



Capstone Project Report
On
“Travel Approval App”

Submitted by:

Name: Botta Naveen
Batch WIP-SF-13
LMS Id: MGSA_523
Date: 27/02/2023

Table of contents

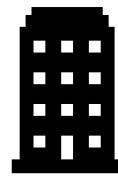
- 1. Abstract.....
- 2. Introduction.....
- 3. Flow of the project.....
- 4. Software Requirements.....
- 5. Screen shots.....
- 6. References.....

Abstract

This report outlines the development of a travel approval app for Salesforce that streamlines the process of requesting and approving business travel requests. The app allows employees to easily submit travel requests through Salesforce, which are then automatically routed to their manager for approval. Managers can review and approve requests in real-time, as well as track the travel expenses and budgets associated with each trip. The app is designed to improve the efficiency of the travel approval process, reduce errors and delays, and provide greater visibility and control over travel expenses. Overall, the travel approval app for Salesforce is a valuable tool for any organization looking to streamline their travel request process and better manage their travel expenses.

Introduction

Travel is an essential component of business operations, but managing the process of requesting and approving travel can be complex and time-consuming. This is especially true for large organizations with numerous employees traveling for work. To address this challenge, a travel approval app for Salesforce has been developed to streamline the travel approval process and improve the efficiency of managing travel requests and expenses.



The travel approval app for Salesforce is a web-based application that integrates with the Salesforce platform to provide a seamless and user-friendly experience for employees and managers. The app automates the travel request process, allowing employees to easily submit requests and managers to quickly approve them. Additionally, the app provides visibility and control over travel expenses and budgets, enabling managers to make informed decisions and ensure compliance with company policies.

This report provides an overview of the development of the travel approval app for Salesforce, including its features, benefits, and implementation process. The report also discusses the potential impact of the app on travel management within organizations and explores future enhancements and developments for the app.

Overall, the travel approval app for Salesforce is a valuable tool for any organization looking to streamline their travel request process and better manage their travel expenses.

Flow of the Project

The travel approval app for Salesforce follows a streamlined process that simplifies the travel request and approval process for employees and managers. The flow of the app can be broken down into several steps, which are outlined below:

1. **Travel Request Submission:** Employees initiate the travel request process by submitting a request through the travel approval app in Salesforce. The request includes details such as the purpose of the trip, travel dates, destinations, and estimated expenses.
2. **Manager Review and Approval:** The travel request is automatically routed to the employee's manager for review and approval. The manager can review the details of the trip, including estimated expenses, and either approve or reject the request.
3. **Budget and Expense Management:** Once a request is approved, the app tracks the expenses associated with the trip, including flights, accommodations, and other travel-related expenses. The app also provides visibility into the available travel budget for the employee or team, ensuring that the trip stays within the approved budget.
4. **Travel Booking and Itinerary Management:** Once a travel request is approved, employees can book their travel and manage their itinerary directly within the travel approval app. The app provides a central location for all travel-related information, making it easy for employees to stay organized and informed about their trip.

5. Travel Expense Reimbursement: After the trip is completed, employees can submit their travel expenses through the travel approval app. The app automatically calculates the total expenses and routes the expense report to the appropriate department for reimbursement.

Overall, the travel approval app for Salesforce streamlines the travel request and approval process, simplifies expense management, and provides visibility and control over travel budgets. By automating these processes, organizations can reduce errors and delays, improve efficiency, and ensure compliance with company policies.

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

Module 1

Exercise 1

Step 1

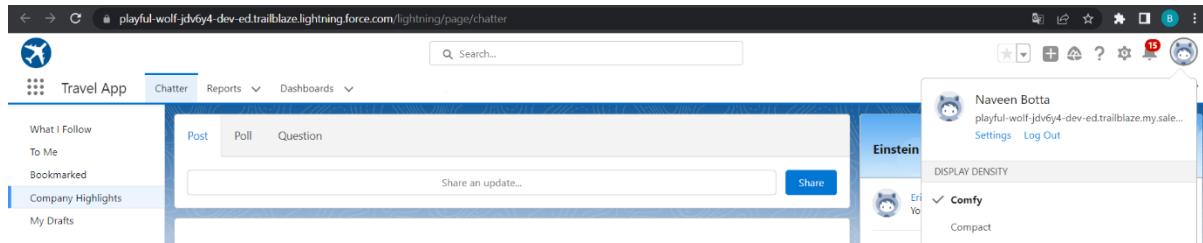
Creating Lightning App: Travel App.

The screenshot shows the 'App Details & Branding' section of the Visualforce Editor. The 'App Name' is set to 'Travel App', the 'Developer Name' is 'Travel_App', and the 'Primary Color Hex Value' is '#0070D2'. The 'Image' field contains a blue airplane icon. The 'Org Theme Options' checkbox is checked. The 'App Launcher Preview' shows the travel app icon and the name 'Travel App'.

Select and add Navigation items to app

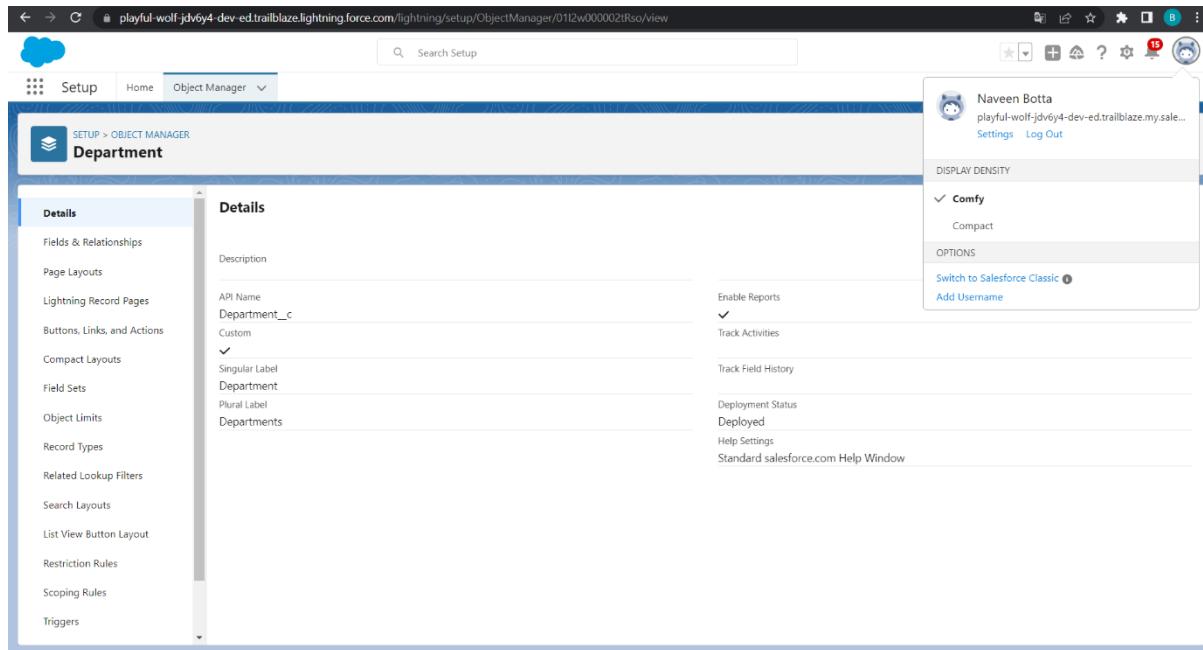
The screenshot shows the 'Navigation Items' section of the Visualforce Editor. The 'Available Items' list includes Accounts, Alert Settings, All Sites, Alternative Payment Methods, App Launcher, Approval Requests, Asset Action Sources, Asset Actions, Asset State Periods, Assets, and Account Creation Logs. The 'Selected Items' list contains Chatter, Reports, and Dashboards.

Giving user profile as System Administrator



Step 2

Custom object Department



Step 3

Creating custom fields in Department object

Department Code Field

The screenshot shows the Salesforce Setup interface for creating a custom field. The left sidebar is titled 'Object Manager' and lists various options like Details, Fields & Relationships, Page Layouts, etc. The main area is titled 'Department Custom Field' and shows the 'Department Code' field definition. The 'Field Information' section includes the field label 'Department Code', field name 'Department_Code', API name 'Department_Code_c', and data type 'Text'. The 'General Options' section has 'Required' checked and 'Case Sensitive' checked. The 'Text Options' section has a length of 10. The top right corner shows a user profile for Naveen Botta and a 'DISPLAY DENSITY' dropdown set to 'Comfy'.

Location Field

The screenshot shows the Salesforce Setup interface for creating a custom field. The left sidebar is titled 'Object Manager' and lists various options like Details, Fields & Relationships, Page Layouts, etc. The main area is titled 'Department Custom Field' and shows the 'Location' field definition. The 'Field Information' section includes the field label 'Location', field name 'Location', API name 'Location__c', and data type 'Picklist'. The 'General Options' section has 'Default Value' set to 'None'. The 'Picklist Options' section has 'Restrict picklist to the values defined in the value set' checked and 'Controlling Field' set to '[New]'. The top right corner shows a user profile for Naveen Botta and a 'DISPLAY DENSITY' dropdown set to 'Comfy'.

Location Picklist Values

The screenshot shows the Salesforce Object Manager interface for the 'Department' object. The 'Fields & Relationships' tab is selected. In the 'Picklist Options' section, there is a checked checkbox 'Restrict picklist to the values defined in the value set'. The 'Controlling Field' is set to '[New]'. The 'Field Dependencies' section lists 'Action' as 'Dependent Field' and 'Data Type' as 'Picklist'. The 'Validation Rules' section shows 'No validation rules defined'. The 'Values' section displays two entries: 'Kolkata' and 'Delhi', both assigned dynamically. The 'Inactive Values' section shows 'No Inactive Values values defined'. A sidebar on the right shows the user's profile (Naveen Botta) and display density settings ('Comfy').

Department Type Field

The screenshot shows the Salesforce Object Manager interface for the 'Department' object. The 'Fields & Relationships' tab is selected. The 'Custom Field Definition Detail' section shows the 'Field Information' for the 'Department Type' field. The field has a label 'Department Type', name 'Department_Type', and API name 'Department_Type_c'. It is a picklist type field. The 'General Options' section shows 'Required' is unchecked and 'Default Value' is set to '[None]'. In the 'Picklist Options' section, there is a checked checkbox 'Restrict picklist to the values defined in the value set'. The 'Controlling Field' is set to 'Location [Champ95]'. The 'Picklist Values Used' section shows 'Active and Inactive picklist values' with a count of '2 (1,000 max)'. A sidebar on the right shows the user's profile (Naveen Botta) and display density settings ('Comfy').

Department Type Picklist

The screenshot shows the Salesforce Object Manager interface for the 'Department' object. The 'Fields & Relationships' tab is selected. In the 'Values' section, there is a table listing department types:

Action	Values	API Name	Default	Chart Colors	Modified By	Modified
<input type="checkbox"/> Edit Del Deactivate	Banking	Banking	<input type="checkbox"/>	Assigned dynamically	Naveen Botta	2/20/2023, 7:38 AM
<input type="checkbox"/> Edit Del Deactivate	Finance	Finance	<input type="checkbox"/>	Assigned dynamically	Naveen Botta	2/20/2023, 7:38 AM
<input type="checkbox"/> Edit Del Deactivate	Education	Education	<input type="checkbox"/>	Assigned dynamically	Naveen Botta	2/20/2023, 7:38 AM
<input type="checkbox"/> Edit Del Deactivate	Energy	Energy	<input type="checkbox"/>	Assigned dynamically	Naveen Botta	2/20/2023, 7:38 AM
<input type="checkbox"/> Edit Del Deactivate	IT	IT	<input type="checkbox"/>	Assigned dynamically	Naveen Botta	2/20/2023, 7:38 AM

Field Dependency

The screenshot shows the 'Edit Field Dependency' screen for the 'Department' object. It displays a grid where rows represent 'Location' and columns represent 'Department Type'. The 'Banking' row is highlighted in yellow, indicating it is selected.

Location:	Kolkata	Delhi
Department Type:	Banking	Banking
	Finance	Finance
	Education	Education
	Energy	Energy
	IT	IT

Step 4

Creating Custom Object Travel Approval

The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main title is 'Travel Approval'. On the left, a sidebar lists various configuration options under '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, and Triggers. The right pane displays the 'Details' section for the 'Travel Approval' object. It shows the API Name as 'Travel_Approval__c', a custom singular label 'Travel Approval', and a plural label 'Travel Approvals'. A 'DISPLAY DENSITY' dropdown is open, set to 'Comfy', with other options like 'Compact' and 'Switch to Salesforce Classic' available. A user profile for Naveen Botta is visible at the top right.

Custom Fields in Travel Approval Object

Contains purpose of trip, status, trip start date, trip end date, out of state, destination state, department(lookup)

The screenshot shows the 'Fields & Relationships' section of the 'Travel Approval' object in the Salesforce Object Manager. The sidebar on the left is identical to the previous screenshot. The main area lists 13 fields and their properties:

Field	Type	Description
Destination State	Text(2)	
Last Modified By	Lookup(User)	
Out-of-State	Checkbox	
Owner	Lookup(User,Group)	
Purpose of Trip	Text Area(255)	
Status	Picklist	
Status Indicator	Formula (Text)	
Total Expenses	Roll-Up Summary (SUM Expense Item)	
Travel Approval #	Name	Auto Number
Trip End Date	Date	
Trip Start Date	Date	

Custom tabs for custom objects

The screenshot shows the Salesforce Setup interface with the 'Custom Tabs' page selected. The left sidebar has 'User Interface' expanded, with 'Tabs' selected. The main content area displays three sections: 'Custom Object Tabs', 'Web Tabs', and 'Visualforce Tabs'. Under 'Custom Object Tabs', there are two entries: 'Departments' (Tab Style: Books) and 'Travel Approvals' (Tab Style: Airplane). Under 'Visualforce Tabs' and 'Lightning Component Tabs', no tabs have been defined.

After creating travel approval object with fields, it looks as shown below.

The screenshot shows the Salesforce Lightning Experience with the 'Travel Approval' object selected. The record ID is TA00001. The 'Details' tab is active, displaying fields such as Travel Approval # (TA00001), Status (New), Out-of-State (checked), Destination State (US), Purpose of Trip (Business Trip to USA), Trip Start Date (2/20/2023), Trip End Date (2/28/2023), Created By (Naveen Botta), and Last Modified By (Naveen Botta). A sidebar on the right shows the user profile (Naveen Botta) and navigation links for DISPLAY DENSITY (Comfy selected), OPTIONS, and Activity (empty). The Activity section includes 'Upcoming & Overdue' and 'Past activity' sections.

Step 5

Importing Department records using data import wizard
And imported record data status can view in Bulk Data Load Jobs as shown

The screenshot shows the Bulk Data Load Jobs page in the Salesforce Setup interface. A specific job for the 'Department' object is detailed:

Job ID	7502w00000C7h1o	Job Type	Bulk V1
Submitted By	Naveen Botta	Operation	Insert
Start Time	2/14/2023, 6:02 AM AKST	Queued Batches	0
End Time	2/14/2023, 6:02 AM AKST	In Progress Batches	0
Time to Complete (hh:mm:ss)	00:01	Completed Batches	1
Object	Department	Failed Batches	0
External ID Field		Progress	100%
Content Type	CSV	Records Processed	16
Concurrency Mode	Parallel	Records Failed	0
API Version	57.0	Retries	0

The 'Batches' section shows one batch with the following details:

View Request	View Result	Batch ID	Start Time	End Time	Total Processing Time (ms)	API Active Processing Time (ms)	Apex Processing Time (ms)	Records Processed	Records Failed	Retry Count	State Message	Status
View Request	View Result	7512w00000UDSG	2/14/2023, 6:02 AM	2/14/2023, 6:02 AM	132	90	0	16	0	0	Completed	

Imported records

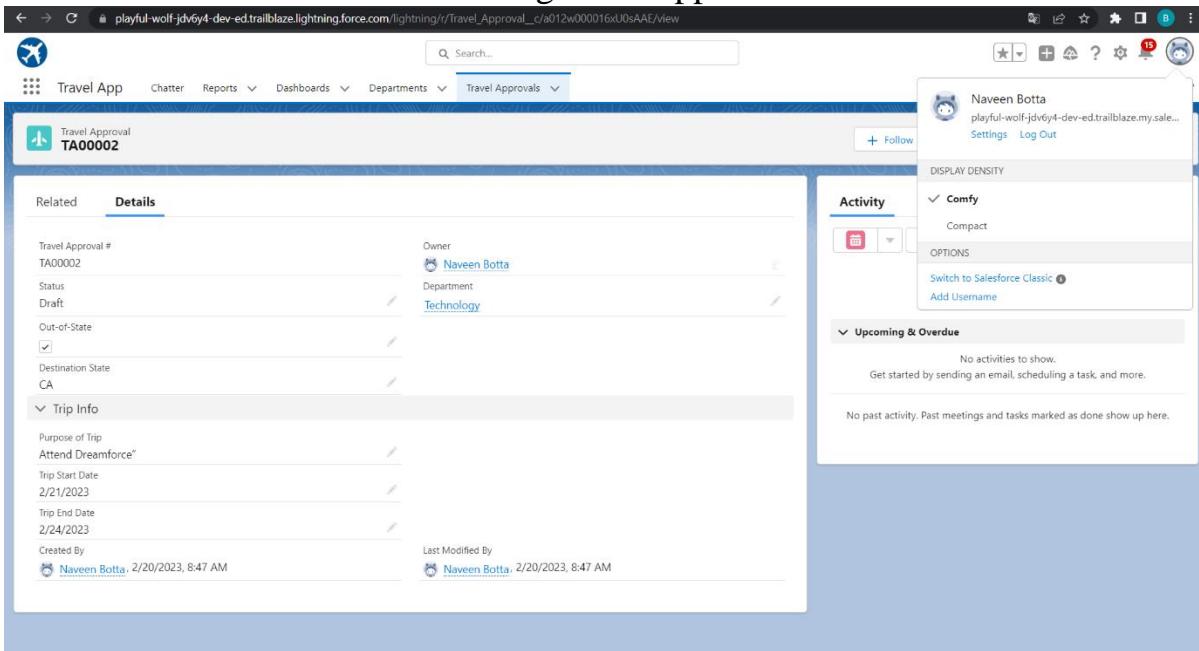
The screenshot shows the Department list page in the Salesforce Travel App. The list includes 17 items, with the first few entries being:

