



Capstone Project Report  
On  
**“Travel Approval App”**

Submitted by:

**Name:** Bodakonda Sravan Kumar

**Batch** WIP-SF-13

**LMS Id:** MGSA\_539

**Date:** 28/02/2023

## **Table of contents**

1.	Abstract.....
2.	Introduction.....
3.	Flow of the project.....
4.	Software Requirements.....
5.	Screen shots.....
6.	Future enhancements.....
7.	References.....

## **Abstract**

Every travel organization requires a Travel Approval Application. This department deals with the approval of travel requests. It also allows for prime levels of control and oversight over travel bookings, policy and budget. The approval process is usually often lengthy and requires a lot of paperwork. It requires so much of time and work hard to complete it. To eradicate human errors and achieve perfect data and procedures, the use of software is considered a smart choice.

Therefore, with the help of Customer Relation Management (CRM) software services provided by Salesforce, we are making a travel approval application. It starts with building a data model where the objects and fields come into picture followed by customizing the user interface where we Use list views and page layouts to streamline an app user's experience.

Every application generally needs a business logic so in this it is achieved by validation rules , formula fields and approval process. Finally, to analyze travel approvals we add reports and dashboards. This application will help them in keeping track of the complete travel approval process.

## Introduction

In today's fast-paced business environment, sales teams often need to travel frequently to meet with clients, attend conferences, and conduct business. However, managing travel requests and approvals can be a time-consuming and error-prone process, particularly when done manually. To address this challenge, many organizations are turning to technology solutions such as the Travel Approval App for Salesforce.

This app is a powerful tool that allows sales teams to easily submit and manage travel requests directly within the Salesforce platform. By automating the travel approval process, the app saves time and reduces the risk of errors and delays. It also provides sales managers with real-time visibility into their teams' travel plans, enabling them to make informed decisions about resource allocation and budgeting.



In this report, we will provide an overview of the Travel Approval App for Salesforce, including its features, benefits, and potential use cases. We will also explore how this app can help organizations improve their travel management processes, increase efficiency, and reduce costs. Finally, we will discuss best practices for implementing and using the app effectively.

## **Flow of the Project**

The flow of the Travel Approval App for Salesforce typically involves several steps, which are as follows:

**Travel Request Submission:** Sales representatives submit travel requests using the Travel Approval App within the Salesforce platform. The request includes details such as travel dates, destinations, budgets, and purpose.

**Automated Routing of Requests:** The app automatically routes the travel request to the appropriate approvers based on predefined rules and workflows. This ensures that requests are reviewed by the right people, in the right order.

**Review and Approval:** Approvers receive notifications when a travel request requires their approval. They can review the request details and approve or reject it within the app.

**Notification and Feedback:** Sales representatives are notified of the approval or rejection of their travel request. If rejected, the app may allow the representative to revise and resubmit the request.

**Real-Time Reporting:** The app provides real-time visibility into travel requests and approvals, allowing sales managers to monitor travel spend and make informed decisions about their teams' travel plans.

Overall, the Travel Approval App for Salesforce automates and streamlines the travel request and approval process, providing a seamless experience for sales representatives and managers while ensuring compliance and cost-effectiveness.

# Software Requirements

For the fastest and most stable experience, we recommend:

- An Octane 2.0 score of 30,000 or greater
- Network latency of 150 ms or less
- Download speed of 3 Mbps or greater
- At least 8 GB of RAM, with 3 GB available for Salesforce browser tabs

Minimum requirements are:

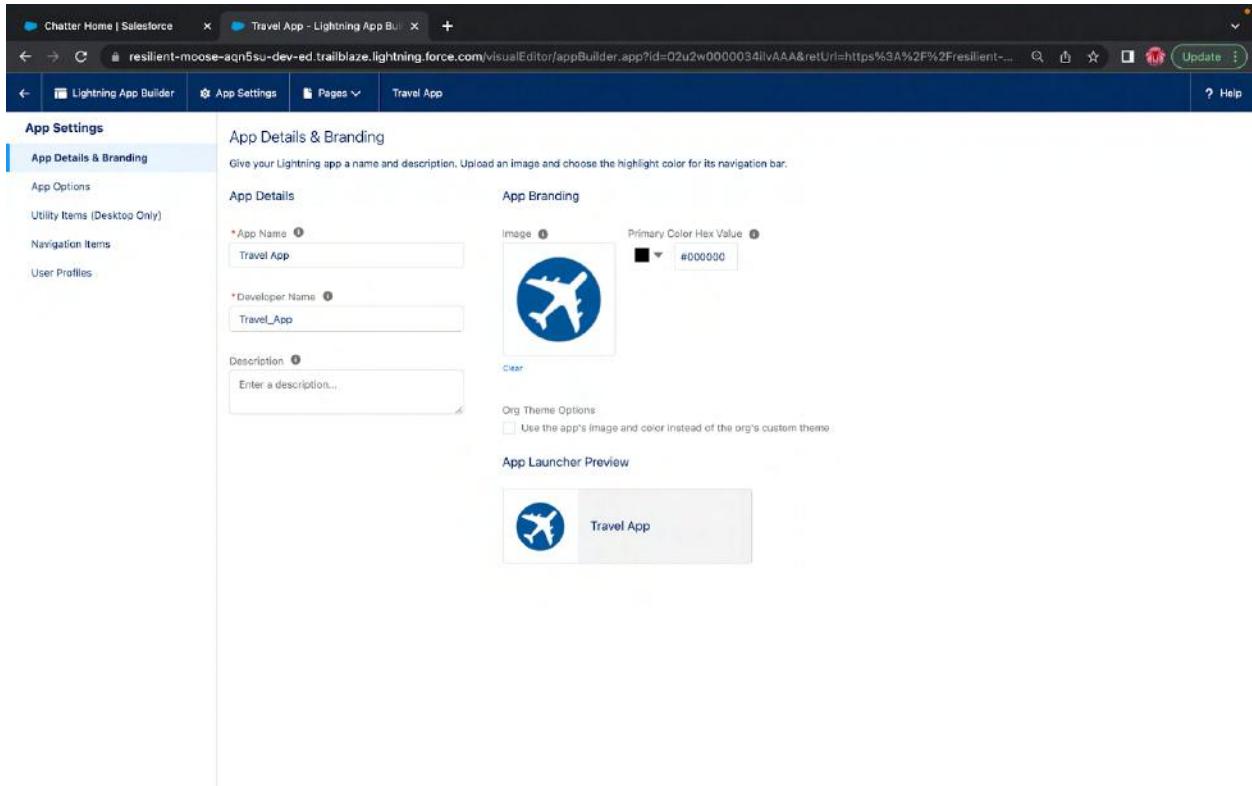
- An Octane 2.0 score of 20,000 or greater
- Network latency of 200 ms or less
- Download speed of 1 Mbps or greater
- At least 5 GB of RAM, with 2 GB available for Salesforce browser tabs

**OR**

Requirements	
Windows	
<b>Operating system</b>	<a href="#">Windows 8.1</a> 64-bit, <a href="#">Windows 8</a> 64-bit, Windows 7 Service Pack 1 64-bit, Windows Vista Service Pack 2 64-bit
<b>CPU</b>	<a href="#">Core 2 Quad</a> Q6600 at 2.4 GHz or <a href="#">AMD Phenom</a> 9850 at 2.5 GHz
<b>Memory</b>	4 GB <a href="#">RAM</a>
<b>Free space</b>	65 GB of free space
<b>Graphics hardware</b>	DirectX 10-compatible <a href="#">GPU</a> : <a href="#">GeForce 9800GT</a> 1GB or <a href="#">ATI Radeon HD 4870</a> 1GB
<b>Sound hardware</b>	<a href="#">DirectX</a> 10 compatible <a href="#">sound card</a>

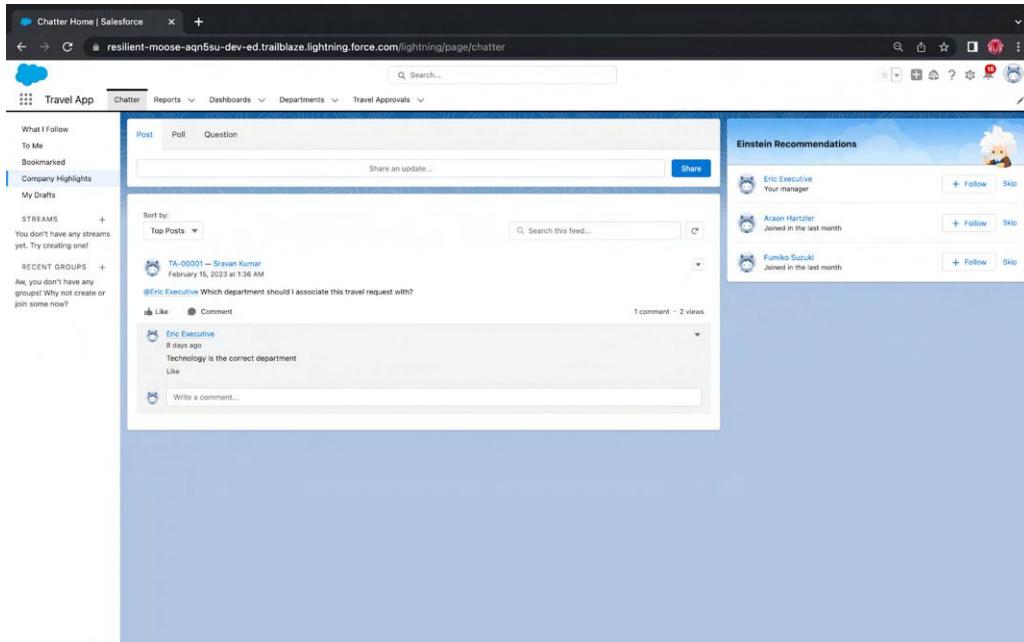
# Module-1 Screenshots

## Travel App:-



## Exercise -1

### Step-1



## Step-2

### Created Department custom object

The screenshot shows the Salesforce Setup interface under 'Object Manager'. A new object named 'Department' has been created. The 'Details' tab is selected, displaying the following information:

Details	
Description	
API Name	Department__c
Singular Label	Department
Plural Label	Departments
Enable Reports	✓
Track Activities	✓
Track Field History	
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

The left sidebar lists various configuration options for the object.

### Created new custom object tab

The screenshot shows the Salesforce Setup interface under 'Tabs'. A new custom object tab for 'Department' has been created. The 'Custom Tab Definition Detail' section shows the following configuration:

Tab Label	Departments
Object	Department
Description	
Created By	Sravan Kumar, 2/14/2023, 11:17 PM
Modified By	Sravan Kumar, 2/14/2023, 11:17 PM

The 'Tab Style' is set to 'Jewel'. The left sidebar shows other tabs and search results.

## Step-3

### Created all the required fields

The screenshot shows the Salesforce Object Manager interface for the 'Department' object. The left sidebar lists various setup options like Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, etc. The main content area displays a table titled 'Fields & Relationships' with 7 items. The columns are 'FIELD LABEL', 'FIELD NAME', 'DATA TYPE', and 'CONTROLLING'. The data includes:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING
Created By	CreatedById	Lookup(User)	
Department Code	Department_Code__c	Text(10) (Unique Case Sensitive)	
Department Name	Name	Text(80)	
Department Type	Department_Type__c	Picklist	
Last Modified By	LastModifiedById	Lookup(User)	
Location	Location__c	Picklist	
Owner	OwnerId	Lookup(User,Group)	

The top right corner shows the user profile of Sravan Kumar and the display density setting is set to 'Comfy'.

## Created Field Dependency

The screenshot shows the 'Edit Field Dependency' page for the 'Department' object. The left sidebar is identical to the previous screenshot. The main area has a title 'Edit Field Dependency' and a sub-section 'Instructions' with a note about selecting cells for visibility. Below is a table for defining dependencies:

Controlling Field	Location	Dependent Field	Department Type

Below the table is a legend for 'Included Value' and 'Excluded Value'. There are two sections for picking values from picklists:

- Location:** Picklist values: Kolkata, Banking, Finance, Education, Energy, IT.
- Department Type:** Picklist values: Debt, Banking, Finance, Education, Energy, IT.

Buttons at the bottom include Save, Cancel, and Preview.

## Step-4

### Created the required Travel Approval Object

The screenshot shows the Salesforce Setup interface for the 'Travel Approval' object. The left sidebar lists various configuration tabs: 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, and Flow Triggers. The main 'Details' tab is selected, displaying the following information:

Field	Value
Description	
API Name	Travel_Approval__c
Custom	✓
Singular Label	Travel Approval
Plural Label	Travel Approvals

On the right, a sidebar shows user information for 'Sarvan Kumar' and settings for 'DISPLAY DENSITY' (set to 'Comfy') and 'OPTIONS' (checkboxes for 'Enable Reports', 'Track Activities', 'Track Field History', 'Deployment Status' (set to 'Deployed'), and 'Help Settings').

### Created the required fields

The screenshot shows the 'Fields & Relationships' section of the 'Travel Approval' object setup. It displays a table of fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING
Created By	CreatedBy	Lookup(User)	
Department	Department__c	Lookup(Department)	
Destination State	Destination_State__c	Text(2)	
Last Modified By	LastModifiedBy	Lookup(User)	
Out-of-State	Out_of_State__c	Checkbox	
Owner	OwnerId	Lookup(User/Group)	
Purpose of Trip	Purpose_of_Trip__c	Text Area(255)	
Status	Status__c	Picklist	
Total Expenses	Total_Expenses__c	Roll-Up Summary (SUM Expense Item)	
Travel Approval #	Name	Auto Number	
Trip End Date	Trip_End_Date__c	Date	
Trip Start Date	Trip_Start_Date__c	Date	

## After testing the app

The screenshot shows a Salesforce Lightning page for a Travel Approval record. The record ID is TA-00001. The details section includes:

- Travel Approval #: TA-00001
- Status: Draft
- Total Expenses: \$1,320.00
- Status Indicator: In-Process
- Created By: Sravan Kumar, 2/15/2023, 12:52 AM
- Last Modified By: Sravan Kumar, 2/15/2023, 12:52 AM
- Trip Info:
  - Trip End Date: 2/28/2023
  - Trip Start Date: 2/15/2023
  - Purpose of Trip: Attend Dreamforce
  - Out-of-State: Yes
  - Destination State: CA

The sidebar on the right shows the user profile of Sravan Kumar and the activity feed, which is currently empty.

## Step-5

Imported departments.csv file using Data Import Wizard

The screenshot shows a list of departments imported from a CSV file. The list includes:

- Audit Services
- Contract Management
- Disability Determination Bureau
- Division of Aging
- Division of Disability and Rehabilitative Services
- Division of Family Resources
- Division of Finance
- Division of Mental Health and Addiction
- Human Resources
- Legislative Services
- Office of Communications and Media
- Office of Early Childhood and Out-of-School Learning
- Office of General Counsel
- Office of Medicaid Policy and Planning
- Quality and Compliance Office
- Technology

The sidebar on the right shows the user profile of Sravan Kumar and the activity feed, which is currently empty.

## Exercise-2

### Step-1

#### Created Travel Approval Record

The screenshot shows a Salesforce page for a Travel Approval record with ID TA-00500. The page has a header with the Travel App logo and navigation links for Chatter, Reports, Dashboards, Departments, and Travel Approvals. The main content area displays the record details under the 'Details' tab. Key fields shown include:

- Travel Approval #: TA-00500
- Status: Draft
- Total Expenses: \$0.00
- Status Indicator: In-Process
- Created By: Sravan Kumar, 2/23/2023, 8:16 AM
- Last Modified By: Sravan Kumar, 2/23/2023, 8:16 AM
- Trip Info:
  - Trip End Date: 2/28/2023
  - Trip Start Date: 2/24/2023
  - Purpose of Trip: Attend Dreamforce
  - Out-of-State: ✓
  - Destination State: CA

The right sidebar shows the user profile (Sravan Kumar) and activity feed settings (Comfy, Compact). A message indicates no upcoming or overdue activities.

### Step-2

#### Created Expense Item Custom Object

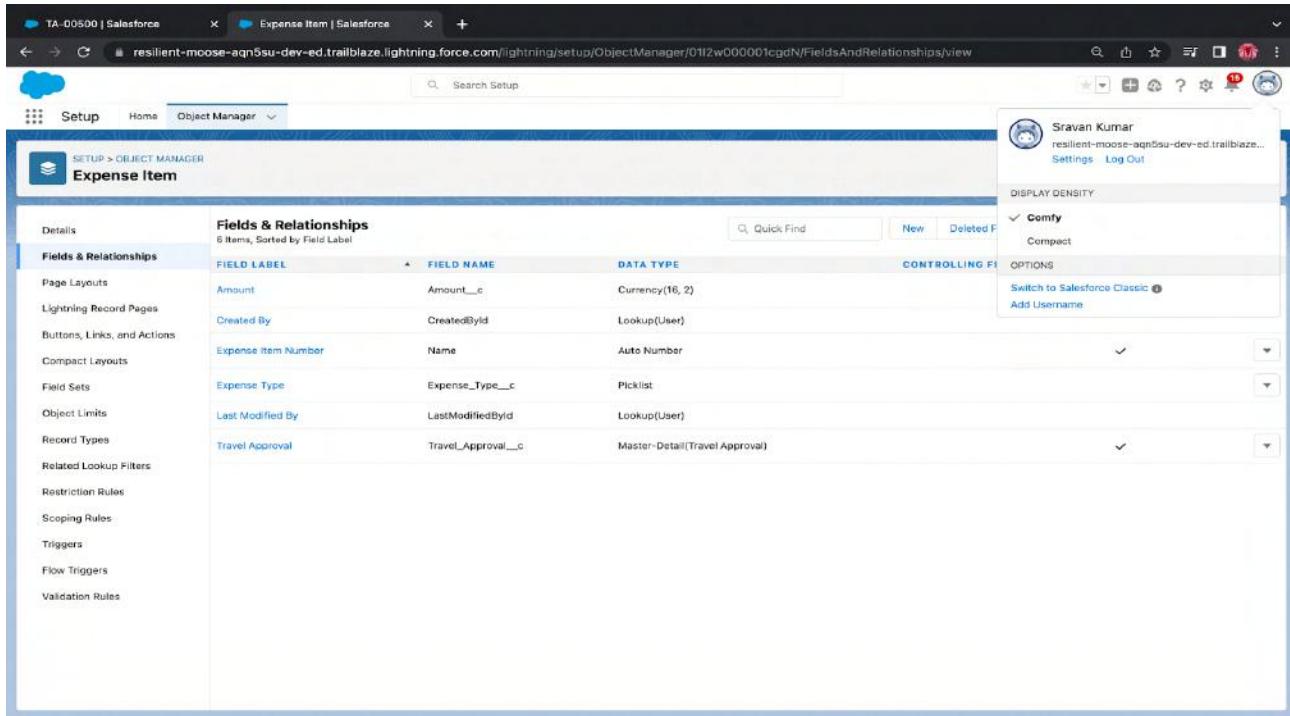
The screenshot shows the Salesforce Setup > Object Manager interface. A new object named "Expense Item" is being created. The "Details" tab is selected, showing the following configuration:

- Description: Expense Item
- API Name: Expense\_Item\_\_c
- Custom: ✓
- Singular Label: Expense Item
- Plural Label: Expense Items
- Enable Reports: ✓
- Track Activities: ✓
- Track Field History: ✓
- Deployment Status: Deployed
- Help Settings: Standard salesforce.com Help Window

The left sidebar lists various object settings like Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Restriction Rules, Scoping Rules, Triggers, Flow Triggers, and Validation Rules.

