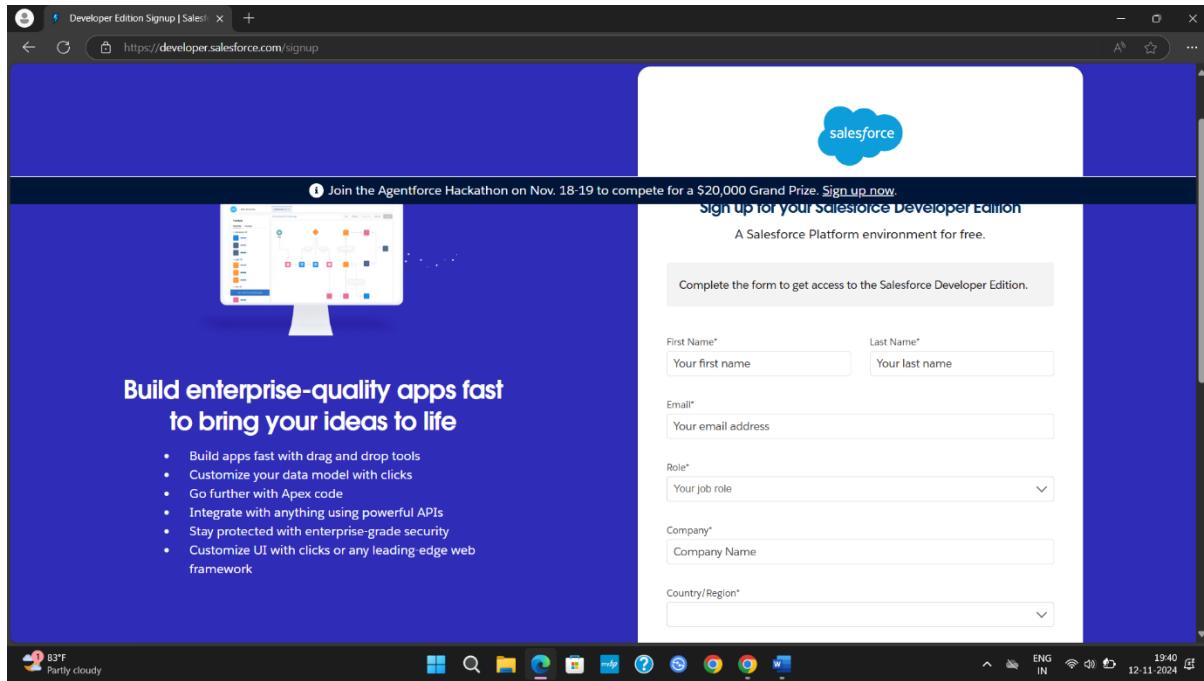
Developer Account Creation

To start working with Salesforce CRM, a developer account is essential. Follow these steps to create an account:

1.Sign-Up Process

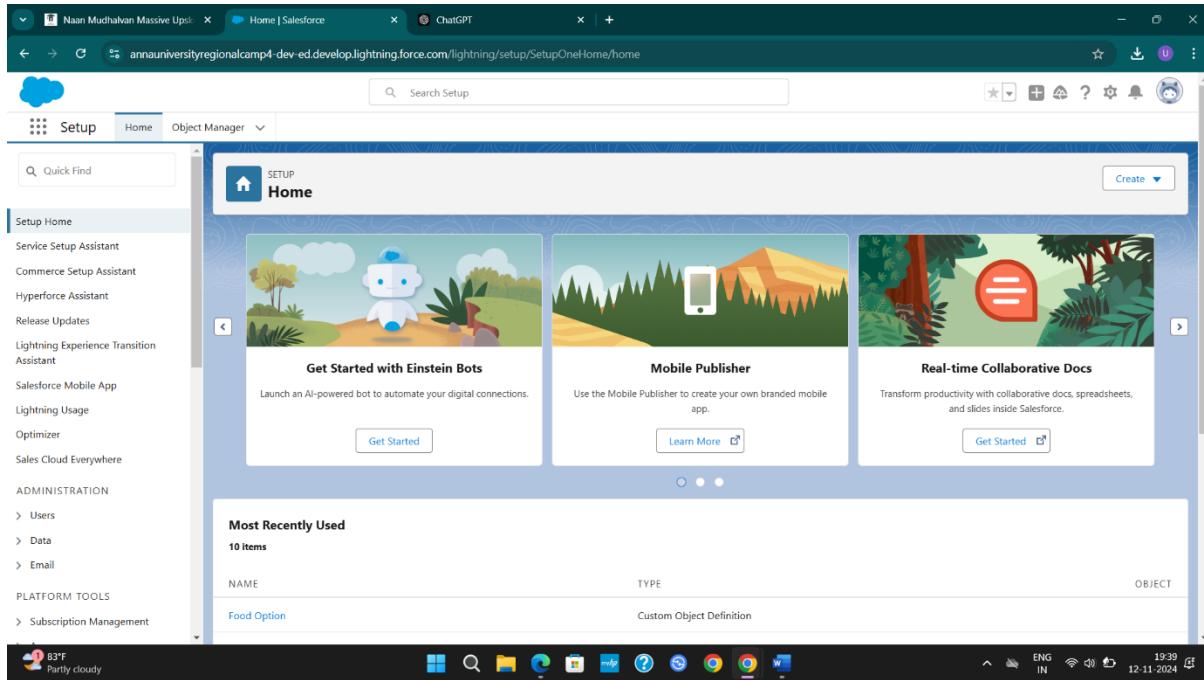
- Go to [Salesforce Developer Sign-Up](#).
- Enter your **First and Last Name**, **Email**, and set **Role** as “Developer.”
- Input your **Company** (College Name), **Country** (India), **Postal Code**, and **Username** (formatted as username@organization.com).

Click **Sign Me Up** after filling out the form



Account Activation

- Open the inbox of the email used for registration, locate the Salesforce verification email, and click **Verify Account**.
- Set a password, choose a security question, and log into your Salesforce account to access the setup page.



Milestone 2 - Objects in Salesforce:

Salesforce objects function as database tables for storing and organizing data relevant to the organization.

- **Standard Objects:** Provided by Salesforce by default (e.g., Accounts, Contacts).
- **Custom Objects:** User-defined objects to store unique organizational data.

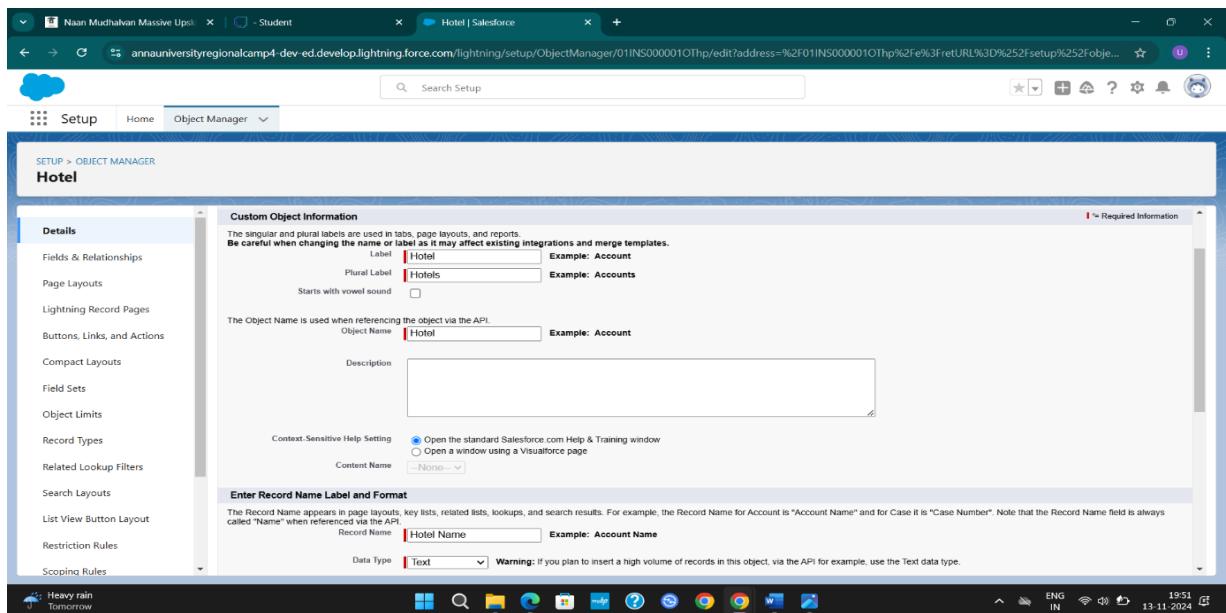
System Development: Custom Objects

Hotel Object:

Hotel Object is created to ensure that when a new Food Option is added or updated with the necessary information

1. Enter label: Hotel
2. Plural Name: Hotels
3. Data Type: (text)
4. Field Name: Hotel Name
5. Click Allow Reports
6. Click Allow Search > Save
- 7.

Purpose: Store data about hotels and update hotel information when new food options are added.



Food Option Object:

1. Enter label: Food Option
2. Plural Name: Food Options
3. Data Type: Auto Number > Display Format > FO - {0000} > Starting Number > 1
4. Field Name: Food Option Name
5. Click Allow Reports
6. Click Allow Search > Save

Purpose: Track food options associated with hotels.

Flight Object:

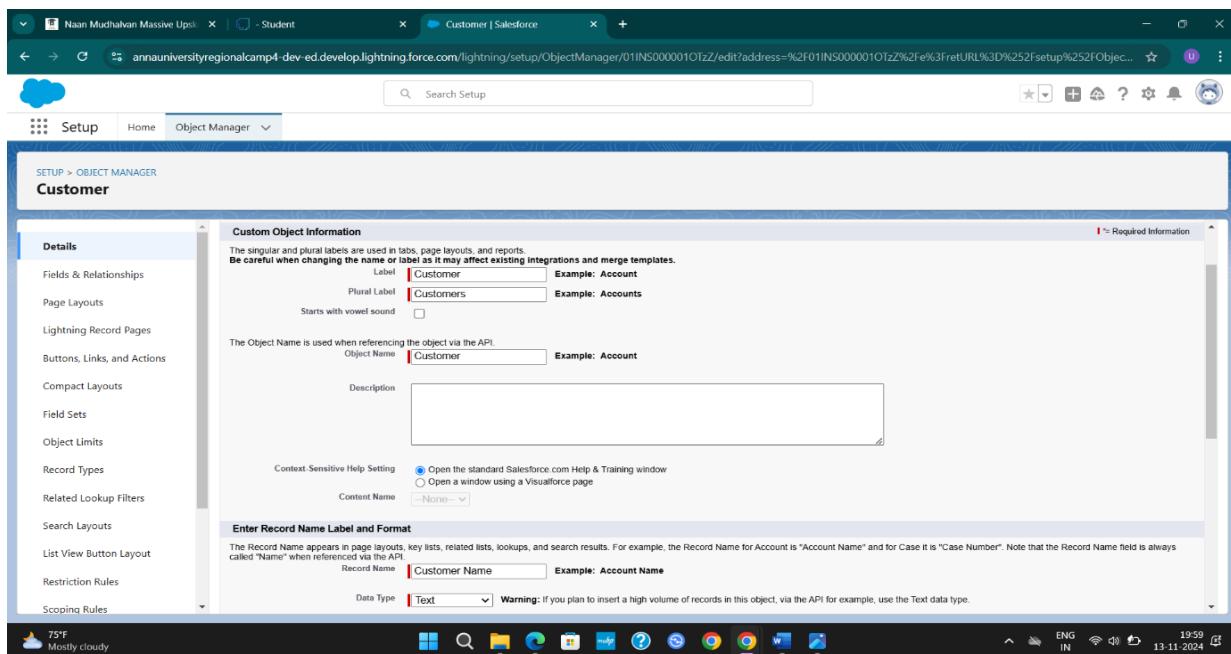
1. Enter label: Flight
2. Plural Name: Flights
3. Data Type: Auto Number >Display Format > FL - {0000} > Starting Number > 1
4. Field Name: Flight Name
5. Click Allow Reports
6. Click Allow Search > Save

Purpose: Track flight bookings and manage customer notifications.

Customer Object:

1. Enter label: Customer
2. Plural Name: Customers
3. Data Type: Text
4. Field Name: Customer Name
5. Click Allow Reports
6. Click Allow Search > Save

Purpose: Manage customer information.

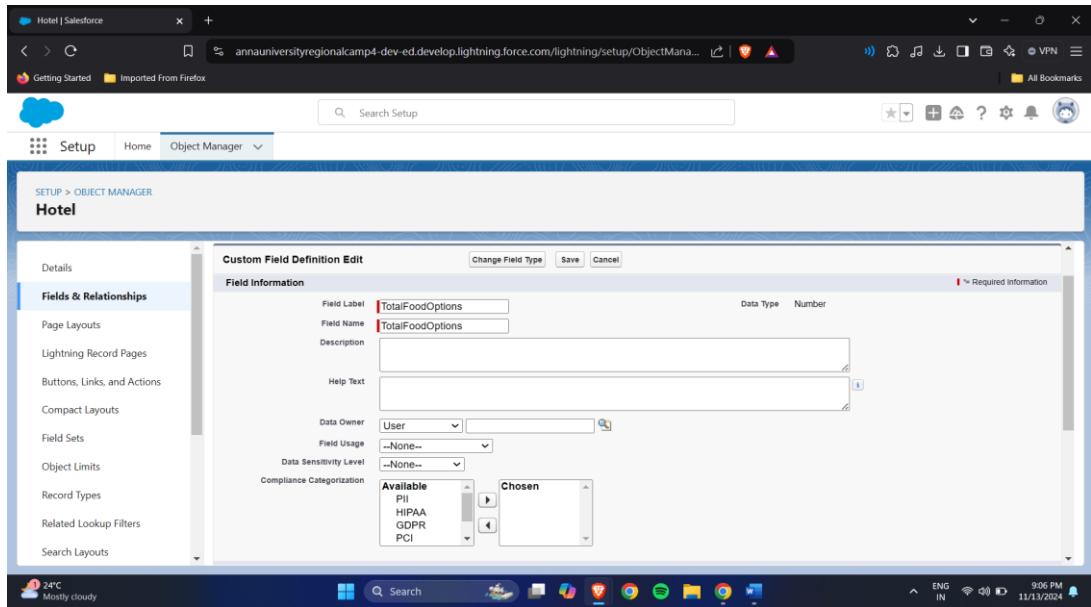


Milestone 3 – Fields:

Create Fields & Relationships for Hotel Object:

Creating fields for a "Hotel" object involves defining the data attributes that represent essential information about a hotel. These fields should capture the details needed to describe and manage the hotel within an application, database, or any system that tracks hotel information.