1	Audit Services
2	Contract Management
3	Disability Determination Bureau
4	Division of Aging
5	Division of Disability and Rehabilitative Services
6	Division of Family Resources
7	Division of Finance
8	Division of Mental Health and Addiction
9	Human Resources
10	Legislative Services
11	Office of Communications and Media
12	Office of Early Childhood and Out-of-School Learning
13	Office of General Counsel
14	Office of Medicaid Policy and Planning
15	Quality and Compliance Office
16	Technology
17	Test1

Exercise 2

Step 1

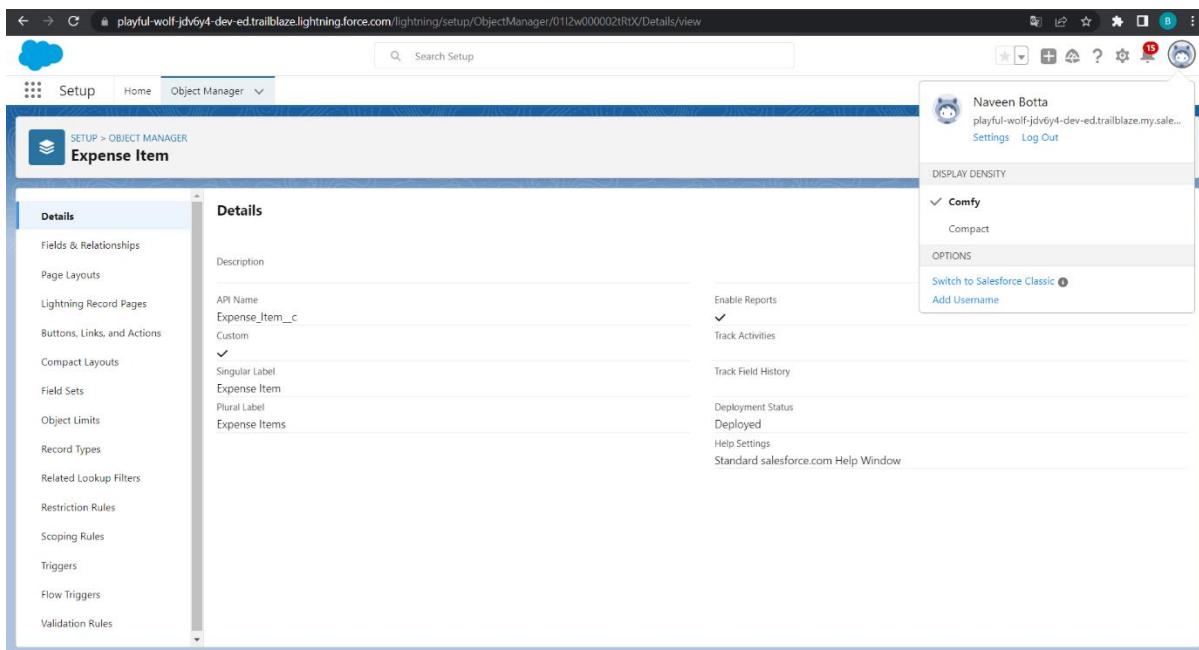
Creating travel approval record



The screenshot shows a Salesforce Lightning page for a Travel Approval record. The URL is playful-wolf-jdv6y4-dev-ed.trailblaze.lightning.force.com/lightning//Travel_Approval_c/a012w000016xJ0sAAE/view. The page title is "Travel Approval TA00002". The left sidebar has "Travel App" selected. The main content area shows the "Details" tab of the Travel Approval record. Fields include: Travel Approval # (TA00002), Status (Draft), Out-of-State (checked), Destination State (CA), Purpose of Trip (Attend Dreamforce), Trip Start Date (2/21/2023), and Trip End Date (2/24/2023). The "Owner" is Naveen Botta, and the "Department" is Technology. The "Last Modified By" field shows Naveen Botta on 2/20/2023 at 8:47 AM. The right sidebar shows the user profile of Naveen Botta and activity feed settings.

Step 2

Creating Expense Item custom object



The screenshot shows the Salesforce Setup interface for creating a new custom object. The URL is playful-wolf-jdv6y4-dev-ed.trailblaze.lightning.force.com/lightning/setup/ObjectManager/01Iw000002tR2X/Details/view. The page title is "Expense Item". The left sidebar shows "SETUP > OBJECT MANAGER". The main content area shows the "Details" tab of the Expense Item object. Fields include: Description, API Name (Expense_Item__c), Singular Label (Expense Item), and Plural Label (Expense Items). The right sidebar shows the user profile of Naveen Botta and setup settings.

Step 3

Custom fields in Expense Item object

The screenshot shows the Salesforce Object Manager interface for the 'Expense Item' object. On the left, a sidebar lists various setup options like Details, Fields & Relationships, Page Layouts, and Lightning Record Pages. The main area is titled 'Fields & Relationships' and displays six items sorted by Field Label. A table provides details for each field:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD
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)	✓

A right-hand sidebar shows user information for Naveen Botta and settings for 'DISPLAY DENSITY' (set to 'Comfy') and 'OPTIONS'.

Step 4

Creating Records in Expense Item

The screenshot shows a Travel Approval record with ID TA00001. The page includes tabs for Related and Details. Under 'Related', there are sections for Notes & Attachments (0), Expense Items (2), and Approval History (0). The Expense Items section lists two items:

Expense Item Number	Expense Type	Amount
E-00004	Airfare	\$450.00
E-00005	Hotel	\$870.00

The right side of the screen shows the user profile for Naveen Botta and a sidebar with 'Activity' and 'Upcoming & Overdue' sections.

Step 5

Creating User with system admin license

The screenshot shows the Salesforce Setup interface under the 'Users' section. A new user record is being created for 'Eric Executive'. The 'User Detail' tab is selected. Key fields filled include:

- Name: Eric Executive
- Alias: exec
- Email: bottanaveen4@gmail.com
- Username: eric.botta@wipro.com
- Nickname: eric.exec
- Title: Manager
- Company: Wipro
- Department: Division
- Address: (GMT+05:30) India Standard Time (Asia/Kolkata)
- Time Zone: (GMT+05:30) India Standard Time (Asia/Kolkata)
- Locale: English (India)
- Language: English
- Delegated Approver: Eric Executive
- Receive Approval Request Emails: Only if I am an approver
- Federation ID: (Connect)
- App Registration: One-Time Password Authenticator (Connect)
- App Registration: Salesforce Authenticator (Connect)
- Security Key (U2F or WebAuthn): (Generate)
- Lightning Login: (Enroll)
- Temporary Verification Code (Expires in 1 to 24 Hours): (Generate)

The 'Role' section shows 'Salesforce System Administrator' checked. The 'User License' section shows 'Active' checked. The 'Profile' section shows 'System Administrator' checked. The 'Options' sidebar on the right indicates the user is 'Comfy' and has selected 'Switch to Salesforce Classic'.

Step 6

Add user eric as manager own account user

The screenshot shows the Salesforce Setup interface under the 'Users' section. A new user record is being created for 'Naveen Botta'. The 'User Detail' tab is selected. Key fields filled include:

- Name: Naveen Botta
- Alias: NBott
- Email: naveen.botta@wipro.com
- Username: naveen.botta@playful-wolf-jdv6y4.com
- Nickname: User16763694006969082773
- Title: Manager
- Company: Wipro
- Department: Division
- Address: IN
- Time Zone: (GMT-09:00) Alaska Standard Time (America/Yakutat)
- Locale: English (United States)
- Language: English
- Delegated Approver: Eric Executive
- Receive Approval Request Emails: Only if I am an approver
- Federation ID: (Connect)
- App Registration: One-Time Password Authenticator (Connect)
- App Registration: Salesforce Authenticator (Connect)
- Security Key (U2F or WebAuthn): (Generate)
- Lightning Login: (Enroll)
- Temporary Verification Code (Expires in 1 to 24 Hours): (Generate)

The 'Role' section shows 'Salesforce System Administrator' checked. The 'User License' section shows 'Active' checked. The 'Profile' section shows 'System Administrator' checked. The 'Options' sidebar on the right indicates the user is 'Comfy' and has selected 'Switch to Salesforce Classic'.

Step 7

Customize the Travel Approval Default search layout

The screenshot shows the 'Edit Search Layout' page for the 'Travel Approval' object. The left sidebar lists various customization options like Details, Fields & Relationships, Page Layouts, and Search Layouts. The main area displays the 'Travel Approval Search Results' layout. It shows two columns: 'Available Fields' and 'Selected Fields'. The 'Available Fields' column lists fields such as Record ID, Out-of-State, Status Indicator, Total Expenses, Owner Alias, Owner First Name, Owner Last Name, Created By Alias, Created Date, Last Modified By Alias, and Last Modified Date. The 'Selected Fields' column lists fields such as Travel Approval #, Purpose of Trip, Department, Status, Destination State, Trip Start Date, and Trip End Date. Below these columns is a checkbox for 'Override the search result column customizations for all users'. On the right, there's a sidebar with user info (Naveen Botta) and display density settings (Comfy selected). Buttons for 'Save' and 'Cancel' are at the bottom.

Step 8

Select fields to display in the Travel Approval “All” List view

The screenshot shows the 'Travel Approvals' list view in the Lightning interface. A modal window titled 'Select Fields to Display' is open, allowing the user to choose which fields to show in the list. The 'Available Fields' section lists fields like Created By Alias, Created Date, Destination State, Last Activity Date, Last Modified By, and Last Modified By Alias. The 'Visible Fields' section lists fields like Department, Created By, Status, Trip Start Date, and Trip End Date. At the bottom of the modal, there are 'Cancel' and 'Save' buttons. The main list view shows a table with columns for Department, Created By, Status, Approved, and Trip End Date, with 50+ items listed.

Step 9

Create Travel approval custom List View “Open Out of State Travel Requests” All users should be able to see this list view.

Travel Approvals
Open Out of State Travel Requests

Department	Created By	Status	Destination State	Trip Start Date	Trip End Date
Audit Services	Naveen Botta	New	CA	2/22/2023	2/27/2023
Technology	Naveen Botta	Draft	CA	2/21/2023	2/24/2023
Test1	Naveen Botta	New	US	2/20/2023	2/28/2023

Filters

- Filter by Owner: All travel approvals
- Matching all of these filters:
 - Out-of-State equals True
 - Status not equal to Approved, Rejected

Add Filter Remove All Add Filter Logic

Step 10

Select fields to display in the Travel Approval “Open Out of State Travel Requests” List view

Select Fields to Display

Available Fields	Visible Fields
Created By Alias	Department
Created Date	Created By
Last Activity Date	Status
Last Modified By	Destination State
Last Modified By Alias	Trip Start Date
Last Modified Date	Trip End Date

Filters

- Filter by Owner: All travel approvals
- Matching all of these filters:
 - Out-of-State equals True
 - Status not equal to Approved, Rejected

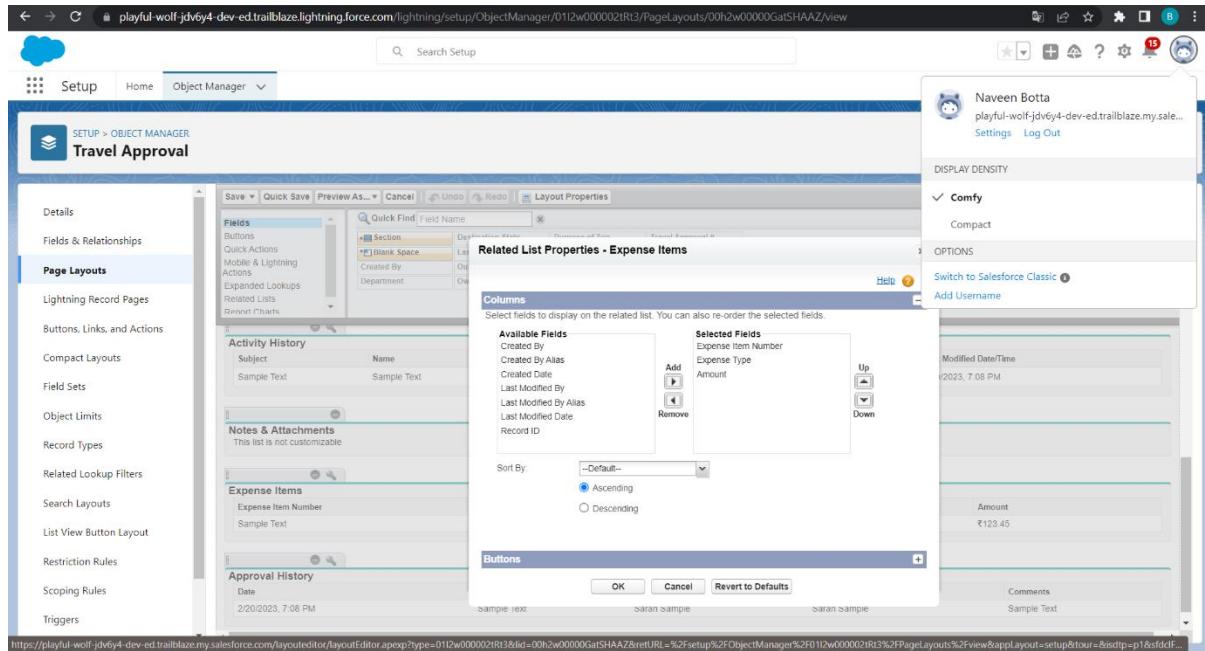
Cancel Save Add Filter Remove All Add Filter Logic

Step 11

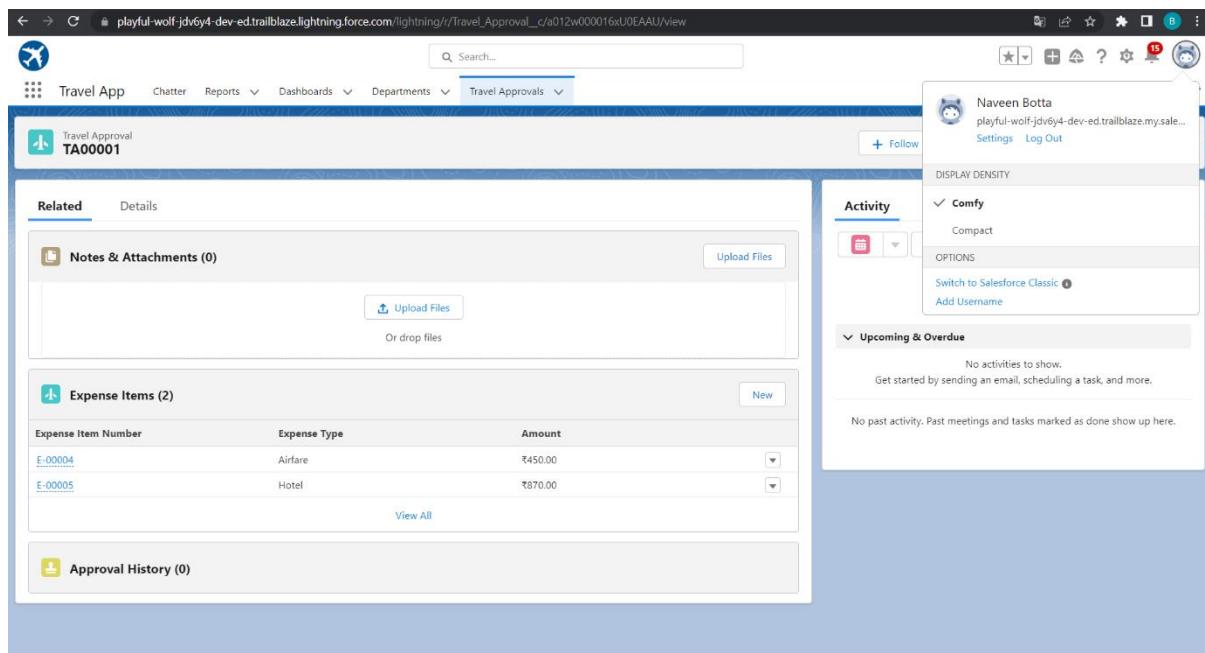
Customizing page layout

Step 12

Customize the Expense Item Related List under the Travel Approval



It looks as shown below



Step 13

Enable feed tracking for travel approval object

The screenshot shows the Salesforce Lightning Feed Tracking home page. At the top, there's a search bar labeled "Search Setup". Below it, a navigation bar includes "Logged in as Eric Executive (eric.botta@wipro.com) Log out as Eric Executive". The main content area has tabs for "Post", "Poll", and "Question", with "Post" selected. A text input field says "Share an update..." and a "Share" button. To the right, there's a "Search this feed..." field and a "Sort by:" dropdown set to "Top Posts". A sidebar on the left titled "Travel App" lists "What I Follow", "To Me", "Bookmarked", and "Company Highlights" (which is currently selected). It also shows "STREAMS" and "RECENT GROUPS". The main feed displays a post from "TA00001 — Naveen Botta" from 10 hours ago, asking "@Eric Executive Which Department should I associate this travel request with?". A comment from "Eric Executive" follows, stating "Technology is the correct department.". On the right, there's an "Einstein Recommendations" sidebar with three items: "Naveen Botta Reports to you", "Integration User Joined in the last week", and "Security User Joined in the last week", each with a "+ Follow" and "Skip" button.

Test Collaboration Mail received for response

The screenshot shows an email collaboration message. The subject line is "Eric Executive commented on your post on Travel Approval: TA00001". The message body starts with a comment from "Eric Executive": "Technology is the correct department." Below this is a blue button with "View/Comment" and "or reply to this email". Underneath, there's a section titled "Original post" showing a comment from "TA00001 — Naveen Botta" from Monday, February 20, 2023, 9:02 AM, asking "@Eric Executive Which Department should I associate this travel request with?". Another comment from "Eric Executive" follows, stating "Technology is the correct department.", timestamped Monday, February 20, 2023, 9:05 AM.

← → C playfull-wolf-jdv6y4-dev-ed.trailblaze.lightning.force.com/lightning/page/chatter

Travel App Chatter Reports Dashboards Departments Travel Approvals

Search...

What I Follow To Me Bookmarked Company Highlights My Drafts

STREAMS + You don't have any streams yet. Try creating one!

RECENT GROUPS + Aw, you don't have any groups! Why not create or join some now?

Post Poll Question Share an update... Share

Sort by: Top Posts Search this feed...

TA000001 — Naveen Botta 10h ago @Eric Executive Which Department should I associate this travel request with?

Like Comment 1 comment · 1 view

Eric Executive 10 hours ago Technology is the correct department.