## Step-3

### Created Custom Fields

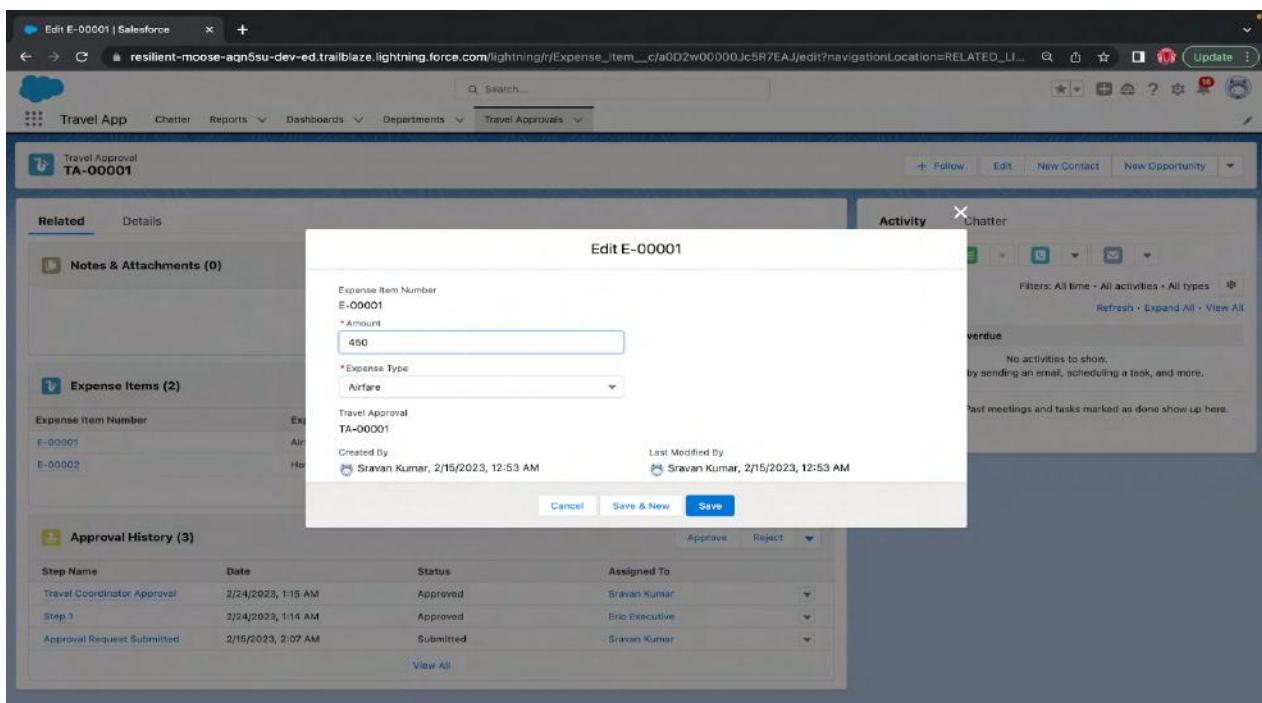


The screenshot shows the Salesforce Setup interface under Object Manager for the 'Expense Item' object. The 'Fields & Relationships' section is displayed, listing six custom fields:

FIELD LABEL	FIELD NAME	DATA TYPE
Amount	Amount__c	Currency(16, 2)
Created By	CreatedById	Lookup(User)
Expense Item Number	Name	Auto Number
Expense Type	Expense_Type__c	Picklist
Last Modified By	LastModifiedById	Lookup(User)
Travel Approval	Travel_Approval__c	Master-Detail(Travel Approval)

## Step-4

### Create Records



The screenshot shows the 'Travel App' interface with a 'Travel Approval' record open. A new 'Expense Item' record is being created with the following details:

- Expense Item Number: E-00001
- Amount: 450
- Expense Type: Airfare
- Travel Approval: TA-00001
- Created by: Sravan Kumar
- Last Modified by: Sravan Kumar

## Created User Eric

## Step-6

### Added Eric Executive as Manager to System Administrator

The screenshot shows the Salesforce Setup interface with the 'Users' tab selected. On the left, a sidebar lists various setup categories like Permission Set Groups, Profiles, and User Management Settings. The main content area displays the 'User Detail' page for a user named 'Eric Executive'. The user's details include Name: Eric Executive, Alias: exec, Email: bodakondasravankumar@gmail.com, Username: bodakondasravankumar@gmail.com, Nickname: eric.exec, Title: , Company: , Department: , Division: , Address: , Time Zone: (GMT-08:00) Pacific Standard Time (America/Los\_Angeles), Locale: English (United States), Language: English, and Delegated Approver: Manager. Under 'User License', 'System Administrator' is checked. The 'OPTIONS' section includes links to 'Switch to Salesforce Classic' and 'Add Username'. A right-hand sidebar shows the user's profile picture and name (Sravan Kumar), and settings for 'DISPLAY DENSITY' (Compact) and 'OPTIONS' (e.g., Marketing User, Offline User, Knowledge User, Flow User, Service Cloud User, Site.com Contributor User, WDC User, Mobile Push Registrations, Data.com User Type, Accessibility Mode, Debug Mode, High-Contrast Palette on Charts, Load Lightning Pages While Scrolling, Send Apex Warning Emails, Salesforce CRM Content User, Receive Salesforce CRM Content Email Alerts, Receive Salesforce CRM Content Alerts as Daily Digest, Make Setup My Default Landing Page, Quick Access Menu, Development Mode).

## Step-7

### Customized the Travel Approval Default Search

The screenshot shows the Salesforce Setup interface with the 'Search Layouts' tab selected. The left sidebar includes sections for Einstein Search (Objects to Always Search (Beta), Promoted Search Terms, Search Layouts, Search Manager, Settings, Synonyms), Objects and Fields (Object Manager), and Global Search. The main content area displays the 'Edit Search Layout' page for 'Travel Approval Search Results'. It shows a list of available fields: Record ID, Out-of-State, Status Indicator, Total Expenses, Owner Alias, Owner First Name, Owner Last Name, Created By Alias, Created By, Created Date, Last Modified By Alias, Travel Approval #, Purpose of Trip, Department, Origin State, Destination State, Trip Start Date, and Trip End Date. The 'Selected Fields' section lists 'Travel Approval #' and 'Purpose of Trip'. Below the field lists are sections for 'Standard Buttons' (none listed) and 'Custom Buttons' (none listed). At the bottom are 'Save' and 'Cancel' buttons.

## Step-8

### Selected all the fields

The screenshot shows a Salesforce Lightning page titled 'Travel Approvals'. A modal window titled 'Select Fields to Display' is open, listing various fields available for selection. The 'Visible Fields' section contains 'Travel Approval #' and 'Status'. Other fields listed in the 'Available Fields' section include 'Created By Alias', 'Created Date', 'Destination State', 'Last Activity Date', 'Last Modified By', and 'Last Modified By Alias'. The 'Save' button at the bottom right of the modal is highlighted in blue.

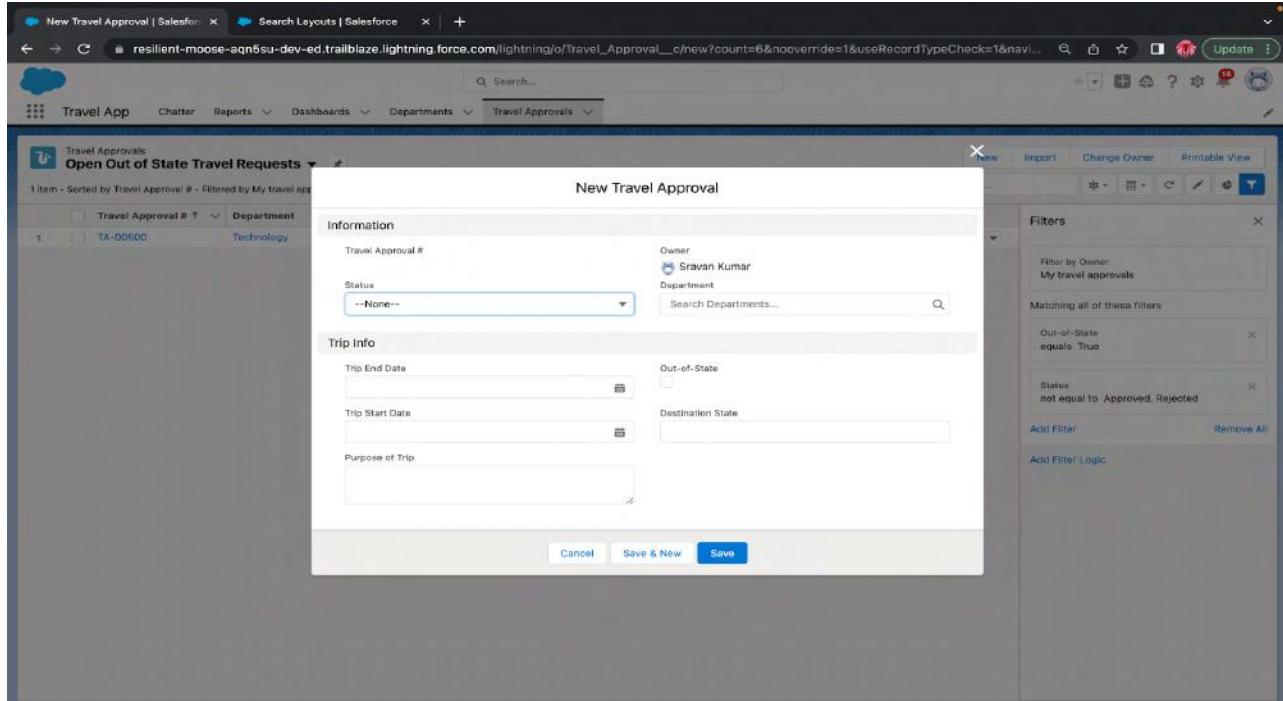
## Step-9

### Created Travel Approval custom list view open out of state Travel Requests

The screenshot shows a Salesforce Lightning page titled 'Travel Approvals' with a custom list view titled 'Open Out of State Travel Requests'. The list view displays one item: a travel approval for 'TA-00650' from 'Technology' department, created by 'Sravan Kumar' in 'Draft' status, destined for 'CA', starting on '2/24/2023' and ending on '2/28/2023'. To the right of the list view is a 'Filters' sidebar containing two filters: 'Out-of-State equals True' and 'Status not equal to Approved, Rejected'. Buttons for 'Add Filter' and 'Remove All' are also visible.

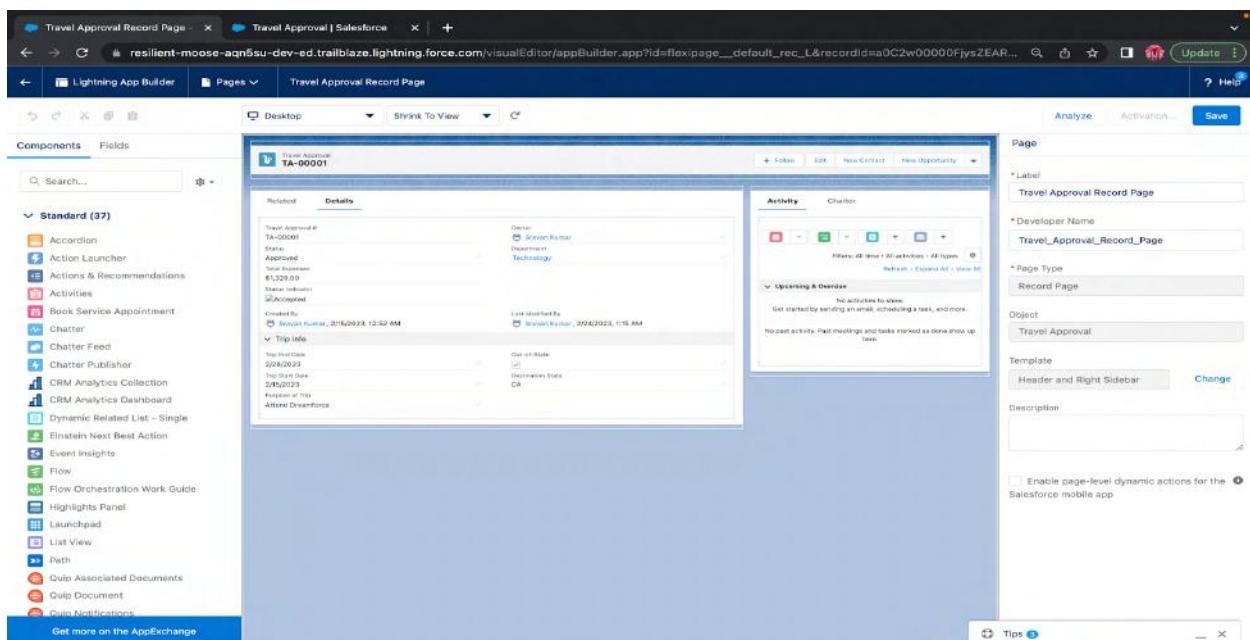
## Step-10

Selected fields to display in the Travel Approval Open Out of State Travel Requests



## Step-11

Customized Travel Approval Page Layout



TA-00001 | Salesforce    Travel Approval | Salesforce    Travel Approval | Salesforce

resilient-moose-aqn5su-dev-ed.trailblaze.lightning.force.com/lightning/setup/ObjectManager/01t2w000001cg8n/PageLayouts/view

Setup Home Object Manager

Search Setup

SETUP > OBJECT MANAGER

## Travel Approval

Page Layouts

1 items, Sorted by Page Layout Name

PAGE LAYOUT NAME	CREATED BY	MODIFIED BY
Travel Approval Layout	Sravan Kumar, 2/14/2023, 11:20 PM	Sravan Kumar, 2/15/2023, 1:59 AM

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

Flow Triggers

Validation Rules

TA-00001 | Salesforce    Travel Approval | Salesforce    Travel Approval | Salesforce

resilient-moose-aqn5su-dev-ed.trailblaze.lightning.force.com/lightning/setup/ObjectManager/01t2w000001cg8n/PageLayouts/00h2w000002epyAAB/view

Setup Home Object Manager

Search Setup

SETUP > OBJECT MANAGER

## Travel Approval

Travel Approval Layout

Save Quick Save Preview As... Cancel Undo Redo Layout Properties

Fields

Section	Destination State	Purpose of Trip	Travel Approval #
Section	Destination State	Purpose of Trip	Travel Approval #
Blank Space	Last Modified By	Status	Trip End Date
Created By	Out-of-State	Status Indicator	Trip Start Date
Department	Owner	Total Expenses	

Custom Console Components Mail Page

DISPLAY DENSITY

✓ Comfy

Compact

OPTIONS

Switch to Salesforce Classic Add Username

SRavan Kumar  
resilient-moose-aqn5su-dev-ed.trailblaze.lightning.force.com/lightning/setup/ObjectManager/01t2w000001cg8n/PageLayouts/00h2w000002epyAAB/view

Settings Log Out

Travel Approval Sample

Highlights Panel

Customize the highlights panel for this page layout...

Quick Actions in the Salesforce Classic Publisher

Actions in this section are currently inherited from the global publisher layout. You can override the global publisher layout to set a customized list of actions for the publisher on pages that use this layout.

Salesforce Mobile and Lightning Experience Actions

Actions in this section are predefined by Salesforce. You can override the predefined actions to set a customized list of actions on Lightning Experience and mobile app pages that use this layout. If you customize the actions in the Quick Actions in the Salesforce Classic Publisher section, and have saved the layout, then this section inherits that set of actions by default when you click to override.

Travel Approval Detail

Standard Buttons

Edit Delete Clone Change Owner Change Record Type Printable View Sharing Sharing Hierarchy Submit for Approval Custom Buttons

Information (Header visible on edit only)

Travel Approval #	GEN-2004-001234
Status	Sample Text
Total Expenses	\$123.45
Status Indicator	Sample Text

Owner Sample Text

Department Sample Text

## Step-12

Customized the Expense Item Related list under Travel Approval page layout

The screenshot shows the Salesforce Lightning Experience for a Travel Approval record. The URL in the browser is [https://resilient-moose-aqn5su-dev-ed.trailblaze.lightning.force.com/lightning/r/Travel\\_Approval\\_\\_c/a0C2w00000FjysZEAR/view](https://resilient-moose-aqn5su-dev-ed.trailblaze.lightning.force.com/lightning/r/Travel_Approval__c/a0C2w00000FjysZEAR/view).

**Related List:**

- Notes & Attachments (0)**: Includes a "Upload Files" button and a "Or drop files" placeholder.
- Expense Items (2)**:

Expense Item Number	Expense Type	Amount
E-00001	Airfare	\$450.00
E-00002	Hotel	\$870.00
- Approval History (3)**:

Step Name	Date	Status	Assigned To
Travel Coordinator Approval	2/24/2023, 1:15 AM	Approved	Sravan Kumar
Step 1	2/24/2023, 1:14 AM	Approved	Eric Executive
Approval Request Submitted	2/15/2023, 2:07 AM	Submitted	Sravan Kumar

**User Profile:** Sravan Kumar (resilient-moose-aqn5su-dev-ed.trailblaze...), Settings, Log Out

**Activity Sidebar:** Comfy (selected), Compact, Options, Switch to Salesforce Classic, Add Username

**Upcoming & Overdue:** No activities to show. Get started by sending an email, scheduling a task, and more. No past activity. Past meetings and tasks marked as done show up here.