Sr. No.	Field Name	Data Type
1	TotalFoodOptions	Number
2	Date	Date



Fields & Relationships of all Hotel Fields:

In a system where you manage hotel data, creating fields and defining relationships for the "Hotel" object is crucial for organizing and retrieving information efficiently. Here's a detailed overview of the fields and relationships typically associated with a "Hotel" object.

Fields & Relationships				
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Date	Date__c	Date		
Hotel Name	Name	Text(80)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		
TotalFoodOptions	TotalFoodOptions__c	Number(18, 0)		

Create Fields for Food Option:

Creating fields for a "Food Option" object is essential when building a system to manage food items, such as a restaurant menu or a hotel's food service options. These fields should cover all the necessary details that define each food item and make it easy for users to search, categorize, and manage food options.

Sr. No.	Field Name	Data Type
1	Name	Text
2	Hotel	Hotel (Lookup)
3	Food Amount	Currency

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Food Amount	Food_Amount__c	Currency(18, 0)		
Food Option Name	Name	Auto Number		✓
Hotel	Hotel__c	Lookup(Hotel)		✓
Last Modified By	LastModifiedById	Lookup(User)		
Name	Name__c	Text(255)		
Owner	OwnerId	Lookup(User,Group)		✓

Create Fields in the Flight Object:

Creating fields in a "Flight" object involves defining essential attributes that represent information about a flight. These fields help to manage and organize flight details within a system for booking, tracking, or scheduling flights. Here's an example of typical fields for a "Flight" object

Sr. No.	Field Name	Data Type
1	Name	Hotel(Lookup)
2	DepartureDateTime	Date/Time
3	ContactEmail	Email

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
ContactEmail	ContactEmail__c	Email		
Created By	CreatedById	Lookup(User)		
DepartureDateTime	DepartureDateTime__c	Date/Time		
Flight Name	Name	Auto Number		✓
Last Modified By	LastModifiedById	Lookup(User)		
Name	Name__c	Lookup(Hotel)		✓
Owner	OwnerId	Lookup(User,Group)		✓

Create Fields in the Customer Object:

Creating fields for a "Customer" object involves defining essential details to identify and understand each customer in the system. Here are common fields typically included

Sr. No.	Field Name	Data Type
1	Customer Name	Text
2	Discount Amount	Currency
3	Discount Percent	Percent

The screenshot shows the Salesforce Setup interface for the Customer object. On the left, there's a sidebar with options like Details, Fields & Relationships (which is selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts. The main area is titled 'Fields & Relationships' and shows a table with 7 items, sorted by Field Label. The table columns are FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The data includes:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Customer Name	Customer_Name__c	Text(255)		
Customer Name	Name	Text(80)		✓
Discount Amount	Discount_Amount__c	Currency(18, 0)		
Discount Percent	Discount_Percent__c	Percent(18, 0)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓

Milestone 4 -Flow:

Flow for Customer Discount Automation:

A Salesforce Flow was created to apply discounts based on the customer's purchase amount. Discounts are granted if the amount exceeds certain thresholds:

Create a new flow variable “TripAdvisor”.

Flow Variables

Create 3 variable:

Variable > Api name > foId > text > Available for Input

Variable > Api name > csId > text > Available for Input

Variable > Api name > discount > Number

Variable > Api name > fold > text > Available for Input

New Resource

* Resource Type
Variable

* API Name fold

Description

* Data Type
Text Allow multiple values (collection) i

Default Value
 Enter value or search resources... Search icon

Availability Outside the Flow
 Available for input Available for output

Cancel Done

Variable > Api name > csId > text > Available for Input

New Resource

* Resource Type
Variable

* API Name csId

Description

* Data Type
Text Allow multiple values (collection) i

Default Value
 Enter value or search resources... Search icon

Availability Outside the Flow
 Available for input Available for output

Cancel Done

Variable > Api name > discount > Number

New Resource

* Resource Type
Variable

* API Name discount

Enter a value.

Description

* Data Type
Number Allow multiple values (collection) i

Decimal Places
 2

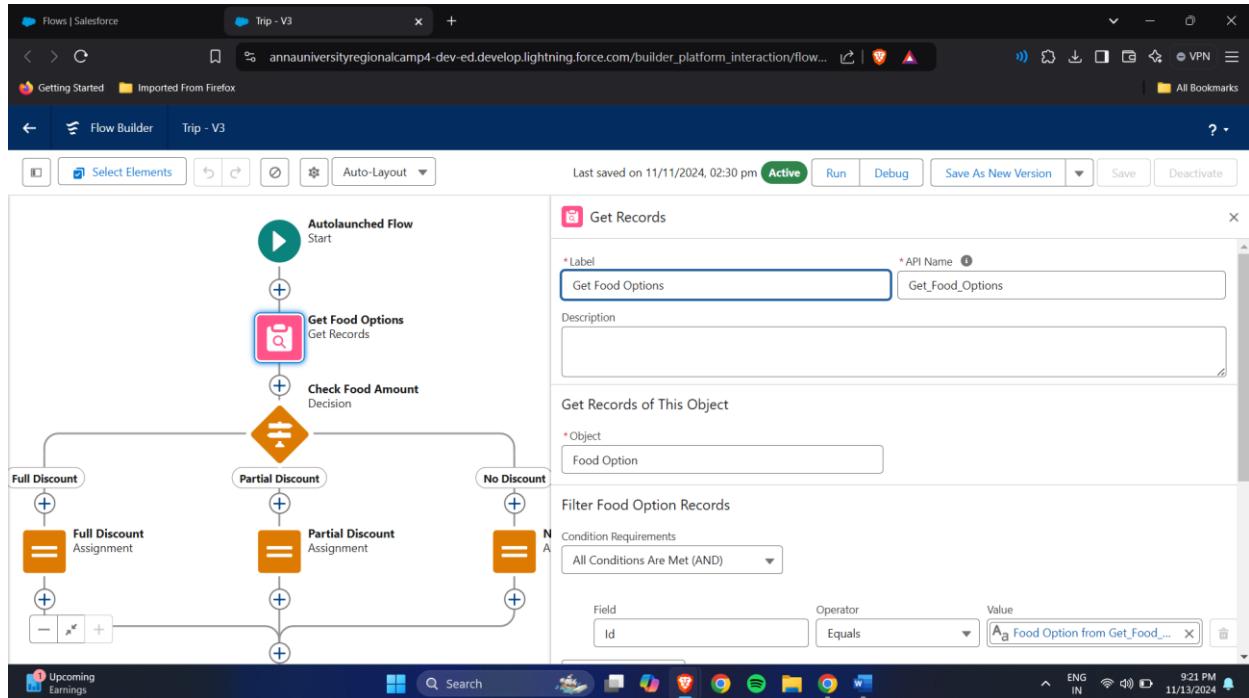
Default Value
 Enter value or search resources... Search icon

Cancel Done

Flow Logic:

Flow Variables are temporary placeholders used within a process or workflow to store and manipulate data as it moves through different stages of execution. These variables enable dynamic data handling, allowing information to be passed from one step to another within a flow.

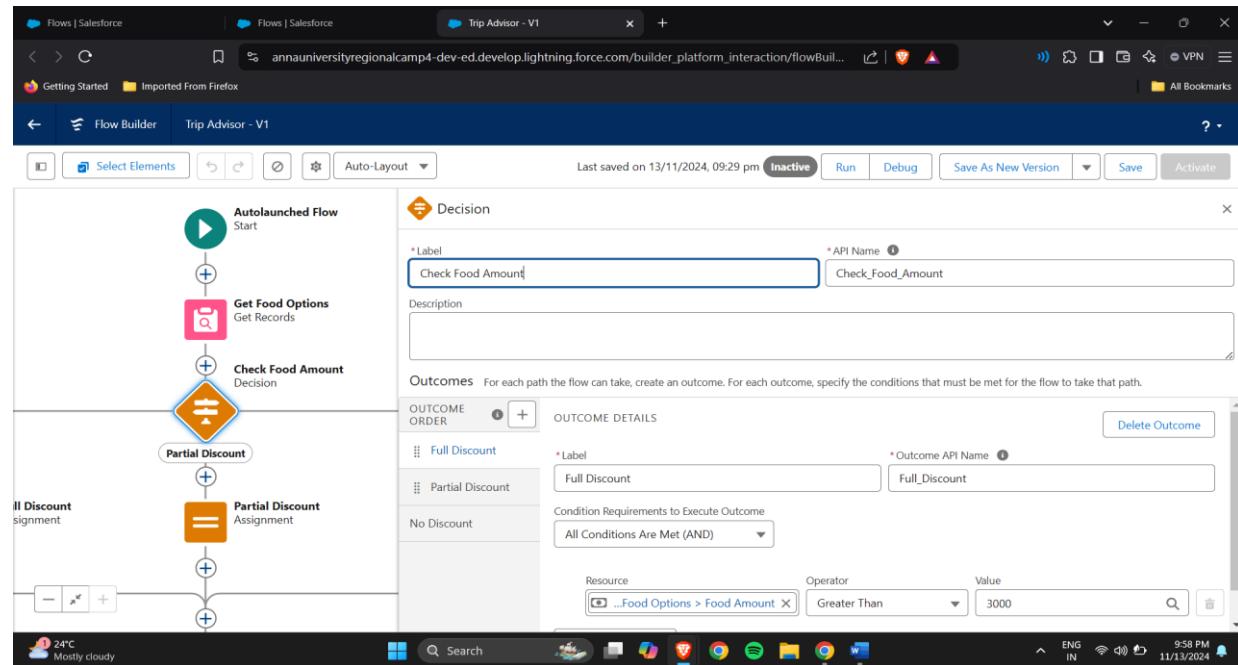
Get Records: Retrieve the necessary customer records.



Purpose:

The "**Get Records**" element in a flow (such as in Salesforce Flow or similar automation platforms) is to retrieve specific records from a database based on defined criteria. This action allows you to fetch data that can be used later in the flow for various purposes, such as updating records, making decisions, or displaying information.

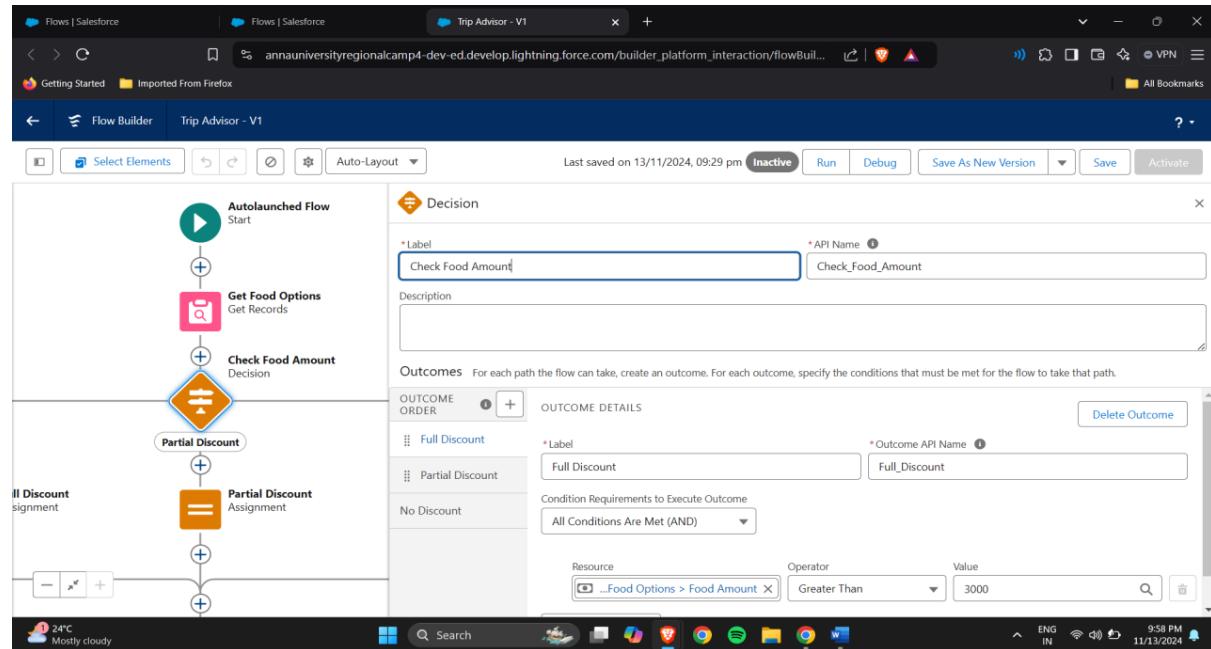
Decision Element: Determine the discount rate based on the purchase amount:



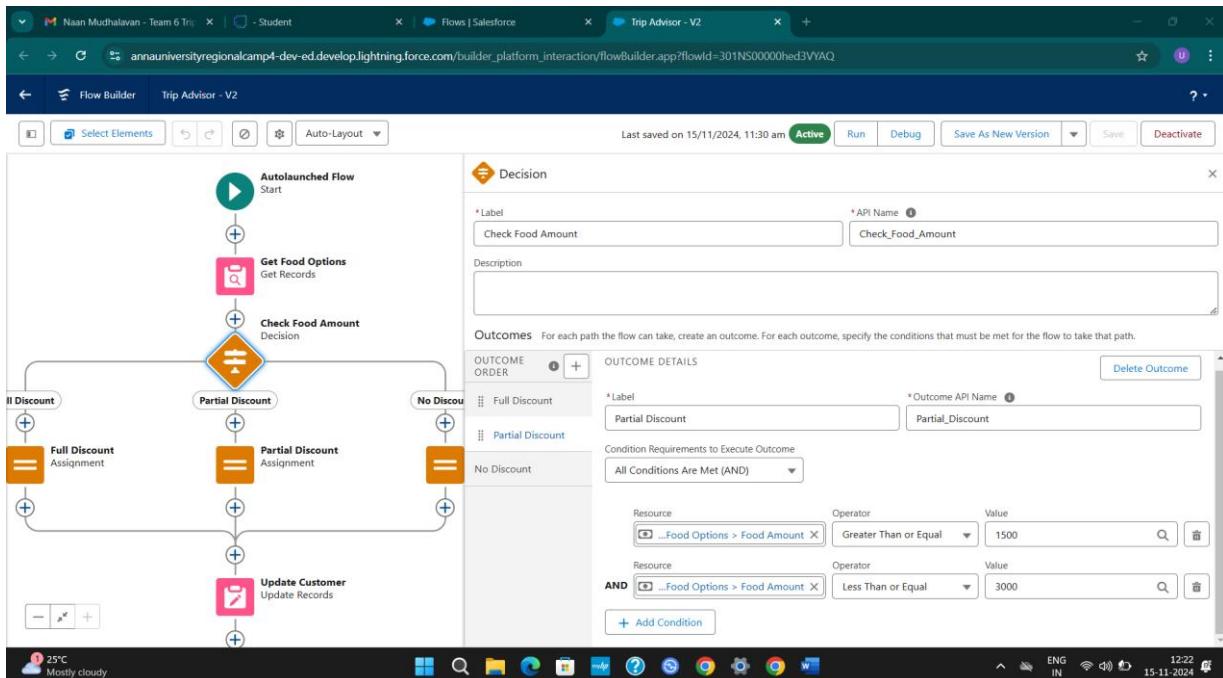
Purpose:

The Decision Element in a flow is used to control the flow's path based on specified conditions. This is particularly useful for creating dynamic, condition-based workflows.