Like Write a comment...

Einstein Naveen Botta playfull-wolf-jdv6y4-dev-ed.trailblaze.my.sale... Settings Log Out

DISPLAY DENSITY Comfy Compact

OPTIONS Switch to Salesforce Classic Add Username Joined in the last week FOLLOW

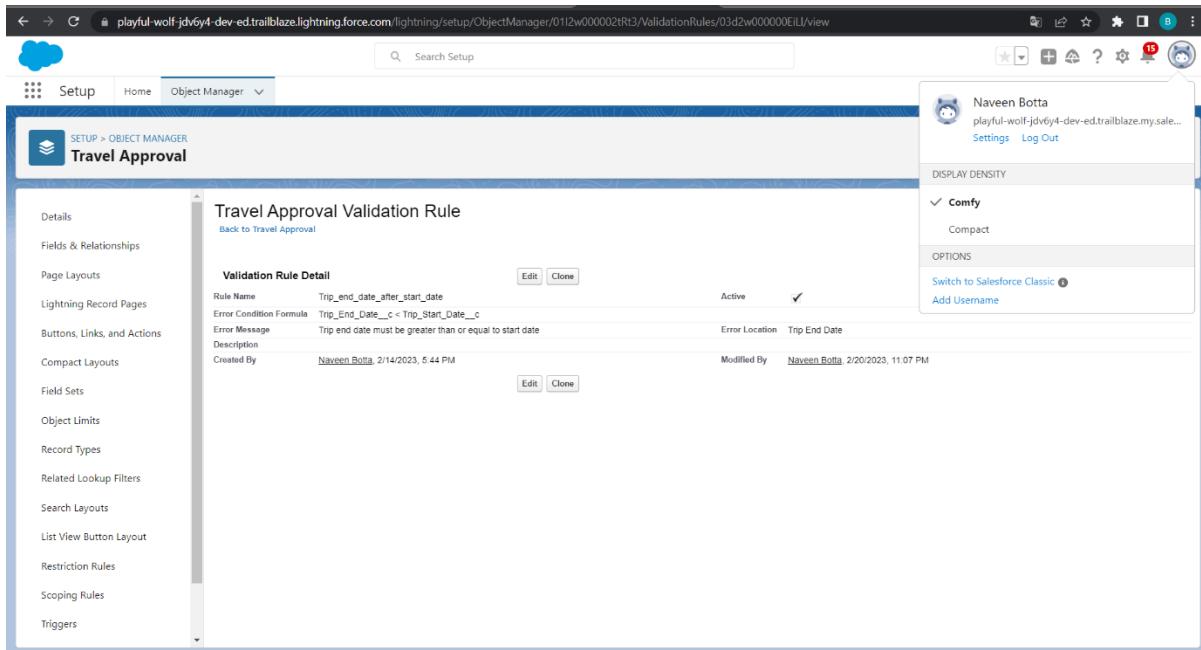
The screenshot shows the Salesforce Lightning Chatter interface. On the left, there's a sidebar with navigation links like 'What I Follow', 'To Me', 'Bookmarked', 'Company Highlights' (which is selected), 'My Drafts', 'STREAMS', and 'RECENT GROUPS'. The main area displays a post from 'TA000001 — Naveen Botta' asking which department to associate a travel request with. A comment from 'Eric Executive' suggests 'Technology is the correct department'. The Einstein sidebar on the right shows profile information for 'Naveen Botta' and options for display density and switching to Salesforce Classic.

Module 2

Exercise 1

Step 1

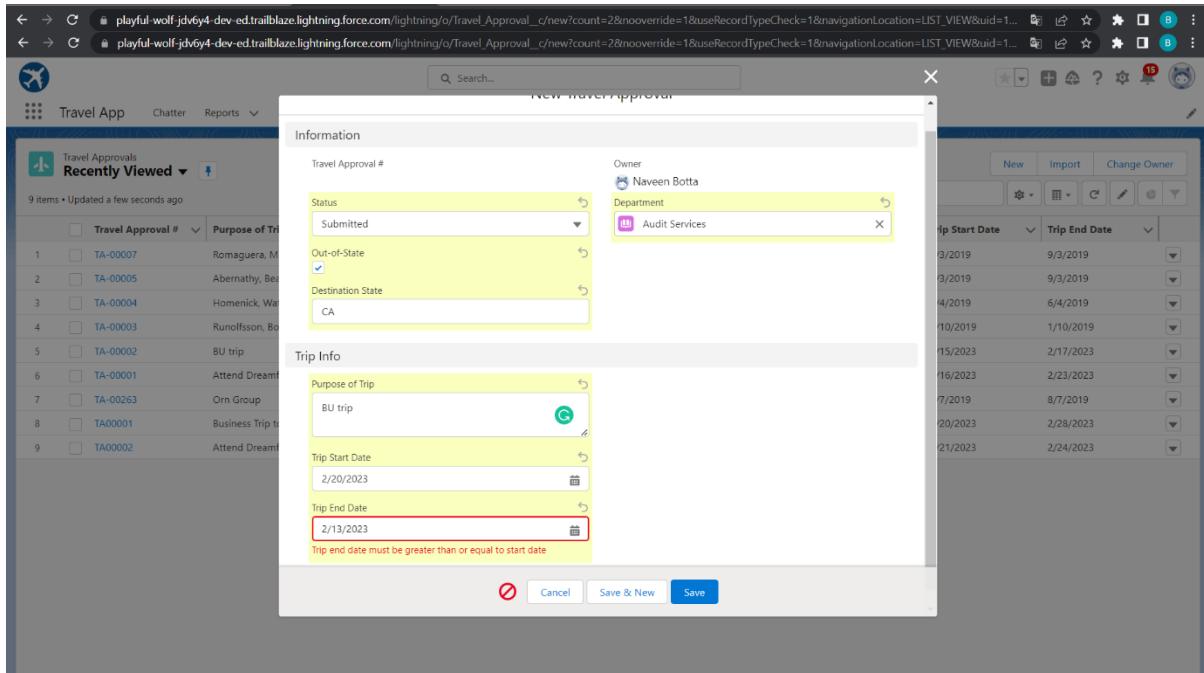
Validation Rule that trip end date always \geq start date



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, and Lightning Record Pages. The main content area displays the 'Travel Approval Validation Rule' detail page. The rule is titled 'Travel Approval Validation Rule' and has the ID '01l2w000002tRt3'. It is active and applies to the 'Trip End Date' field. The validation formula is 'Trip_End_Date__c < Trip_Start_Date__c'. The error message is 'Trip end date must be greater than or equal to start date'. The rule was created by Naveen Botta on 2/14/2023 at 5:44 PM and modified by Naveen Botta on 2/20/2023 at 11:07 PM. A right-hand sidebar shows user information (Naveen Botta) and display density settings (Comfy).

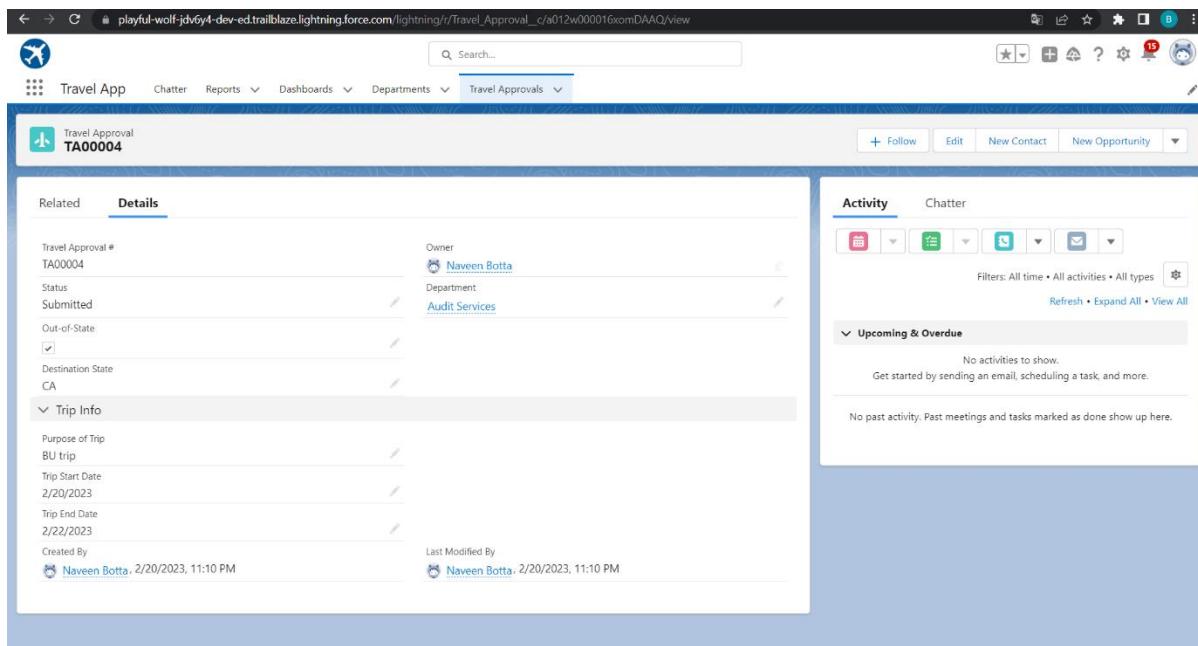
Working of validation rule

In end date if we give date less than start date it show error



Error message

If we give greater than start date it will save not throw error



Step 2

Create Roll-up Summary Field on travel Approval object

The screenshot shows the Salesforce Setup interface for creating a custom field. The left sidebar lists various setup options like Details, Fields & Relationships, Page Layouts, and Buttons, Links, and Actions. The main area displays the 'Travel Approval Custom Field Total Expenses' configuration. The 'Field Information' section includes the field label 'Total Expenses', field name 'Total_Expenses', API name 'Total_Expenses__c', and a description. The 'Roll-Up Summary Options' section specifies the data type as 'Roll-Up Summary', the summarized object as 'Expense Item', the field to aggregate as 'Expense Item Amount', and the summary type as 'SUM'. The right sidebar shows user information (Naveen Botta) and display density settings.

Step 3

Creating Formula Fields
Static Resource with name StatusImages

The screenshot shows the Salesforce Setup interface for creating a static resource. The left sidebar shows a search bar with 'static' and a 'Custom Code' section. The main area displays the 'Static Resource StatusImages' configuration. The 'Static Resource Detail' section includes the name 'StatusImages', namespace prefix, description, MIME type (application/x-zip-compressed), cache control (Public), size (39,130 bytes), and a 'View file' link. The right sidebar shows user information (Naveen Botta) and display density settings.