## Step-13

### Enable Feed Tracking

The screenshot shows the Salesforce Setup interface for Feed Tracking. The left sidebar navigation includes 'Feed Item Actions', 'Feed Item Layouts', 'Feed Tracking' (which is selected), 'Triggers', 'Feeditem Triggers', 'Service', and 'Feed Filters'. The main content area is titled 'Feed Tracking' and contains a sub-section 'Fields in travel approvals'. It lists various fields like 'Problem', 'Project', 'Report', 'Site', 'Topic', and 'Travel Approval'. For each field, there is a checkbox labeled 'Enable Feed Tracking'. A note says 'You can select up to 9 fields.' On the right side, there are sections for 'DISPLAY DENSITY' (set to 'Compact') and 'OPTIONS' (with 'Switch to Salesforce Classic' and 'Add Username' buttons). A user profile for 'Sravan Kumar' is visible at the top right.

The screenshot shows the 'New Travel Approval' page in the Travel App. The title bar indicates 'Travel Approvals' and 'Travel Approvals'. The main form has tabs for 'Information' and 'Trip Info'. In the 'Information' tab, fields include 'Travel Approval #' (set to 'TA-00001'), 'Purpose of Trip' (set to 'Attend Direct Sales Meeting'), 'Status' (set to 'None'), 'Owner' (set to 'Sravan Kumar'), and 'Department' (set to 'Search Departments...'). In the 'Trip Info' tab, fields include 'Trip End Date' (set to '2/28/2023'), 'Out-of-State' (unchecked), 'Trip Start Date' (set to '2/28/2023'), 'Destination State' (empty), and 'Purpose of Trip' (empty). At the bottom are 'Cancel', 'Save & New', and 'Save' buttons.

TA-00001 | Salesforce    Travel Approval | Salesforce    Feed Tracking | Salesforce

resilient-moose-aqn5su-dev-ed.trailblaze.lightning.force.com/lightning/r/Travel\_Approval\_\_c/a0C2w00000FjysZEAR/view

Travel App Chatter Reports Dashboards Departments Travel Approvals

Travel Approval TA-00001

Related Details

Travel Approval # TA-00001

Status Approved

Total Expenses \$1,320.00

Status Indicator Accepted

Created By Sravan Kumar, 2/15/2023, 12:52 AM

Last Modified By Sravan Kumar, 2/24/2023, 1:15 AM

Trip Info

Trip End Date 2/28/2023

Trip Start Date 2/15/2023

Purpose of Trip Attend Dreamforce

Out-of-State

Destination State CA

Activity Chatter

Post Poll Question

Share an update... Share

Search this feed...

Sravan Kumar updated this record. 15m ago

Status Draft to Approved

Like Comment

Write a comment...

Sravan Kumar February 15, 2023 at 1:38 AM

@Eric Executive Which department should I associate this travel request with?

1 comment • 2 views

Like Comment

## Module 2 Screenshots

### Exercise-1

#### Step-1

Created Trip end date after start date Validation Rule

The screenshot shows the Salesforce Setup interface for the 'Travel Approval' object. On the left, a sidebar lists various configuration options like Details, Fields & Relationships, Page Layouts, etc. The main area is titled 'Travel Approval Validation Rule'. It shows a validation rule named 'Trip\_end\_date\_after\_start\_date' with the formula `Trip_End_Date__c < Trip_Start_Date__c`. A tooltip for the ABS function is visible. The error message is set to 'Trip end date must be greater than or equal to start date'.

The screenshot shows the 'Validation Rule Detail' page for the 'Trip\_end\_date\_after\_start\_date' rule. It displays the rule name, formula, error message ('Trip end date must be greater than or equal to start date'), and location ('Trip End Date'). The rule was created by 'Shrawan Kumar' on 2/15/2023 at 1:40 AM and modified by the same user on the same date and time.

## Step-2

### Created a Roll Up summary field Total Expenses

The screenshot shows the Salesforce Setup interface for the 'Travel Approval' object. On the left, a sidebar lists various setup categories like Fields & Relationships, Page Layouts, Lightning Record Pages, etc. The main panel displays the 'Custom Field Definition Detail' for the 'Total Expenses' field. Key details shown include:

- Field Label:** Total Expenses
- Field Name:** Total\_Expenses
- API Name:** Total\_Expenses\_\_c
- Description:** Null Text
- Data Owner:** Field Usage
- Data Sensitivity Level:** Compliance Categories
- Compliance Category:** None
- Created By:** Sravan Kumar, 2/15/2023, 1:42 AM
- Modified By:** Sravan Kumar, 2/15/2023, 1:42 AM
- Roll-Up Summary Options:** Data Type: Roll-Up Summary, Subsumed Object: Expense Item, Field to Aggregate: Expense Item Amount, Filter Criteria: None
- Summary Type:** SUM

## Step-3

### Created a formula field StatusImages

The screenshot shows the Salesforce Setup interface for creating a static resource named 'StatusImages'. The left sidebar shows the 'Static Resources' section under 'Data'. The main panel shows the 'Static Resource Edit' screen for 'StatusImages'. The fields filled are:

- Name:** StatusImages
- Description:** (empty)
- File:** Choose file: StatusImages.zip
- Cache Control:** Public

The screenshot shows the Salesforce Setup interface under the Static Resources section. A specific resource named "StatusImages" is selected. The details pane shows the following information:

Name	StatusImages
Namespace Prefix	
Description	
MIME Type	application/zip
Cache Control	Public
Size	57,029 bytes
<a href="#">View file</a>	
Created By	Sravan Kumar, 2/15/2023, 1:45 AM
Last Modified By	Sravan Kumar, 2/15/2023, 1:45 AM

Below the details, there are "Edit", "Delete", and "Where is this used?" buttons.

## Step-4

### Created a formula field status indicator

The screenshot shows the Salesforce Setup interface under the Object Manager section. A custom field named "Status Indicator" is selected for the "Travel Approval" object. The details pane shows the following information:

**Custom Field Definition Detail**

Field Information	Object Name
Field Label: Status Indicator Field Name: Status_Indicator API Name: Status_Indicator_c Description: Help Text Data Owner: Field Usage Data Sensitivity Level: Compliance Category Created By: Sravan Kumar, 2/15/2023, 1:48 AM	Travel Approval
Modified By: Sravan Kumar, 2/15/2023, 1:48 AM	

**Formula Options**

Data Type: Formula

```
If([ISNULL(Status__c), 'Approved', IMAGE('resource>Status/Images/thumb-up.png', 'Accepted', 20, 20), If([ISNULL(Status__c), 'Rejected', IMAGE('resource>Status/Images/thumb-down.png', 'Rejected', 20, 20), IMAGE('resource>Status/Images/draft.png', 'In-Process', 20, 20))])
```

# Step-5

## Created a Record -Triggered Flow Decision Block

The screenshot shows the Salesforce Flow Builder interface. At the top, there are three tabs: "Chatter Home | Salesforce", "Travel Approval | Salesforce", and "Out of State Travel Flag - V1". The "Out of State Travel Flag - V1" tab is active. The main area displays a flow diagram with a single "Start" node. A tooltip for the "Start" node indicates it is a "Record-Triggered Flow" for the "Travel Approval" object, triggered by "A record is created or updated". The "Toolbox" on the left lists various flow elements under "Logic" and "Data".

The screenshot shows the "Edit Decision" dialog box for a decision block named "Is Travel Out of State? (is\_Travel\_Out\_of\_State)". The dialog box is divided into sections: "OUTCOME ORDER" (with outcomes "Yes Out of State" and "In State"), "OUTCOME DETAILS" (with outcome API name "Yes\_Out\_of\_State" and condition "All Conditions Are Met: (AND)"), and "Resource" (with condition "A\_B \$Record > Destination State" and value "TX"). The "When to Execute Outcome" section contains two options: "If the condition requirements are met" (radio button) and "Only if the record that triggered the flow to run is updated to meet the condition requirements" (checkbox). The "Done" button is visible at the bottom right.

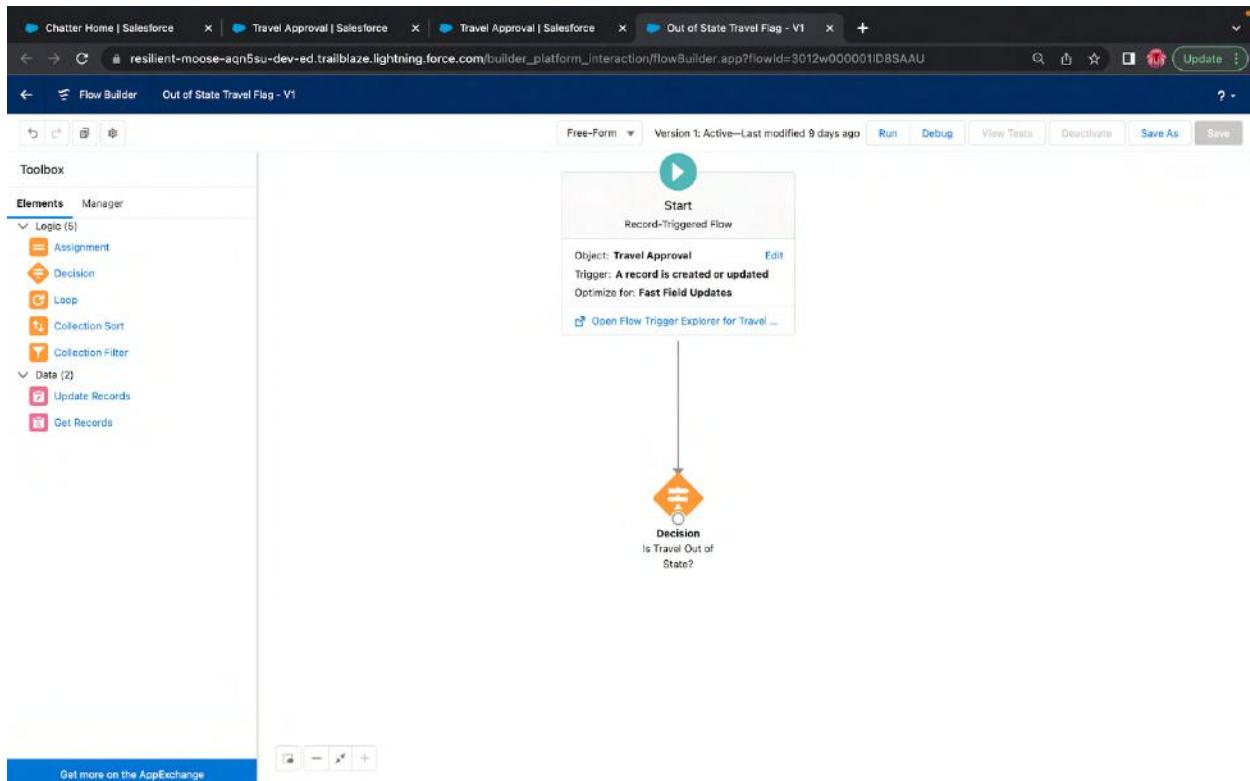
# Updated Travel Approval Record (Yes Out Of State)

The screenshot shows the Salesforce Flow Builder interface. The top navigation bar includes tabs for Chatter Home, Travel Approval, Travel Approval, and Out of State Travel Flag - V1. The main title is "Out of State Travel Flag - V1". The status bar indicates "Free-Form" flow type, "Version 1: Active—Last modified 9 days ago", and buttons for Run, Debug, View Tests, Deactivate, Save As, and Save.

The left sidebar is titled "Toolbox" and contains sections for Elements Manager, Logic (5), Data (2), and a list of specific components: Assignment, Decision, Loop, Collection Sort, Collection Filter, Update Records, and Get Records.

The central workspace displays the "Edit Update Records" dialog. It includes instructions for finding records to update and setting their values. A note states: "Because this flow runs before a record is saved, you can only update the record that triggered the flow to run. To update other records, configure the trigger to run the flow after the record is saved." Below this is a "Set Filter Conditions" section with a dropdown menu set to "None—Always Update Record".

The main configuration area is titled "Set Field Values for the Travel Approval Record". It shows a table with one row: "Field" (Out\_of\_State\_\_c) and "Value" (True). There is also a "Done" button at the bottom right.

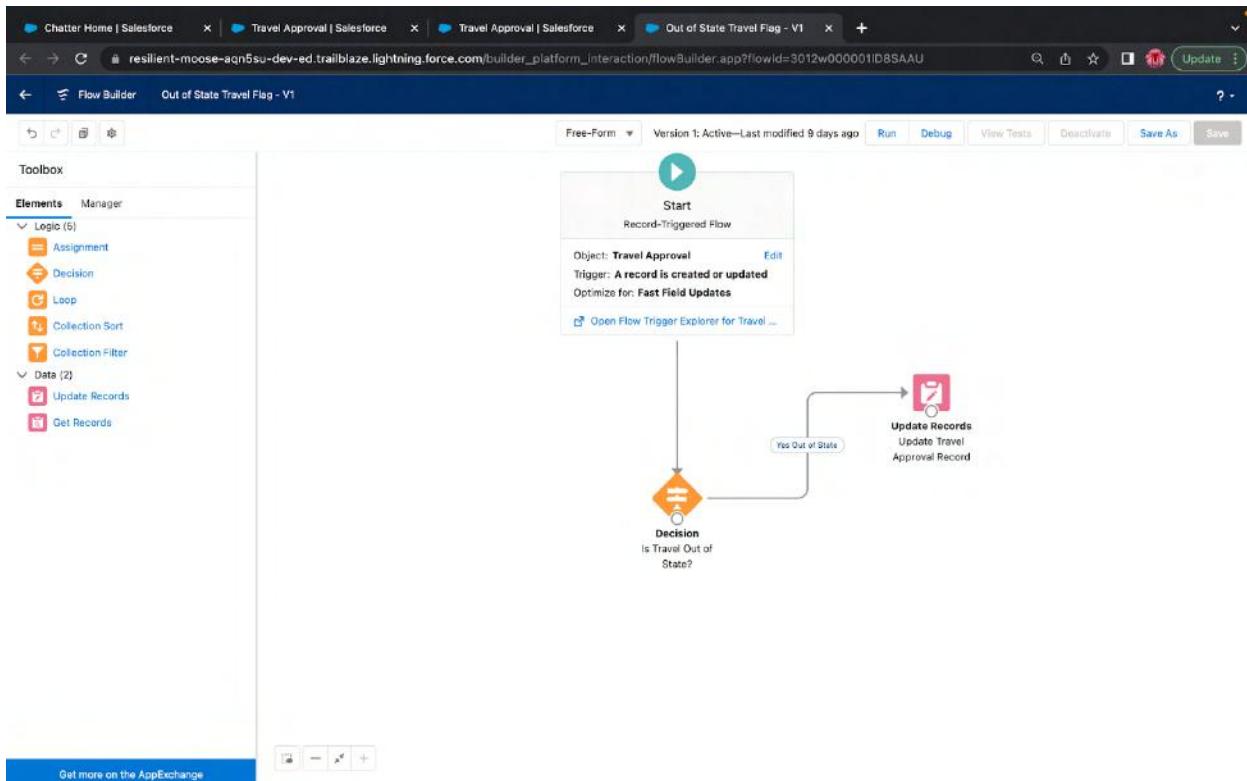


# Updated Travel Approval Record(In State)

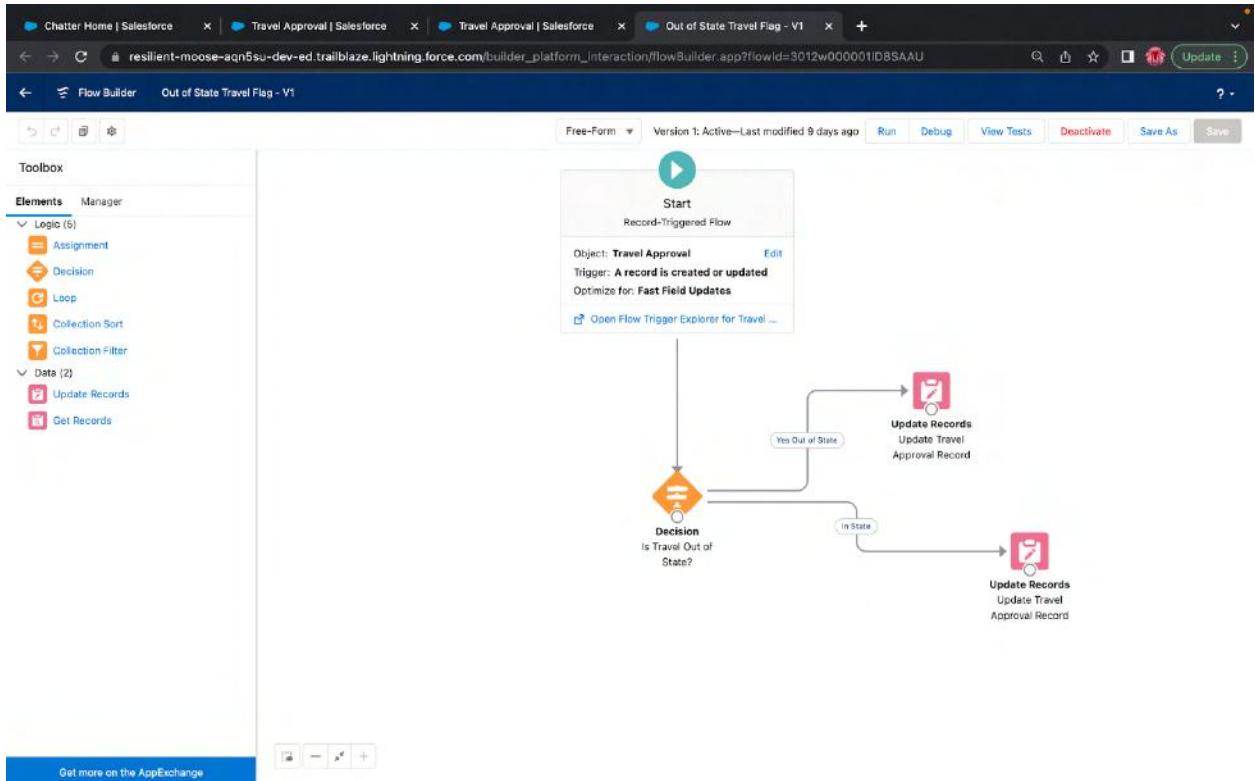
The screenshot shows the 'Edit Update Records' configuration in the Flow Builder. The flow is triggered by a record creation or update on the 'Travel Approval' object. The configuration includes:

- How to Find Records to Update and Set Their Values:**
  - Use the travel approval record that triggered the flow
  - Update records related to the travel approval 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
- A note: "Because this flow runs before a record is saved, you can only update the record that triggered the flow to run. To update other records, configure the trigger to run the flow after the record is saved."
- Set Filter Conditions:** Condition Requirements to Update Record: None—Always Update Record.
- Set Field Values for the Travel Approval Record:**

Field	Value
Out_of_State__c	False



# Out Of State Travel Flag(flow)



## Step-6

### Created an approval process Travel Approval Request

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

- Page:** Approval Processes
- Process Name:** Travel Approval: Travel Approval Request
- Status:** Active (checked)
- Description:** Travel Approval: Total Expenses GREATER THAN 0
- Entry Criteria:** Travel Approval: Total Expenses GREATER THAN 0
- Record Editability:** Administrator ONLY
- Approval Assignment Email Template:** Initial Submitters: Travel Approval Owner
- Created By:** Sravan Kumar, 2/15/2023, 2:03 AM
- Modified By:** Sravan Kumar, 2/15/2023, 2:03 AM
- Initial Submission Actions:**
  - Action Type: Record Lock
  - Description: Lock the record from being edited
- Approval Steps:**

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions   Edit	1	Step 1			Manager	Final Rejection
Show Actions   Edit	2	Travel Coordinator Approval		Travel Approval: Out-of-State EQUALS True	User Sravan Kumar	Final Rejection
- Final Approval Actions:**
  - Action Type: Record Lock
  - Description: Lock the record from being edited
  - Action Type: Field Update
  - Description: Set Status to Approved

The screenshot shows the Approval Processes page in the Salesforce Setup. The process is named "Travel Approval Request". It has one step, "Step 1", which is "Travel Coordinator Approval". The criteria for this step is "Travel Approval: Out-of-State equals True". The assigned approver is "Manager" (User: Sravan Kumar). The process is active and was created by Sravan Kumar on 2/15/2023 at 1:59 AM.