Full Discount: Amount exceeds 3000.



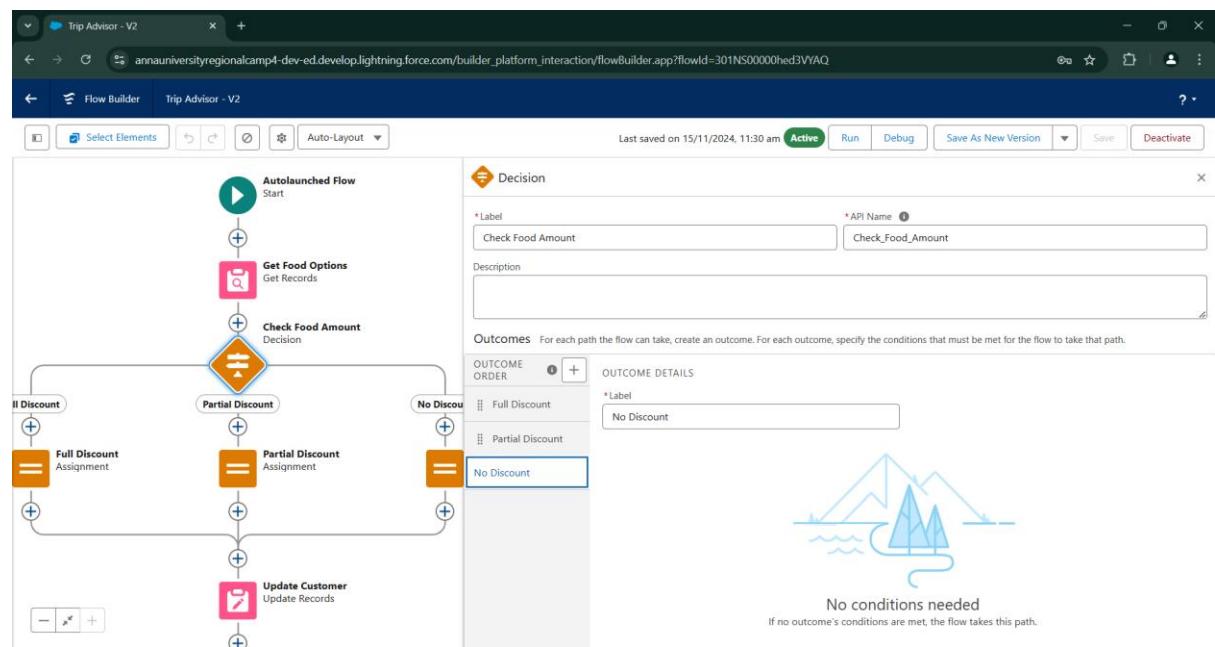
Partial Discount: Amount In-between 1500 to 3000.



Purpose:

The Full Discount in a flow (such as a sales or customer service process) is to Provide Complete Financial Relief, Streamline Issue Resolution, Enhance Customer Loyalty and Support Promotional Strategies.

No Discount:



Purpose:

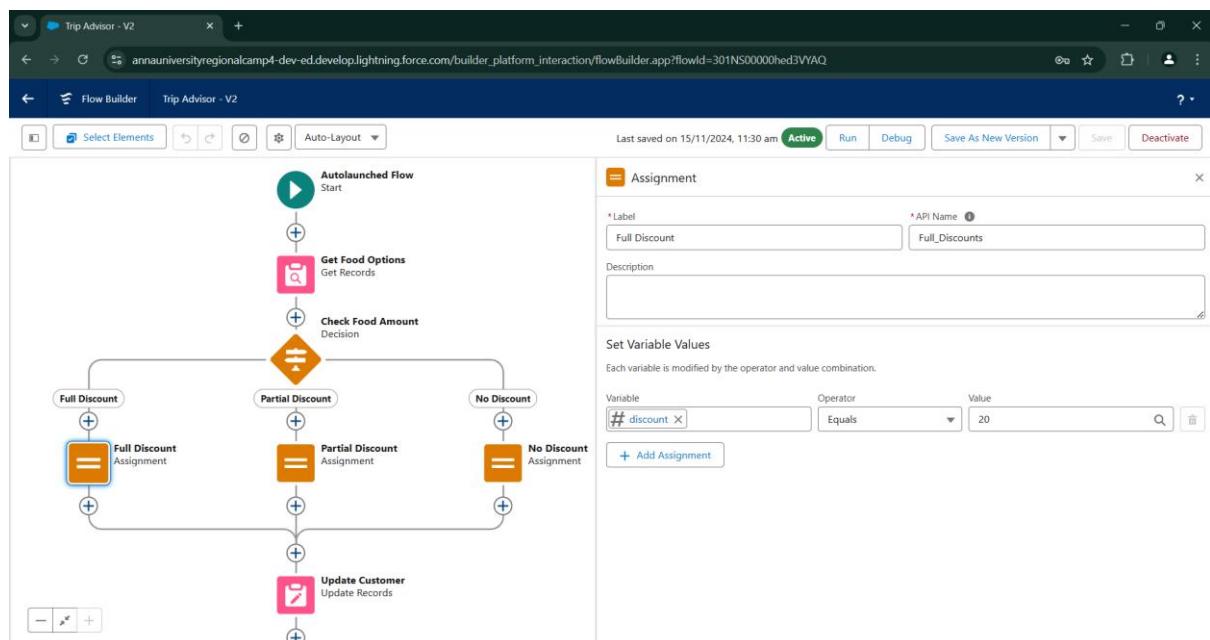
A "No Discount" option in a flow is to provide flexibility in scenarios where certain products, services, or customers may not be eligible for any discount.

Assignment:

Full Discount: Variable > discount

Operator > Equals

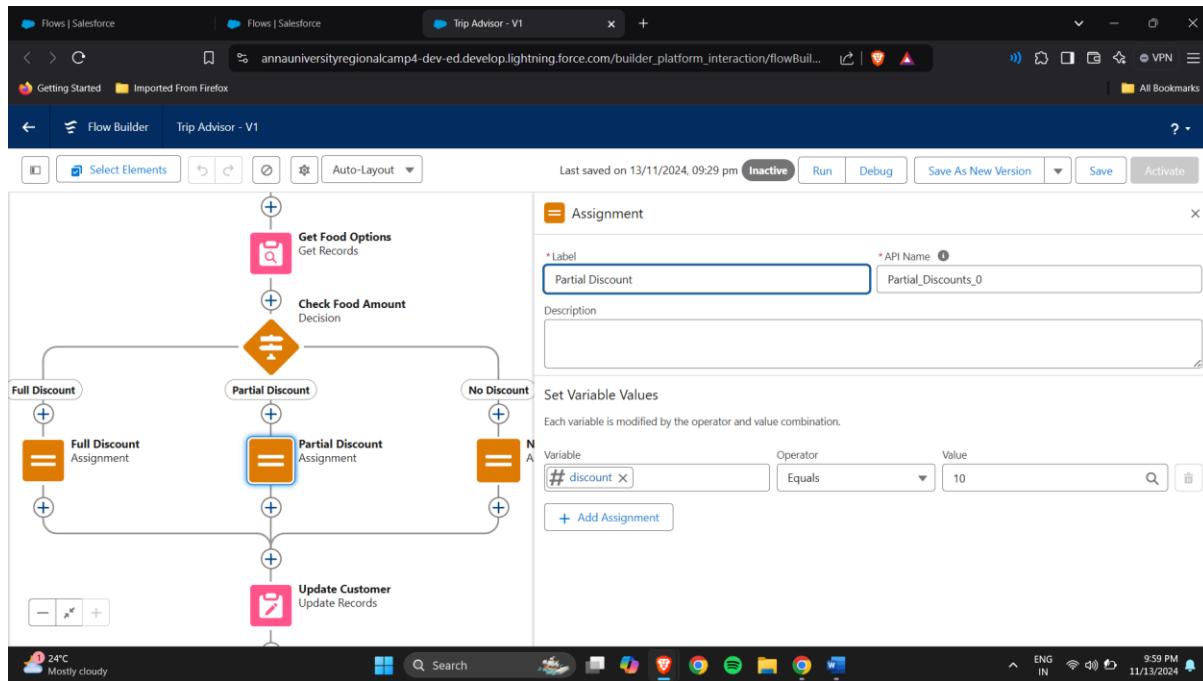
Value > 20



Partial Discount: Variable > discount

Operator > Equals

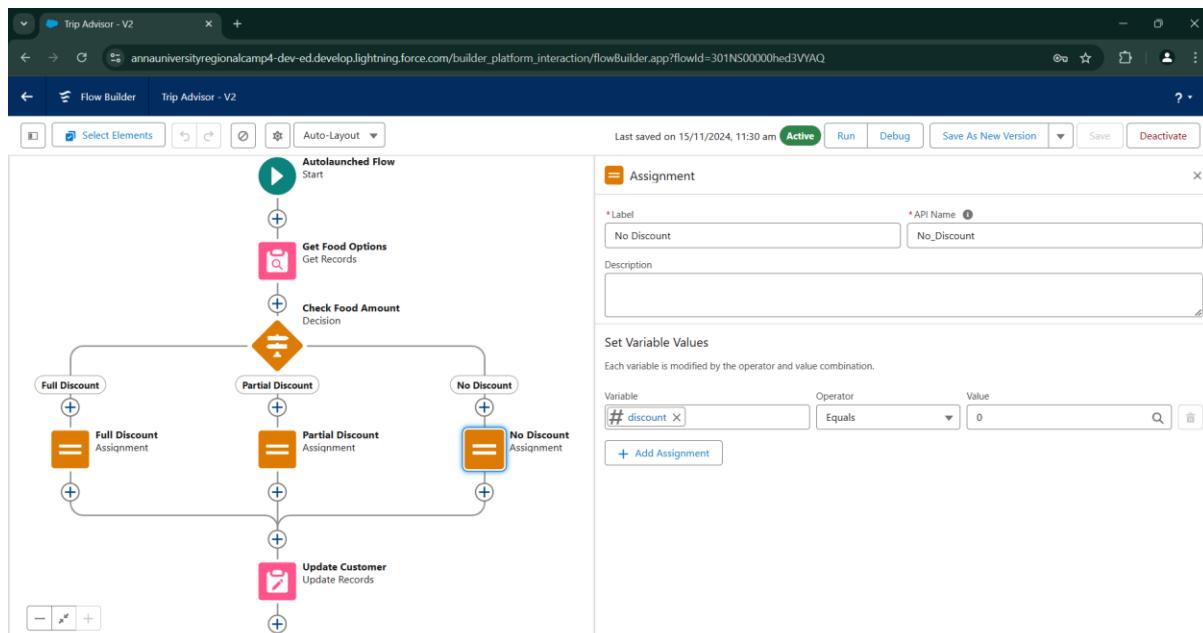
Value > 10



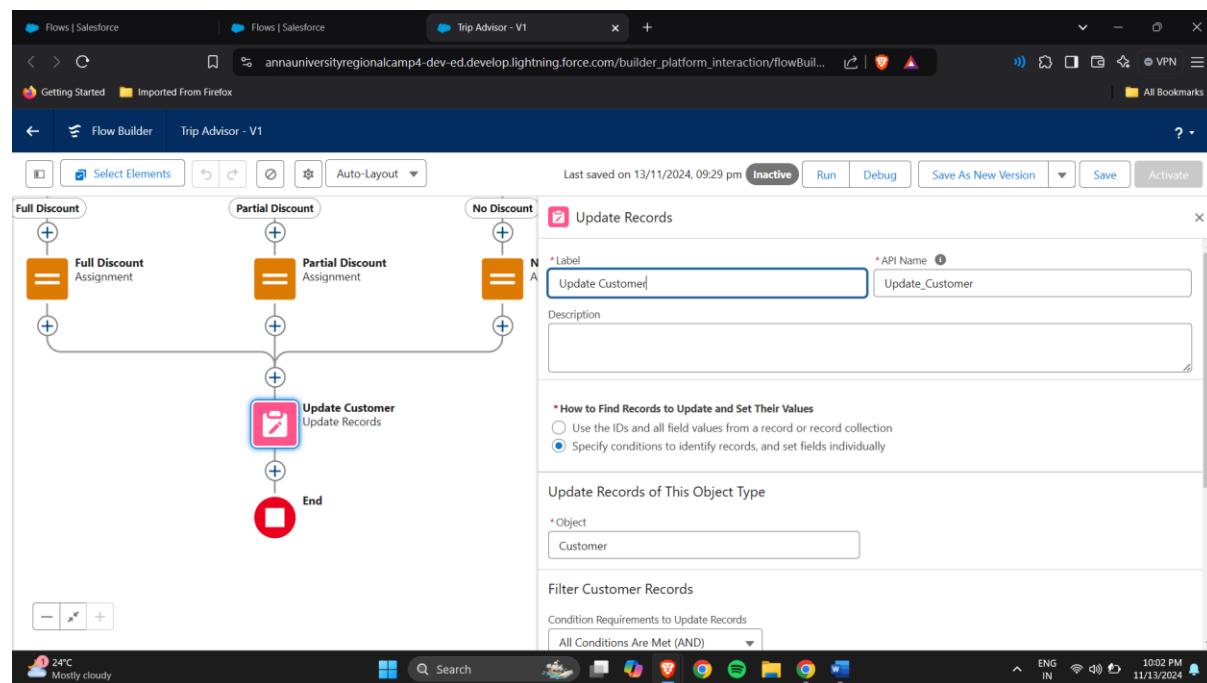
No Discount: Variable > discount

Operator > Equals

Value > 0



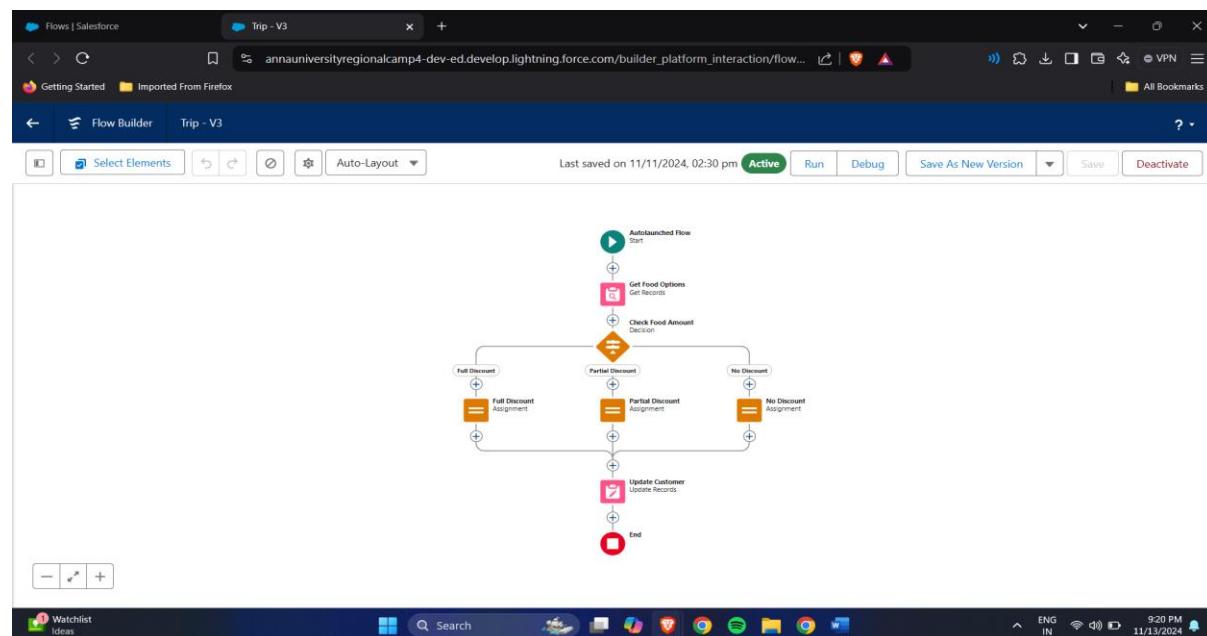
Update Record: Assign discount amounts and update records accordingly.



Purpose:

An "Update Record" element in a flow (commonly in CRM platforms like Salesforce) is to modify existing records in the database based on specified criteria and conditions. This element allows automation of record updates within workflows, saving time and ensuring data consistency.

Final Output of the Flow activate:



Milestone 5 - Tabs:

Tabs in Salesforce provide a user interface for managing and viewing records.

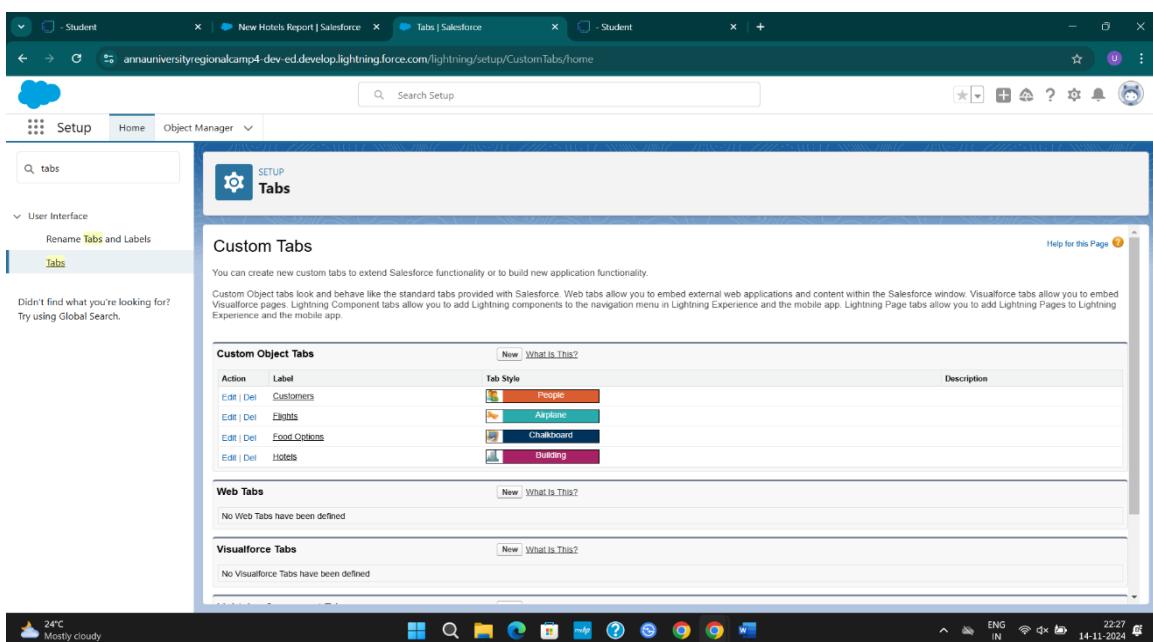
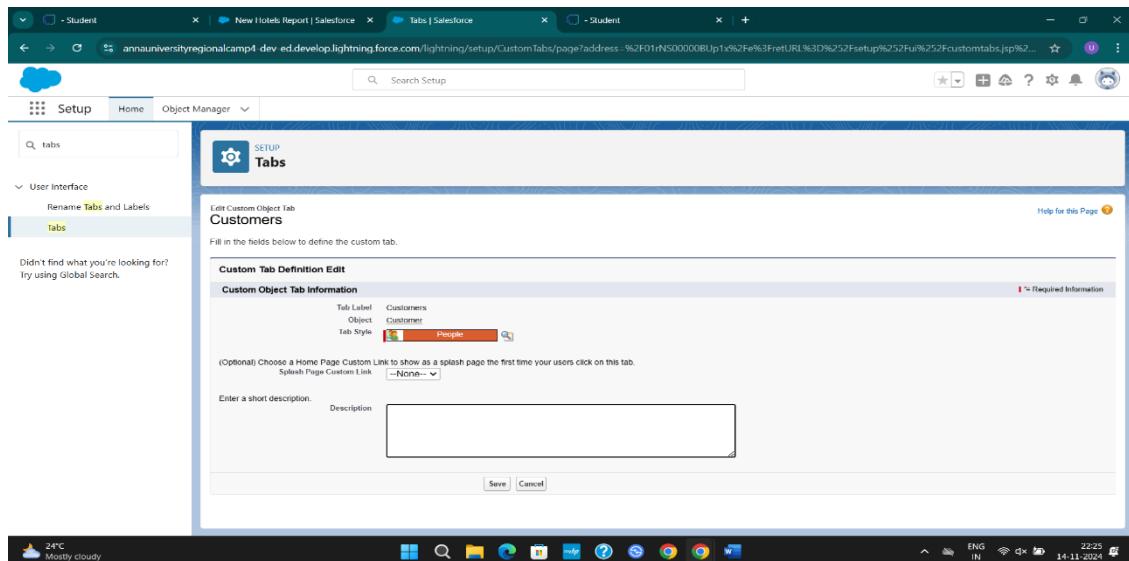
1. Types of Tabs:

- Custom Tabs: Specific to custom objects.
- Web Tabs: Display web content.
- Visualforce Tabs: Display Visualforce pages.
- Lightning Component Tabs: Add Lightning components to the navigation.
- Lightning Page Tabs: Add Lightning Pages to mobile app navigation.

Use Case:

Creating objects and storing TripAdvisor E-management data is the first step to meet their requirements. To enable employees to access stored data efficiently, the admin needs to create dedicated tabs. By designing specific tabs, the organization can enhance the user experience, streamline navigation features. This approach helps employees find and manage data efficiently, supporting better service and operational effectiveness within TripAdvisor E-management

Creating a Custom Tab



- From Setup, search Tabs and select New (Custom Object Tab).
- Choose Opportunity Automobile and complete the setup.

Milestone 6 - Lightning App:

An app is a collection of items that work together to serve a particular function. In Lightning Experience, Lightning apps gives users access to sets of objects, tabs, and other items all in one convenient bundle in the navigation bar.

Lightning apps let you brand your apps with a custom color and logo. You can even include a utility bar and Lightning page tabs in your Lightning app. Members of your org can work more efficiently by easily switching between apps.

Use case:

Well done! You're close to meeting the requirements of TripAdvisor E-Management by creating objects to store the organization's data effectively. However, building a database alone is not enough to fully meet organizational needs. The real challenge lies in ensuring that users within TripAdvisor E-Management can easily access and interact with the objects you've created for them.

As the Admin for TripAdvisor E-Management, it's your responsibility to ensure that every user in the organization has appropriate access to the data modelling structure, enabling them to retrieve, update, and manage the data they need seamlessly. This will help TripAdvisor E-Management operate efficiently and provide users with a smooth experience as they engage with the system.

Activity 1:

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.

App Name	Developer Name	Description	Last Modified Date	Type	Visible
11 Lightning Usage App	LightningInstrumentation	View Adoption and Usage Metrics for Lightning Experience	28/08/2024, 8:57 am	Lightning	
12 Marketing CRM Classic	Marketing	Track sales and marketing efforts with CRM objects.	28/08/2024, 8:57 am	Classic	
13 Platform	Platform	The fundamental Lightning Platform	28/08/2024, 8:57 am	Classic	
14 Queue Management	QueueManagement	Create and manage queues for your business.	28/08/2024, 8:57 am	Lightning	
15 Sales	Sales	The world's most popular sales force automation (SFA) solution	28/08/2024, 8:57 am	Classic	
16 Sales	LightningSales	Manage your sales process with accounts, leads, opportunities, and more	28/08/2024, 8:57 am	Lightning	
17 Sales Console	LightningSalesConsole	(Lightning Experience) Lets sales reps work with multiple records on one screen	28/08/2024, 8:57 am	Lightning	
18 Salesforce Chatter	Chatter	The Salesforce Chatter social network, including profiles and feeds	28/08/2024, 8:57 am	Classic	
19 Salesforce Scheduler S...	LightningScheduler	Set up personalized appointment scheduling.	28/08/2024, 8:59 am	Lightning	
20 Service	Service	Manage customer service with accounts, contacts, cases, and more	28/08/2024, 8:57 am	Classic	
21 Service Console	LightningService	(Lightning Experience) Lets support agents work with multiple records across custo...	28/08/2024, 8:57 am	Lightning	
22 Site.com	Sites	Build pixel-perfect, data-rich websites using the drag-and-drop Site.com applicatio...	28/08/2024, 8:57 am	Classic	
23 Subscription Managem...	RevenueCloudConsole	Get started automating your revenue processes	28/08/2024, 8:57 am	Lightning	
24 TripAdvisor E-Manage...	TripAdvisor_EManagement	An integrated solution for managing trips, enhancing user experience, and ensurin...	13/11/2024, 10:11 pm	Lightning	

2.Fill the app name in app details and branding as follow

App Name: TripAdvisor E-Management.