Step 4

Create Status formula field on travel approval

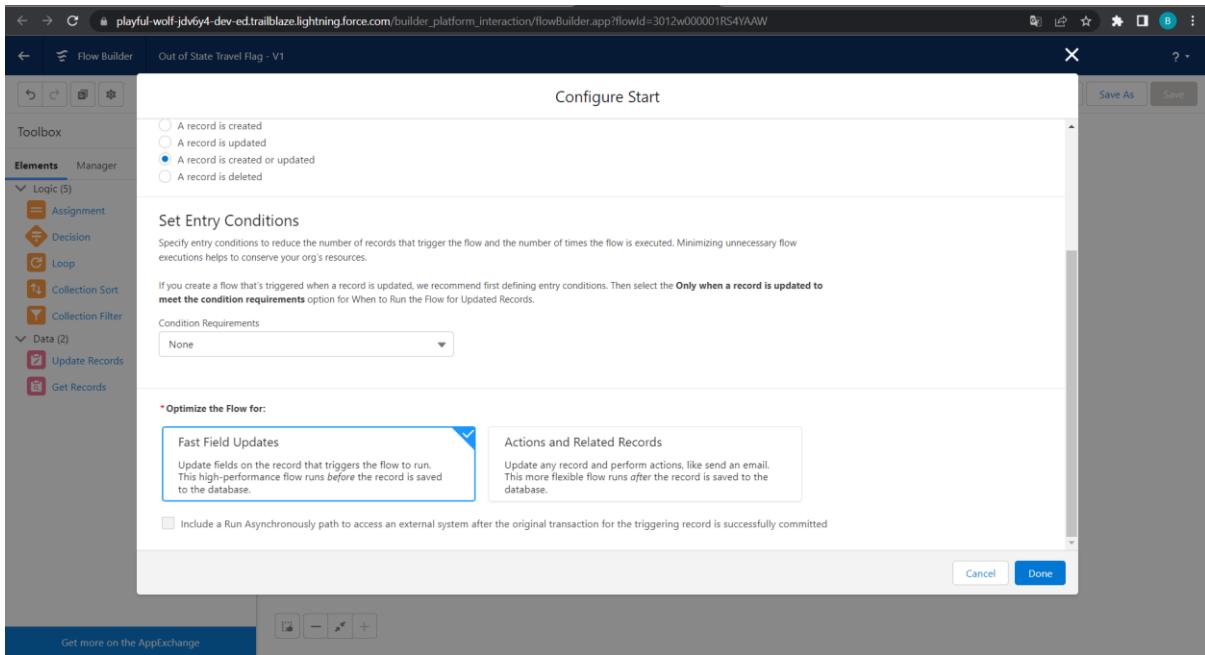
The screenshot shows the Salesforce Setup interface for creating a custom field. The object is 'Travel Approval'. The field name is 'Status Indicator' (Status_Indicator). The formula is:

```
IF (ISPICKVAL(Status__c, 'Approved'), IMAGE('/resource>Status/images/thumb-up.png', 'Accepted', 20, 20), IF (ISPICKVAL(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

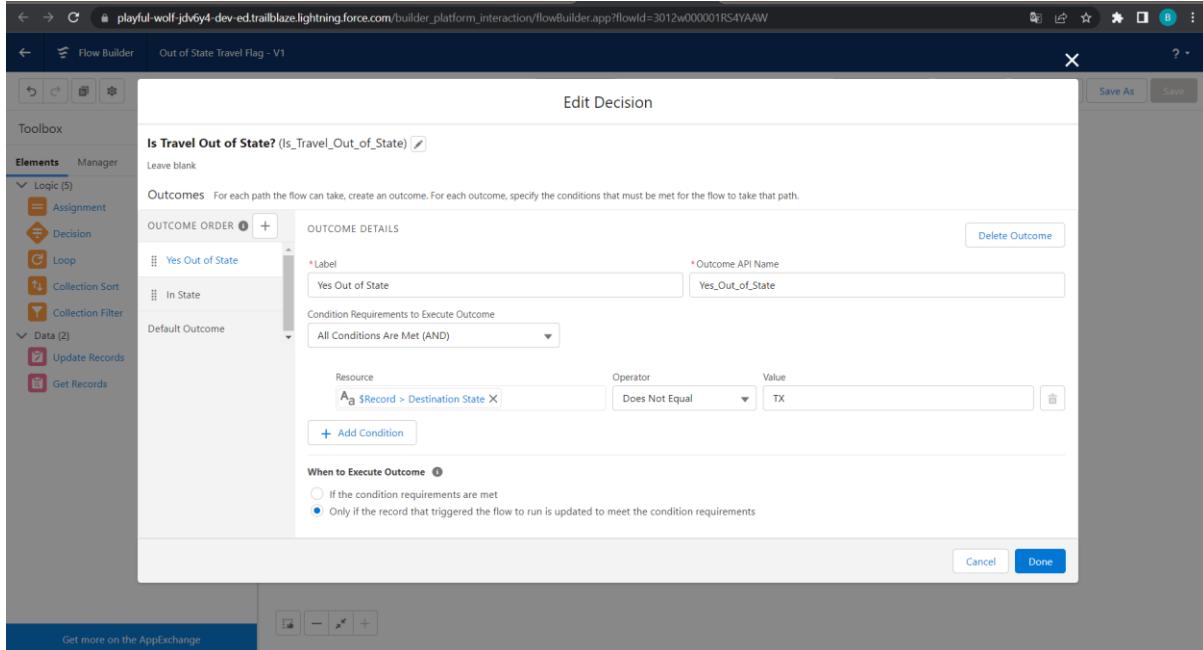
Record Triggered Flow

The screenshot shows the Flow Builder interface for configuring a flow. The trigger is set to 'A record is created or updated'. The entry condition is 'None'.

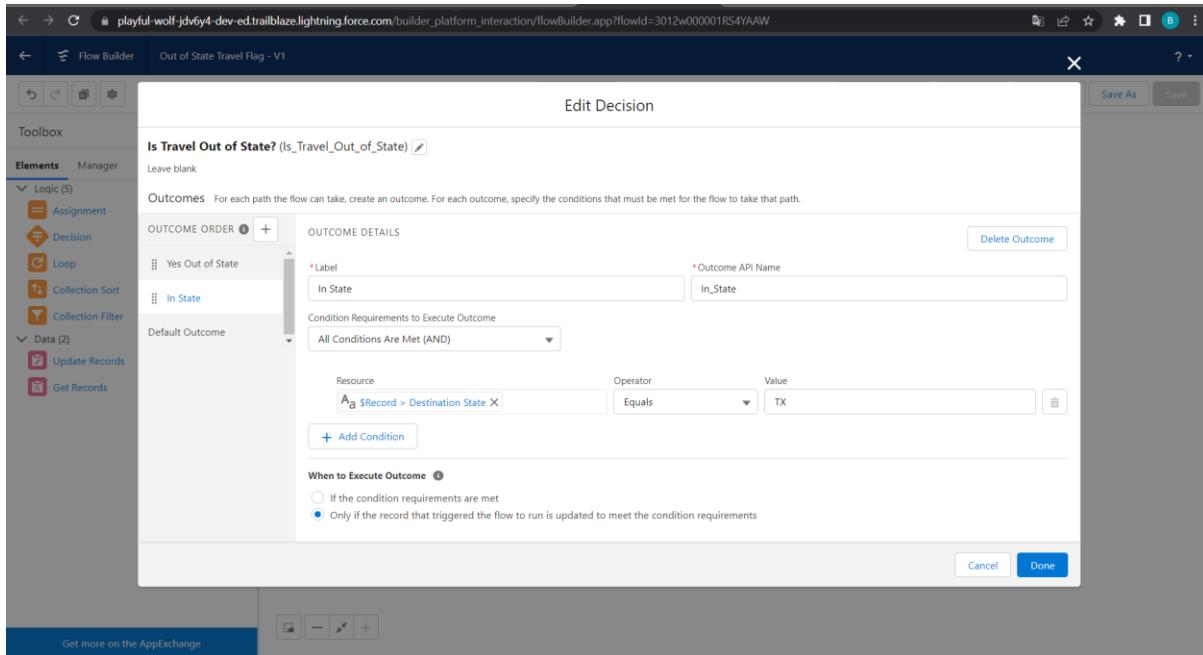


And add Decision element to flow

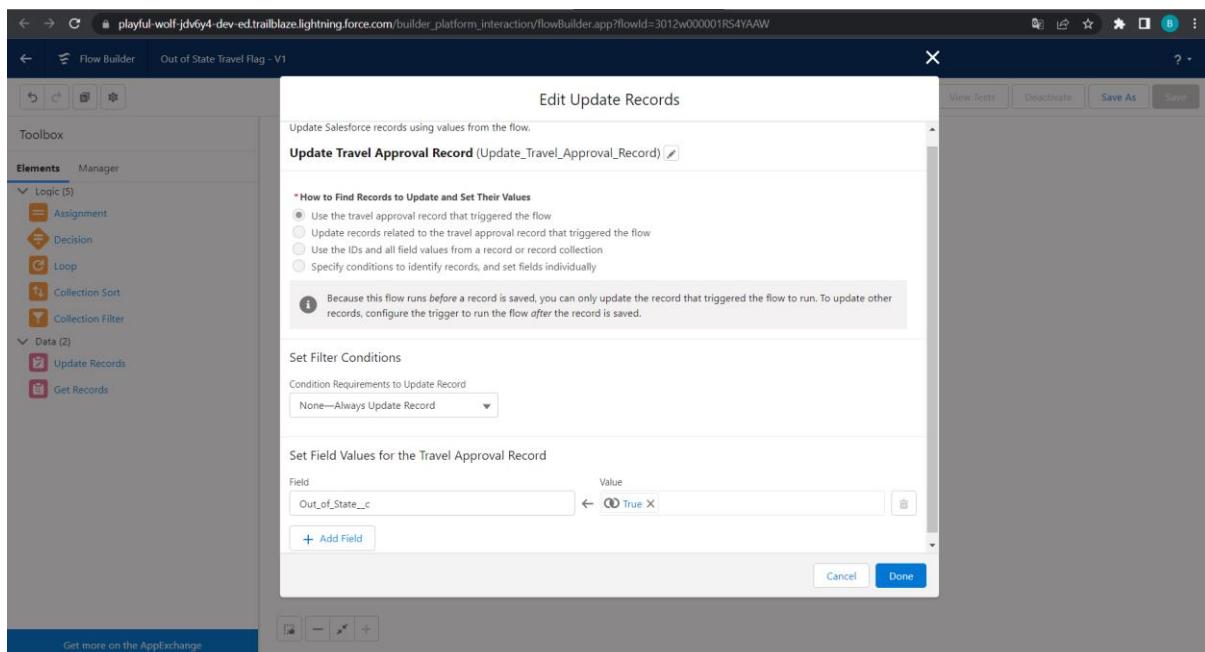
1. For Is travel out of state?



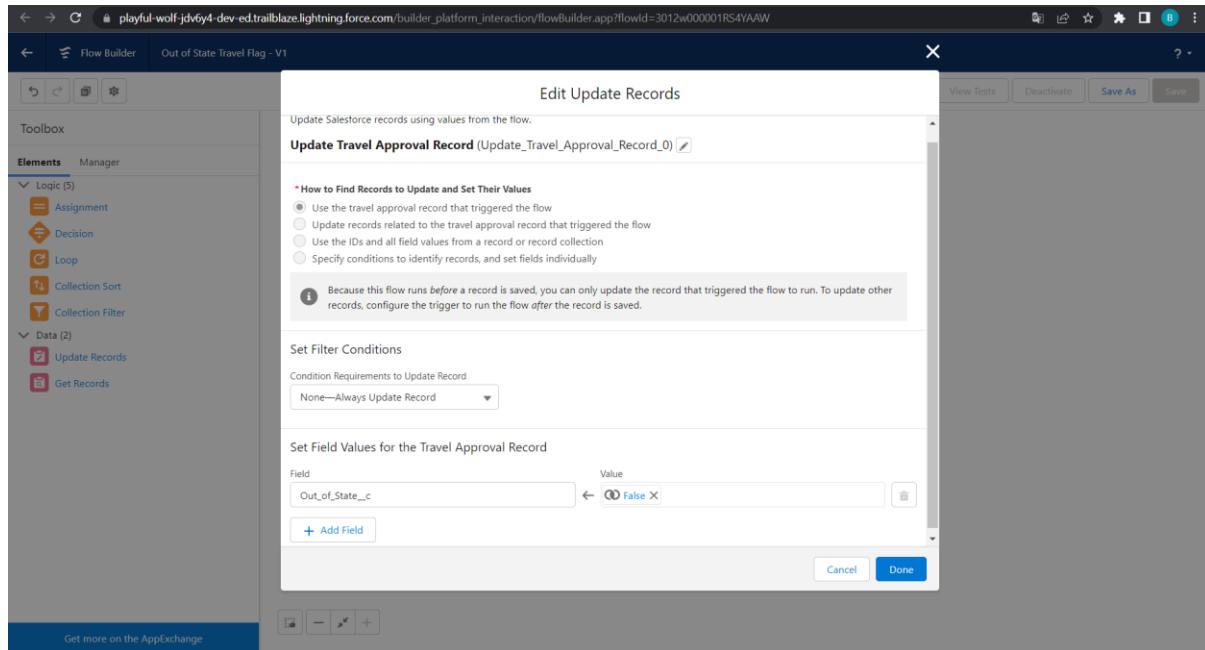
2. For in state



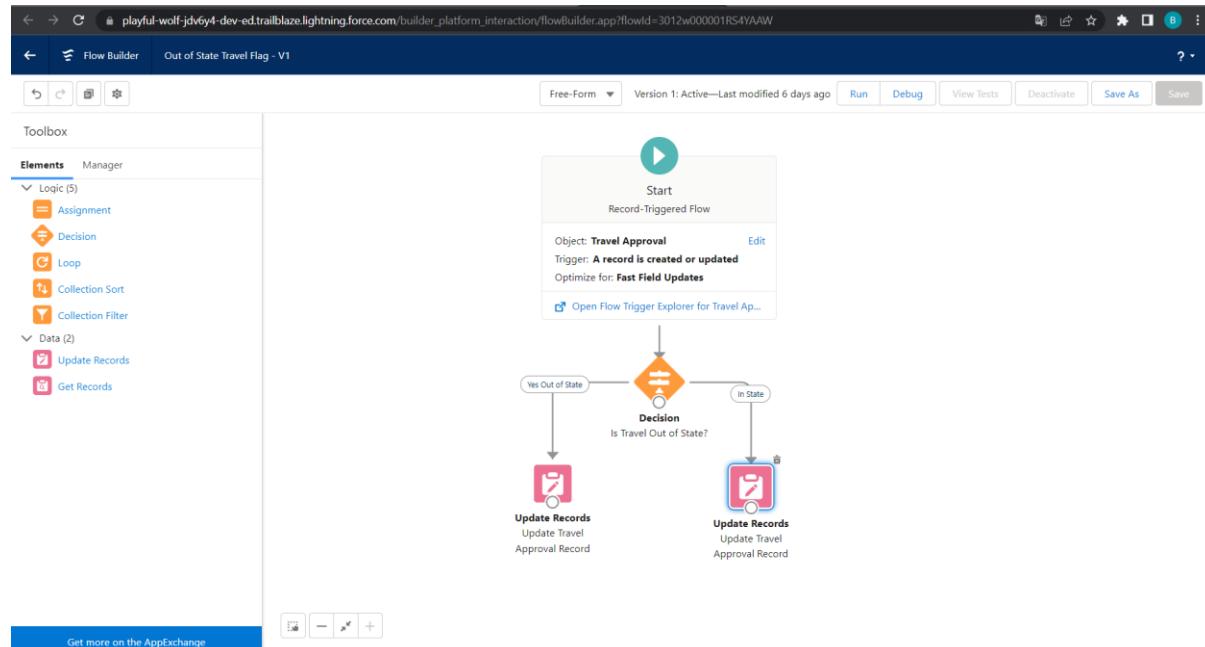
Creating action for flow using Update Record Element
This action work if decision- Is out of state?



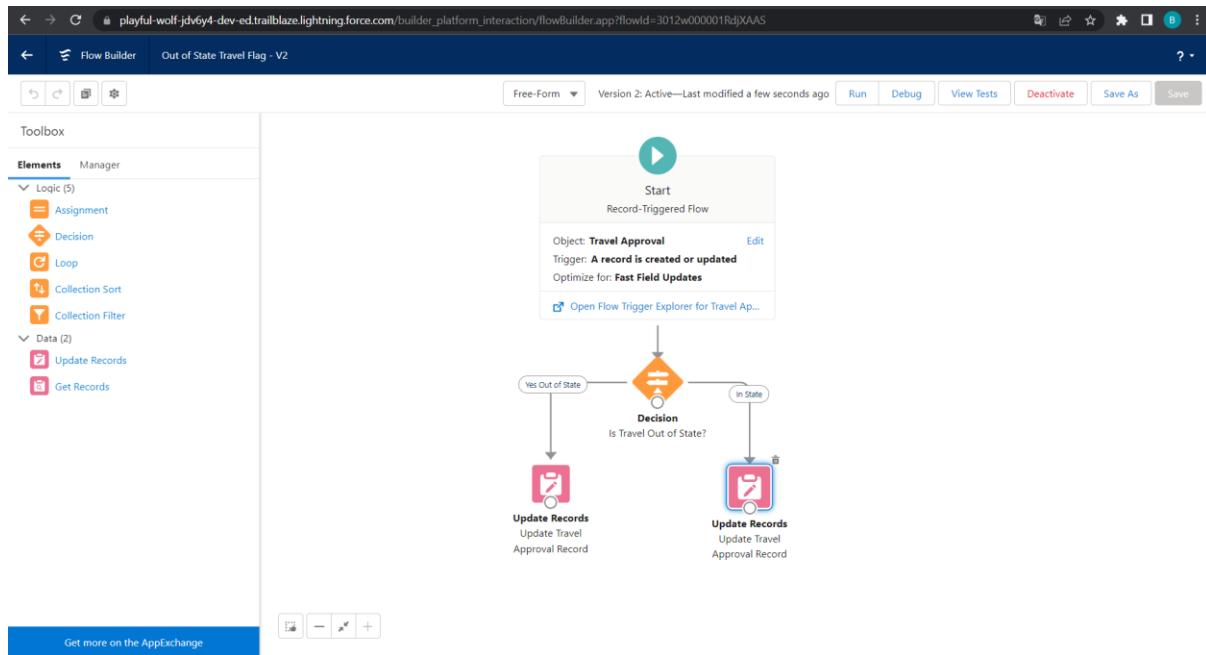
This decision works for- In State



Finally, the flow looks as shown



Save and activate the flow



Step 6

Creating an Approval Process to send travel approvals to manager or travel coordinator.

The screenshot shows the "Approval Processes" setup page for a process named "Travel Approval: Travel Approval Request". The process definition details include:

- Process Name: Travel Approval Request
- Unique Name: Travel_Approval_Request
- Description: Travel Approval Request
- Entry Criteria: TRUE
- Record Editability: Administrator ONLY
- Approval Assignment Email Template: Initial Submitters
- Initial Submitters: Travel Approval Owner
- Created By: Naveen Botta, 2/14/2023, 6:07 PM
- Modified By: Naveen Botta, 2/20/2023, 7:28 PM

The "Initial Submission Actions" section contains a "Record Lock" action with the description "Lock the record from being edited". The "Approval Steps" section lists two 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 Eric Executive	Final Rejection

The screenshot shows the 'Approval Processes' page in Salesforce. A modal window titled 'Travel Coordinator Approval' is open under 'Step 1. Enter Name and Description'. In the 'Enter Name and Description' section, the 'Name' field is set to 'Travel Coordinator Approval' and the 'Unique Name' field is set to 'Travel_Coordinator_Approval'. The 'Description' field is empty. At the bottom right of the modal are 'Save', 'Next', and 'Cancel' buttons.

Criteria are met : out of state = true

The screenshot shows the 'Approval Processes' page in Salesforce. A modal window titled 'Step 2. Specify Step Criteria' is open. Under 'Specify Step Criteria', the option 'Enter this step if the following [criteria are met]:' is selected. Below this, a table defines a criterion: 'Travel Approval: Out-of-State' is compared to 'True' using the 'equals' operator. This row is preceded by four rows of 'None' values. The table includes columns for 'Field', 'Operator', and 'Value'. At the bottom right of the modal are 'Previous', 'Save', 'Next', and 'Cancel' buttons.

Selecting eric as manager

The screenshot shows the 'Approval Processes' setup page. In the sidebar, 'Approval Processes' is selected under 'Process Automation'. The main area displays 'Step 3. Select Assigned Approver'. It asks to specify the user who should approve records that enter this step. A table shows previous approval step information: Step Number: 1, Name: Step 1, Criteria: (empty), and Assign To: Manager. Below this, a 'Select Approver' section allows choosing approvers manually or automatically. The 'Automatically assign to approver(s)' option is selected, and 'Eric Executive' is chosen from a dropdown. Under 'When multiple approvers are selected', the 'Approve or reject based on the FIRST response' option is selected. A note states: 'The approver's delegate may also approve this request.' A 'Reject Behavior' section is at the bottom.

Final Approval Action

The screenshot shows the 'Field Updates' setup page. In the sidebar, 'Approval Processes' is selected under 'Process Automation'. The main area displays 'Set Status to Approved' under 'Field Update Detail'. The details are: Name: Set Status to Approved, Unique Name: Set_Status_to_Approved, Object: Travel Approval, Field to Update: Travel Approval: Status, Field Data Type: Picklist, and New Field Value: Approved. Below this, sections for 'Rules Using This Field Update', 'Approval Processes Using This Field Update', and 'Entitlement Processes Using This Field Update' are shown. The 'Approval Processes Using This Field Update' table has one row: Action: Edit | Del, Approval Process Name: Travel Approval Request, Description: Travel Approval, Type: Travel Approval, and State: Active. A note at the bottom says: 'Always show me ▾ more records per related list'.

Final Rejection Action

The screenshot shows the Salesforce Setup interface with the 'Field Updates' page selected. A field update named 'Set Status to Rejected' is displayed. The details are as follows:

Name	Set Status to Rejected
Unique Name	Set_Status_to_Rejected
Description	Travel Approval
Object	Travel Approval
Field to Update	Travel Approval: Status
Field Data Type	Picklist
Re-evaluate Workflow Rules after Field Change	<input type="checkbox"/>
New Field Value	Rejected

Below the details, there are sections for 'Rules Using This Field Update', 'Approval Processes Using This Field Update', and 'Entitlement Processes Using This Field Update'. The 'Approval Processes Using This Field Update' section shows one entry:

Action	Approval Process Name	Description	Type	State
Edit Del	Travel Approval Request	Travel Approval	Travel Approval	Active

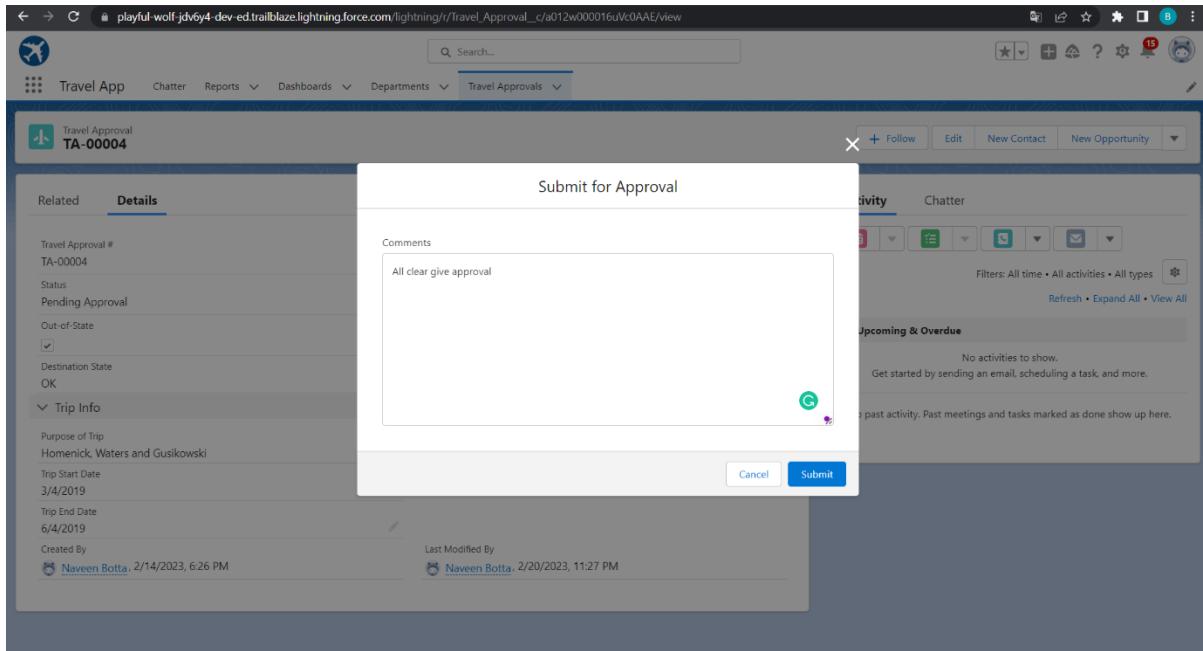
Testing the approval process
Status is pending here

The screenshot shows the Travel App interface with a travel approval record. The record ID is TA-00004. The status is listed as 'Pending Approval'. The 'Details' tab is selected, showing the following fields:

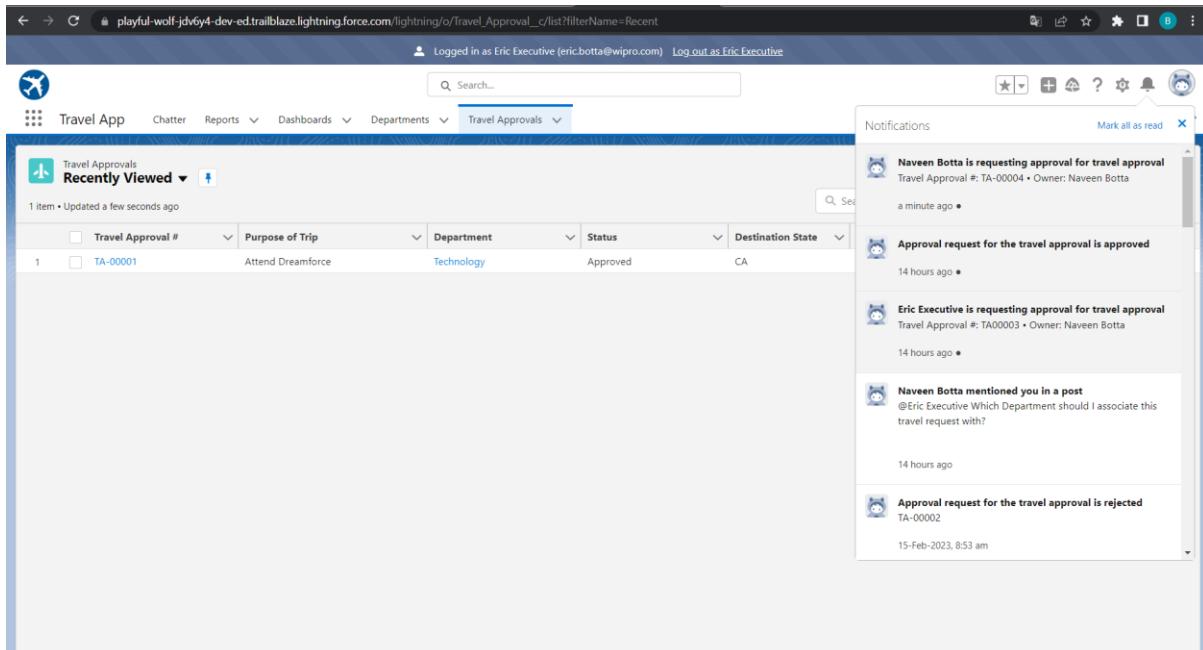
Travel Approval #	TA-00004
Status	Pending Approval
Out-of-State	<input checked="" type="checkbox"/>
Destination State	OK
Trip Info	
Purpose of Trip	Homenick, Waters and Gusikowski
Trip Start Date	3/4/2019
Trip End Date	6/4/2019
Created By	Naveen Botta, 2/14/2023, 6:26 PM
Last Modified By	Naveen Botta, 2/20/2023, 11:27 PM

On the right side, there is an 'Activity' section which is currently empty.

Submit for approval with comment



Login as Eric to give Approval



Eric Can Approve or Reject

The screenshot shows a Salesforce Lightning page for a travel approval request. At the top, the URL is `playful-wolf-jdv6y4-dev-ed.trailblaze.lightning.force.com/lightning/r/ProcessInstanceWorkItem/04i2w00000418ex/AA/view`. The user is logged in as Eric Executive (eric.botta@wipro.com). The page title is "Travel Approval Approval" with a status of "Pending". The header includes links for Chatter, Reports, Dashboards, Departments, Travel Approvals, and a search bar. On the right, there are buttons for Approve, Reject, and Reassign.

Details

Submitter: Naveen Botta Date Submitted: 21-Feb-2023 Actual Approver: Eric Executive Assigned To: Eric Executive

Approval Details

Travel Approval #: TA-00004 Owner: Naveen Botta

Submitter Comments

Naveen Botta: At; clear give approval 21-Feb-2023, 1:58:53 pm

After giving Approval

The screenshot shows the same travel approval record after it has been approved. The status is now "Approved". The "Owner" field shows Naveen Botta. The "Department" field shows "Division of Disability and Rehabilitative Services". The "Trip Info" section includes details like "Purpose of Trip: Homenick, Waters and Gusikowski", "Trip Start Date: 3/4/2019", "Trip End Date: 6/4/2019", and "Created By: Naveen Botta, 2/14/2023, 6:26 PM". The "Last Modified By" field also shows Naveen Botta, dated 2/20/2023, 11:31 PM. A sidebar on the right displays the user profile of Naveen Botta, the display density settings (set to "Comfy"), and activity logs, which are currently empty.

Exercise 2

Step 1

Using Data import wizard to import travel approval records

The screenshot shows the 'Choose data' step of the Data Import Wizard. A progress bar at the top indicates 'Getting closer.' The left panel lists 'Standard objects' and 'Custom objects'. Under 'Standard objects', 'Travel Approvals' is selected and highlighted with a green checkmark. The middle panel, titled 'What do you want to do?', contains fields for 'Match by:' (set to '-None-'), 'Which User field in your file designates record owners?' (set to '-None-'), 'Which Department field in your file do you want to match against to set the Department lookup field?' (set to '-None-'), and 'Trigger workflow rules and processes?' (unchecked). The right panel, titled 'Where is your data located?', shows a CSV file named 'TravelApprovals.csv' uploaded, with settings for 'Character Code' (ISO-8859-1), 'Values Separated By' (Comma), and 'File Encoding' (UTF-8). Navigation buttons at the bottom include 'Cancel', 'Previous', and a large green 'Next' button.

The screenshot shows the 'Edit mapping' step of the Data Import Wizard. A progress bar at the top indicates 'Almost done'. The left panel shows the 'Edit Field Mapping: Travel Approvals' section, stating that the file has been auto-mapped to existing Salesforce fields. The right panel displays a table of field mappings:

Edit	Mapped Salesforce Object	CSV Header	Example	Example	Example
Change	Department	Department	Office of Commu	Disability Determ	Division of Disability and Rehabilitative Services
Change	Destination State	Destination State	FL	OK	OK
Change	Purpose of Trip	Purpose of Trip	White-Kutch	Runolfsson, Bog	Homenick, Waters and Gusikowski
Change	Trip Start Date	Trip Start Date	6/14/19	10-01-2019	04-03-2019
Change	Trip End Date	Trip End Date	6/15/19	10-01-2019	04-06-2019
Change	Status	Status	Approved	Approved	Rejected

Navigation buttons at the bottom include 'Cancel', 'Previous', and a large green 'Next' button.

The screenshot shows the Salesforce Data Management Data Importer home page. At the top, there are tabs for 'Setup' (selected), 'Home', and 'Object Manager'. A search bar says 'Search Setup'. On the left, a progress bar has 'Choose data' on the left and 'Edit mapping' on the right. To the right, a sidebar shows the user's profile: Naveen Botta, playful-wolf-jdv6y4-dev-ed.trailblaze.lightning.force.com, with options for 'Settings' and 'Log Out'. Below the sidebar, 'DISPLAY DENSITY' is set to 'Comfy' (selected) and 'Compact'. Under 'OPTIONS', there are links to 'Switch to Salesforce Classic' and 'Add Username'. The main area is titled 'Review & Start Import' with the sub-instruction 'Review your import information and click Start Import.' It shows three sections: 'Your selections:' containing 'Travel Approvals ✓', 'Add new records ✓', and 'TravelApprovals.csv ✓'; 'Your import will include:' showing 'Mapped fields' (6) and 'Unmapped fields' (0); and 'Your import will not include:' which is empty. At the bottom right are 'Cancel', 'Previous', and a large green 'Start Import' button.

Records imported successfully

The screenshot shows the Travel App interface. The top navigation bar includes 'Travel App', 'Chatter', 'Reports', 'Dashboards', 'Departments', 'Travel Approvals' (selected), and a specific record 'TA-00004 Approval'. The main content area displays a list of travel approvals. The header of the list table includes columns: 'Department' (with a sorting arrow pointing up), 'Created By', 'Status', 'Trip Start Date', and 'Trip End Date'. The table body contains 18 rows, each representing a travel approval record. The first few rows show the following data:

Department	Created By	Status	Trip Start Date	Trip End Date
Audit Services	Naveen Botta	Approved	3/9/2019	3/9/2019
Audit Services	Naveen Botta	Approved	7/8/2019	7/8/2019
Audit Services	Naveen Botta	Approved	4/11/2019	7/11/2019
Audit Services	Naveen Botta	Approved	3/9/2019	3/9/2019
Audit Services	Naveen Botta	Approved	9/7/2019	9/7/2019
Audit Services	Naveen Botta	Approved	7/8/2019	7/8/2019
Audit Services	Naveen Botta	Approved	4/11/2019	7/11/2019
Audit Services	Naveen Botta	Approved	3/9/2019	3/9/2019
Audit Services	Naveen Botta	Approved	7/8/2019	7/8/2019
Audit Services	Naveen Botta	Approved	8/7/2019	8/7/2019
Audit Services	Naveen Botta	Approved	3/9/2019	3/9/2019
Audit Services	Naveen Botta	Approved	8/7/2019	9/7/2019
Audit Services	Naveen Botta	Approved	7/8/2019	7/8/2019
Audit Services	Naveen Botta	Approved	4/11/2019	7/11/2019
Audit Services	Naveen Botta	Submitted	2/20/2023	2/22/2023
Contract Management	Naveen Botta	Rejected	2/15/2023	2/17/2023
Contract Management	Naveen Botta	Approved	9/2/2019	9/2/2019
Contract Management	Naveen Botta	Approved	4/7/2019	4/7/2019

Step 2

Creating Travel Request by Department report

REPORT ▾ Travel Requests by Department ▾ Travel Approvals

Fields > Outline Filters 1

Groups

- GROUP ROWS
- Add group...

Department x Status x

GROUP COLUMNS

- Add group...

Columns

- Add column...
- # Out-of-State
- Destination State
- Trip Start Date
- Trip End Date

Department Status Out-of-State Destination State Trip Start Date Trip End Date