## Final Approval Actions

The screenshot shows the Field Updates page in the Salesforce Setup. A new field update named "Set Status to Approved" has been created for the "Travel Approval" object. The status field is being updated to "Approved". This field update is currently not used by any rules or entitlement processes.

# Final Rejection Actions

The screenshot shows the Salesforce Field Updates page. A specific field update named "Set Status to Rejected" has been created for the "Travel Approval" object. The "Field Update Detail" section shows the following configuration:

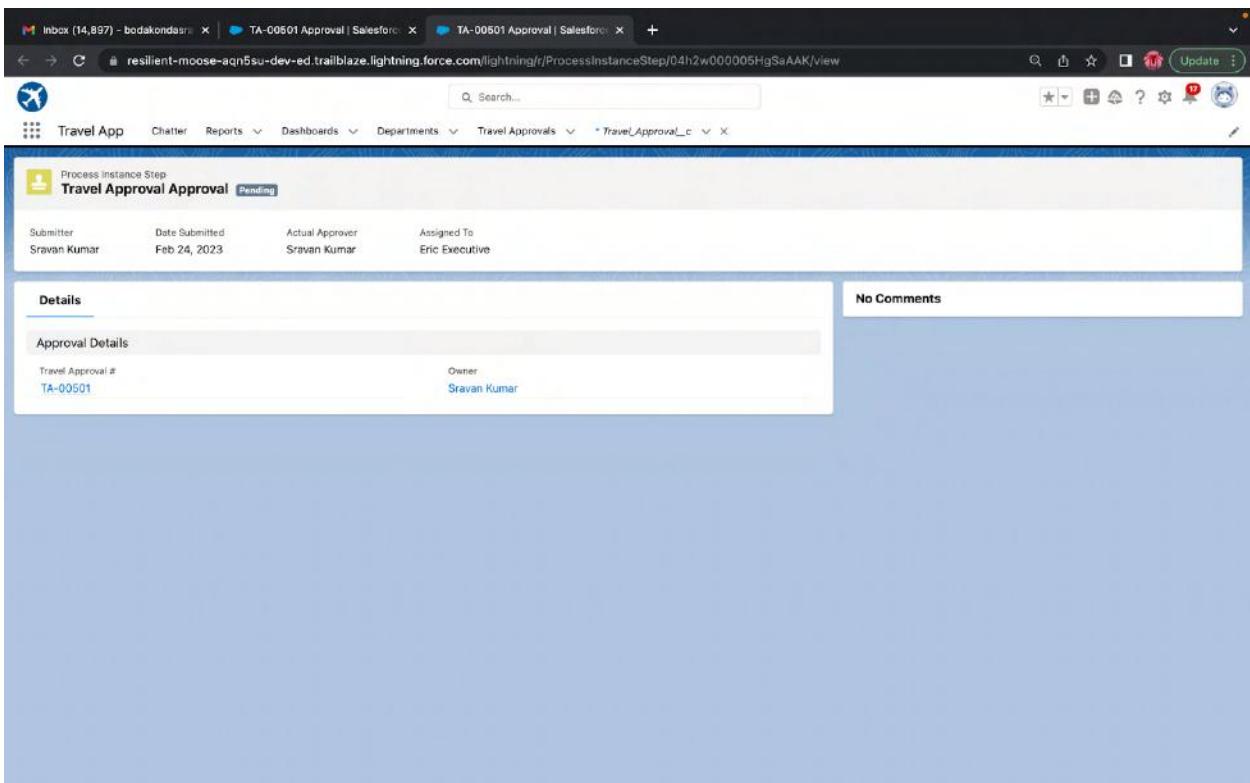
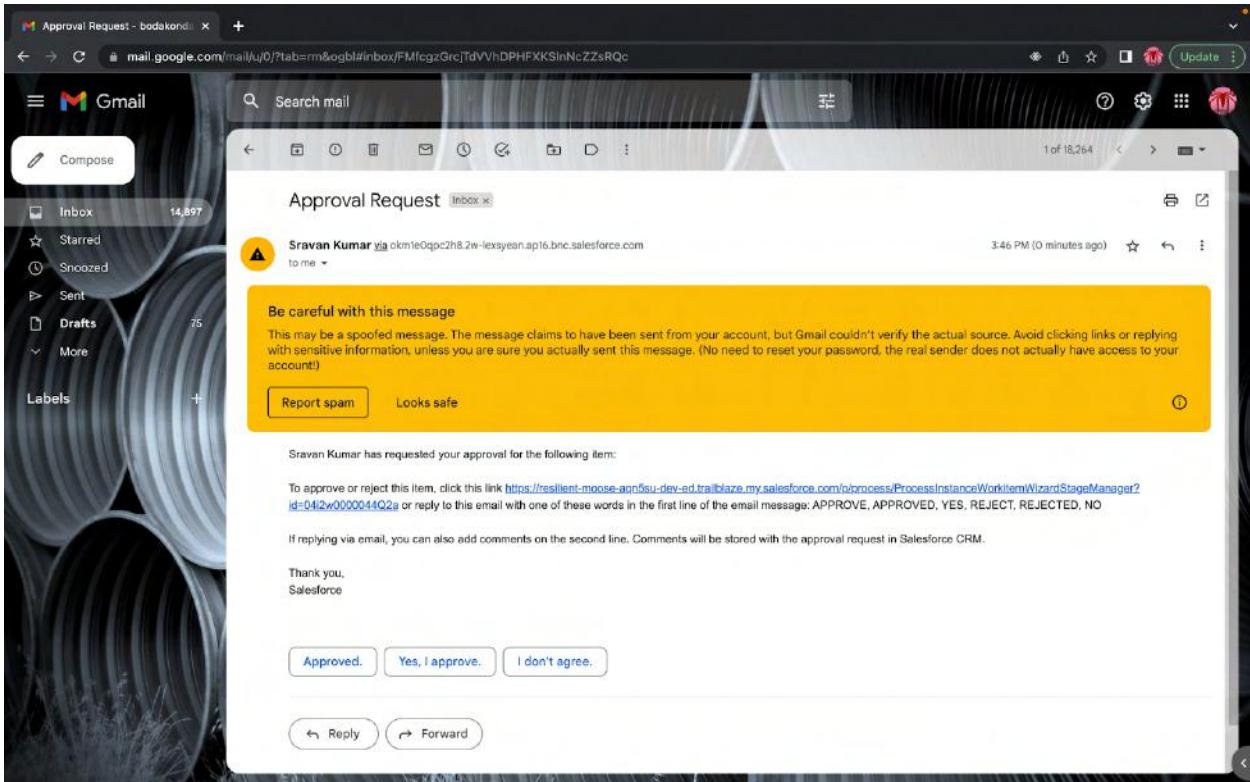
- Name:** Set Status to Rejected
- Unique Name:** Set\_Status\_\_o\_Rejected
- Description:** Travel Approval
- Object:** Travel Approval
- Field to Update:** Travel Approval: Status
- Field Data Type:** Picklist
- Re-evaluate Workflow Rules after Field Change:**
- New Field Value:** Rejected

Below the detail section, there are three sections: "Rules Using This Field Update", "Approval Processes Using This Field Update", and "Entitlement Processes Using This Field Update". Each section contains a note stating that the field update is currently not used by any rules, processes, or entitlements.

The screenshot shows the Process Automation Settings page. Under the "Process Automation" section, the "Approval Processes" tab is selected. The "Email Approval Sender" dropdown is set to "Approval Submitter". The "Enable email approval response" checkbox is checked. A yellow warning box states: "By enabling the email approval response feature, you agree to allow Salesforce to process email approval responses, update approval requests for all active users in your organization, and update the approval object on behalf of your organization's users." Other settings include "Let users pause flows" (unchecked), "Let users resume interviews" (checked), "Flows launched from a URL or from Setup use the Lightning runtime experience instead of the classic runtime experience" (checked), and "Allow Apex code to set and remove approval process locks" (checked).

# Testing the Approval Process

## Received an email



# Assigned to Eric Executive

Process instance Step  
Travel Approval Approval **Approved**

Submitter: Sravan Kumar Date Submitted: Feb 24, 2023 Actual Approver: Sravan Kumar Assigned To: Sravan Kumar

**Details**

Approval Details

Travel Approval #: TA-00501 Owner: Sravan Kumar

No Comments

## Exercise-2

### Step-1

#### Imported Travel Approval records

Travel Approvals

All

150+ items · Sorted by Travel Approval # · Filtered by All travel approvals · Updated a few seconds ago

#	Travel Approval #	Department	Created By	Status	Trip Start Date	Trip End Date
1	TA-00001	Technology	Sravan Kumar	Approved	2/15/2023	2/28/2023
2	TA-00201	Office of Communications and Media	Sravan Kumar	Approved	6/14/2019	6/15/2019
3	TA-00202	Disability Determination Bureau	Sravan Kumar	Approved	10/1/2019	10/1/2019
4	TA-00203	Division of Disability and Rehabilitative Services	Sravan Kumar	Rejected	4/3/2019	4/6/2019
5	TA-00204	Technology	Sravan Kumar	Approved	3/3/2019	3/20/2019
6	TA-00205	Human Resources	Sravan Kumar	Approved	4/27/2019	5/5/2019
7	TA-00206	Division of Finance	Sravan Kumar	Rejected	3/9/2019	3/9/2019
8	TA-00207	Contract Management	Sravan Kumar	Approved	12/22/2019	1/1/2020
9	TA-00208	Division of Aging	Sravan Kumar	Approved	4/23/2019	4/28/2019
10	TA-00209	Audit Services	Sravan Kumar	Rejected	8/9/2019	8/15/2019
11	TA-00210	Division of Aging	Sravan Kumar	Approved	11/5/2019	11/11/2019
12	TA-00211	Disability Determination Bureau	Sravan Kumar	Approved	3/22/2019	3/22/2019
13	TA-00212	Technology	Sravan Kumar	Approved	3/6/2019	3/9/2019
14	TA-00213	Office of General Counsel	Sravan Kumar	Approved	1/2/2019	1/26/2019
15	TA-00214	Office of Communications and Media	Sravan Kumar	Approved	3/7/2019	3/19/2019
16	TA-00215	Division of Aging	Sravan Kumar	Approved	8/6/2019	8/14/2019
17	TA-00216	Legislative Services	Sravan Kumar	Approved	4/7/2019	4/14/2019
18	TA-00217	Contract Management	Sravan Kumar	Approved	5/8/2019	5/14/2019
19	TA-00218	Division of Disability and Rehabilitative Services	Sravan Kumar	Approved	11/6/2019	11/12/2019
20	TA-00219	Audit Services	Sravan Kumar	Approved	2/28/2019	3/6/2019
21	TA-00220	Disability Determination Bureau	Sravan Kumar	Rejected	3/6/2019	3/7/2019
22	TA-00221	Office of Communications and Media	Sravan Kumar	Approved	7/9/2019	7/13/2019
23	TA-00222	Legislative Services	Sravan Kumar	Rejected	6/14/2019	6/26/2019

## Step-2

### Created a Report on Travel Requests by Department

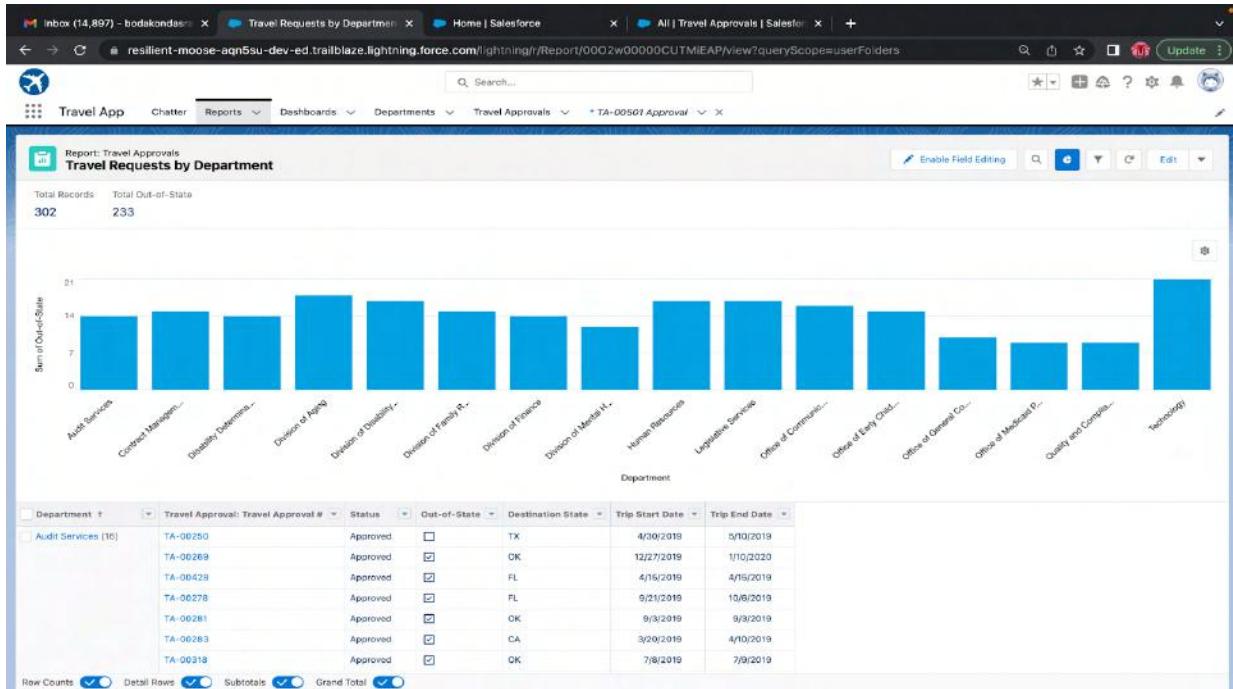
The screenshot shows the Salesforce Report Builder interface with the following details:

- Report Name:** Travel Requests by Department
- Report Type:** Travel Approvals
- Filters:** Department, Status, Out-of-State, Destination State, Trip Start Date, Trip End Date.
- Data:** The report lists travel approvals for various departments. Key entries include:
  - Division of Aging: TA-00412 (Rejected)
  - Division of Family Resources: TA-00410 (Approved)
  - Division of Finance: TA-00406 (Approved)
  - Division of Mental Health and Addiction: TA-00421 (Approved)
  - Human Resources: TA-00409 (Approved)
  - Office of Communications and Media: TA-00416 (Approved)
  - Office of Early Childhood and Out-of-School Learning: TA-00408 (Approved)
  - Office of General Counsel: TA-00417 (Rejected), TA-00420 (Approved), TA-00493 (Approved)
  - Office of Medicaid Policy and Planning: TA-00422 (Approved)
  - Quality and Compliance Office: TA-00407 (Rejected)
- Report Options:** Outline, Filters, Row Counts, Detail Rows, Subtotals, Grand Total, Conditional Formatting.