Developer Name: this will auto be populated

Description: Give a meaningful description

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

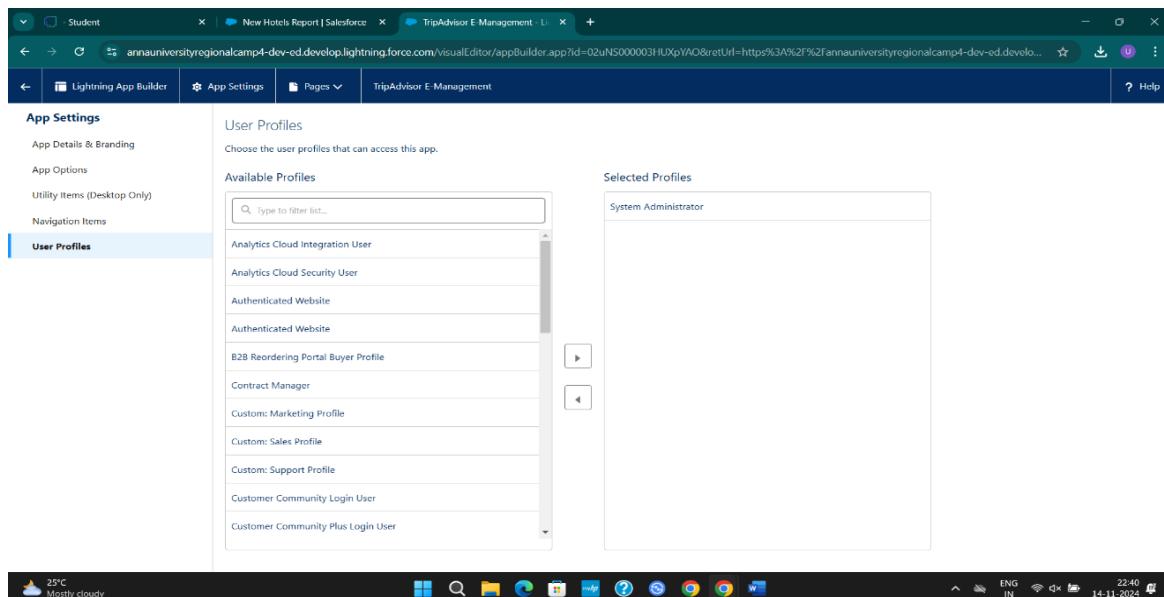
Primary colour hex value: keep this default

The screenshot shows the 'App Details & Branding' section of the Lightning App Builder. The 'App Name' field contains 'TripAdvisor E-Management'. The 'Primary Color Hex Value' is set to '#0070D2'. A preview of the app launcher shows the icon and name. The status bar at the bottom indicates it's 22:39 on 14-11-2024.

3.Then click Next → (App option page) keep it as default → Next → (Utility Items) keep it as default → Next.

The screenshot shows the 'Navigation Items' section of the Lightning App Builder. The 'Selected Items' list includes 'Hotels', 'Food Options', 'Flights', 'Customers', 'Reports', and 'Dashboards'. The status bar at the bottom indicates it's 22:39 on 14-11-2024.

4.To Add Navigation Items:



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

Milestone 7 - OWD Setting:

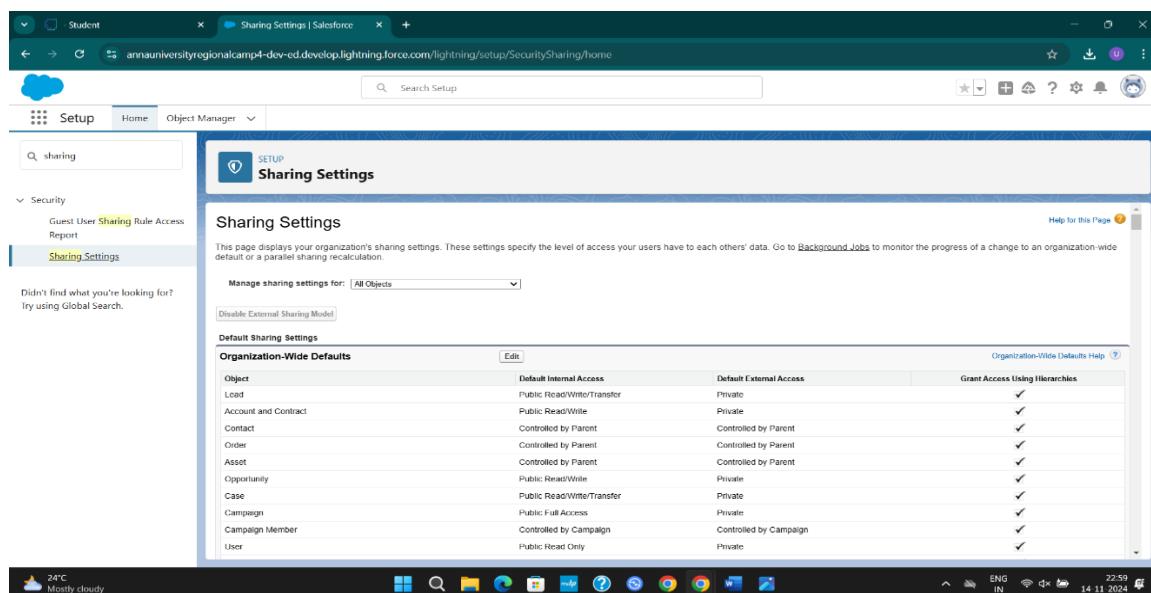
Use case:

TripAdvisor E-Management, **Organization-Wide Defaults (OWDs)** are the foundational security settings that determine access to data across the system. OWDs are used to control who can access specific information within the platform. You can extend or restrict access through additional methods such as sharing rules, role hierarchies, team structures, and account groups, as well as manual sharing options.

Activity 1:

Create OWD Setting

1. Go to Set Up → in the Quick Find box type Sharing Settings → click on it.
2. Click Edit in the Organization-Wide Defaults area.



3. Search for the Employee object.
4. Under default internal access and default external access change the options to “Private” and under grant access using hierarchies select the check box.
5. Click on save.

Customer	Private	Private	<input checked="" type="checkbox"/>
Flight	Public Read/Write	Private	<input checked="" type="checkbox"/>
Food Option	Private	Private	<input checked="" type="checkbox"/>
Hotel	Public Read/Write	Private	<input checked="" type="checkbox"/>

Other Settings

Standard Report Visibility Manager Groups

Buttons: Save | Cancel

6. This Setting is for all the Users Which have been Created.

Milestone 8 - User Adoption:

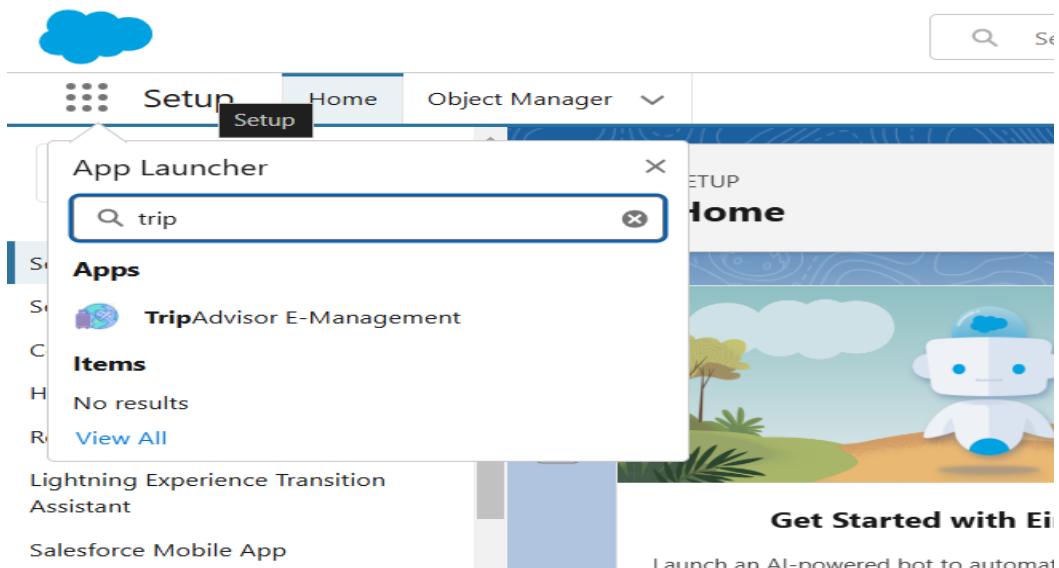
Use Case:

As a new Administrator in the TripAdvisor E-management system, you handle user management tasks such as creating and editing user accounts, resetting passwords, assigning permissions, configuring access to travel data, and more. In this unit, you will learn about managing users and adding them to your TripAdvisor E-management platform.

Activity 1:

Create a Record (Hotel)

1. Click on App Launcher on the left side of the screen.
2. Search TripAdvisor E-Management System & click on it.



3. Click on the Hotel tab.
4. Click New.
5. Fill the Details and click on Save.

Edit Jothi Hotel

* = Required Information

* Hotel Name

Owner
 Umar Ahmed Khan A

TotalFoodOptions

Date



Created By

 Umar Ahmed Khan A, 14/11/2024, 11:02 am

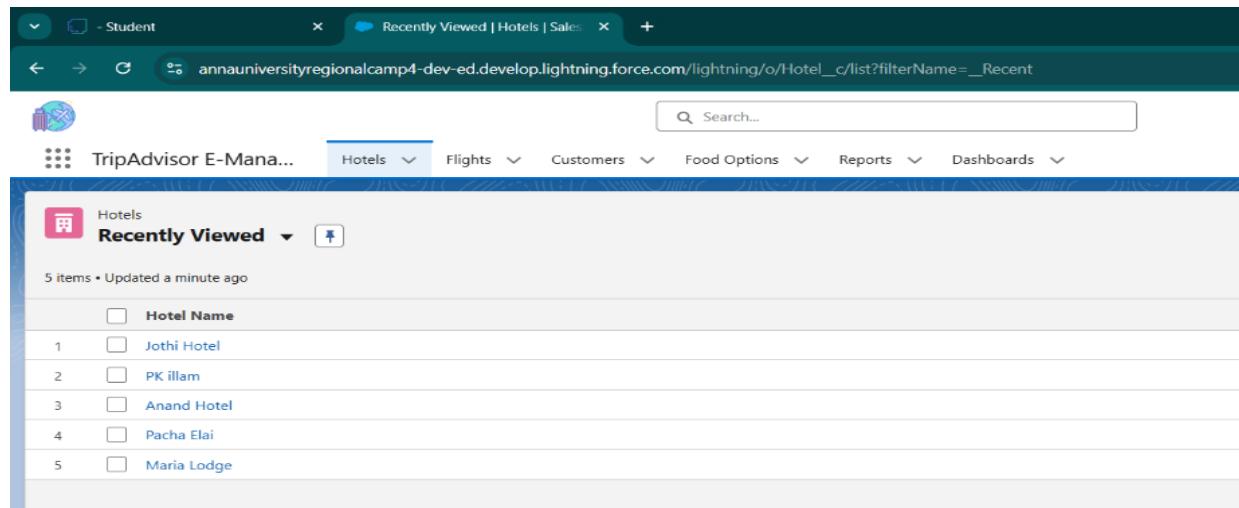
Last Modified By

 Umar Ahmed Khan A, 14/11/2024, 11:24 am

Activity 2:

Create at least 10 records for each of the objects: Hotel, Flight, Customers, Food Options.

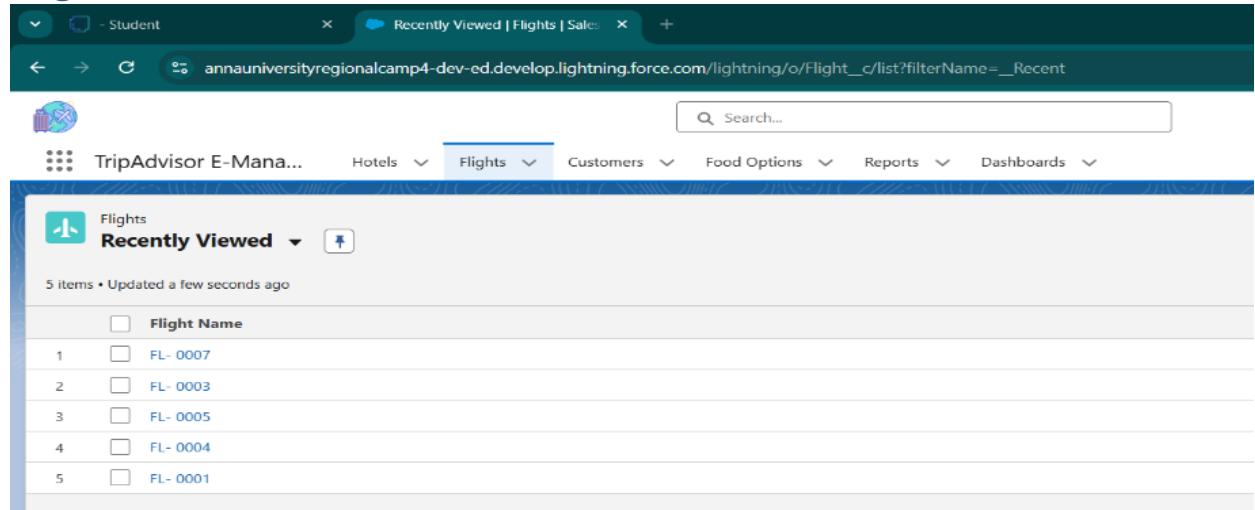
Hotel Records:



The screenshot shows a list of recently viewed hotels. At the top, there is a header with a house icon and the text "Hotels Recently Viewed". Below this, it says "5 items • Updated a minute ago". The list contains five items, each with a checkbox and a link to the hotel's detail page:

- 1 [Jothi Hotel](#)
- 2 [PK illam](#)
- 3 [Anand Hotel](#)
- 4 [Pacha Elai](#)
- 5 [Maria Lodge](#)

Flight Records:



The screenshot shows a list of recently viewed flights. At the top, there is a header with an airplane icon and the text "Flights Recently Viewed". Below this, it says "5 items • Updated a few seconds ago". The list contains five items, each with a checkbox and a link to the flight's detail page:

- 1 [FL- 0007](#)
- 2 [FL- 0003](#)
- 3 [FL- 0005](#)
- 4 [FL- 0004](#)
- 5 [FL- 0001](#)

Customer Records:

The screenshot shows a Salesforce Lightning interface for managing customer records. The top navigation bar includes tabs for 'Student' (selected), 'Recently Viewed | Customers', and a '+' button. Below the bar, a breadcrumb trail shows the current path: 'annauniversityregionalcamp4-dev-ed.develop.lightning.force.com/lightning/o/Customer__c/list?filterName=_Recent'. The main content area features a search bar and a navigation menu with links to 'Hotels', 'Flights', 'Customers' (selected), 'Food Options', 'Reports', and 'Dashboards'. A sidebar on the left displays a 'Recently Viewed' section for 'Customers' with 5 items, updated a few seconds ago. The list includes: 1. Siva, 2. Tamil, 3. Prasanth, 4. Subash, and 5. elayabarathi. Each item is preceded by a small checkbox icon.