Contract Management (1) Approved (1) OK 11/5/2019 11/5/2019

Subtotal 1

Subtotal 1

Disability Determination Bureau (2) Approved (1) OK 1/10/2019 1/10/2019

Subtotal 1

Rejected (1) OK 6/3/2019 7/3/2019

Subtotal 1

Subtotal 2

Row Counts Detail Rows Subtotals Grand Total Conditional Formatting

DISPLAY DENSITY

Comfy Compact

OPTIONS

Switch to Salesforce Classic Add Username

Naveen Bottu playful-wolf-jdv6y4-dev-ed.trailblaze.lightning.force.com/lightning/r/Report/00O2w00000CMExeEAP/edit?queryScope=userFolders

Chatter Reports Dashboards Departments Travel Approvals TA-00004 Approval

Adding Chart properties

REPORT ▾ Travel Requests by Department ▾ Travel Approvals

Fields > Outline Filters 1

Groups

- GROUP ROWS
- Add group...

Department x Status x

GROUP COLUMNS

- Add group...

Columns

- Add column...
- # Out-of-State
- Destination State
- Trip Start Date
- Trip End Date

Department Status Out-of-State Destination State Trip Start Date Trip End Date

Contract Management (1) Approved (1) OK 11/5/2019 11/5/2019

Subtotal 1

Subtotal 1

Disability Determination Bureau (2) Approved (1) OK 1/10/2019 1/10/2019

Subtotal 1

Rejected (1) OK 6/3/2019 7/3/2019

Subtotal 1

Subtotal 2

Row Counts Detail Rows Subtotals Grand Total

Update Preview Automatically

Bar Column Stacked Bar Stacked Column Line Donut Funnel Scatter Plot

Chart Attributes

Chart Title

X-Axis Department

+ Group

Y-Axis Record Count

Show Reference Line

+ Measure

Show Values

Save & Run Save Close Run

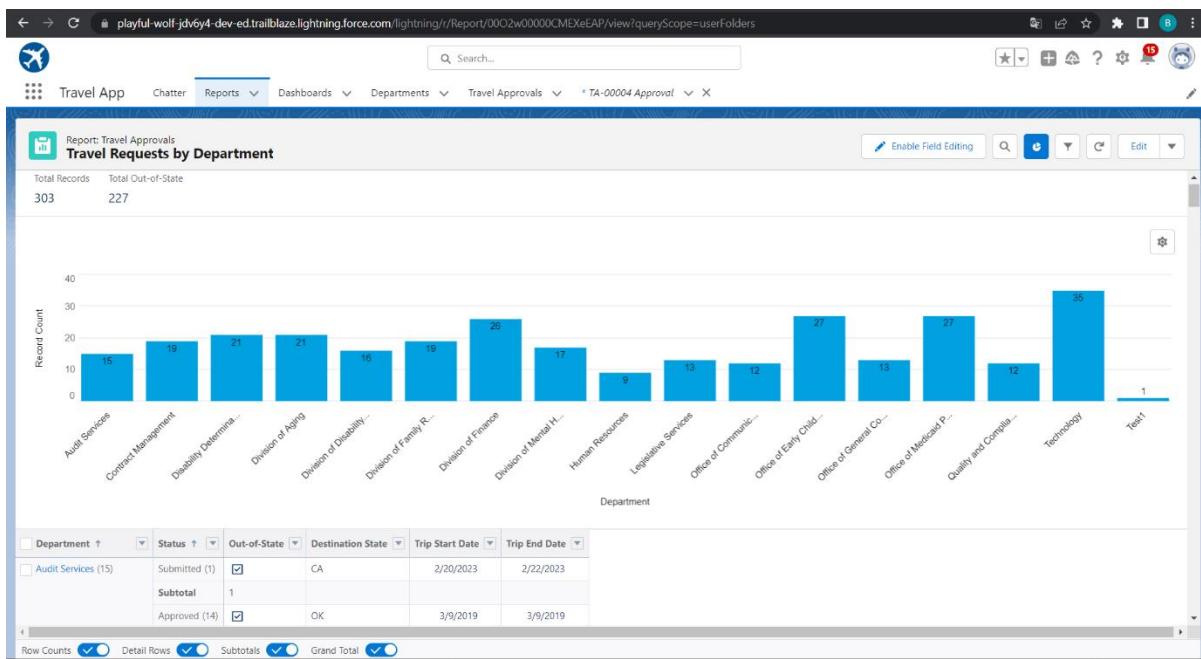
playful-wolf-jdv6y4-dev-ed.trailblaze.lightning.force.com/lightning/r/Report/00O2w00000CMExeEAP/edit?queryScope=userFolders

Chatter Reports Dashboards Departments Travel Approvals TA-00004 Approval

Saving in public folder

The screenshot shows the Salesforce Lightning interface with the URL <playful-wolf-jdv6y4-dev-ed.trailblaze.lightning.force.com/lightning/r/Report/00O2w00000CMExeEAP/edit?queryScope=userFolders>. A modal window titled "Save Report As" is open, prompting the user to enter a report name ("Copy of Travel Requests by Department"), a unique name ("Copy_of_Travel_Requests_by_Department_U9s"), and a description. The report is categorized under the "Public Reports" folder. The background shows the report configuration interface with various fields and columns defined.

Final Report



Step 3

Creating Travel request by Month Report

The screenshot shows a Salesforce Lightning report titled "Travel Requests by Month". The report displays travel data grouped by month, with subtotals for each group. The columns include Trip End Date, Out-of-State, Department, Status, Destination State, and Trip Start Date. The data shows travel requests for various departments across different states, with some trips being approved and others rejected.

Trip End Date	Out-of-State	Department	Status	Destination State	Trip Start Date
January 2019 (2)	(1)	Division of Family Resources	Approved	TX	1/4/2019
	(1)	Disability Determination Bureau	Approved	OK	1/10/2019
	Subtotal				
March 2019 (1)	(1)	Legislative Services	Approved	TX	3/6/2019
	Subtotal				
April 2019 (1)	(1)	Office of Early Childhood and Out-of-School Learning	Approved	FL	4/4/2019
	Subtotal				
May 2019 (1)	(1)	Quality and Compliance Office	Approved	CA	5/2/2019
	Subtotal				
June 2019 (1)	(1)	Division of Disability and Rehabilitative Services	Rejected	OK	3/4/2019
	Subtotal				
July 2019 (2)	(2)	Disability Determination Bureau	Rejected	OK	6/3/2019
	Subtotal				

Final report

The screenshot shows the final version of the Salesforce Lightning report titled "Travel Requests by Month". The report has been optimized, showing only the total number of records (303) and the detailed travel data for January 2019. The data for January includes trips from various departments such as Division of Family Resources, Division of Finance, and Office of General Counsel, all of which were approved and destined for Texas. The report also includes a subtotal for the month and a final grand total.

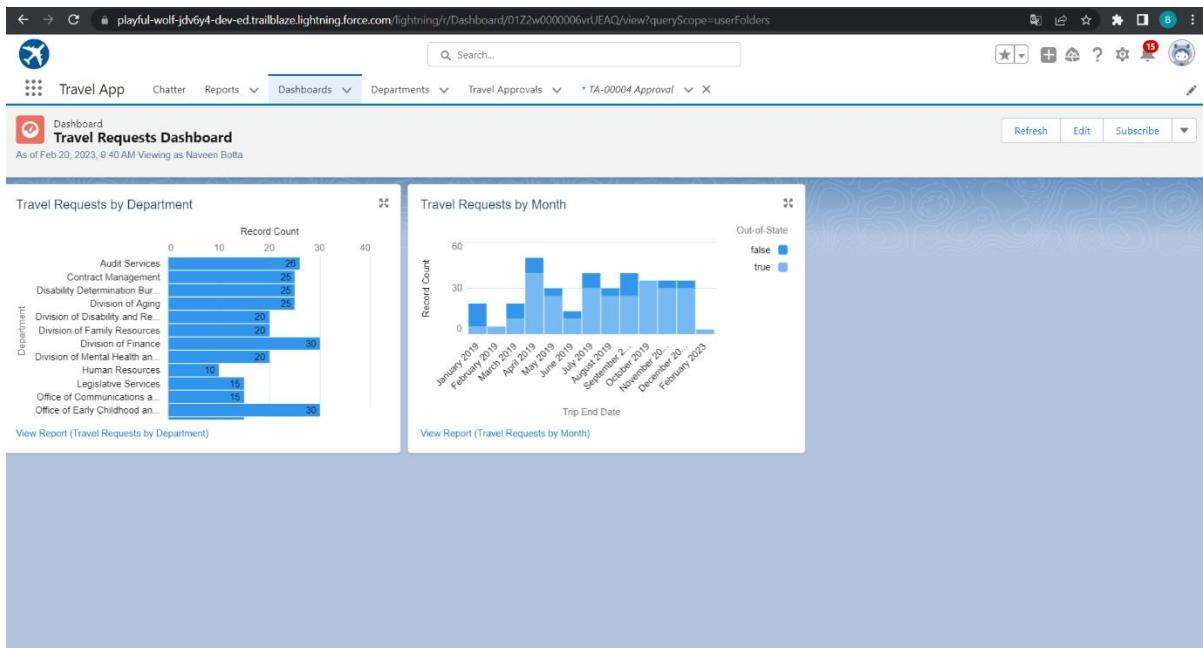
Trip End Date	Out-of-State	Department	Status	Destination State	Trip Start Date
January 2019 (18)	(13)	Division of Family Resources	Approved	TX	1/4/2019
		Division of Finance	Approved	TX	1/10/2019
		Office of General Counsel	Approved	TX	1/4/2019
		Division of Family Resources	Approved	TX	1/4/2019
		Division of Finance	Approved	TX	1/10/2019
		Office of General Counsel	Approved	TX	1/4/2019
		Office of General Counsel	Approved	TX	1/4/2019
		Division of Family Resources	Approved	TX	1/4/2019
		Division of Finance	Approved	TX	1/10/2019
		Office of General Counsel	Approved	TX	1/4/2019
		Division of Family Resources	Approved	TX	1/4/2019
		Division of Finance	Approved	TX	1/10/2019
	Subtotal	(5)			
		Disability Determination Bureau	Approved	OK	1/10/2019
		Disability Determination Bureau	Approved	OK	1/10/2019

Step 4

Dash board saved in private folder

The screenshot shows the Salesforce Lightning interface for the Travel App. The top navigation bar includes Chatter, Reports, Dashboards, Departments, Travel Approvals, and a search bar. A sidebar on the left lists categories like Dashboards, Recent, Folders, and Favorites. The main content area displays a table titled 'Dashboards' with columns for Dashboard Name, Description, Folder, Created By, and Created On. One entry is visible: 'Travel Requests Dashboard' in the 'Private Dashboards' folder, created by Naveen Botta on 2/14/2023 at 7:36 PM. A context menu is open on the right side of the dashboard table, showing options for 'DISPLAY DENSITY' (set to 'Comfy'), 'OPTIONS' (with 'Switch to Salesforce Classic' and 'Add Username'), and user information for Naveen Botta.

Final Dashboard

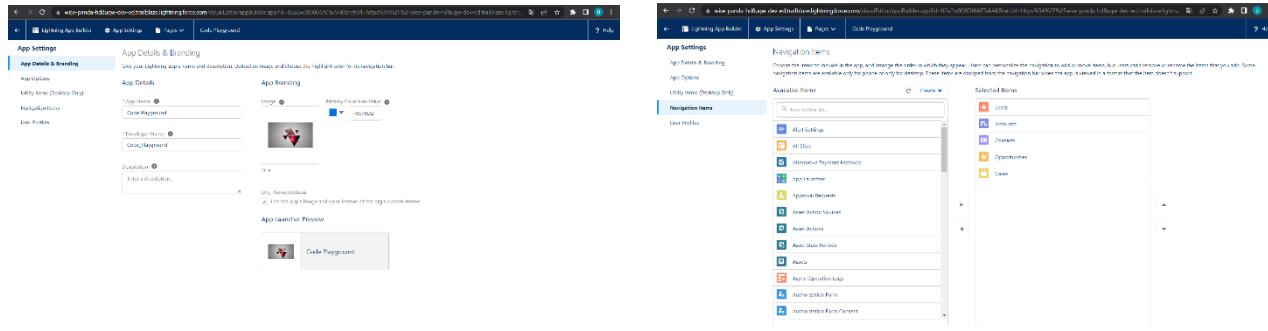


Module 3

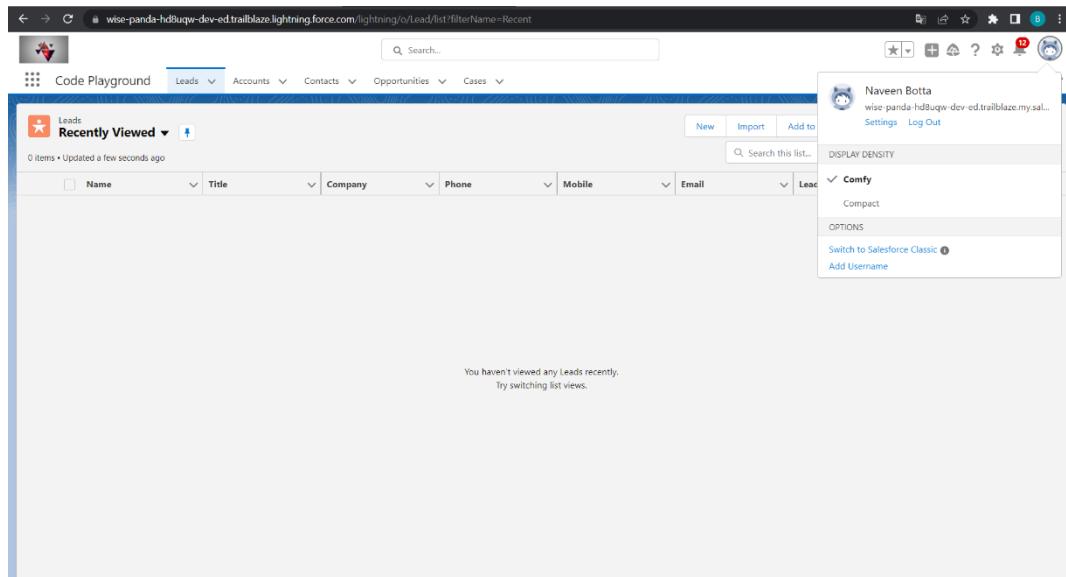
Code Playground App

Exercise 1

Creating Code Playground Lightning App with Lead, Account, Contact, Opportunity, Case to navigation profile as system admin.



Finally, it looks Like as shown



Creating custom objects

The screenshot shows the Salesforce Object Manager interface. On the left, a sidebar lists various setup options like Fields & Relationships, Page Layouts, and Buttons, Links, and Actions. The main area is titled 'Customer' under 'Details'. It shows the API Name as 'Customer__c', the singular label as 'Customer', and the plural label as 'Customers'. A right-hand sidebar displays user information (Naveen Botta) and settings, including a 'Comfy' theme option.

Custom fields on Customer custom object

The screenshot shows the 'Fields & Relationships' section for the 'Customer' object. It lists seven fields: Active (checkbox), Created By (lookup to User), Customer Name (text), Customer Type (picklist), Description (text area), Last Modified By (lookup to User), and Owner (lookup to User or Group). The 'Customer Name' field is indexed, indicated by a checkmark in the 'INDEXED' column.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Active	Active__c	Checkbox		
Created By	CreatedById	Lookup(User)		
Customer Name	Name	Text(80)		✓
Customer Type	Customer_Type__c	Picklist		
Description	Description__c	Text Area(255)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User.Group)		✓

Creating Billing Object

The screenshot shows the Salesforce Object Manager interface. On the left, a sidebar lists various configuration options like Fields & Relationships, Page Layouts, and Record Types. The main area is titled "Details" and shows the following information for the "Billing" object:

- Description:** API Name: Billing__c, Custom: ✓, Singular Label: Billing, Plural Label: Billings.
- Enable Reports:** ✓
- Track Activities:** ✓
- Track Field History:** ✓
- Deployment Status:** Deployed
- Help Settings:** Standard salesforce.com Help Window

The top right corner shows the user profile of Naveen Botta and various display density and options settings.

Tab only for code playground app

The screenshot shows the Salesforce Tabs setup page. The sidebar on the left includes links like Setup Home, Service Setup Assistant, and Platform Tools. The main area is titled "Tabs" and lists various standard tabs:

- Sample Console (standard__ServiceConsole)
- High Volume Customer Portal User
- Authenticated Website User
- App Launcher (standard__AppLauncher)
- Community (standard__Community)
- Site.com (standard__Sites)
- Salesforce Chatter (standard__Chatter)
- Content (standard__Content)
- Analytics Studio (standard__Insights)
- Sales (standard__LightningSales)
- Sales Console (standard__LightningSalesConsole)
- Service Console (standard__LightningService)
- Lightning Usage App (standard__LightningInstrumentation)
- Digital Experiences (standard__SalesforceCMS)
- Commerce (standard__Commerce)
- Data Manager (standard__DataManager)
- Bolt Solutions (standard__LightningBolt)
- Code Playground (Code_Playground) - This tab has a checked checkbox below it: "Append tab to users' existing personal customizations".

The top right corner shows the user profile of Naveen Botta and various display density and options settings.

Tabs for Customer and Billing Objects

The screenshot shows the Salesforce Setup interface with the 'Custom Tabs' page selected. The left sidebar has 'Tabs' highlighted under 'User Interface'. The main content area displays sections for 'Custom Object Tabs', 'Web Tabs', 'Visualforce Tabs', and 'Lightning Component Tabs', each with a table showing tabs and their styles.

Action	Label	Tab Style	Description
Edit Del	Billing	Desk	
Edit Del	Customers	Radar dash	

Action	Label	Tab Style	Description
New	What Is This?		

Action	Label	Tab Style	Description
New	What Is This?		

Action	Label	Tab Style	Description
New	What Is This?		

Custom fields on billing

The screenshot shows the Salesforce Setup interface with the 'Fields & Relationships' page for the 'Billing' object selected. The left sidebar lists various configuration options like Page Layouts, Lightning Record Pages, and Field Sets. The main content area displays a table of fields and their relationships.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD
Amount Paid	Amount_Paid__c	Currency(18, 0)	
Bill Number	Name	Auto Number	
Created By	CreatedBy	Lookup(User)	
Customer	Customer__c	Master-Detail(Customer)	
Customer Type	Customer_Type__c	Picklist	
Last Modified By	LastModifiedBy	Lookup(User)	
Status	Status__c	Picklist	

Finally, it looks as shown below

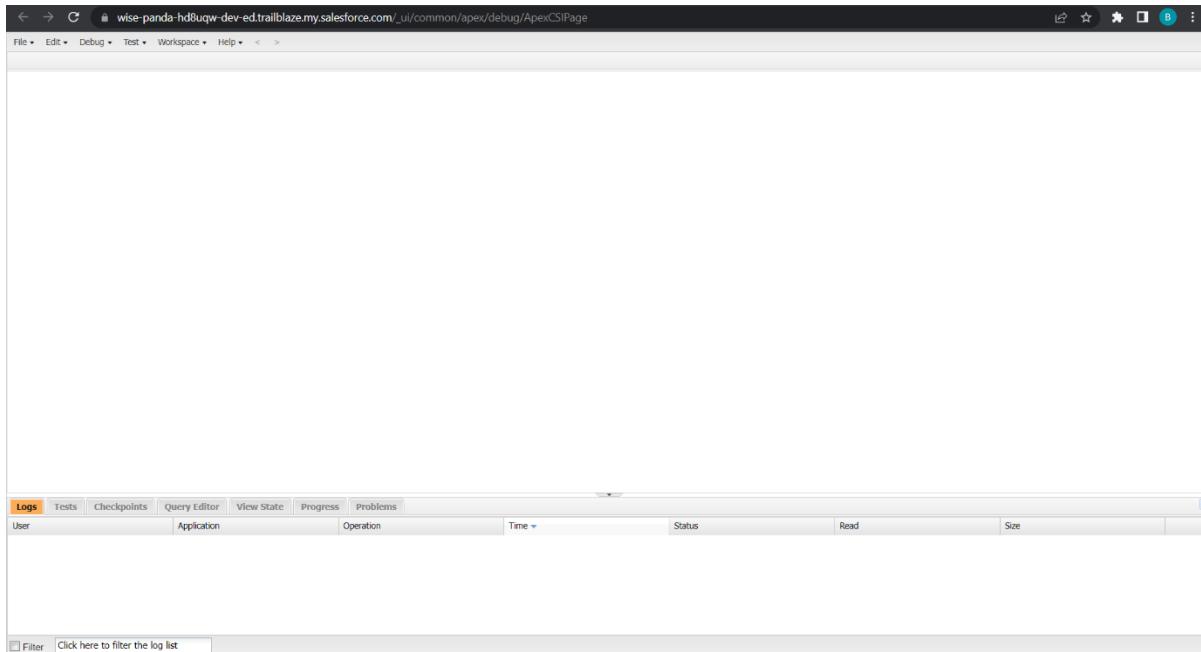
The screenshot shows a Salesforce Lightning interface for the Accounts page. The main content area displays a list of 13 accounts, each with a checkbox, account name, account site, billing state/province, phone number, and type. The accounts listed are:

Rank	Account Name	Account Site	Billing State/Province	Phone	Type
1	Burlington Textiles Corp of America		NC	(336) 222-7000	Customer - Direct
2	Dickenson plc		KS	(785) 241-6200	Customer - Channel
3	Edge Communications		TX	(512) 757-6000	Customer - Direct
4	Express Logistics and Transport		OR	(503) 421-7800	Customer - Channel
5	GenePoint		CA	(650) 867-3450	Customer - Channel
6	Grand Hotels & Resorts Ltd		IL	(312) 596-1000	Customer - Direct
7	Pyramid Construction Inc.			(014) 427-4427	Customer - Channel
8	Sample Account for Entitlements				autoproc
9	sForce		CA	(415) 901-7000	NBott
10	United Oil & Gas Corp.		NY	(212) 842-5500	Customer - Direct
11	United Oil & Gas, Singapore		Singapore	(650) 450-8810	Customer - Direct
12	United Oil & Gas, UK		UK	+44 191 4956203	Customer - Direct
13	University of Arizona		AZ	(520) 773-9050	Customer - Direct

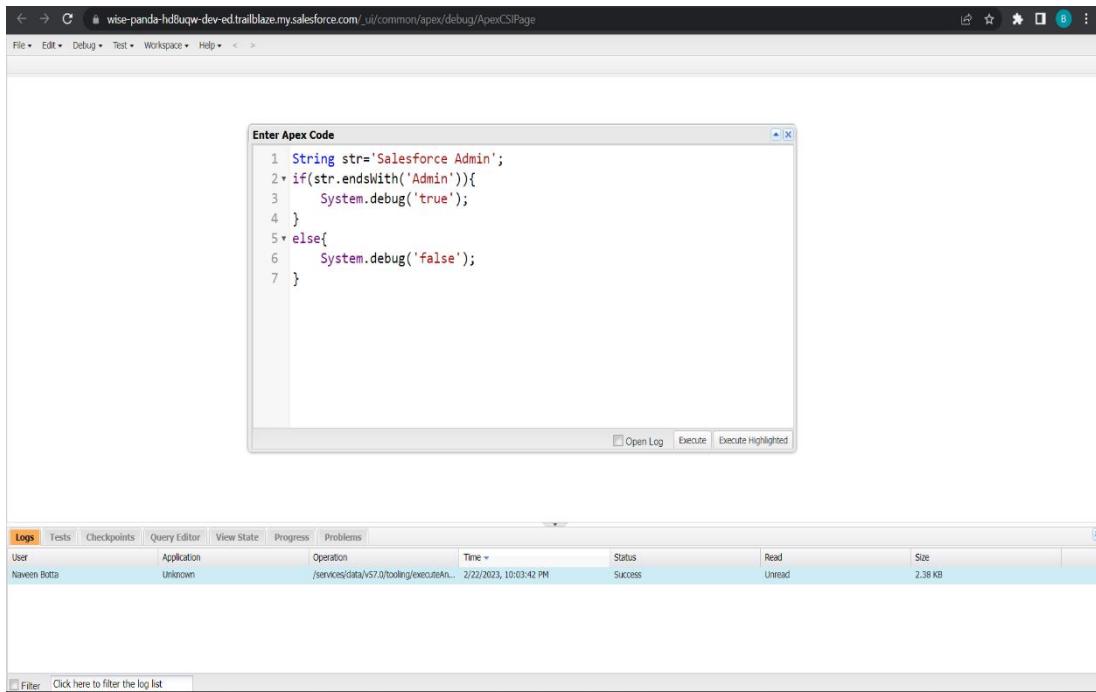
The right sidebar shows a user profile for Naveen Bott (wise-panda-hd8uqw-dev-ed.trailblaze.my.salesforce.com), with options to switch to Salesforce Classic or add a username. The sidebar also includes a "DISPLAY DENSITY" dropdown set to "Comfy" and a "Compact" option.

Exercise 2

Use Execute Anonymous to define and execute the following code:
Developer Console



1. Define a String Variable & use string method ‘endsWith’ to display the output.
By clicking on Debug and select open Anonymous window



Output

This screenshot shows the Salesforce Apex CS Page. At the top, there's a header bar with the URL "wise-panda-hd@ugw-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSPage". Below the header is a toolbar with options like File, Edit, Debug, Test, Workspace, and Help. A banner at the top center says "Log executeAnonymous @2/22/2023, 10:04:15 PM".

The main area has two tabs: "Execution Log" and "Logs". The "Execution Log" tab is active, showing a table with columns: Timestamp, Event, and Details. One entry is visible: "22:04:15:004 USER_DEBUG [3]||DEBUG|true".

The "Logs" tab is also visible, showing a table with columns: User, Application, Operation, Time, Status, Read, and Size. It lists two log entries:

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:04:15 PM	Success	Read	2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:03:42 PM	Success	Unread	2.38 KB

At the bottom of the page, there are filters for "This Frame", "Executable", "Debug Only", and "Filter", along with a link "Click here to filter the log".

This screenshot shows the "Enter Apex Code" dialog box, which is part of the Salesforce Apex CS Page. The dialog contains the following Apex code:

```
1 String str='Salesforce Admin';
2 if(str.endsWith('ad')){
3     System.debug('true');
4 }
5 else{
6     System.debug('false');
7 }
```

Below the code editor are three buttons: "Open Log", "Execute", and "Execute Highlighted".

This screenshot shows the Salesforce Apex CS Page, similar to the one above but with different log entries.

The "Execution Log" table shows one entry: "22:05:51:004 USER_DEBUG [5]||DEBUG|false".

The "Logs" table shows three log entries:

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:05:51 PM	Success	Read	2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:04:15 PM	Success	Read	2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:03:42 PM	Success	Unread	2.38 KB

At the bottom, there are filters for "This Frame", "Executable", "Debug Only", and "Filter", along with a link "Click here to filter the log list".

2. Define 2 Date type variables, use Date method today() & addDays(30) to display the output

The screenshot shows the Salesforce Apex Dev Console interface. At the top, there are tabs for 'File', 'Edit', 'Debug', 'Test', 'Workspace', and 'Help'. Below the tabs, three log entries are listed:

- 22:13:40:004 USER_DEBUG [3]DEBUG todays date : 22/02/2023
- 22:13:40:005 USER_DEBUG [5]DEBUG Adding days :12/07/2000

In the center, there is a modal window titled 'Enter Apex Code' containing the following code:

```

1 Date todayDate=Date.today();
2 String strDate=Date.today().format();
3 System.debug('todays date : '+strDate);
4 Date dt = Date.newInstance(2000, 06, 12);
5 System.debug('Adding days :'+dt.addDays(30).format());

```

At the bottom of the modal are buttons for 'Open Log', 'Execute', and 'Execute Highlighted'. Below the modal, the main console area shows a table of logs:

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:13:40 PM	Success		3.21 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:05:51 PM	Success		2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:04:15 PM	Success		2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:03:42 PM	Success	Unread	2.38 KB

3. Display the output of an Integer variable from string '10' and then add 20 to it

The screenshot shows the Salesforce Apex Dev Console interface. At the top, there are tabs for 'File', 'Edit', 'Debug', 'Test', 'Workspace', and 'Help'. Below the tabs, three log entries are listed:

- 22:17:38:005 USER_DEBUG [3]DEBUG Sum :30

In the center, there is a modal window titled 'Enter Apex Code' containing the following code:

```

1 String num1 = '10';
2 Integer value=Integer.valueOf(num1);
3 System.debug('Sum :'+(value+20));

```

At the bottom of the modal are buttons for 'Open Log', 'Execute', and 'Execute Highlighted'. Below the modal, the main console area shows a table of logs:

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:17:38 PM	Success		2.48 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:13:40 PM	Success		3.21 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:05:51 PM	Success		2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:04:15 PM	Success		2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:03:42 PM	Success	Unread	2.38 KB

4. Define a String Variable & use string method length() to display the output

The screenshot shows the Salesforce Apex Dev Console. In the top right, there are tabs for various log entries. Below them is the 'Execution Log' section with columns for 'Timestamp', 'Event', and 'Details'. A single entry is shown: '22:19:57:005 USER_DEBUG [1]DEBUG|Length of string :6'. To the right is an 'Enter Apex Code' window containing the following code:

```
1 String toFindLen ='naveen';
2 Integer value= toFindLen.length();
3 System.debug('Length of string :'+value);
```

At the bottom of the page is a log viewer with tabs for 'Logs', 'Tests', 'Checkpoints', 'Query Editor', 'View State', 'Progress', and 'Problems'. The 'Logs' tab is selected, showing a list of log entries from a user named 'Naveen Botta'. The log entries are:

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:19:57 PM	Success		2.52 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:17:38 PM	Success		2.48 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:13:40 PM	Success		3.21 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:05:51 PM	Success		2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:04:15 PM	Success		2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 10:03:42 PM	Success	Unread	2.38 KB

5. Define a List of integer and display the output using add(), get(), set(), clear(), methods

The screenshot shows the Salesforce Apex Dev Console. In the top right, there are tabs for various log entries. Below them is the 'Execution Log' section with columns for 'Timestamp', 'Event', and 'Details'. Four entries are shown: '22:24:53:005 USER_DEBUG [5]DEBUG|(1, 2, 3)', '22:24:53:005 USER_DEBUG [7]DEBUG|2', '22:24:53:005 USER_DEBUG [9]DEBUG|(5, 2, 3)', and '22:24:53:005 USER_DEBUG [11]DEBUG|()'. To the right is an 'Enter Apex Code' window containing the following code:

```
1 List<Integer> list1=new List<Integer>();
2 list1.add(1);
3 list1.add(2);
4 list1.add(3);
5 System.debug(list1);
6 Integer fetch = list1.get(1);
7 System.debug(fetch);
8 list1.set(0,5);
9 system.debug(list1);
10 list1.clear();
11 system.debug(list1);
```

At the bottom of the page is a log viewer with tabs for 'Logs', 'Tests', 'Checkpoints', 'Query Editor', 'View State', 'Progress', and 'Problems'. The 'Logs' tab is selected, showing a list of log entries from a user named 'Naveen Botta'. The log entries are:

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/22/2023, 10:24:53 PM	Success		3.52 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/22/2023, 10:19:57 PM	Success		2.52 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/22/2023, 10:17:38 PM	Success		2.48 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/22/2023, 10:13:40 PM	Success		3.21 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/22/2023, 10:05:51 PM	Success		2.38 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/22/2023, 10:04:15 PM	Success		2.38 KB

Exercise 2

Use Execute Anonymous to define and execute the following code to display the value of x = 0 to 9

The screenshot shows the Salesforce Apex CS Page interface. At the top, there are several tabs labeled with log entries: 'Log executeAnonymous @2/22/2023, 10:13:40 PM', 'Log executeAnonymous @2/22/2023, 10:17:38 PM', 'Log executeAnonymous @2/22/2023, 10:19:57 PM', 'Log executeAnonymous @2/22/2023, 10:24:53 PM', and 'Log executeAnonymous @2/22/2023, 10:28:54 PM'. Below the tabs is a toolbar with icons for File, Edit, Debug, Test, Workspace, Help, and a back/forward arrow. The main area is divided into two sections: 'Execution Log' on the left and 'Enter Apex Code' on the right. The 'Execution Log' table shows a series of debug logs from 22:28:54:003 to 22:28:54:004, all labeled 'USER_DEBUG' and showing the value of 'x' from 0 to 9. The 'Enter Apex Code' pane contains the following code:

```
1 Integer x=20;
2
3 for(integer x=0;x<10;x+=1){
4     system.debug('Value of x is : '+x);
5 }
```

At the bottom of the page are three buttons: 'Open Log', 'Execute', and 'Execute Highlighted'.

Below the main page is a separate 'Logs' section with a table of operations. The table has columns for User, Application, Operation, Time, Status, Read, and Size. It lists several successful operations by 'User' Naveen Botta, mostly related to '/services/data/v57.0/tooling/executeA...', with sizes ranging from 2.48 KB to 18.51 MB.

Exercise 3

Answer the following in True or False:

Integer myunluckyNumber = 7;

Integer myluckyNumber = 15; → myluckyNumber != myunluckyNumber + 8.

The screenshot shows the Salesforce Apex CS Page interface. At the top, there are several tabs labeled with log entries: 'Log executeAnonymous @2/22/2023, 10:17:38 PM', 'Log executeAnonymous @2/22/2023, 10:19:57 PM', 'Log executeAnonymous @2/22/2023, 10:24:53 PM', 'Log executeAnonymous @2/22/2023, 10:28:54 PM', and 'Log executeAnonymous @2/22/2023, 10:33:06 PM'. Below the tabs is a toolbar with icons for File, Edit, Debug, Test, Workspace, Help, and a back/forward arrow. The main area is divided into two sections: 'Execution Log' on the left and 'Enter Apex Code' on the right. The 'Execution Log' table shows a single log entry at 22:33:06:003, labeled 'USER_DEBUG' and showing the value 'false'. The 'Enter Apex Code' pane contains the following code:

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

At the bottom of the page are three buttons: 'Open Log', 'Execute', and 'Execute Highlighted'.

Below the main page is a separate 'Logs' section with a table of operations. The table has columns for User, Application, Operation, Time, Status, Read, and Size. It lists several successful operations by 'User' Naveen Botta, mostly related to '/services/data/v57.0/tooling/executeA...', with sizes ranging from 2.48 KB to 18.51 MB.

Exercise 4

Answer the following in True Or False:

Boolean isFalse = false;

Boolean isTrue = True; (isTrue || isFalse)

The screenshot shows the Salesforce Debug 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 is a tab bar with Execution Log, Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Execution Log tab is selected, showing a table of logs with columns for Timestamp, Event, Details, User, Application, Operation, Time, Status, Read, and Size. There are six log entries from the user 'Naveen Botta'. The Logs tab is also selected, showing a table of logs with similar columns. In the center, there's an 'Enter Apex Code' window with the following code:

```
1 Boolean isTrue = True;
2 Boolean isFalse = False;
3 if(isTrue || isFalse){
4     system.debug('True');
5 }
6 else{
7     system.debug('False');
8 }
```

At the bottom of the page are buttons for Open Log, Execute, and Execute Highlighted.

Exercise 5

Answer the following in True Or False:

Date tomorrow = Date.today().addDays(1);

Date today = Date.today();

today != tomorrow.

The screenshot shows the Salesforce Debug 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 is a tab bar with Execution Log, Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Execution Log tab is selected, showing a table of logs with columns for Timestamp, Event, Details, User, Application, Operation, Time, Status, Read, and Size. There are six log entries from the user 'Naveen Botta'. The Logs tab is also selected, showing a table of logs with similar columns. In the center, there's an 'Enter Apex Code' window with the following code:

```
1 Date today = Date.today();
2 Date tomorrow = Date.today().addDays (1);
3 if(today!=tomorrow){
4     system.debug('True');
5 }
6 else{
7     system.debug('False');
8 }
```

At the bottom of the page are buttons for Open Log, Execute, and Execute Highlighted.

Exercise 6

Write a program and execute to demo the use of “If..else if...else”

The screenshot shows the Salesforce ApexCSLPage interface. At the top, there's a navigation bar with links like File, Edit, Debug, Test, Workspace, Help, and a search bar. Below it is a tab bar with several tabs labeled with log entries. The main area has two panes: the left pane is titled "Execution Log" and lists a single entry for "USER_DEBUG" at timestamp 22:43:13:003; the right pane is titled "Enter Apex Code" and contains the following Apex code:

```
1 Integer Score =81;
2 if(Score == 100){
3     System.debug('Grade A+');
4 }else if(Score >= 90){
5     System.debug('Grade A');
6 }else if(Score >= 80){
7     System.debug('Grade B');
8 }else{
9     System.debug('Grade : Failed');
10 }
```

At the bottom of the code editor are three buttons: Open Log, Execute, and Execute Highlighted.

Below the code editor is a log viewer titled "Logs". It has a toolbar with filters for This Frame, Executable, Debug Only, and Filter. The "Logs" tab is selected. The log table has columns for User, Application, Operation, Time, Status, Read, and Size. There are six log entries from the user "Naveen Botta" with various operation details and sizes ranging from 2.68 KB to 18.51 MB. The last entry shows an "Apex CPU time limit exceeded" error.

Exercise 7

Write a program to execute and demo the use of “Apex – for Loop”

The screenshot shows the Salesforce Lightning interface. At the top, there's a navigation bar with links for Leads, Accounts, Contacts, Opportunities, Cases, Customers, and Billings. The main content area displays a list of bills under the "Billings" tab. The list is titled "Recently Viewed" and shows four items, each with a checkbox and a bill number: B - 0004, B - 0003, B - 0002, and B - 0001. To the right of the list is a user profile sidebar for "Naveen Botta" with options for Settings and Log Out. The sidebar also includes sections for DISPLAY DENSITY (Compact), OPTIONS (Switch to Salesforce Classic, Add Username), and a message indicating the user is online.

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < ▾

11:17:38 AM | TestDisqualifyTestLeads.apxc | DisqualifyTestLeads.apxc | BillingStatus.apxc | Log executeAnonymous @2/25/2023, 8:18:56 PM | Log executeAnonymous @2/25/2023, 8:19:55 PM | Log executeAnonymous @2/25/2023, 8:20:39 PM

Code Coverage: None ▾ API Version: 57 ▾ Go To

```

1 public class BillingStatus {
2     public static void viewStatus(){
3         List<Billing__c> billList = [SELECT ID, NAME, STATUS__C FROM BILLING__C ];
4         List<String> billRecords = new List<String>();
5         for(Billing__c bill : billList)
6         {
7
8             if(bill.Status__c == 'Paid'){
9                 billRecords.add(bill.Name);
10                System.debug('Value of Current Record in the Loop '+bill);
11            }
12        }
13    }
14
15    System.debug('Value of Paid BillingList IDs '+billRecords);
16
17 }
18 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:20:39 PM	Success		7.54 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:19:55 PM	Success		8.95 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:18:56 PM	Success		7.11 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:17:51 PM	Success		7.83 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:17:01 PM	Success		7.81 KB

Filter Click here to filter the log list Stop sharing Hide

Output

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < ▾

11:17:38 AM | TestDisqualifyTestLeads.apxc | DisqualifyTestLeads.apxc | BillingStatus.apxc | Log executeAnonymous @2/25/2023, 8:18:56 PM | Log executeAnonymous @2/25/2023, 8:19:55 PM | Log executeAnonymous @2/25/2023, 8:20:39 PM

Execution Log

Timestamp	Event	Details
20:20:39:018	USER_DEBUG	[10][DEBUG]Value of Current Record in the Loop Billing__c:(Id=a012w000017u9tAA, Name=B - 0011, Status__c=Paid]
20:20:39:018	USER_DEBUG	[10][DEBUG]Value of Current Record in the Loop Billing__c:(Id=a012w000017d4rAAE, Name=B - 0007, Status__c=Paid]
20:20:39:018	USER_DEBUG	[10][DEBUG]Value of Current Record in the Loop Billing__c:(Id=a012w000017d4rAAE, Name=B - 0009, Status__c=Paid]
20:20:39:019	USER_DEBUG	[15][DEBUG]Value of Paid BillingList IDs (B - 0011, B - 0007, B - 0009, B - 0010)

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
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:20:39 PM	Success		7.54 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:19:55 PM	Success		8.95 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:18:56 PM	Success		7.11 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:17:51 PM	Success		7.83 KB
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeA...	2/25/2023, 8:17:01 PM	Success		7.81 KB

Filter Click here to filter the log list Stop sharing Hide

Exercise 8

Write a Class to demo the use of Constants in Apex

```
1 public class DiscountClass {  
2     public static Decimal calculateDiscount(Integer price){  
3         Final Decimal regularDiscount = 0.1;  
4         Decimal finalPrice = price - price * regularDiscount;  
5         return finalPrice;  
6     }  
7 }  
  
1 Decimal finalPrice= DiscountClass.calculateDiscount(100);  
2 system.debug('finalprice: '+finalprice);
```

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 11:23:26 PM	Success		3.71 KB

Output

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/22/2023, 11:23:26 PM	Success		3.71 KB

Exercise 9

Write a Class to demo the use of Interface in Apex

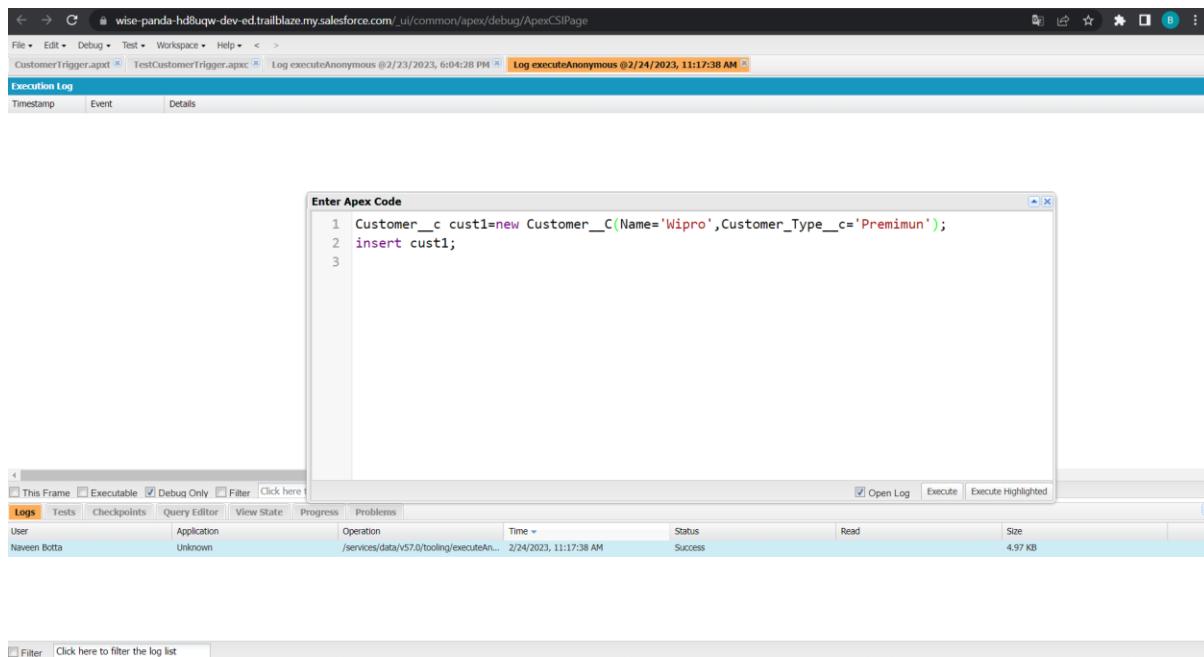


```
1 public class DemoInterface {
2
3     public interface InterfaceExample{
4         Double percentageDiscountToBeApplied();
5     }
6
7     public class PremiumCustomer implements InterfaceExample{
8         public Double percentageDiscountToBeApplied(){
9             return 0.30;
10        }
11    }
12
13    public class normalCustomer implements InterfaceExample{
14        public Double percentageDiscountToBeApplied(){
15            return 0.10;
16        }
17    }
18 }
19 }
```

Exercise 10

Demo on DML Insert Operation Using Database methods

1. Insert a Customer Record First using simple DML Statement



The screenshot shows the Salesforce developer console with the following details:

- Execution Log:** A modal window titled "Enter Apex Code" contains the following Apex code:

```
1 Customer__c cust1=new Customer__c(Name='Wipro',Customer_Type__c='Premiumun');
2 insert cust1;
3
```
- Logs:** The main console displays the execution log for the trigger.

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/24/2023, 11:17:38 AM	Success		4.97 KB

Result in UI record inserted into customer.

The screenshot shows a Salesforce Lightning interface. At the top, there's a navigation bar with links for Leads, Accounts, Contacts, Opportunities, Cases, Customers, and Billings. Below the navigation is a search bar and a user profile for 'Naveen Botta'. The main content area displays a 'Customer Wipro' record. The 'Details' tab is selected, showing fields like Customer Name (Wipro), Owner (Naveen Botta), Active status (unchecked), Customer Type (Premium), and a short description. It also shows the creation and last modification details. A sidebar on the right provides display density options (Comfy, Compact) and links to switch to Salesforce Classic or add a username.

2. Insert Billing record using Database methods

Here use record id to insert customer.