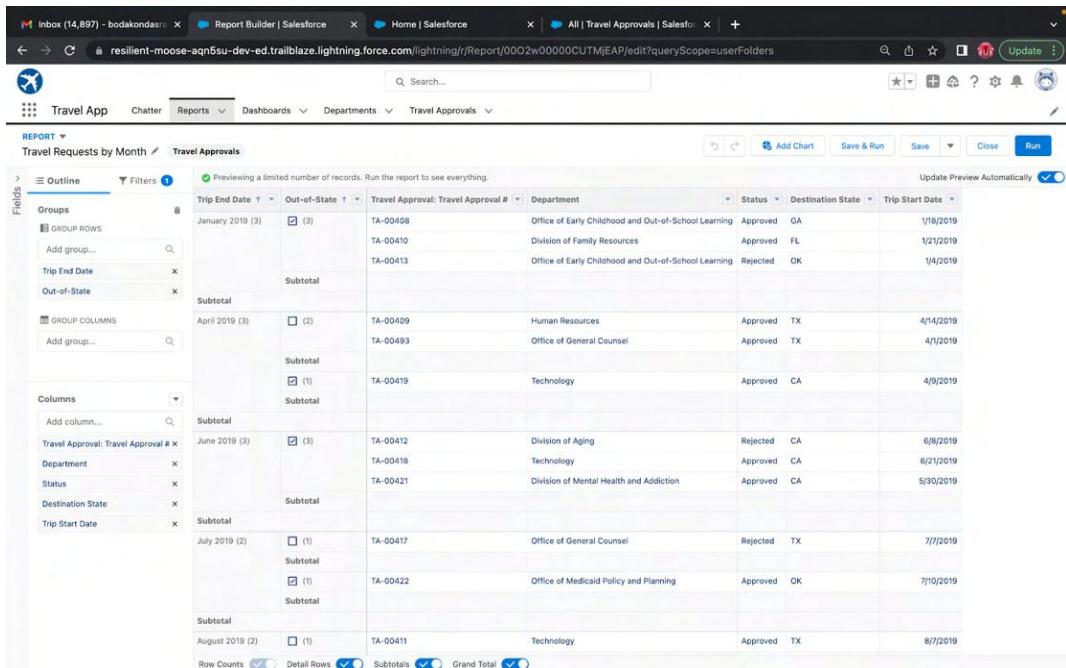
The screenshot shows the Salesforce report view with the following details:

- Report Name:** Travel Requests by Department
- Report Type:** Travel Approvals
- Filters:** Department, Status, Out-of-State, Destination State, Trip Start Date, Trip End Date.
- Data:** The report lists travel approvals for various departments. Key entries include:
  - Audit Services: TA-00250 (Approved), TA-00289 (Approved), TA-00428 (Approved), TA-00278 (Approved), TA-00281 (Approved), TA-00283 (Approved), TA-00318 (Approved), TA-00491 (Rejected), TA-00344 (Approved), TA-00370 (Approved), TA-00374 (Approved), TA-00391 (Rejected), TA-00399 (Approved), TA-00404 (Rejected), TA-00209 (Rejected), TA-00219 (Approved).
  - Contract Management: TA-00496 (Approved), TA-00499 (Approved), TA-00225 (Approved).
- Report Options:** Enable Field Editing, Row Counts, Detail Rows, Subtotals, Grand Total.

# After Testing



## Step-3 Created a Travel Requests by Monthly Report



Inbox (14,897) - bodakondasri | Travel Requests by Month | Sales | Home | Salesforce | All | Travel Approvals | Salesforce | +

resilient-moose-aqn5su-dev-ed.trailblaze.lightning.force.com/lightning/r/Report/00O2w00000CUTMjEAP/view?queryScope=userFolders

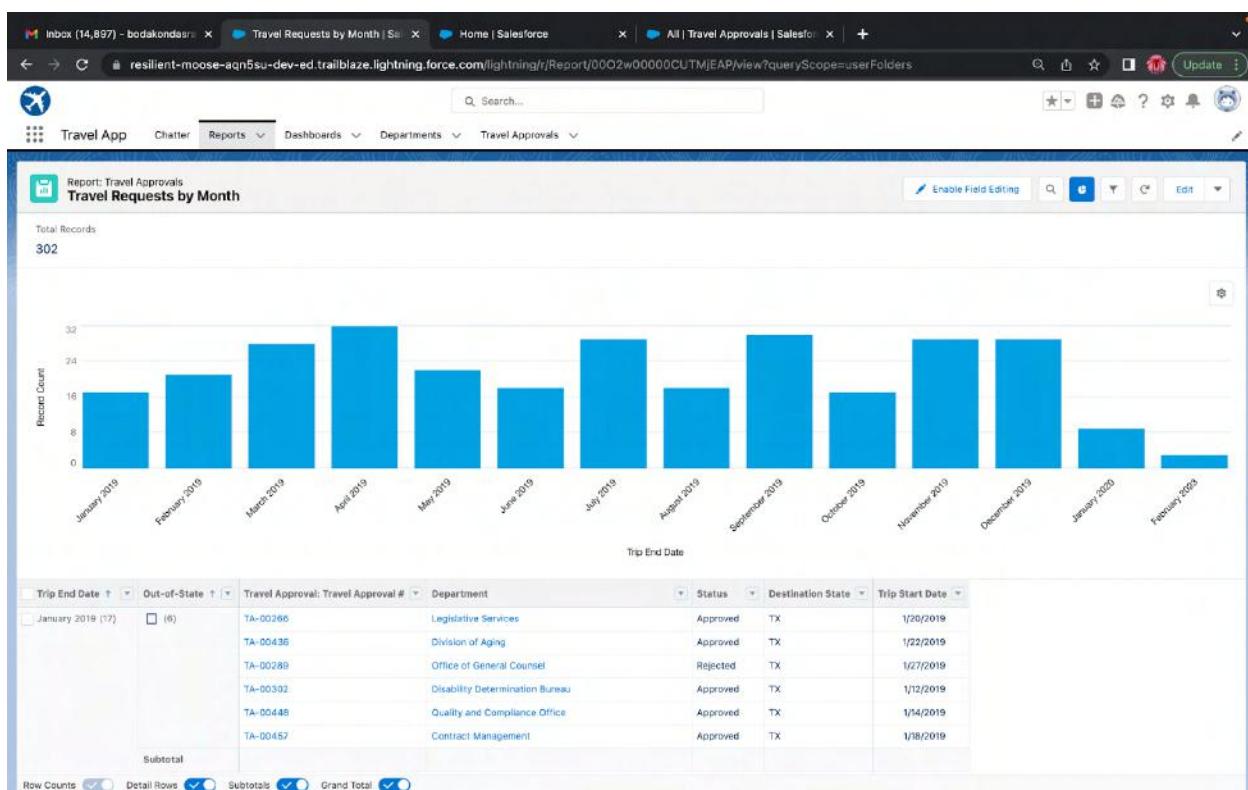
Travel App Chatter Reports Dashboards Departments Travel Approvals

Report: Travel Approvals  
Travel Requests by Month

Total Records 302

Trip End Date Out-of-State Travel Approval: Travel Approval # Department Status Destination State Trip Start Date

Trip End Date	Out-of-State	Travel Approval: Travel Approval #	Department	Status	Destination State	Trip Start Date
January 2019 (17)	(6)	TA-00266	Legislative Services	Approved	TX	1/20/2019
		TA-00436	Division of Aging	Approved	TX	1/22/2019
		TA-00289	Office of General Counsel	Rejected	TX	1/27/2019
		TA-00392	Disability Determination Bureau	Approved	TX	1/12/2019
		TA-00448	Quality and Compliance Office	Approved	TX	1/14/2019
		TA-00457	Contract Management	Approved	TX	1/18/2019
Subtotal		TA-00408	Office of Early Childhood and Out-of-School Learning	Approved	GA	1/18/2019
Subtotal		TA-00410	Division of Family Resources	Approved	FL	1/21/2019
Subtotal		TA-00419	Office of Early Childhood and Out-of-School Learning	Rejected	OK	1/4/2019
Subtotal		TA-00223	Office of Communications and Media	Rejected	FL	1/21/2019
Subtotal		TA-00229	Division of Disability and Rehabilitative Services	Approved	OK	1/23/2019
Subtotal		TA-00244	Division of Aging	Approved	OK	1/24/2019
Subtotal		TA-00304	Division of Aging	Approved	FL	1/3/2019
Subtotal		TA-00314	Division of Mental Health and Addiction	Approved	OK	1/3/2019
Subtotal		TA-00315	Contract Management	Approved	GA	1/30/2019
Subtotal		TA-00359	Division of Disability and Rehabilitative Services	Approved	FL	1/29/2019
Subtotal		TA-00395	Contract Management	Approved	FL	1/29/2019
Subtotal						
Row Counts		Detail Rows	Subtotals	Grand Total		



## Step-4

### Created a Report and dashboard for Travel Approvals

The screenshot shows the Salesforce Reports page. On the left, there's a sidebar with sections for Reports, Recent, Folders, and Favorites. The main area displays a table of recent reports with columns for Report Name, Description, Folder, Created By, and Created On. One report, "Travel Requests by Month", is highlighted.

REPORTS	Report Name	Description	FOLDER	Created By	Created On
Recent	Travel Requests by Month		Public Reports	Sravan Kumar	2/15/2023, 2:27 AM
Created by Me	Travel Requests by Department		Public Reports	Sravan Kumar	2/15/2023, 2:21 AM
Private Reports	Sample Flow Report: Screen Flows	Which flows run, what's the status of each interview, and how long do users take to complete the screens?	Public Reports	Automated Process	1/5/2023, 8:52 PM
Public Reports					
All Reports					

The screenshot shows the Travel Requests Dashboard. It features two main charts: "Travel Requests by Department" and "Travel Requests by Month".

**Travel Requests by Department:**

Department	Record Count
Audit Services	18
Contract Management	29
Disability Determination Bureau	11
Division of Aging	20
Division of Disability and Rehabilitation	21
Division of Family Resources	18
Division of Finance	19
Division of Mental Health and Substance Abuse	17
Human Resources	16
Legislative Services	19
Office of Communications and Outreach	23
Office of Early Childhood and Family Support	19

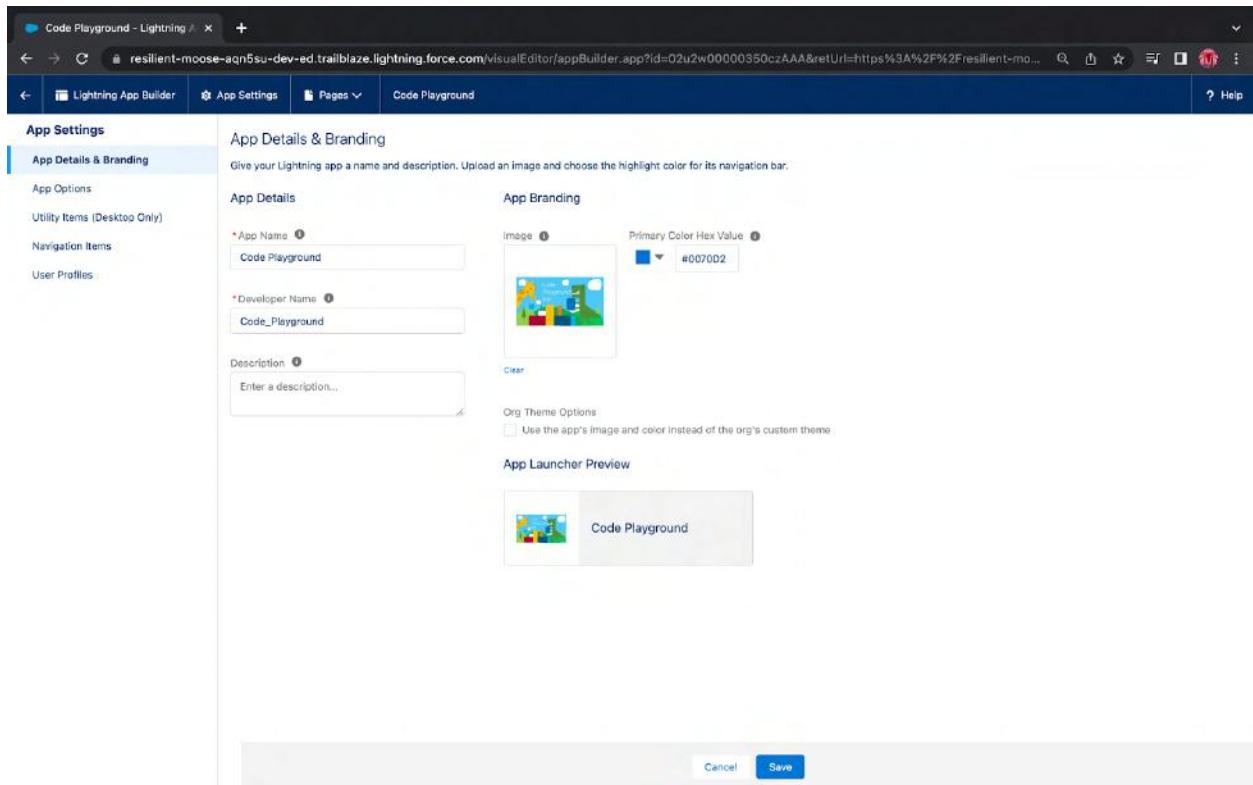
**Travel Requests by Month:**

Trip End Date > Out-of-State	Record Count
Jan.-False	8
Feb.-False	1
Mar.-True	3
Mar.-False	16
Apr.-True	21
Apr.-False	26
May-False	5
Jun.-True	17
Jun.-False	3
Jul.-True	15
Jul.-False	6
Aug.-True	23
Aug.-False	15
Sep.-True	11
Sep.-False	24

# Module-3 Screenshots

## Exercise - 1

### Created Code Playground App



## Created the Custom Object Customer as Customer

The screenshot shows the Salesforce Object Manager interface for the 'Customer' object. The left sidebar lists various configuration tabs: 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, Flow Triggers, and Validation Rules. The main content area displays the 'Details' tab for the 'Customer' object. It includes fields for API Name (Customer\_\_c), Singular Label (Customer), Plural Label (Customers), and Deployment Status (Deployed). There are also sections for Description, Enable Reports, Track Activities, Track Field History, Help Settings, and Standard salesforce.com Help Window.

## Created all the Custom

The screenshot shows the 'Fields & Relationships' section of the Salesforce Object Manager for the 'Customer' object. It lists seven fields: Active, Created By, Customer Name, Customer Type, Description, Last Modified By, and Owner. The table includes columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED status. The 'Customer Name' field is indexed.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Active	Active_c	Checkbox		
Created By	CreatedById	Lookup(User)		
Customer Name	Name	Text(60)		✓
Customer Type	Customer_Type__c	Picklist		
Description	Description__c	Text Area(255)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User/Group)		✓

## Created a Custom Object named Billing

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes tabs for Setup, Home, and Object Manager. The main title is "Billing". The left sidebar lists various configuration options: 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, Flow Triggers, and Validation Rules. The right panel displays the "Details" section for the Billing object, which includes fields like API Name (Billing\_\_c), Singular Label (Billing), Plural Label (Billings), and Deployment Status (Deployed). It also shows sections for Description, Enable Reports, Track Activities, Track Field History, Help Settings, and Standard Help Window.

## Create Playground App

The screenshot shows a custom application named "Code Playground" running on Salesforce. The top navigation bar includes tabs for Leads, Accounts, Contacts, Opportunities, Cases, Customers, and Billings. The main page displays a list of accounts under the heading "All Accounts". The list includes columns for Account Name, Account Site, Billing State/Province, Phone, Type, and Account Owner Alias. The data shows 50+ items, all owned by "SKuma". The account names are listed sequentially from 1 to 23, such as "Acidency Bulk Company", "Accufarm Bulk Company", etc.

Account Name	Account Site	Billing State/Province	Phone	Type	Account Owner Alias
1 Acidency Bulk Company					SKuma
2 Accufarm Bulk Company					SKuma
3 Accupharm Bulk Company					SKuma
4 Accusage Bulk Company					SKuma
5 Aclima Bulk Company					SKuma
6 Acumentor Bulk Company					SKuma
7 Acusage Bulk Company					SKuma
8 Aeris Bulk Company					SKuma
9 Anache Bulk Company					SKuma
10 Anarce Bulk Company					SKuma
11 Anivet Bulk Company					SKuma
12 Anocha Bulk Company					SKuma
13 Apex Bulk Company					SKuma
14 Apexia Bulk Company					SKuma
15 Apextri Bulk Company					SKuma
16 Applidic Bulk Company					SKuma
17 Applideck Bulk Company					SKuma
18 Aquacine Bulk Company					SKuma
19 Aquafire Bulk Company					SKuma
20 Aquasseur Bulk Company					SKuma
21 Aquavive Bulk Company					SKuma
22 Arctic Bulk Company					SKuma
23 Asimilne Bulk Company					SKuma

## Exercise - 2

String Variable using endsWith() method

Log executeAnonymous @21/02/2023, 11:36:45

Execution Log

Timestamp	Event	Details
11:36:45:002	USER_DEBUG	[2] DEBUG true

Enter Apex Code

```
1 String str='apex class';
2 System.debug(str.endsWith('s'));
```

Open Log    Execute    Execute Highlighted

Used Date Method and Displayed the Date

Log executeAnonymous @21/02/2023, 11:38:34

Execution Log

Timestamp	Event	Details
11:38:34:003	USER_DEBUG	[1] DEBUG 2023-02-20 00:00:00
11:38:34:003	USER_DEBUG	[1] DEBUG 2023-03-22 00:00:00

Enter Apex Code

```
1 Date str1=Date.today();
2 System.debug(str1);
3 Date str2=str1.addDays(30);
4 System.debug(str2);
```

Open Log    Execute    Execute Highlighted

Converted the string to integer and added 20 to it

Log executeAnonymous @21/02/2023, 11:40:06

Execution Log

Timestamp	Event	Details
11:40:06:002	USER_DEBUG	[3] DEBUG 10
11:40:06:002	USER_DEBUG	[4] DEBUG 30

Enter Apex Code

```
1 String str='10';
2 Integer x=Integer.ValueOf(str);
3 System.debug(x);
4 System.debug(x+20);
```

Open Log    Execute    Execute Highlighted

Used Length method to display String length

Log executeAnonymous @21/02/2023, 11:41:31

Execution Log

Timestamp	Event	Details
11:41:31:002	USER_DEBUG	[2] DEBUG size:20

Enter Apex Code

```
1 String str='Salesforce Developer';
2 System.debug('size:'+str.length());
```

Open Log    Execute    Execute Highlighted

Defined a list and displayed the outputs using different methods

The screenshot shows the Salesforce IDE interface. At the top, there is a header bar with the text "Log executeAnonymous @21/02/2023, 11:43:39". Below it is a "Execution Log" table with columns: "Timestamp", "Event", and "Details". The log entries are:

Timestamp	Event	Details
11:43:39:003	USER_DEBUG	[7] DEBUG After using add() method(0, 1, 2, 3, 4)
11:43:39:003	USER_DEBUG	[8] DEBUG After using get() method0
11:43:39:003	USER_DEBUG	[9] DEBUG After using set() method(0, 100, 2, 3, 4)
11:43:39:003	USER_DEBUG	[10] DEBUG After clearing all elements()

Below the log is an "Enter Apex Code" window containing the following Apex code:

```
1 List<Integer> ll= new List<Integer>();  
2 ll.add(0);  
3 ll.add(1);  
4 ll.add(2);  
5 ll.add(3);  
6 ll.add(4);  
7 System.debug('After using add() method'+ll);  
8 system.debug('After using get() method'+ll.get(0));  
9 ll.set(1,100);system.debug('After using set() method'+ll);  
10 ll.clear();system.debug('After clearing all elements'+ll);  
11
```

At the bottom of the interface, there are buttons for "Open Log", "Execute", and "Execute Highlighted".