Food Options Records:

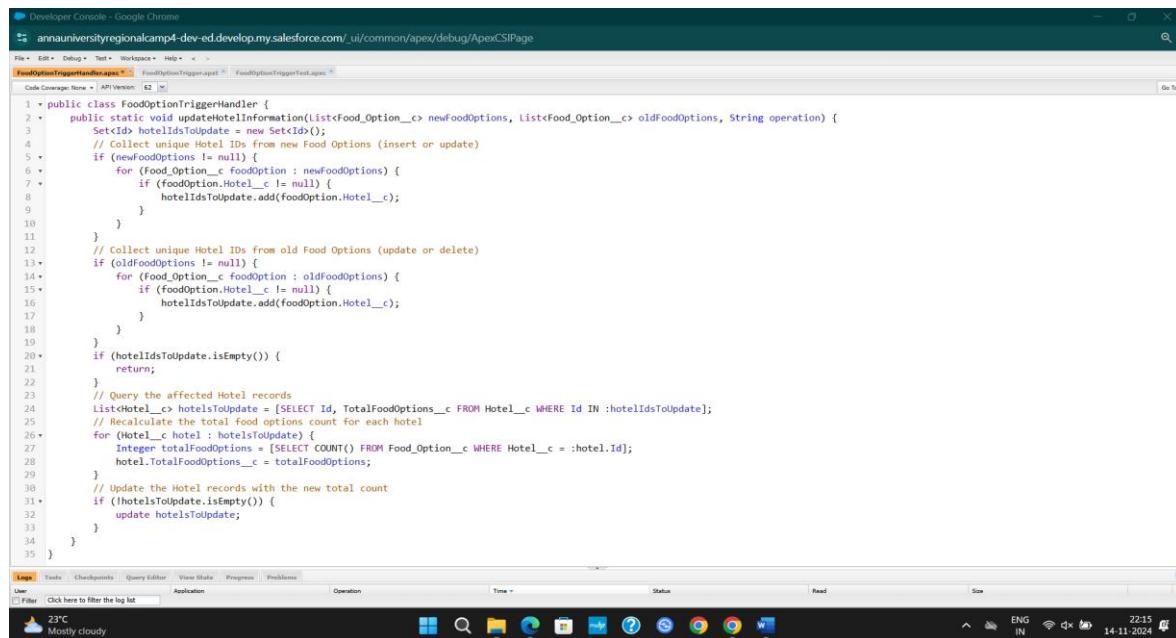
The screenshot shows a Salesforce Lightning interface for managing food options. The top navigation bar includes tabs for 'Student' (selected), 'Recently Viewed | Food Option', and a '+' button. Below the bar, a breadcrumb trail shows the current path: 'annauniversityregionalcamp4-dev-ed.develop.lightning.force.com/lightning/o/Food_Option__c/list?filterName=_Recent'. The main content area features a search bar and a navigation menu with links to 'Hotels', 'Flights', 'Customers', 'Food Options' (selected), 'Reports', and 'Dashboards'. A sidebar on the left displays a 'Recently Viewed' section for 'Food Options' with 33 items, updated a few seconds ago. The list includes: 1. FO - 0033, 2. FO - 0032, 3. FO - 0031, 4. FO - 0030, 5. FO - 0029, 6. FO - 0028, 7. FO - 0027, 8. FO - 0026, 9. FO - 0025, 10. FO - 0024, 11. FO - 0023, 12. FO - 0022, and 13. FO - 0021. Each item is preceded by a small checkbox icon.

Milestone 9 - Apex Trigger:

Apex Trigger for Food Options Management

An Apex Trigger was implemented to update hotel information whenever a new food option is added or updated. This ensures the hotel's total food options count reflects all associated food options.

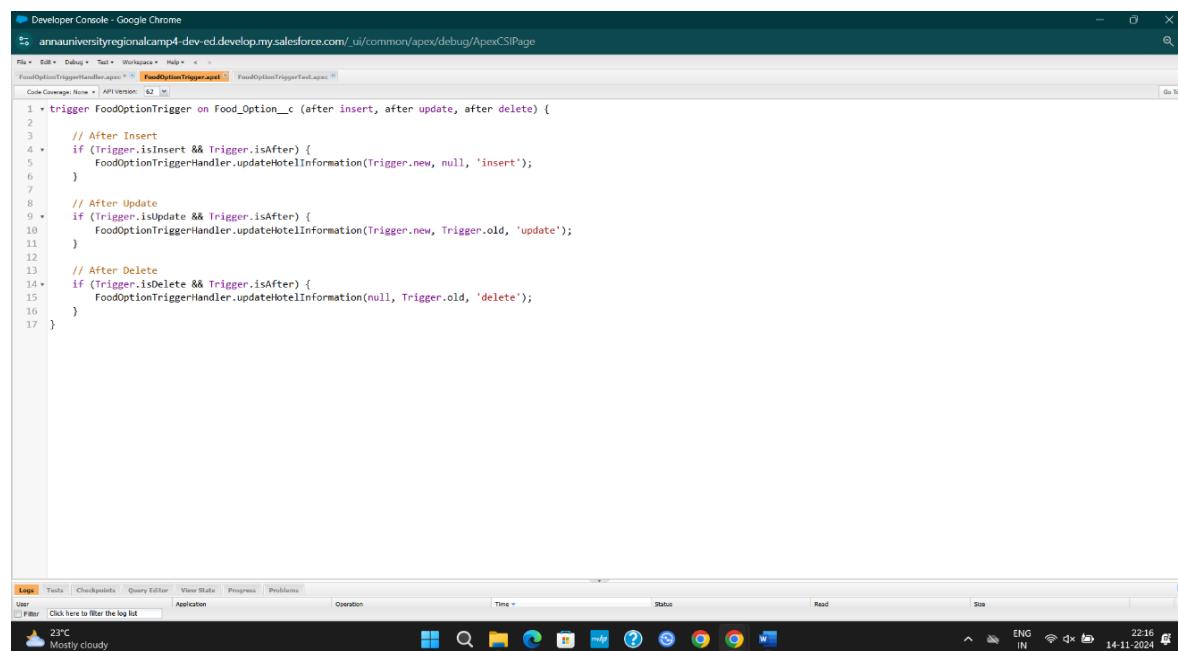
Apex Trigger Handler:



```
1 public class FoodOptionTriggerHandler {
2     public static void updateHotelInformation(List<Food_Option__c> newFoodOptions, List<Food_Option__c> oldFoodOptions, String operation) {
3         Set<Id> hotelIdsToUpdate = new Set<Id>();
4         // Collect unique Hotel IDs from new Food Options (insert or update)
5         if (newFoodOptions != null) {
6             for (Food_Option__c foodOption : newFoodOptions) {
7                 if (foodOption.Hotel__c != null) {
8                     hotelIdsToUpdate.add(foodOption.Hotel__c);
9                 }
10            }
11        }
12        // Collect unique Hotel IDs from old Food Options (update or delete)
13        if (oldFoodOptions != null) {
14            for (Food_Option__c foodOption : oldFoodOptions) {
15                if (foodOption.Hotel__c != null) {
16                    hotelIdsToUpdate.add(foodOption.Hotel__c);
17                }
18            }
19        }
20        if (hotelIdsToUpdate.isEmpty()) {
21            return;
22        }
23        // Query the affected Hotel records
24        List<Hotel__c> hotelsToUpdate = [SELECT Id, TotalFoodOptions__c FROM Hotel__c WHERE Id IN :hotelIdsToUpdate];
25        // Recalculate the total food options count for each hotel
26        for (Hotel__c hotel : hotelsToUpdate) {
27            Integer totalFoodOptions = [SELECT COUNT() FROM Food_Option__c WHERE Hotel__c = :hotel.Id];
28            hotel.TotalFoodOptions__c = totalFoodOptions;
29        }
30        // Update the Hotel records with the new total count
31        if (!hotelsToUpdate.isEmpty()) {
32            update hotelsToUpdate;
33        }
34    }
35 }
```

An **Apex Trigger Handler** is a design pattern used to organize and manage the logic of an Apex trigger. It helps in maintaining clean, reusable, and easily maintainable code. Instead of placing the logic directly within the trigger, it delegates it to a handler class

Trigger:

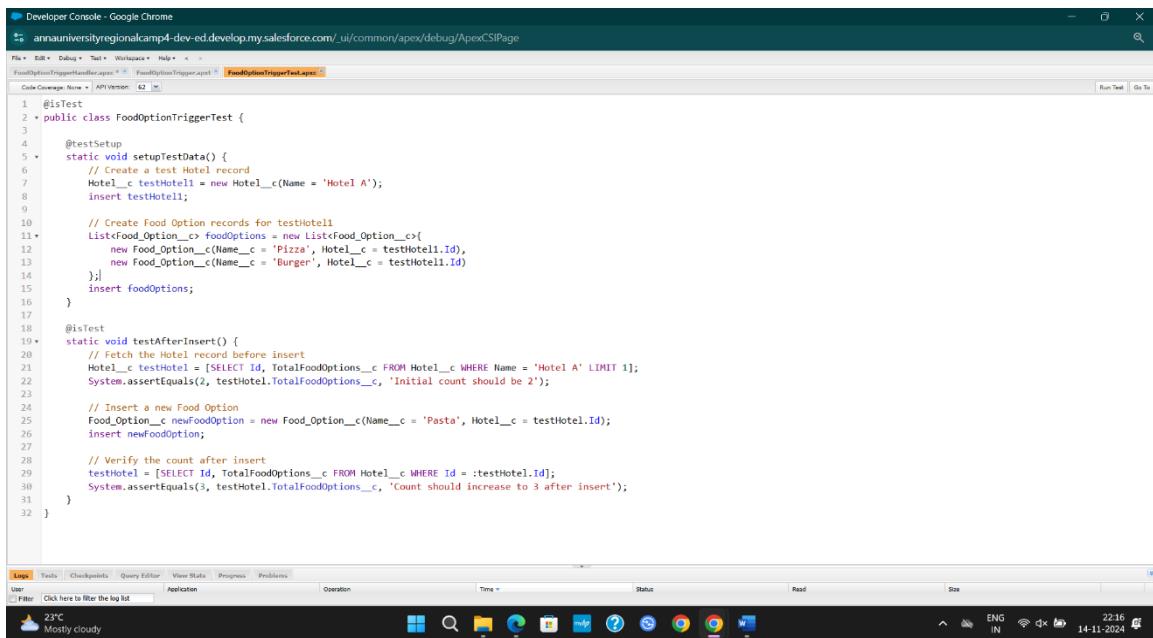


```
1 trigger FoodOptionTrigger on Food_Option__c (after insert, after update, after delete) {
2
3     // After Insert
4     if (Trigger.isInsert && Trigger.isAfter) {
5         FoodOptionTriggerHandler.updateHotelInformation(Trigger.new, null, 'insert');
6     }
7
8     // After Update
9     if (Trigger.isUpdate && Trigger.isAfter) {
10        FoodOptionTriggerHandler.updateHotelInformation(Trigger.new, Trigger.old, 'update');
11    }
12
13     // After Delete
14     if (Trigger.isDelete && Trigger.isAfter) {
15        FoodOptionTriggerHandler.updateHotelInformation(null, Trigger.old, 'delete');
16    }
17 }
```

A **Trigger** in Salesforce is a piece of code that automatically executes (or "fires") when a specific event occurs on a record in Salesforce, such as creating, updating, or deleting a record.

It allows developers to add custom logic to standard operations, providing more control over data and business processes.

Test Trigger:



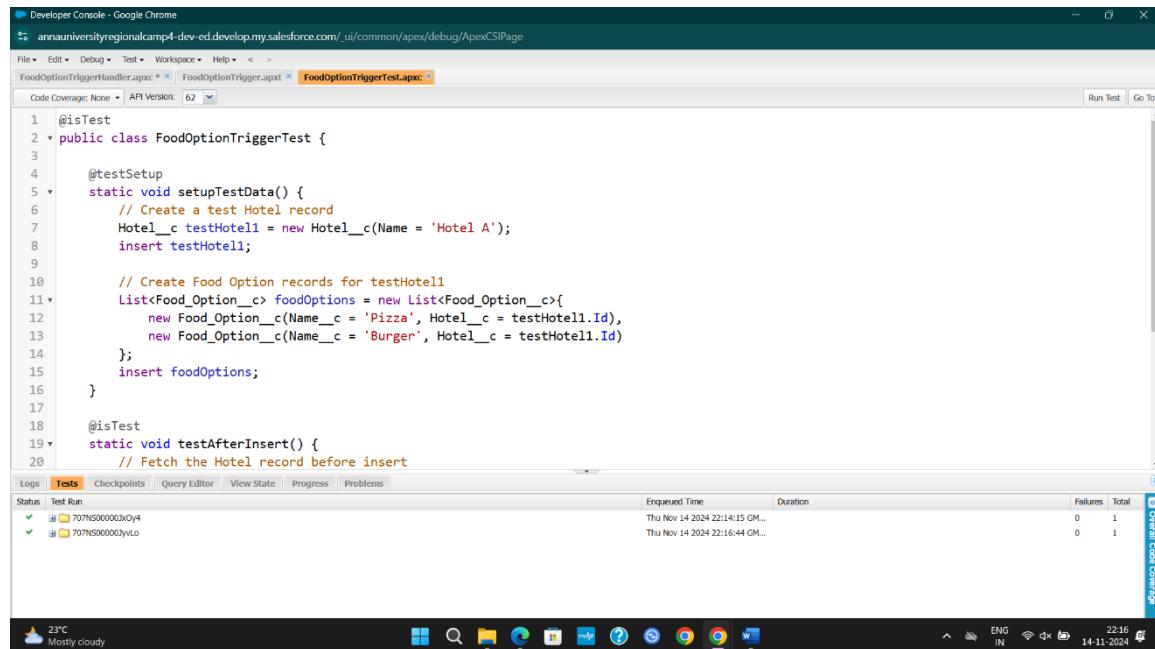
```

1  @isTest
2  +public class FoodOptionTriggerTest {
3
4      @testSetup
5      static void setupTestData() {
6          // Create a test Hotel record
7          Hotel__c testHotel1 = new Hotel__c(Name = 'Hotel A');
8          insert testHotel1;
9
10         // Create Food Option records for testHotel1
11         List<Food_Option__c> foodOptions = new List<Food_Option__c>{
12             new Food_Option__c(Name__c = 'Pizza', Hotel__c = testHotel1.Id),
13             new Food_Option__c(Name__c = 'Burger', Hotel__c = testHotel1.Id)
14         };
15         insert foodOptions;
16     }
17
18     @isTest
19     static void testAfterInsert() {
20         // Fetch the Hotel record before insert
21         Hotel__c testHotel = [SELECT Id, TotalFoodOptions__c FROM Hotel__c WHERE Name = 'Hotel A' LIMIT 1];
22         System.assertEquals(2, testHotel.TotalFoodOptions__c, 'Initial count should be 2');
23
24         // Insert a new Food Option
25         Food_Option__c newFoodOption = new Food_Option__c(Name__c = 'Pasta', Hotel__c = testHotel.Id);
26         insert newFoodOption;
27
28         // Verify the count after insert
29         testHotel = [SELECT Id, TotalFoodOptions__c FROM Hotel__c WHERE Id = :testHotel.Id];
30         System.assertEquals(3, testHotel.TotalFoodOptions__c, 'Count should increase to 3 after insert');
31     }
32 }

```

A **Test Trigger** in Salesforce is used to validate that the trigger behaves as expected under different conditions. It is written using Apex test methods to simulate various scenarios, ensuring that triggers perform the correct operations, like inserting, updating, or deleting records.