Printed numbers between 0 to 9

The screenshot shows the Salesforce IDE interface. At the top, there are two header bars: "Log executeAnonymous @21/02/2023, 11:43:39" and "Log executeAnonymous @21/02/2023, 11:49:10". Below them is a "Execution Log" table with columns: "Timestamp", "Event", and "Details". The log entries are:

Timestamp	Event	Details
11:49:10:003	USER_DEBUG	[4] DEBUG Value of a is: 0
11:49:10:003	USER_DEBUG	[4] DEBUG Value of a is: 1
11:49:10:003	USER_DEBUG	[4] DEBUG Value of a is: 2
11:49:10:003	USER_DEBUG	[4] DEBUG Value of a is: 3
11:49:10:003	USER_DEBUG	[4] DEBUG Value of a is: 4
11:49:10:003	USER_DEBUG	[4] DEBUG Value of a is: 5
11:49:10:003	USER_DEBUG	[4] DEBUG Value of a is: 6
11:49:10:003	USER_DEBUG	[4] DEBUG Value of a is: 7
11:49:10:004	USER_DEBUG	[4] DEBUG Value of a is: 8
11:49:10:004	USER_DEBUG	[4] DEBUG Value of a is: 9

Below the log is an "Enter Apex Code" window containing the following Apex code:

```
1 Integer a=20;  
2 while(a>12){  
3     for(a=0;a<10;a++){  
4         system.debug('Value of a is: '+a);  
5     }  
6 }
```

At the bottom of the interface, there are buttons for "Open Log", "Execute", and "Execute Highlighted".

### Exercise - 3

The screenshot shows the Salesforce Dev Console interface. At the top, there are three tabs: "Log executeAnonymous @21/02/2023, 11:43:39", "Log executeAnonymous @21/02/2023, 11:49:10", and "Log executeAnonymous @21/02/2023, 11:51:50". The third tab is highlighted. Below the tabs is a header bar with "Execution Log" and three columns: "Timestamp", "Event", and "Details". A single log entry is shown: "Timestamp: 11:51:50:002", "Event: USER\_DEBUG", and "Details: [7]|DEBUG|True". To the right of the log is an "Enter Apex Code" window containing the following code:

```
1 Integer myunluckyNumber = 7;
2 Integer myluckyNumber = 15;
3 if(myluckyNumber != myunluckyNumber + 8){
4     system.debug('False');
5 }
6 else{
7     system.debug('True');
8 }
```

At the bottom of the "Enter Apex Code" window are three buttons: "Open Log" (with a checked checkbox), "Execute", and "Execute Highlighted".

### Exercise - 4

The screenshot shows the Salesforce Dev Console interface. At the top is a single tab: "Log executeAnonymous @21/02/2023, 11:52:51". Below the tab is a header bar with "Execution Log" and three columns: "Timestamp", "Event", and "Details". A single log entry is shown: "Timestamp: 11:52:51:003", "Event: USER\_DEBUG", and "Details: [3]|DEBUG|true". To the right of the log is an "Enter Apex Code" window containing the following code:

```
1 Boolean.isTrue = True;
2 Boolean.isFalse = false;
3 System.debug(isTrue || isFalse);
```

At the bottom of the "Enter Apex Code" window are three buttons: "Open Log" (with a checked checkbox), "Execute", and "Execute Highlighted".

## Exercise - 5

The screenshot shows the Salesforce IDE interface. At the top, there is a header bar with the text "Log executeAnonymous @21/02/2023, 11:55:12". Below this is a "Execution Log" section with columns for "Timestamp", "Event", and "Details". A single log entry is shown: "11:55:12:002 USER\_DEBUG [3] DEBUG|true". To the right of the log is an "Enter Apex Code" window containing the following Apex code:

```
1 Date today = Date.today();
2 Date tomorrow = Date.today().addDays(1);
3 System.debug(today != tomorrow);
```

At the bottom of the "Enter Apex Code" window are three buttons: "Open Log" (with a checked checkbox), "Execute", and "Execute Highlighted".

## Exercise - 6

Program for finding the grade based on score

The screenshot shows the Salesforce IDE interface. At the top, there is a header bar with the text "Log executeAnonymous @21/02/2023, 12:00:02". Below this is a "Execution Log" section with columns for "Timestamp", "Event", and "Details". A single log entry is shown: "12:00:02:004 USER\_DEBUG [6] DEBUG|Grade: A". To the right of the log is an "Enter Apex Code" window containing the following Apex code:

```
1 Integer score= 95;
2
3 if (score == 100) {
4     System.debug('Grade: A+');
5 } else if (score >= 90) {
6     System.debug('Grade: A');
7 } else if (score >= 80) {
8     System.debug('Grade: B');
9 } else {
10     System.debug('Grade: Failed');
11 }
```

At the bottom of the "Enter Apex Code" window are three buttons: "Open Log" (with a checked checkbox), "Execute", and "Execute Highlighted".

## Exercise - 7

Program to execute Apex- for loop

The screenshot shows the Salesforce Developer Console interface. At the top, the URL is `resilient-moose-aqn5su-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage`. Below the URL, there are tabs for `BillingObj.apxc` (which is selected), `BillingCustomerRecordInsertion.apxc`, and `Log executeAnonymous @21/02/2023, 12:53:21`. The main area displays the Apex code for `BillingObj.apxc`:

```
1 * public class BillingObj {
2     public static void records(){
3         Billing__c b = new Billing__c();
4         b.Amount_Paid__c=15000;
5         b.Status__c ='Paid';
6         Billing__c b1= new Billing__c();
7         b1.Amount_Paid__c=15000;
8         b1.Status__c ='Paid';
9         Billing__c b2= new Billing__c();
10        b2.Amount_Paid__c=15000;
11        b2.Status__c ='Paid';
12        List<Billing__c> bill = new List<Billing__c>();
13        bill.add(b);
14        bill.add(b1);
15        bill.add(b2);
16        try{
17            insert bill;
18        }
19        catch(Exception e){
20            System.debug('Invalid Data');
21        }
22        bill=[Select Id,Name,Status__c FROM Billing__c WHERE CreatedDate = today];
23        system.debug('Value of Billing List: '+bill);
24        for(Billing__c bb: bill){
25            if(bb.status__c == 'Paid'){
26                system.debug('Value of Current record over which current record is iterating is: '+bb );
27                system.debug('Billing Records are: '+bb.Name);
28            }
29        }
30    }
31 }
```

Below the code, there are tabs for `Logs`, `Tests`, `Checkpoints`, `Query Editor`, `View Status`, `Progress`, and `Problems`. The `Problems` tab is selected. The bottom part of the screenshot shows the `Execution Log` for the anonymous apex execution:

Timestamp	Event	Details
12:54:12:077	USER_DEBUG	[23]DEBUG Value of Billing List: (Billing__c:(Id=a0F2w00000AsCqxEAF, Name=B-0003, Status__c=Paid), Billing__c:(Id=a0F2w00000AsCqxEAF, Name=B-0004, Status__c=Paid), Billing__c:(Id=a0F2w00000AsCqxEAF, Name=B-0005, Status__c=Paid))
12:54:12:077	USER_DEBUG	[25]DEBUG Value of Current record over which current record is iterating is: Billing__c:(Id=a0F2w00000AsCqxEAF, Name=B-0003, Status__c=Paid)
12:54:12:077	USER_DEBUG	[27]DEBUG Billing Records are: B-0003
12:54:12:077	USER_DEBUG	[26]DEBUG Value of Current record over which current record is iterating is: Billing__c:(Id=a0F2w00000AsCqxEAF, Name=B-0004, Status__c=Paid)
12:54:12:077	USER_DEBUG	[27]DEBUG Billing Records are: B-0004
12:54:12:078	USER_DEBUG	[26]DEBUG Value of Current record over which current record is iterating is: Billing__c:(Id=a0F2w00000AsCqxEAF, Name=B-0005, Status__c=Paid)
12:54:12:078	USER_DEBUG	[27]DEBUG Billing Records are: B-0005
12:54:12:078	USER_DEBUG	[26]DEBUG Value of Current record over which current record is iterating is: Billing__c:(Id=a0F2w00000AsCpEAV, Name=B-0001, Status__c=Paid)
12:54:12:078	USER_DEBUG	[27]DEBUG Billing Records are: B-0001
12:54:12:078	USER_DEBUG	[26]DEBUG Value of Current record over which current record is iterating is: Billing__c:(Id=a0F2w00000AsCheEAf, Name=B-0002, Status__c=Paid)
12:54:12:078	USER_DEBUG	[27]DEBUG Billing Records are: B-0002

The screenshot shows the Salesforce Developer Console interface. At the top, the URL is `resilient-moose-aqn5su-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage`. Below the URL, there are tabs for `InterfaceExample.apxc` (which is selected), `BillingObj.apxc`, and `Log executeAnonymous @21/02/2023, 12:54:12`. The main area displays the Apex code for `InterfaceExample.apxc`:

```
1 * public class InterfaceExample {
2     public void testMethod() {
3         System.out.println('Hello World');
4     }
5 }
```

Below the code, there are tabs for `Logs`, `Tests`, `Checkpoints`, `Query Editor`, `View Status`, `Progress`, and `Problems`. The `Problems` tab is selected. The bottom part of the screenshot shows the `Execution Log` for the anonymous apex execution:

Timestamp	Event	Details
12:54:12:077	USER_DEBUG	[23]DEBUG Hello World

## Exercise - 8

Created a Class to demonstrate constants in Apex

The screenshot shows the Salesforce IDE interface. At the top, there are tabs for 'Log executeAnonymous @21/02/2023, 12:10:25' and 'DiscountClass.apxc'. Below the tabs, there are dropdowns for 'Code Coverage: None' and 'API Version: 57'. The main area contains the following Apex code:

```
1 public class DiscountClass {  
2     Decimal regularDiscount = 1.56;  
3     Decimal finalPrice;  
4     public Decimal calculateDiscount(Integer price){  
5         finalPrice = price - price*regularDiscount;  
6         System.debug('finalPrice :'+finalPrice);  
7         return finalPrice;  
8     }  
9 }
```

To the right of the code editor, a modal window titled 'Enter Apex Code' is open, containing the following code:

```
1 DiscountClass x= new DiscountClass();  
2 x.calculateDiscount(200);
```

At the bottom of the modal are three buttons: 'Open Log' (with a checked checkbox), 'Execute', and 'Execute Highlighted'.

The screenshot shows the Salesforce IDE interface. At the top, there are tabs for 'File', 'Edit', 'Debug', 'Test', 'Workspace', 'Help', and two logs: 'Log executeAnonymous @21/02/2023, 12:10:25' and 'Log executeAnonymous @21/02/2023, 12:29:23'. The 'Log executeAnonymous @21/02/2023, 12:29:23' tab is active, showing the 'Execution Log' section with the following table:

Timestamp	Event	Details
12:29:23:016	USER_DEBUG	[6] DEBUG finalPrice :-112.00

To the right of the log, a modal window titled 'Enter Apex Code' is open, containing the same code as the previous modal:

```
1 DiscountClass x= new DiscountClass();  
2 x.calculateDiscount(200);
```

At the bottom of the modal are three buttons: 'Open Log' (with a checked checkbox), 'Execute', and 'Execute Highlighted'.

## Exercise - 9

The screenshot shows the Salesforce Developer Console interface. At the top, the URL is `resilient-moose-aqn5su-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage`. Below the URL, there are tabs for `InterfaceExample.apxc`, `PremiumCustomer.apxc`, and `normalCustomer.apxc`. The status bar indicates `Log executesAnonymous @21/02/2023, 14:50:10`. The code editor contains the following Apex code:

```
1 public interface InterfaceExample {
2     Double percentageDiscountToBeApplied();
3 }
4 }
```

Below the code editor is a table titled "Logs" showing log entries:

User	Application	Operation	Time	Status	Read	Size
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:50:10	Success		3.87 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:48:38	Success		3.86 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:05:04	Success		4.41 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 13:03:15	Success		4.47 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:54:12	Success		10.82 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:29:23	Success		4.47 KB

At the bottom, there is a "Logs" button and a "Click here to filter the log list" link.

The screenshot shows the Salesforce Developer Console interface. At the top, the URL is `resilient-moose-aqn5su-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage`. Below the URL, there are tabs for `InterfaceExample.apxc`, `PremiumCustomer.apxc`, and `normalCustomer.apxc`. The status bar indicates `Log executesAnonymous @21/02/2023, 14:50:10`. The code editor contains the following Apex code:

```
1 public class PremiumCustomer implements InterfaceExample{
2     public Double percentageDiscountToBeApplied(){
3         return 0.30;
4     }
5 }
```

A modal window titled "Enter Apex Code" is open, containing the following code:

```
1 PremiumCustomer x= new PremiumCustomer();
2 Double y=x.percentageDiscountToBeApplied();
3 system.debug(y);
```

Below the code editor is a table titled "Logs" showing log entries:

User	Application	Operation	Time	Status	Read	Size
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:50:10	Success		3.87 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:48:38	Success		3.86 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:05:04	Success		4.41 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 13:03:15	Success		4.47 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:54:12	Success		10.82 KB
Shravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:29:23	Success		4.47 KB

At the bottom, there is a "Logs" button and a "Click here to filter the log list" link.

Developer Console

File Edit Debug Test Workspace Help

InterfaceExample.apxc PremiumCustomer.apxc normalCustomer.apxc Log executeAnonymous @21/02/2023, 14:50:10 Log executeAnonymous @21/02/2023, 14:58:58 Log executeAnonymous @21/02/2023, 15:00:44

**Execution Log**

Timestamp	Event	Details
15:00:44:007	USER_DEBUG	[3] DEBUG 0.3

**Enter Apex Code**

```
1 PremiumCustomer x= new PremiumCustomer();
2 Double y=x.percentageDiscountTobeApplied();
3 system.debug(y);
```

Open Log

This Frame Executable  Debug Only Filter Click here to filter the log

**Logs** Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 15:00:44	Success		3.82 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:58:58	Success		3.85 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:50:10	Success		3.87 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:48:38	Success		3.86 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:05:04	Success		4.41 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 13:03:15	Success		4.47 KB

Filter Click here to filter the log list

Developer Console

File Edit Debug Test Workspace Help

InterfaceExample.apxc PremiumCustomer.apxc normalCustomer.apxc Log executeAnonymous @21/02/2023, 14:50:10 Log executeAnonymous @21/02/2023, 14:58:58 Log executeAnonymous @21/02/2023, 15:00:44

Code Coverage: None API Version: 57 Go To

```
1 public class normalCustomer implements InterfaceExample{
2     public Double percentageDiscountTobeApplied(){
3         return 0.10;
4     }
5 }
```

**Enter Apex Code**

```
1 normalCustomer x= new normalCustomer();
2 Double y=x.percentageDiscountTobeApplied();
3 system.debug(y);
```

Open Log

**Logs** Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 15:00:44	Success		3.82 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:58:58	Success		3.85 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:50:10	Success		3.87 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:48:38	Success		3.86 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:05:04	Success		4.41 KB
Sravan Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 13:03:15	Success		4.47 KB

Filter Click here to filter the log list

The screenshot shows the Salesforce Developer Console interface. At the top, there's a navigation bar with links like File, Edit, Debug, Test, Workspace, Help, and a search bar. Below the navigation bar, there are tabs for InterfaceExample.apxc, PremiumCustomer.apxc, and normalCustomer.apxc. A status bar at the bottom indicates "Log executeAnonymous @21/02/2023, 14:50:10" and "Log executeAnonymous @21/02/2023, 14:58:58".

**Execution Log:**

Timestamp	Event	Details
14:58:58:013	USER_DEBUG	[3]:DEBUG 0.1

**Enter Apex Code:**

```

1  normalCustomer x= new normalCustomer();
2  Double y=x.percentageDiscountToBeApplied();
3  system.debug(y);

```

Buttons below the code editor: Open Log, Execute, Execute Highlighted.

**Logs:**

User	Application	Operation	Time	Status	Read	Size
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:58:58	Success		3.85 KB
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:50:10	Success		3.87 KB
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:48:38	Success		3.86 KB
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 14:05:04	Success		4.41 KB
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 13:03:15	Success		4.47 KB
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:51:12	Success		10.82 KB

Filter: Click here to filter the log list

## Exercise - 10

### Demo on DML Insert Operation Using Database Methods

The screenshot shows the Salesforce Developer Console interface. At the top, there's a navigation bar with links like File, Edit, Debug, Test, Workspace, Help, and a search bar. Below the navigation bar, there are tabs for BillingCustomerRecordInsertion.apxc. A status bar at the bottom indicates "Log executeAnonymous @22/02/2023, 11:07:53".

**Logs:**

User	Application	Operation	Time	Status	Read	Size
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	22/02/2023, 11:07:53	Success		7.01 KB
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	22/02/2023, 11:07:04	Insert failed. First exception on row ...	Unread	3.95 KB
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	22/02/2023, 11:06:40	Insert failed. First exception on row ...	Unread	3.97 KB

**Logs:**

User	Application	Operation	Time	Status	Read	Size
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	22/02/2023, 11:07:53	Success		7.01 KB
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	22/02/2023, 11:07:04	Insert failed. First exception on row ...	Unread	3.95 KB
Shravani Kumar	Unknown	/services/data/v57.0/tooling/execut...	22/02/2023, 11:06:40	Insert failed. First exception on row ...	Unread	3.97 KB

Filter: Click here to filter the log list

The screenshot shows the Salesforce Developer Console with the URL [resilient-moose-aqn5su-dev-ed.trailblaze.my.salesforce.com/\\_ui/common/apex/debug/ApexCSIPage](https://resilient-moose-aqn5su-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage). The title bar indicates the log was executed at 22/02/2023, 11:07:53. The Execution Log table has columns: Timestamp, Event, and Details. A single entry is shown: 11:07:53:111 USER\_DEBUG [15] DEBUG Record Inserted , Billing ID: a0F2w00000Auk03EAV.

## Exercise - 11

Displaying Records According to the required conditions

The screenshot shows the Salesforce Developer Console with the URL [brave-shark-bb5n2h-dev-ed.trailblaze.my.salesforce.com/\\_ui/common/apex/debug/ApexCSIPage](https://brave-shark-bb5n2h-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage). The title bar indicates the query was run at 11:38 AM. The query is: Select AccountId, Amount, Name, StageName, Account.Industry, Account.Website FROM Opportunity where Account.Industry = 'Energy' AND Account.AnnualRevenue > 5000. The results table has columns: AccountId, Amount, Name, StageName, Account.Industry, and Account.Website. There are 10 rows returned, each representing an opportunity from the United Oil company.

AccountId	Amount	Name	StageName	Account.Industry	Account.Website
0012w000011jDLeANN	125000	United Oil Portable Generators	Negotiation/Review	Energy	<a href="http://www.uos.com">http://www.uos.com</a>
0012w000011jDLeANN	270000	United Oil Refinery Generators	Proposal/Price Quote	Energy	<a href="http://www.uos.com">http://www.uos.com</a>
0012w000011jDLeANN	120000	United Oil SLA	Closed Won	Energy	<a href="http://www.uos.com">http://www.uos.com</a>
0012w000011jDLeANN	270000	United Oil Installations	Negotiation/Review	Energy	<a href="http://www.uos.com">http://www.uos.com</a>
0012w000011jDLeANN	270000	United Oil Installations	Closed Won	Energy	<a href="http://www.uos.com">http://www.uos.com</a>
0012w000011jDLeANN	915000	United Oil Refinery Generators	Closed Won	Energy	<a href="http://www.uos.com">http://www.uos.com</a>
0012w000011jDLeANN	235000	United Oil Installations	Closed Won	Energy	<a href="http://www.uos.com">http://www.uos.com</a>
0012w000011jDLeANN	440000	United Oil Emergency Generators	Closed Won	Energy	<a href="http://www.uos.com">http://www.uos.com</a>
0012w000011jDLeANN	120000	United Oil Standby Generators	Closed Won	Energy	<a href="http://www.uos.com">http://www.uos.com</a>
0012w000012jDLeANN	675000	United Oil Plant Standby Generators	Needs Analysis	Energy	<a href="http://www.uos.com">http://www.uos.com</a>

Query Grid: Save Rows | Insert Row | Delete Row | Refresh Grid | Access in Salesforce: Create New | Open Detail Page | Edit Page

Logs | Tests | Checkpoints | **Query Editor** | View State | Progress | Problems

Select AccountId,Amount,Name,StageName,Account.Industry,Account.Website FROM Opportunity where Account.Industry = 'Energy' AND Account.AnnualRevenue > 5000

Any query errors will appear here...

History

Executed

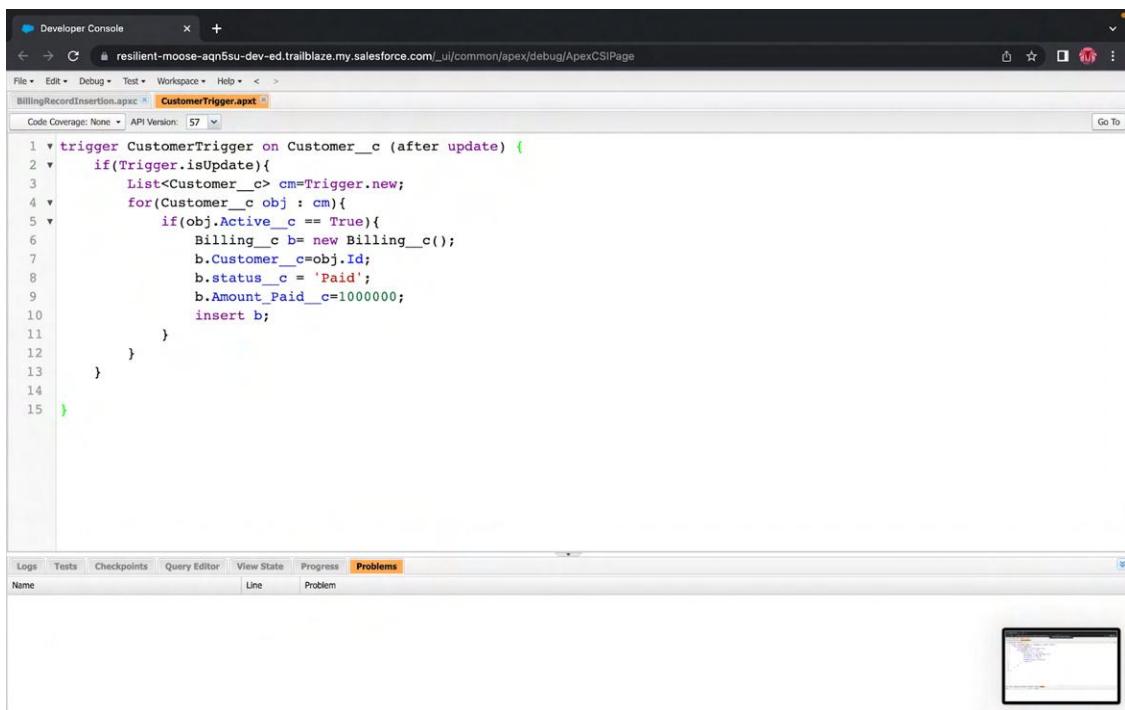
SELECT Id,Name from Account

Select AccountId,Amount,Name,StageName,Account.Industry,Account.Website...

Execute  Use Tooling API

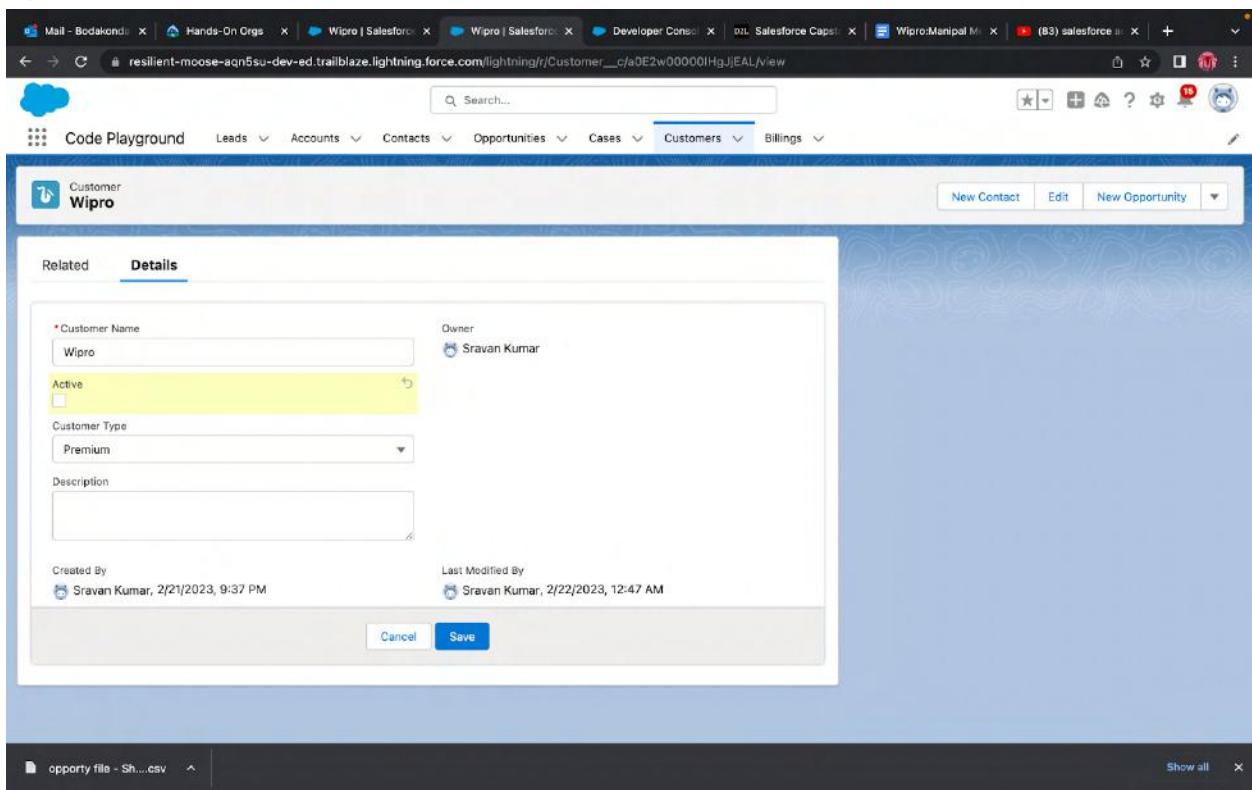
## Exercise - 12

Created a Trigger for Billing named CustomTrigger

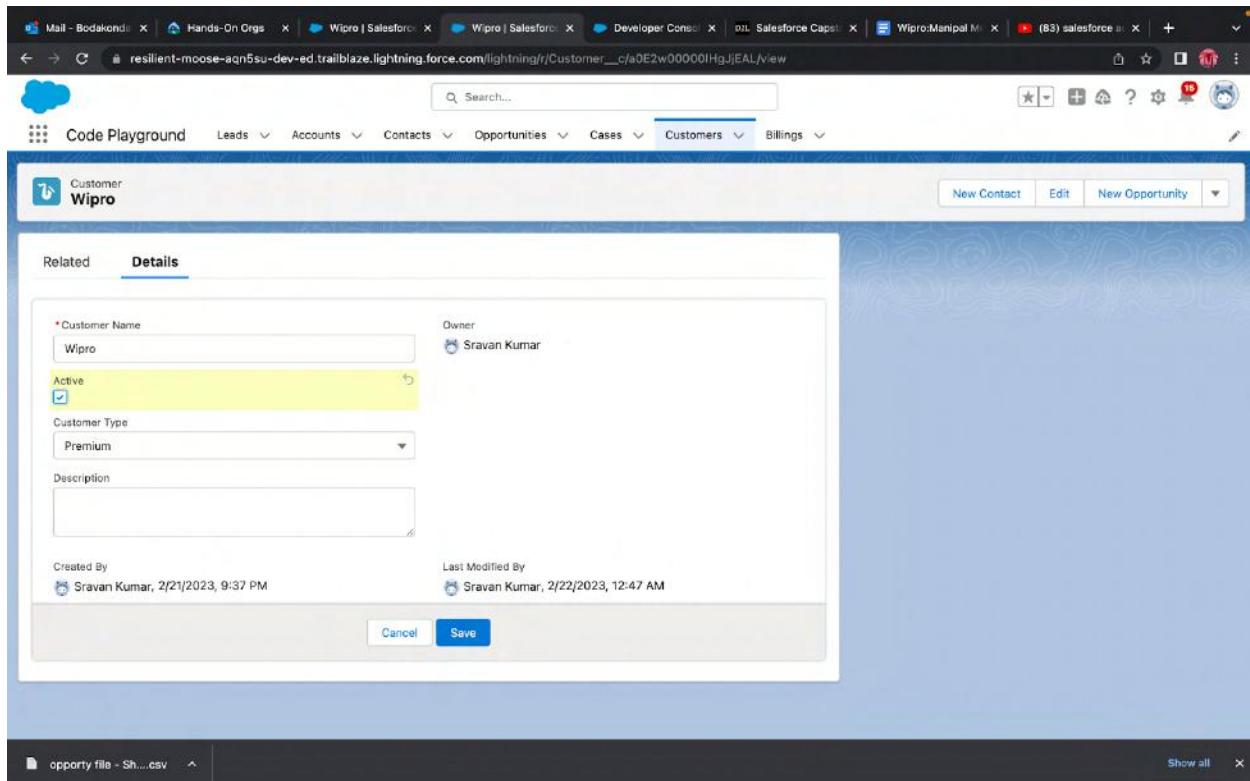


```
trigger CustomerTrigger on Customer__c (after update) {
    if(Trigger.isUpdate){
        List<Customer__c> cm=Trigger.new;
        for(Customer__c obj : cm){
            if(obj.Active__c == True){
                Billing__c b = new Billing__c();
                b.Customer__c=obj.Id;
                b.status__c = 'Paid';
                b.Amount_Paid__c=1000000;
                insert b;
            }
        }
    }
}
```

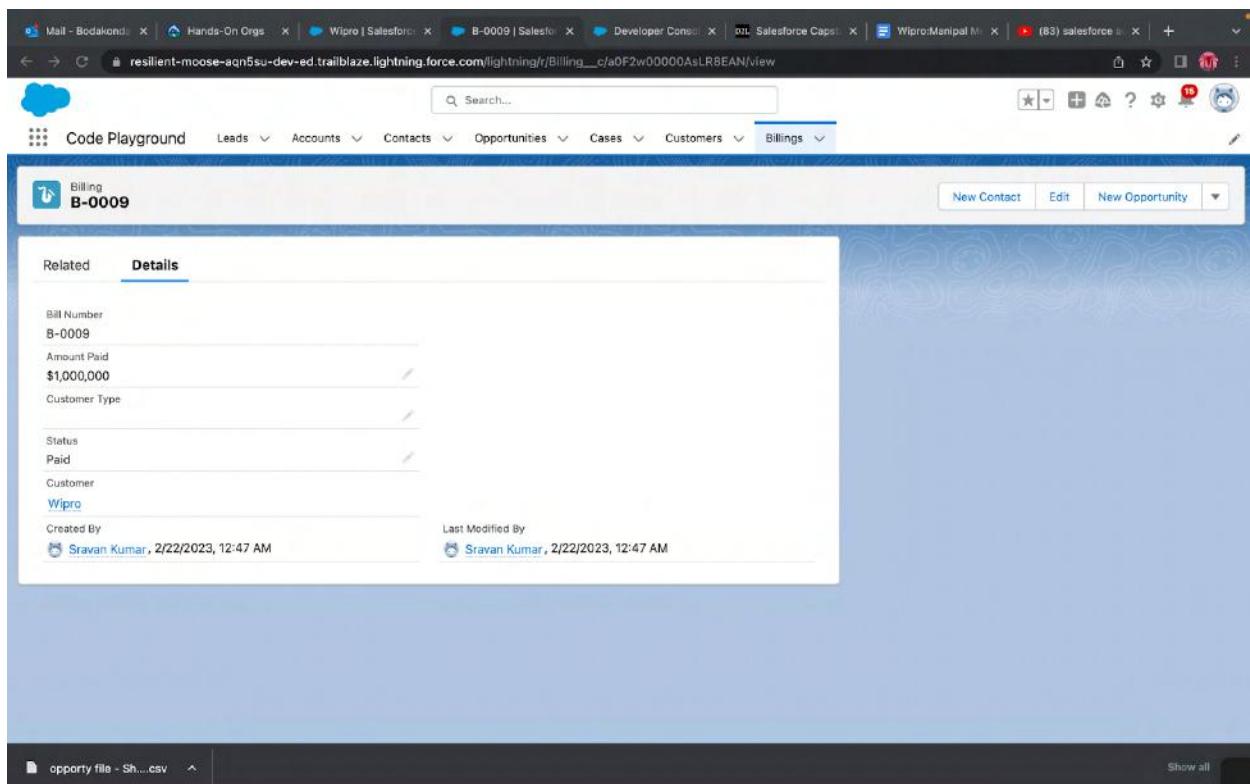
Update Salesforce Customer from in active to active



The screenshot shows a Salesforce Lightning Experience page for a Customer record. The customer name is "Wipro" and the status is "Active". The "Active" checkbox is checked. The "Customer Type" is set to "Premium". There is a "Description" field which is empty. The "Created By" field shows "Sravan Kumar, 2/21/2023, 9:37 PM". The "Last Modified By" field shows "Sravan Kumar, 2/22/2023, 12:47 AM". At the bottom of the form, there are "Cancel" and "Save" buttons.