Test case Result:



Status	Test Run	Enqueued Time	Duration	Failures	Total
✓	707NS00000D0y4	Thu Nov 14 2024 22:14:15 GM...		0	1
✓	707NS00000DyvL0	Thu Nov 14 2024 22:16:44 GM...		0	1

A "Test Trigger case run successfully" means that a trigger (an automated process or function) was executed, and it completed without errors or failures.

- **Trigger:** A piece of code that runs automatically in response to specific events (like creating, updating, or deleting a record).
- **Test Case:** A scenario designed to verify that the trigger works as expected under certain conditions.

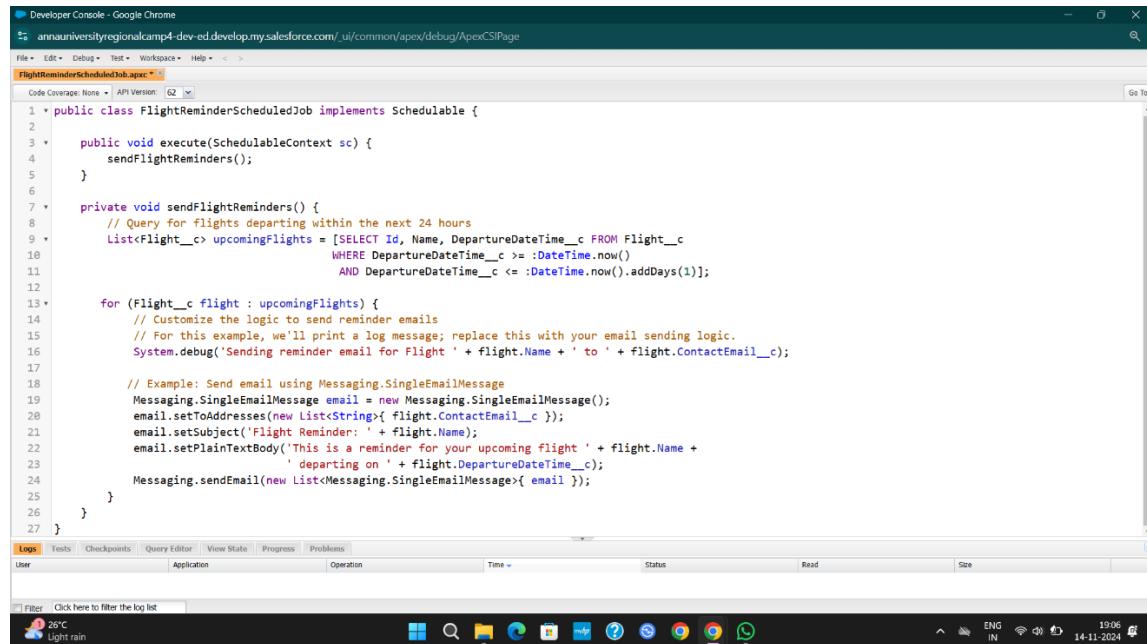
- **Successful Run:** The trigger was executed correctly, and the desired results were achieved, with no issues encountered during testing (e.g., data was updated correctly, no errors occurred).

Milestone 10 - Apex Scheduled:

Apex Scheduled Class for Flight Reminders

The Apex Schedulable class, FlightReminderScheduledJob, was created to send reminder emails to customers who have booked flights, 24 hours prior to departure.

Scheduled Class Code:



```

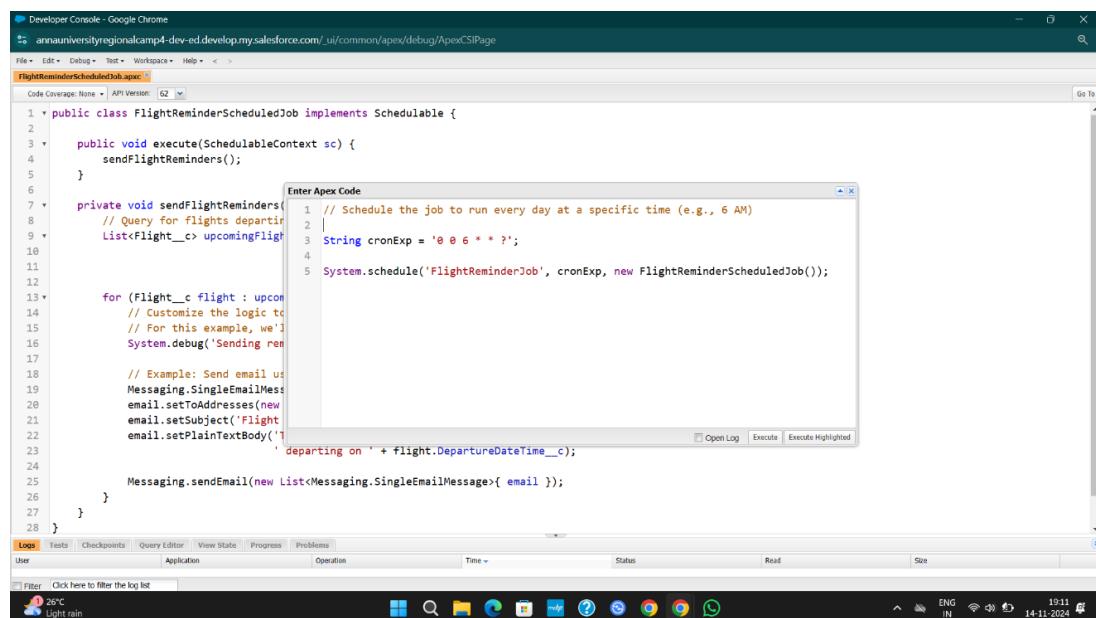
1 * public class FlightReminderScheduledJob implements Schedulable {
2
3     public void execute(SchedulableContext sc) {
4         sendFlightReminders();
5     }
6
7     private void sendFlightReminders() {
8         // Query for flights departing within the next 24 hours
9         List<Flight__c> upcomingFlights = [SELECT Id, Name, DepartureDateTime__c FROM Flight__c
10                                         WHERE DepartureDateTime__c >= :DateTime.now()
11                                         AND DepartureDateTime__c <= :DateTime.now().addDays(1)];
12
13         for (Flight__c flight : upcomingFlights) {
14             // Customize the logic to send reminder emails
15             // For this example, we'll print a log message; replace this with your email sending logic.
16             System.debug('Sending reminder email for Flight ' + flight.Name + ' to ' + flight.ContactEmail__c);
17
18             // Example: Send email using Messaging.SingleEmailMessage
19             Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
20             email.setToAddresses(new List<String>{ flight.ContactEmail__c });
21             email.setSubject('Flight Reminder: ' + flight.Name);
22             email.setPlainTextBody('This is a reminder for your upcoming flight ' + flight.Name +
23                                   ' departing on ' + flight.DepartureDateTime__c);
24             Messaging.sendEmail(new List<Messaging.SingleEmailMessage>{ email });
25         }
26     }
27 }

```

The screenshot shows the Salesforce Developer Console with the code for FlightReminderScheduledJob.apc. The code implements the Schedulable interface and contains a private method sendFlightReminders() which queries for flights departing within the next 24 hours and sends a reminder email using the Messaging.SingleEmailMessage class. The developer console interface includes tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems, along with a status bar at the bottom.

A **Scheduled Class** in platforms like Salesforce allows you to automate and schedule the execution of Apex classes at specified times or intervals. It is particularly useful for recurring tasks, such as sending emails, updating records, or integrating external systems, without manual intervention.

Scheduling the Job:



The screenshot shows the Salesforce Developer Console with the code for FlightReminderScheduledJob.apc. A modal dialog titled "Enter Apex Code" is open, containing the cron expression for scheduling the job: "String cronExp = '0 0 6 * *';". Below the code, there are "Open Log" and "Execute" buttons. The developer console interface includes tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems, along with a status bar at the bottom.

Scheduling a Job Code refers to setting up an automated process to run at a specific time or interval without manual intervention. This is commonly used in systems like CRM platforms, databases, or job scheduling tools. The purpose is to perform repetitive tasks (e.g., data updates, reports generation) at predefined times.

Output of the job Scheduling:

Developer Console - Google Chrome

File Edit Debug Test Workspace Help < > Go To

FlightReminderScheduledJob.apc

Code Coverage: None API Version: G2

```
1 public class FlightReminderScheduledJob implements Scheduler {
2
3     public void execute(SchedulableContext sc) {
4         sendFlightReminders();
5     }
6
7     private void sendFlightReminders() {
8         // Query for flights departing within the next 24 hours
9         List<Flight__c> upcomingFlights = [SELECT Id, Name, DepartureDateTime__c FROM Flight__c
10                                         WHERE DepartureDateTime__c >= :DateTime.now()
11                                         AND DepartureDateTime__c <= :DateTime.now().addDays(1)];
12
13     for (Flight__c flight : upcomingFlights) {
14         // Customize the logic to send reminder emails
15         // For this example, we'll print a log message; replace this with your email sending logic.
16         System.debug('Sending reminder email for Flight ' + flight.Name + ' to ' + flight.ContactEmail__c);
17
18         // Example: Send email using Messaging.SingleEmailMessage
19         Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
20         email.setToAddresses(new List<String>{ flight.ContactEmail__c });
21         email.setSubject('Flight Reminder: ' + flight.Name);
22         email.setPlainTextBody('This is a reminder for your upcoming flight ' + flight.Name +
23                               ' departing on ' + flight.DepartureDateTime__c);
24     }
25 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
Umar Ahmed Khan A	Unknown	/services/data/v62.0/tooling/executeAnonymous	11/14/2024, 7:12:54 PM	Success		3.24 KB

Filter Click here to filter the log list

1 26°C Light rain

ENG IN 19:12 14-Nov-2024

A Scheduled Job Code is a task or process that runs automatically at predefined times or intervals. When it runs successfully, it means the scheduled job has executed without errors, completing the intended task as per the schedule.

All job scheduled has been updated:

The screenshot shows the Salesforce Setup interface with the 'Scheduled Jobs' tab selected. The main content area displays a table of scheduled jobs, including their names, submitter, status, and cron trigger ID. A sidebar on the left lists various setup categories like Feature Settings, Sales, Products, and Jobs.

Action	Job Name	Submitted By	Submitted	Started	Next Scheduled Run	Type	Cron Trigger ID
Manage Del Pause Job	FlightReminderJob	A. Umar Ahmed Khan	14/11/2024, 7:12 pm		15/11/2024, 6:00 am	Scheduled Apex	08eNS00000039jP
Del	Metalytics Data Loader Job for Org 00D0N000002eiJU	User_Interaction	28/08/2024, 8:59 am	14/11/2024, 3:47 am	15/11/2024, 3:47 am	Autonomous Data Loader Job	08eNS000000yxtz
	Program Milestone Computation Cron Job	Process, Automated	28/08/2024, 8:59 am	14/11/2024, 1:29 pm	14/11/2024, 8:29 pm	Program Milestone Computation Cron Job	08eNS000000yxxt
	Program Status Update Cron Job	Process, Automated	28/08/2024, 8:59 am	14/11/2024, 6:30 pm	15/11/2024, 9:30 am	Program Status Update Cron Job	08eNS000000yxtx

"All jobs scheduled have been updated" means that the tasks or processes that were planned or set to run at specific times have been modified or refreshed. This could involve changing the timing, parameters, or details of the scheduled jobs to ensure they align with new requirements, improve efficiency, or reflect updated information.

Milestone 11 - Reports:

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

Types of Reports in Salesforce

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

Use Case:

The CEO of TripAdvisor wants a concise overview of employee activities, current projects, project progress, and assigned assets, along with the condition of these assets. This data will provide a clear snapshot of the organization's status, enabling informed decision-making. The CEO has tasked you with presenting this information in an organized, easily interpretable format.

Let's create a Report.

Activity 1:

Create Report

1. Go to the app → click on the reports tab
2. Click New Report
3. Select report type from category or from report type panel or from search panel → click on start report.

The screenshot shows the 'Create Report' screen in Salesforce. On the left, there is a sidebar with a 'Category' section containing 'Recently Used' and 'All' (which is selected). Below this are various report categories: Accounts & Contacts, Opportunities, Customer Support Reports, Leads, Campaigns, Activities, Contracts and Orders, Price Books, Products and Assets, Administrative Reports, File and Content Reports, and Individuals. In the center, there is a search bar labeled 'Select a Report Type' with the placeholder 'Search results for hotel'. Below the search bar is a table with columns 'Report Type Name' and 'Category'. Two rows are visible: 'Hotels' under 'Standard' and 'Food Options with Hotel' under 'Standard'. On the right, there is a 'Details' pane. It shows a summary for 'Hotels' (Standard Report Type), a 'Start Report' button, and sections for 'Created By You' (listing 'New Hotels Report'), 'Created By Others' (listing 'No Reports Yet'), and 'Objects Used in Report Type' (listing 'Owner').

4. Customize your report and add fields from left pane as shown below
5. Save or run it.

REPORT ▾

New Hotels Report ▾ Hotels

Outline ▾ Filters 1

Groups

GROUP ROWS

Add group... ▾

Columns

Add column... ▾

Hotel: Hotel Name ▾

TotalFoodOptions ▾

Date ▾

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

	Hotel: Hotel Name	TotalFoodOptions	Date
1	Maria Lodge	7	14/11/2024
2	PK illam	6	14/11/2024
3	Jothi Hotel	8	14/11/2024
4	Pacha Elai	6	14/11/2024
5	Anand Hotel	6	14/11/2024
6		33	

Reports for Hotel, Flight, Customer, and Food Option have been created using a standardized method, streamlining data management and analysis for improved decision-making and business operations.

New Hotel Report:

Report: Hotels

New Hotels Report

Total Records 5 Total TotalFoodOptions 33

	Hotel: Hotel Name	TotalFoodOptions	Date
1	Maria Lodge	7	14/11/2024
2	PK illam	6	14/11/2024
3	Jothi Hotel	8	14/11/2024
4	Pacha Elai	6	14/11/2024
5	Anand Hotel	6	14/11/2024
6		33	

A new hotel report has been created, streamlining data management and providing valuable insights for decision-making.

New Flight Report:

Report: Flights

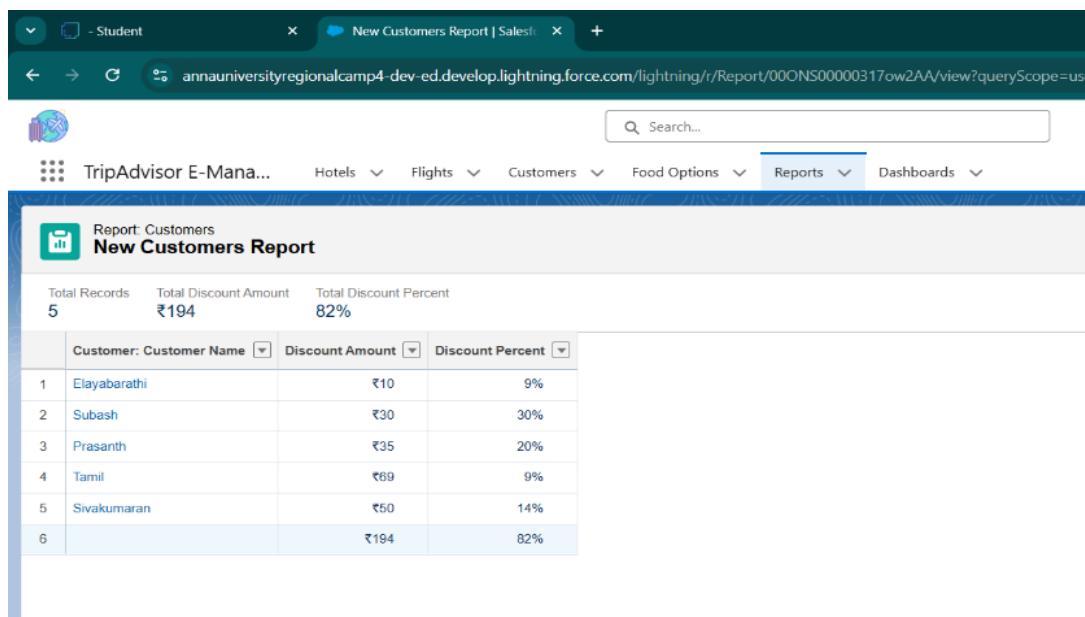
New Flights Report

Total Records 5

	Flight: Flight Name	ContactEmail	Name	DepartureDateTime
1	FL- 0001	umarahmedkhan29@gmail.com	Jothi Hotel	15/11/2024, 12:00 pm
2	FL- 0003	sac@gmail.com	Maria Lodge	15/11/2024, 12:00 pm
3	FL- 0004	dam143@gmail.com	Anand Hotel	16/11/2024, 12:00 pm
4	FL- 0005	pachasat231@gmail.com	Pacha Elai	17/11/2024, 10:30 pm
5	FL- 0007	pkilai221@gmail.com	PK illam	18/11/2024, 11:00 am

A new flight report has been created, providing updated and relevant data for better analysis and decision-making.

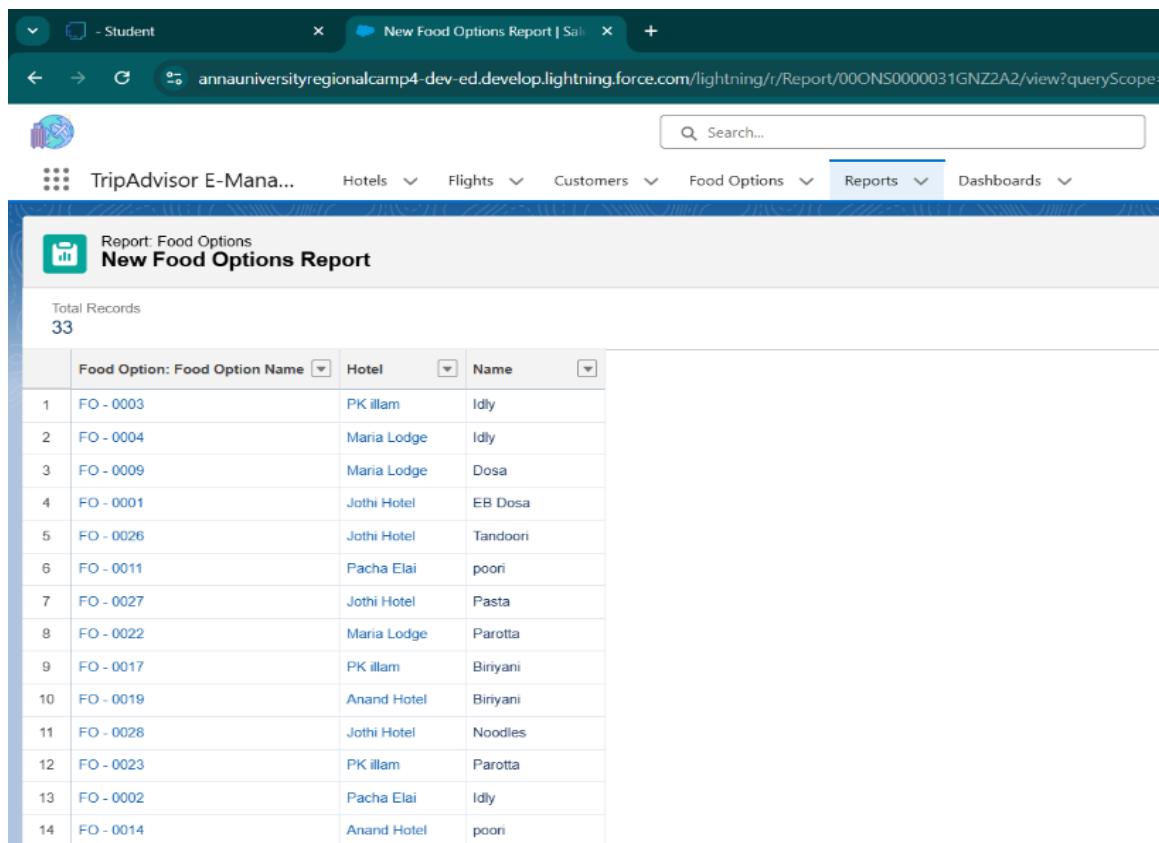
New Customers Report:



	Customer: Customer Name	Discount Amount	Discount Percent
1	Elayabarathi	₹10	9%
2	Subash	₹30	30%
3	Prasanth	₹35	20%
4	Tamil	₹69	9%
5	Sivakumaran	₹50	14%
6		₹194	82%

A new customer report has been created to manage and analyse customer data efficiently.

New Food Options Report:



	Food Option: Food Option Name	Hotel	Name
1	FO - 0003	PK illam	Idly
2	FO - 0004	Maria Lodge	Idly
3	FO - 0009	Maria Lodge	Dosa
4	FO - 0001	Jothi Hotel	EB Dosa
5	FO - 0026	Jothi Hotel	Tandoori
6	FO - 0011	Pacha Elai	poori
7	FO - 0027	Jothi Hotel	Pasta
8	FO - 0022	Maria Lodge	Parotta
9	FO - 0017	PK illam	Biryani
10	FO - 0019	Anand Hotel	Biryani
11	FO - 0028	Jothi Hotel	Noodles
12	FO - 0023	PK illam	Parotta
13	FO - 0002	Pacha Elai	Idly
14	FO - 0014	Anand Hotel	poori

A new Food Option report has been created to streamline the management and analysis of food-related data.

Total Reports

The screenshot shows a web browser window titled "Recent | Reports | Salesforce" with the URL "annauniversityregionalcamp4-dev-ed.lightning.force.com/lightning/o/Report/home?queryScope=mru". The page is titled "TripAdvisor E-Mana..." and has a navigation bar with links for Hotels, Flights, Customers, Food Options, Reports, and Dashboards. On the left, there's a sidebar with sections for Reports (Recent, Created by Me, Private Reports, Public Reports, All Reports), Folders (All Folders), and a search bar for "Search...". The main content area displays a table of recent reports:

Report Name	Description	Folder	Created By	Created On	Subscribed
New Hotels Report		Trip	Umar Ahmed Khan A	14/11/2024, 11:58 am	
New Flights Report		Trip	Umar Ahmed Khan A	14/11/2024, 12:02 pm	
New Customers Report		Trip	Umar Ahmed Khan A	14/11/2024, 12:04 pm	
New Food Options Report		Trip	Umar Ahmed Khan A	14/11/2024, 12:05 pm	

Milestone 12 - Dashboards:

Dashboards provide a visual summary of real-time data, enabling users to quickly understand business trends, monitor performance, and make informed decisions. They allow easy access to report data through visual components.

UseCase:

As an Admin for TripAdvisor E-Management, you continually strive to meet business requirements, driving the organization toward peak performance. Your dedication and effective data visualization in reports have greatly impressed your superiors, making it effortless for the CEO to access and view essential data during meetings without having to search.

Activity 1:

Create Dashboard

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

The dialog is titled "New Dashboard". It contains fields for "Name" (set to "Dashboard 1"), "Description" (empty), "Folder" (set to "Private Dashboards" with a "Select Folder" button), and two buttons at the bottom: "Cancel" and "Create".

2. Give a Name and click on Create.

3. Select add component.

The dialog is titled "Select Report". It has a sidebar with sections for Reports (Recent, Created by Me, Private Reports, Public Reports, All Reports) and Folders (Created by Me, Shared with Me, All Folders). The main area shows a list of recent reports with a search bar "Search Reports and Folders...". At the bottom are "Cancel" and "Select" buttons.

Report Name	Description	Folder
New Food Options Report	Umar Ahmed Khan A - 14-Nov-2024, 12:12 pm · Trip	
New Customers Report	Umar Ahmed Khan A - 14-Nov-2024, 12:04 pm · Trip	
New Flights Report	Umar Ahmed Khan A - 15-Nov-2024, 12:02 am · Trip	
New Hotels Report	Umar Ahmed Khan A - 15-Nov-2024, 12:00 am · Trip	

4. Select a Report and click on select.

The screenshot shows a Salesforce Lightning dashboard titled "November | Salesforce". The top navigation bar includes links for Student, Anna University Regional Camp4 - dev-ed Develop, and the current page, November | Salesforce. The dashboard has a search bar and various icons for refresh, edit, and notifications. The main content area is divided into four sections:

- New Hotels Report:** A table showing hotel names and their total food options. Data:

Hotel	Hotel Name	TotalFoodOptions
Anand Hotel		6
Jothi Hotel		8
Maria Lodge		7
Pacha Elai		6
PK ilam		6
- New Flights Report:** A donut chart titled "Record Count" showing the distribution of flight records across three dates: 15/11/2024 (1 record), 16/11/2024 (2 records), and 17/11/2024 (2 records).
- New Customers Report:** A table showing customer names, discount amounts, and discount percentages. Data:

Customer	Customer Name	Discount Amount	Discount Percent
Elayabharathi		₹10	9%
Prasanth		₹35	20%
Sivakumaran		₹50	14%
Subash		₹30	30%
Tamil		₹69	9%
- New Food Options Report:** A table showing food option names, amounts, and associated hotels. Data:

Food Option	Food Option Name	Food Amount	Hotel	Name
FO - 0001		₹90	Jothi Hotel	EB Dosa
FO - 0002		₹30	Pacha Elai	Idly
FO - 0003		₹20	PK ilam	Idly

At the bottom, there are links for "View Report (New Hotels Report)", "View Report (New Flights Report)", and "View Report (New Customers Report)". The system status bar at the bottom right shows "Upcoming Earnings", system icons, and the date/time "15-11-2024 00:32".

Key Scenarios Addressed by Salesforce in the Implementation Project

1. Automating Hotel Data Updates:

Triggers handle changes to food options without manual intervention.

2. Customer Discount Management:

Flows automate discount calculation and application, enhancing customer satisfaction.

3. Flight Reminder Notifications:

Schedulable Apex ensures timely communication, reducing customer queries.

Conclusion:

Summary of Achievements

The *TripAdvisor E-Management* Salesforce project successfully streamlined the management of travel-related services by implementing a comprehensive solution that integrates automation, custom user interfaces, and real-time notifications. Key achievements include:

- Developed a user-friendly CRM tailored to manage hotel, flight, and food option data efficiently.
- Automated discount calculations and flight reminder notifications, significantly reducing manual processes and improving customer satisfaction.
- Provided actionable insights into hotel occupancy, food option availability, and flight booking trends, supporting better business decisions.

This solution is scalable and adaptable, providing a robust foundation for future enhancements, such as advanced customer personalization or integration with third-party travel platforms.