## New Record Created in Billing Object with Salesforce as Customer



## Exercise - 13

Created a Test class for CustomTrigger

The screenshot shows the Salesforce Developer Console with the code editor open for `CustomerTriggerTest.apex`. The code implements a test class for a custom trigger named `CustomerTrigger`. It includes logic for creating and updating a `Customer__c` record, and then asserting that the billing information has been updated correctly after the update.

```
1  @isTest
2  * public class CustomerTriggerTest {
3      @isTest
4      static void testCustomer(){
5          //Creating Customer
6          Customer__c cust1 = new Customer__c();
7          cust1.Name='test1';
8          cust1.Customer_Type__c = 'Premium';
9          insert cust1;
10
11         //Updating Customer
12         cust1.Active__c = True;
13         update cust1;
14
15         //After Updating, changing the billing information
16         if(cust1.Active__c == True){
17             Test.startTest();
18             Billing__c billing = new Billing__c();
19             billing.Status__c = 'Paid';
20             billing.Amount_Paid__c = 1000000;
21             Test.stopTest();
22             system.assertEquals('Paid', billing.Status__c);
23             system.assertEquals(1000000, billing.Amount_Paid__c);
24         }
25     }
26 }
```

Below the code editor is the log tab, which displays the following log entries:

User	Application	Operation	Time	Status	Read	Size
Steven Kumar	Unknown	/services/data/v57.0/tooling/runTestsSyn...	22/02/2023, 14:09:30	Success	Unread	8.71 KB
Steven Kumar	Unknown	/services/data/v57.0/tooling/runTestsSyn...	22/02/2023, 14:28:23	Success	Unread	2.86 KB
Steven Kumar	Browser	/aura	22/02/2023, 14:17:08	Success	Unread	4.46 KB
Steven Kumar	Browser	/aura	22/02/2023, 14:17:03	Success	Unread	3.52 KB
Steven Kumar	Browser	/aura	22/02/2023, 14:14:20	Success	Unread	4.56 KB
Steven Kumar	Unknown	/services/data/v57.0/tooling/executeAste...	22/02/2023, 14:09:27	Success	Unread	5.56 KB

## Exercise - 14

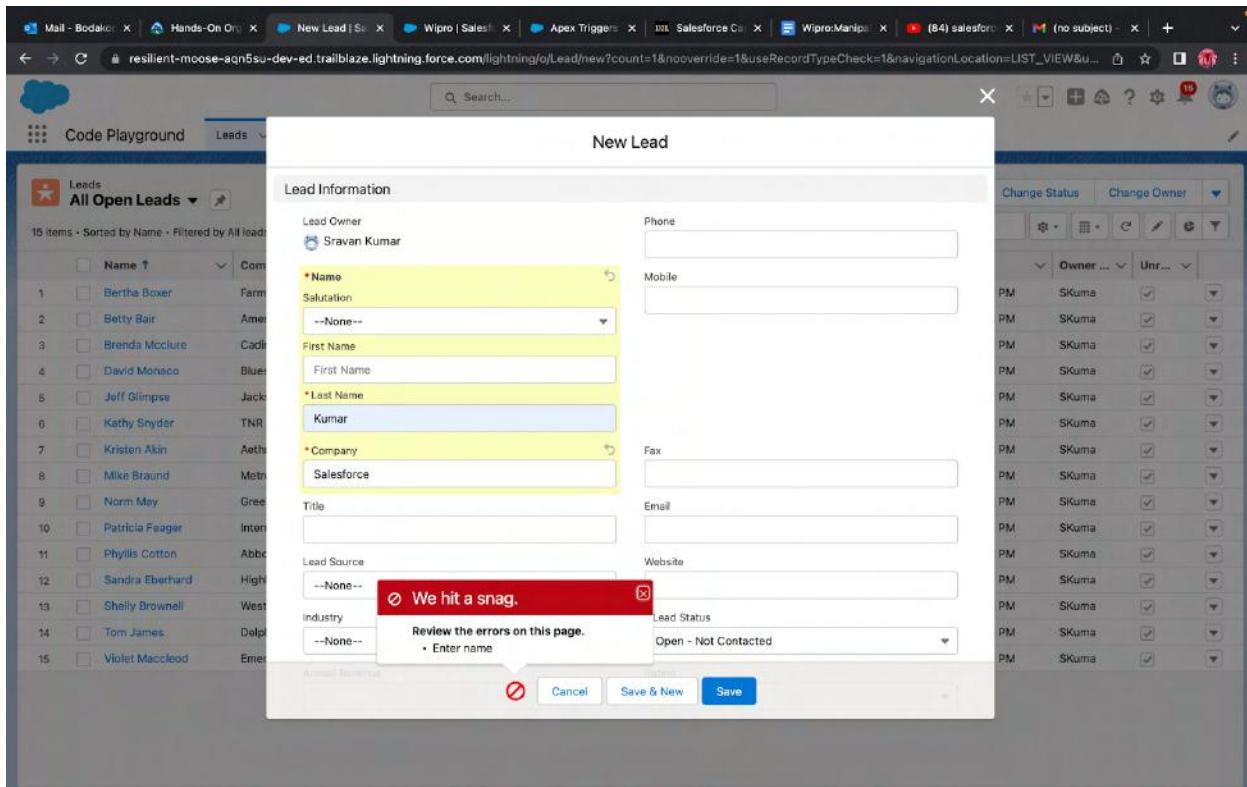
Created a Trigger for Leads Named DisqualifyTestLeads

The screenshot shows the Salesforce Developer Console with the code editor open for `DisqualifyTestLeads.apex`. The code implements a trigger named `DisqualifyTestLeads` that runs on the `Lead` object. It checks if the first name or last name of a lead starts with 'test'. If it does, the lead is marked as 'Disqualified'.

```
1  * trigger DisqualifyTestLeads on Lead (after insert,before insert,before update) {
2      List<Lead> ld= new List<Lead>();
3      for(Lead mylead:trigger.new){
4          if(mylead.FirstName=='test' && mylead.FirstName.length()!=0 || mylead.LastName=='test' && mylead.LastName.length()!=0){
5              System.debug(mylead.FirstName+' '+mylead.LastName+' '+ 'Will be Disqualified');
6              ld.add(mylead);
7          }
8          if(mylead.FirstName==null){
9              mylead.addError('Enter name');
10         }
11     }
12     for(Lead mylead2:ld){
13         mylead2.Status ='Disqualified';
14     }
15 }
```

Below the code editor is the log tab, which displays the following log entries:

User	Application	Operation	Time	Status	Read	Size
Steven Kumar	Unknown	ApexTestHandler	22/02/2023, 15:14:39	Success	Unread	3012 bytes
Steven Kumar	Unknown	ApexTestHandler	22/02/2023, 15:14:38	Success	Unread	4.03 KB
Steven Kumar	Unknown	ApexTestHandler	22/02/2023, 15:14:35	Success	Unread	8.75 KB
Steven Kumar	Unknown	ApexTestHandler	22/02/2023, 15:14:33	Assertion Failed: incorrect ticker symbol ...	Unread	9.04 KB
Steven Kumar	Unknown	ApexTestHandler	22/02/2023, 15:14:35	Success	Unread	20.97 KB
Steven Kumar	Unknown	ApexTestHandler	22/02/2023, 15:14:35	Insert failed. First exception on row 0; f...	Unread	326.81 KB



## Exercise - 15

Created a test class for DisqualifyTestLeads trigger

```

trigger DisqualifyTestLeads on Lead (after insert, before update) {
    List<Lead> ld = new List<Lead>();
    for(Lead mylead:Trigger.new){
        if(mylead.FirstName=='test' && mylead.FirstName.length()!=0 || mylead.LastName=='test' && mylead.LastName.length()!=0){
            System.debug(mylead.FirstName+' '+mylead.LastName+' will be Disqualified');
            ld.add(mylead);
        }
        if(mylead.FirstName==null){
            mylead.addError('Enter name');
        }
    }
    for(Lead mylead2:ld){
        mylead2.Status = 'Disqualified';
    }
}

```

Status	Test Run	Enqueued Time	Duration	Failures	Total
Logs	7972v000000L5jw	Fri Feb 23 2023 14:50:54 GM...		1	1
Test	7972v000000L5jw			0	1
Checkpoints	7972v000000L5jw			0	1
Query Editor	7972v000000L5jw			0	1
View Status	7972v000000L5jw			0	1
Progress	7972v000000L5jw			0	1
Problems	7972v000000L5jw			0	1

Overall Code Coverage	Percent	Lines
Class	100%	26/28
DisqualifyTestLeads	100%	6/6
OrderEventTrigger	0%	0/1
NewLeadController	0%	6/6
NormalController	0%	6/6
OrderEventTrigger	0%	6/6
OrderEventTrigger	88%	26/28

```

1
2  @isTest
3  public class DisqualifyTestLeads {
4
5      @isTest
6      private static void insertData(){
7          Lead l1 = new Lead();
8          l1.FirstName = 'Test';
9          l1.LastName = 'Test';
10         l1.Company = 'Test';
11         Test.startTest();
12         insert l1;
13         Test.stopTest();
14         System.assertEquals('Test', l1.FirstName);
15         System.assertEquals('Test', l1.LastName);
16         System.assertEquals('Test', l1.Company);
17     }
18     @isTest
19     private static void insertData2(){
20         Lead l2 = new Lead();
21         l2.LastName = 'Test';
22         l2.Company = 'Wipro';
23         Test.startTest();
24         insert l2;
25         Test.stopTest();
26         System.assertEquals(null, l2.FirstName);
27         System.assertEquals('Test', l2.LastName);
28         System.assertEquals('Test', l2.Company);
29     }
30
31

```

Logs Tests Checkpoints Query Editor View State Progress Problems

Status	Test Run	Enqueued Time	Duration	Failures	Total	Overall Code Coverage		
*	TestRun @ 2:48:31 pm	Thu Feb 23 2023 14:50:54 GMT+05:30 (India Standard Time)		1	1	<b>Class</b>	<b>Percent</b>	<b>Lines</b>
○	7072w00001Lduo			0	1	DisqualifyTestLeads	100%	0/0
○	Dashboard_DashboardPageTest			0	1	LeadProcessor	100%	0/0
○	WhatisIt_UtControllerTest			0	2	NewLeadController	0%	0/0
*	AccountsServiceTest			1	1	NameCustomer	0%	0/2
○	BillingCalloutServiceMock			0	0	OrderEventTrigger	0%	0/0
○	BillingCalloutServiceTest			0	2	OrderEventTrigger	0%	0/0
○	PushMessageProcessorTest			0	1	OrderEventTrigger	0%	0/0

## Exercise - 16

Created a VFP OppView and displayed four apex: outfields

```

1 <apex:page standardController="Opportunity">
2     <apex:pageBlock title="Opportunities">
3         <apex:pageBlockSection>
4             <apex:outputField value="{! Opportunity.Name}"/>
5             <apex:outputField value="{! Opportunity.Amount}"/>
6             <apex:outputField value="{! Opportunity.CloseDate}"/>
7             <apex:outputField value="{! Opportunity.Account.Name}"/>
8         </apex:pageBlockSection>
9     </apex:pageBlock>
10 </apex:page>

```

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
Sravan Kumar	Browser	/apex/OppView	22/02/2023, 15:23:37	Success	Unread	1.19 KB
Sravan Kumar	Browser	/apex/OppView	22/02/2023, 15:22:47	Success	Unread	1.19 KB
Sravan Kumar	Browser	/apex/OppView	22/02/2023, 15:22:42	Success	Unread	1.19 KB
Sravan Kumar	Browser	/apex/OppView	22/02/2023, 15:21:50	Success	Unread	1.19 KB
Sravan Kumar	Browser	/apex/OppView	22/02/2023, 15:21:41	Success	Unread	1.2 KB
Sravan Kumar	Browser	/apex/OppView	22/02/2023, 15:21:39	Success	Unread	1.2 KB

Filter Click here to filter the log list

Opportunities

Opportunity Name	Close Date	Amount	Account Name

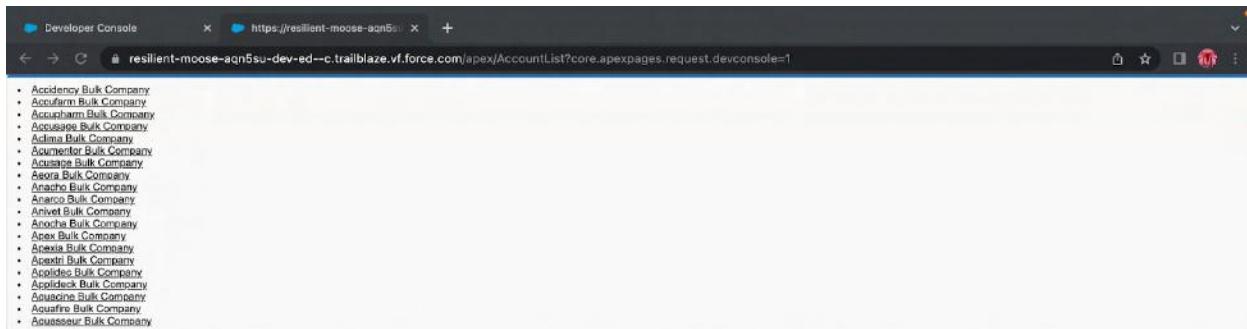
## Exercise - 17

Created a VFP that shows a list of Accounts linked to their record pages

```
1 <apex:page standardController="Account" recordSetVar="Accounts" >
2   <apex:pageBlock>
3     <apex:repeat var="a" value="{!Accounts}" rendered="true" id="account_list">
4       <li>
5         <apex:outputLink value="/{!a.ID}">
6           <apex:outputText value="{!a.Name}" />
7         </apex:outputLink>
8       </li>
9     </apex:repeat>
10   </apex:pageBlock>
11 
12 </apex:page>
```

Logs Tests Checkpoints Query Editor View State Progress Problems

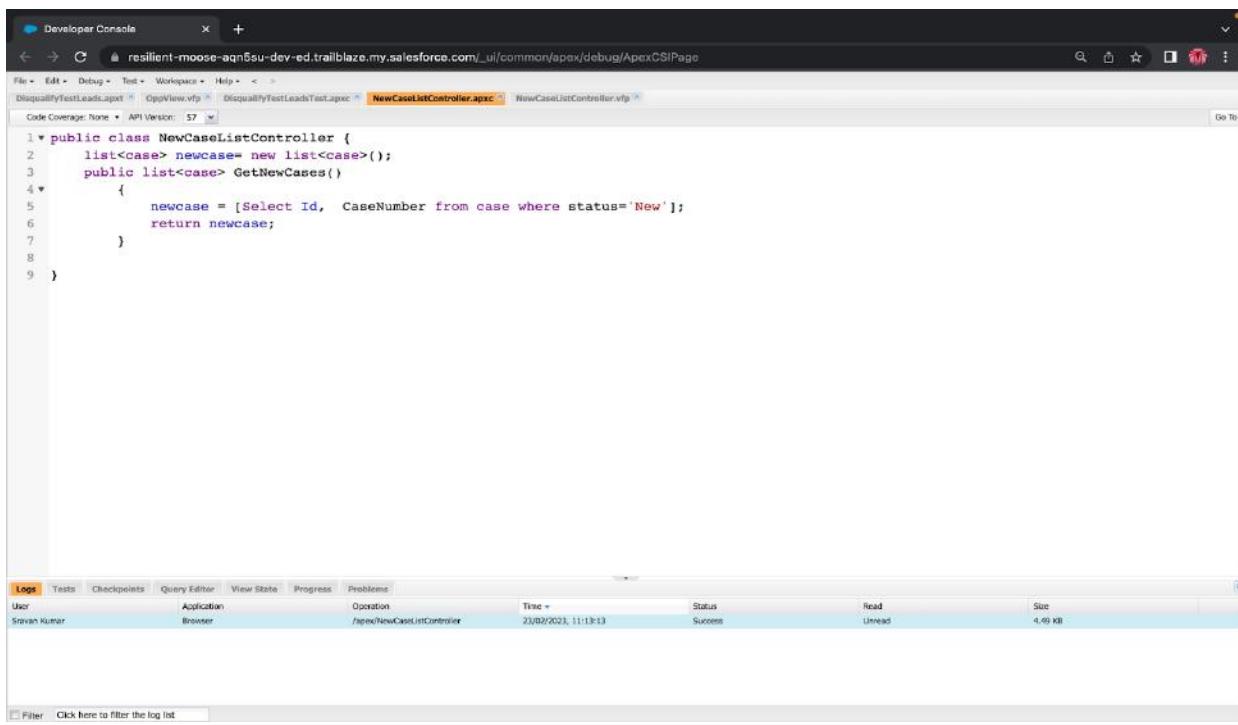
Name	Line	Problem
Dis qualifyTestLeadsTest	10	Method does not exist or incorrect signature: void add(Integer) from the type List
Dis qualifyTestLeadsTest	14	Method does not exist or incorrect signature: void insert(Integer, Boolean) from the type Database
Dis qualifyTestLeadsTest	15	Method does not exist or incorrect signature: void assert(String, String) from the type System



A screenshot of a web browser window titled "Developer Console". The address bar shows the URL: <https://resilient-moose-aqn5su-dev-ed-c.trailblaze.vf.force.com/apex/AccountList?core.apexpages.request.devconsole=1>. The page content displays a list of company names, all of which appear to be variations of "Acme Bulk Company" followed by a hyphen and a random string of letters (e.g., "AcmebulkCompany-12345").

## Exercise - 18

Created a VFP that uses an apex class to display a list of cases with the status of 'New'.



A screenshot of the Salesforce Developer Console. The top navigation bar includes "File", "Edit", "Debug", "Test", "Workspaces", "Help", and tabs for "Logs", "Tests", "Checkpoints", "Query Editor", "View State", "Progress", and "Problems". The main area shows the code for the `NewCaseListController` class:

```
1 * public class NewCaseListController {
2     list<case> newcase= new list<case>();
3     public list<case> GetNewCases()
4     {
5         newcase = [Select Id, CaseNumber from case where status='New'];
6         return newcase;
7     }
8 }
```

Below the code, a "Logs" tab is selected, showing a single log entry:

User	Application	Operation	Time	Status	Read	Size
Sreeram Kumar	Browser	/apex/NewCaseListController	23/02/2023, 11:13:13	Success	Unread	4.46 KB

At the bottom, there is a "Filter" input field with the placeholder "Click here to filter the log list".

The screenshot shows the Salesforce Developer Console interface. The top navigation bar includes File, Edit, Debug, Test, Workspace, Hop, and a search bar. Below the bar, several tabs are visible: DisqualifyTestLeads.apex, ObjView.vfp, DisqualifyTestLeadsTest.apc, NewCaseListController.apc, and NewCaseListController.vfp (which is currently selected). A dropdown menu indicates the API Version is 57. The main content area displays the Apex Visualforce page code:

```
1 * <apex:page controller="NewCaseListController" >
2 *     <apex:repeat var="case" value="{!!NewCases}">
3 *         <li>
4             <apex:outputLink value="/{!case.id}">{!case.id}</apex:outputLink>
5             {!case.CaseNumber}
6         </li>
7     </apex:repeat>
8 </apex:page>
```

This screenshot shows the Log tab in the Salesforce developer console. The log table has columns for User, Application, Operation, Time, Status, Read, and Size. One entry is listed:

User	Application	Operation	Time	Status	Read	Size
Sreeram Kumar	Browser	/apex/NewCaseListController	23/02/2023, 11:13:13	Success	Unread	4.49 KB

A preview window on the right shows a screenshot of the application interface.

This screenshot shows the browser history in the Salesforce Developer Console. The address bar shows the URL: https://resilient-moose-aqn5su-dev-ed-c.trailblaze.vf.force.com/apex/NewCaseListController?core.apexpages.request.devconsole=1. The history list contains three entries:

- 5002ad00000d0CyAAE 00001002
- 5002ad00000d0CyAAE 00001016
- 5002ad00000d0Cz5AAE 00001024

## **References**

- 1.[Salesforce](#)
- 2.[SalesForce Admin](#)
- 3.[Trailhead Salesforce](#)
- 4.[Apex](#)