The screenshot shows the Salesforce Dev Console. At the top, it displays the file path 'CustomerTrigger.apxt' and the log entry 'Log executeAnonymous @2/23/2023, 6:04:28 PM'. Below this is the 'Execution Log' section, which shows a single log entry: '18:04:28:044 USER_DEBUG [6] DEBUG|Successfully inserted bill. Billing ID = a012w000017dnHaAAM'. The main area contains the Apex code for inserting a Billing record:

```
1 List<Billing__c> billList = new List<Billing__c>{new Billing__c(Status__c='Paid',Amount_Paid__c=500000,Customer__c='a002w00000Yp2Z2AAJ')};
2 
3 Database.SaveResult[] srList = Database.insert(billList,false);
4 for(Database.SaveResult sr:srList){
5     if(sr.isSuccess()){
6         system.debug('Successfully inserted bill. Billing ID = '+sr.getId());
7     }
8     else{
9         for(Database.Error err:sr.getErrors()){
10             system.debug('Error is '+err.getMessage());
11             system.debug('Fields effected by the error = '+err.getFields());
12         }
13     }
14 }
```

At the bottom, there's a 'Logs' tab showing a table of log entries. The table has columns for User, Application, Operation, Time, Status, Read, and Size. There are three entries from 'Naveen Botta' related to the Apex test execution.

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	/services/data/v57.0/tooling/executeAn...	2/23/2023, 6:04:28 PM	Success		4.64 KB
Naveen Botta	Unknown	ApexTestHandler	2/23/2023, 5:41:24 PM	Success	Unread	976 bytes
Naveen Botta	Unknown	ApexTestHandler	2/23/2023, 5:41:24 PM	Success	Unread	10.62 KB

Output in UI

The screenshot shows a Salesforce Lightning interface. At the top, there's a navigation bar with links for Leads, Accounts, Contacts, Opportunities, Cases, Customers, and Billings. Below the navigation is a search bar. The main content area displays a 'Billing' record with the ID 'B - 0010'. The 'Details' tab is selected, showing fields like Bill Number (B - 0010), Amount Paid (\$50,000.00), Customer Type (Smith), Status (Paid), and Created By (Naveen Botta). On the right side, there's a user profile for Naveen Botta, options for DISPLAY DENSITY (Comfy or Compact), and links to Switch to Salesforce Classic or Add Username.

Exercise 11

Write and execute SOQL queries from Developer Console
Display ID, Amount, Stage, Account Name, Account Industry, Account Website
From Opportunity
AND “Account Annual Revenue > 5000”
Add a Where Clause “Account Industry = Energy”

The screenshot shows the Salesforce Developer Console. At the top, it displays a URL: wise-panda-hd8uqw-dev-ed.trailblaze.lightning.force.com/lightning//Billing__c/a012w000017dNHAAAM/view. Below the URL is a navigation bar with File, Edit, Debug, Test, Workspace, Help, and a dropdown for Opportunity.obj. The main area shows a query results grid titled 'Query Results - Total Rows: 10'. The columns are Id, Amount, StageName, Account.Name, Account.Industry, and Account.Website. The results show 10 rows of data. Below the grid is the 'Query Editor' pane, which contains the SOQL query: 'SELECT Id, Amount, StageName, Account.Name, Account.Industry, Account.website FROM Opportunity WHERE ACCOUNT.INDUSTRY = \'ENERGY\' AND ACCOUNT.ANNUALREVENUE > 5000'. The 'History' pane on the right shows the query was executed successfully.

Exercise 12

Write an Apex Trigger, Name = CustomerTrigger.

Create a Billing Record when on Customer Object the 'Status' field changes to Active from Inactive.

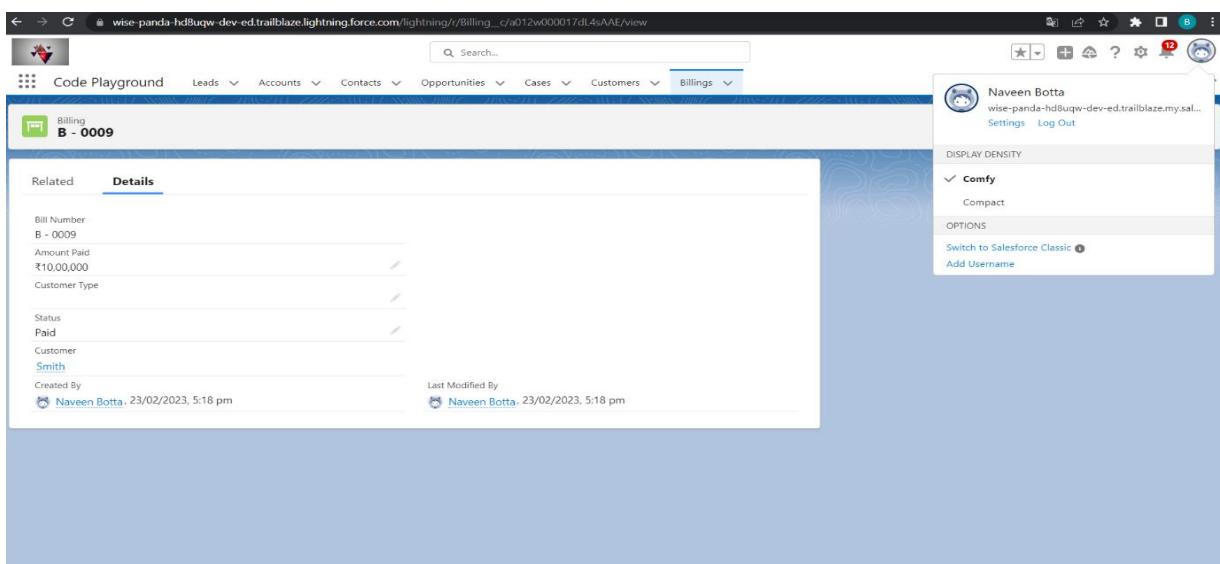
Billing "Status" must be: Paid, Amount Paid = 1000000

Perform the DML on a List.

```
trigger CustomerTrigger on Customer__c (after insert, after update) {
    List<Billing__c> newBill = new List<Billing__c>();
    for(Customer__c cust : trigger.new){
        if(cust.Active__c == True){
            Billing__c bi = new Billing__c();
            bi.Status__c = 'Paid';
            bi.Amount_Paid__c = 1000000;
            bi.Customer__c='a002w00000YpZ22AAJ';
            newBill.add(bi);
        }
        if(newBill.size()>0){
            insert newBill;
        }
    }
}
```

User	Application	Operation	Time	Status	Read	Size
Naveen Botta	Unknown	ApexTestHandler	2/23/2023, 5:41:24 PM	Success	Unread	976 bytes
Naveen Botta	Unknown	ApexTestHandler	2/23/2023, 5:41:24 PM	Success	Unread	10.62 kB

A new record is created when we created a customer record with active is true.



Exercise 13

Write a Test Class for CustomerTrigger.

Create the Data for the Customer custom Object

Use Test.startTest() & Test.stopTest().

Use assert statement to validate the output, Test Class must have full code coverage.

The screenshot shows the Salesforce Apex test editor. The code editor contains the following Apex test class:

```
1  @isTest
2  *public class TestCustomerTrigger {
3    @isTest static void testName(){
4      Customer__c com = new Customer__c();
5      com.Active__c = False;
6      insert com;
7      Test.startTest();
8      com.Active__c = True;
9      update com;
10     Test.stopTest();
11   }
12 }
```

The tabs at the bottom are Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Tests tab is selected. The status bar shows "Test Run". The table below shows the test results:

Status	Test Run	Enqueued Time	Duration	Failures	Total
✓	7072w00008KKQX0	Thu Feb 23 2023 17:31:49 GM...		0	3
✓	7072w00008KKQBK	Thu Feb 23 2023 17:18:52 GM...		0	3
✓	7072w00008KKQ88	Thu Feb 23 2023 17:16:38 GM...		0	3
✓	7072w00008KKQJ4	Thu Feb 23 2023 17:12:22 GM...		0	3
✓	7072w00008KKPM0	Thu Feb 23 2023 17:08:54 GM...		0	3
✗	7072w00008KKQ8B	Thu Feb 23 2023 17:04:08 GM...		1	3
✓	7072w00008KKQ8B	Thu Feb 23 2023 17:41:23 GM...		0	3

Overall Code Coverage table:

Class	Percent	Lines
Overall	40%	
BillingStatus	0%	0/7
CustomerTrigger	100%	10/10
DemoInterface	0%	0/4
DiscountClass	0%	0/4

Exercise 14

The screenshot shows the Salesforce Apex test editor. The code editor contains the following trigger:

```
1  *trigger DisqualifyTestLeads on Lead (before insert,before update,after insert) {
2
3    List<Lead> li = new List<Lead>();
4    if(Trigger.isBefore && Trigger.isInsert){
5      for(Lead myLead : Trigger.New){
6        if(( String.isNotBlank(myLead.FirstName) && String.isNotBlank(myLead.LastName)) && (myLead.FirstName=='Test' || myLead.LastName=='Test')){
7          System.debug(myLead.FirstName + ' ' + myLead.LastName + ' Will be disqualified!');
8          li.add(myLead);
9        }
10      }
11    }
12
13    for(Lead myLead : li){
14      myLead.Status = 'Disqualified';
15    }
16  }
```

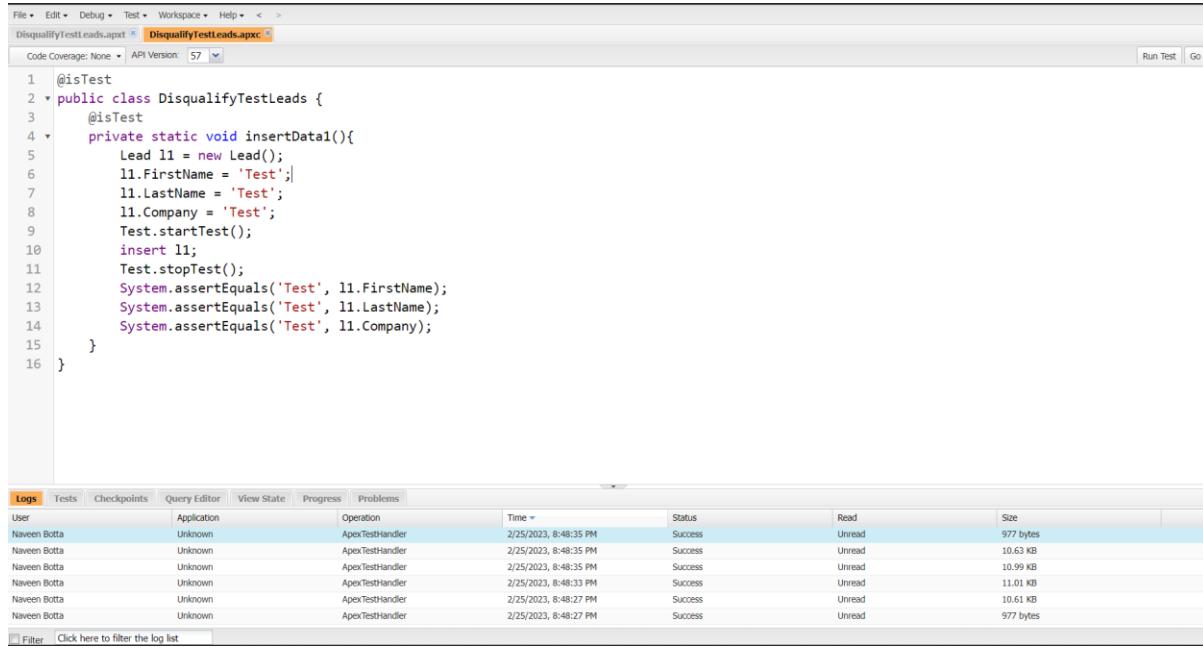
The tabs at the bottom are Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Tests tab is selected. The status bar shows "Test Run". The table below shows the test results:

Status	Test Run	Enqueued Time	Duration	Failures	Total
✓	7072w00008B0M2R	Sun Feb 26 2023 21:34:50 GM...		0	4
✓	7072w00008B0M2R	Sun Feb 26 2023 21:35:23 GM...		0	4

Overall Code Coverage table:

Class	Percent	Lines
Overall	48%	
BillingStatus	0%	0/7
CustomerTrigger	100%	10/10
DemoInterface	0%	0/4
DiscountClass	0%	0/4
DisqualifyTestLeads	100%	7/7

Exercise 15



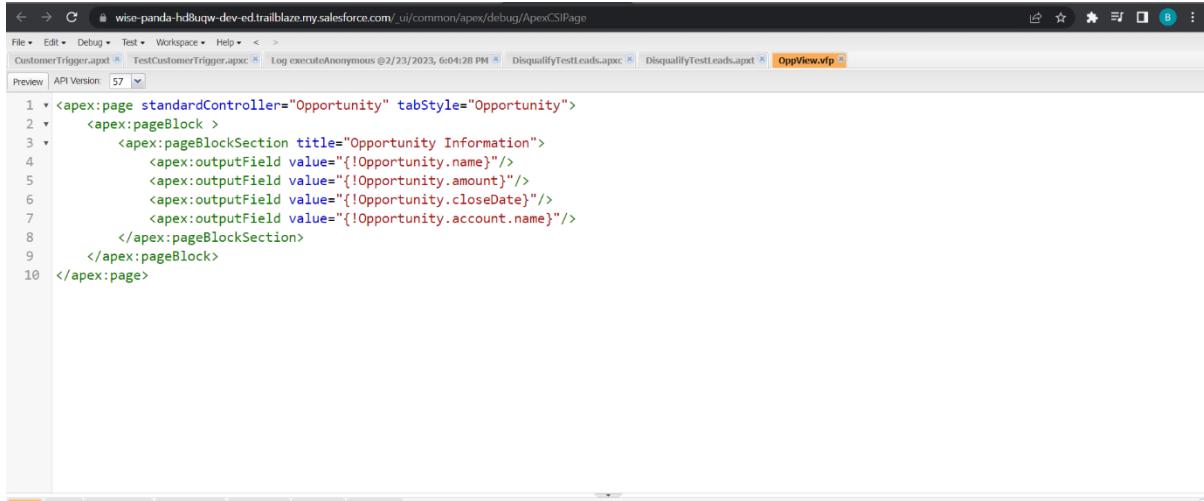
```
File • Edit • Debug • Test • Workspace • Help • < >
DisqualifyTestLeads.apxc
Code Coverage: None • API Version: 57
Run Test | Go To

1  @isTest
2  public class DisqualifyTestLeads {
3      @isTest
4      private static void insertData1(){
5          Lead l1 = new Lead();
6          l1.FirstName = 'Test';
7          l1.LastName = 'Test';
8          l1.Company = 'Test';
9          Test.startTest();
10         insert l1;
11         Test.stopTest();
12         System.assertEquals('Test', l1.FirstName);
13         System.assertEquals('Test', l1.LastName);
14         System.assertEquals('Test', l1.Company);
15     }
16 }
```

Logs	Tests	Checkpoints	Query Editor	View State	Progress	Problems
User Naveen Botta	Application Unknown	Operation ApexTestHandler	Time 2/25/2023, 8:48:35 PM	Status Success	Read Unread	Size 977 bytes
Naveen Botta	Unknown	ApexTestHandler	2/25/2023, 8:48:35 PM	Success	Unread	10.63 KB
Naveen Botta	Unknown	ApexTestHandler	2/25/2023, 8:48:35 PM	Success	Unread	10.99 KB
Naveen Botta	Unknown	ApexTestHandler	2/25/2023, 8:48:33 PM	Success	Unread	11.01 KB
Naveen Botta	Unknown	ApexTestHandler	2/25/2023, 8:48:27 PM	Success	Unread	10.61 KB
Naveen Botta	Unknown	ApexTestHandler	2/25/2023, 8:48:27 PM	Success	Unread	977 bytes

Filter Click here to filter the log list

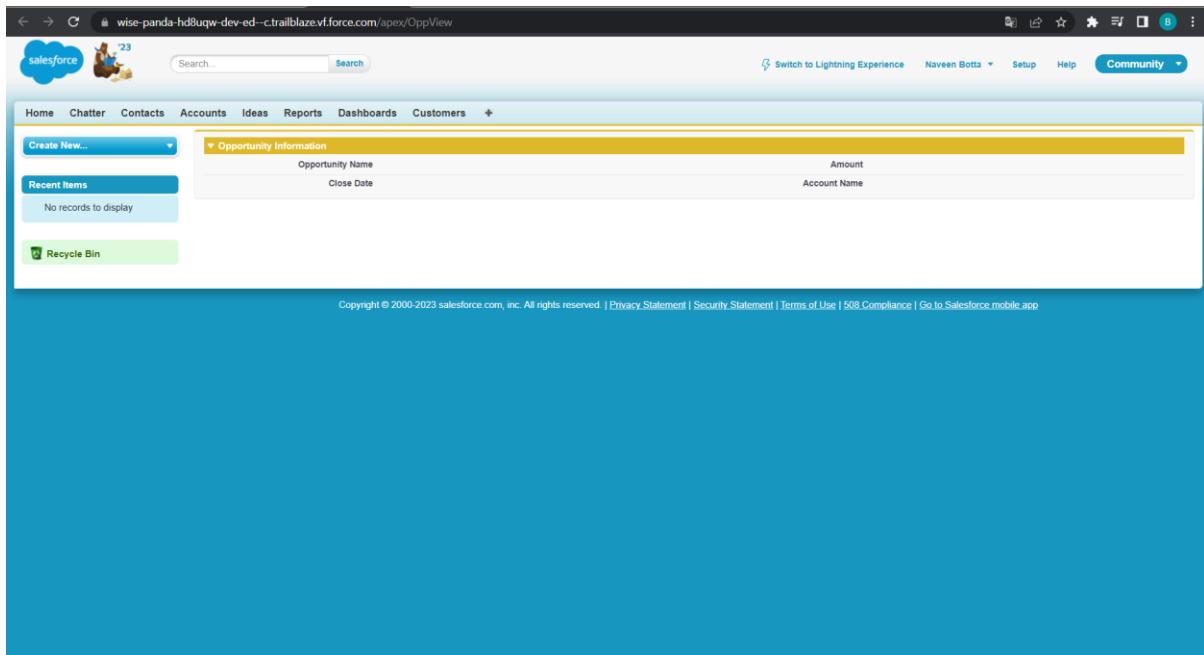
Exercise 16



The screenshot shows the Salesforce developer console with the URL `wise-panda-hd8uqw-dev-ed.trailblaze.my.salesforce.com/ui/common/apex/debug/ApexCSIPage`. The page displays the Visualforce page code:

```
1 <apex:page standardController="Opportunity" tabStyle="Opportunity">
2   <apex:pageBlock>
3     <apex:pageBlockSection title="Opportunity Information">
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>
```

Output in Visualforce Page

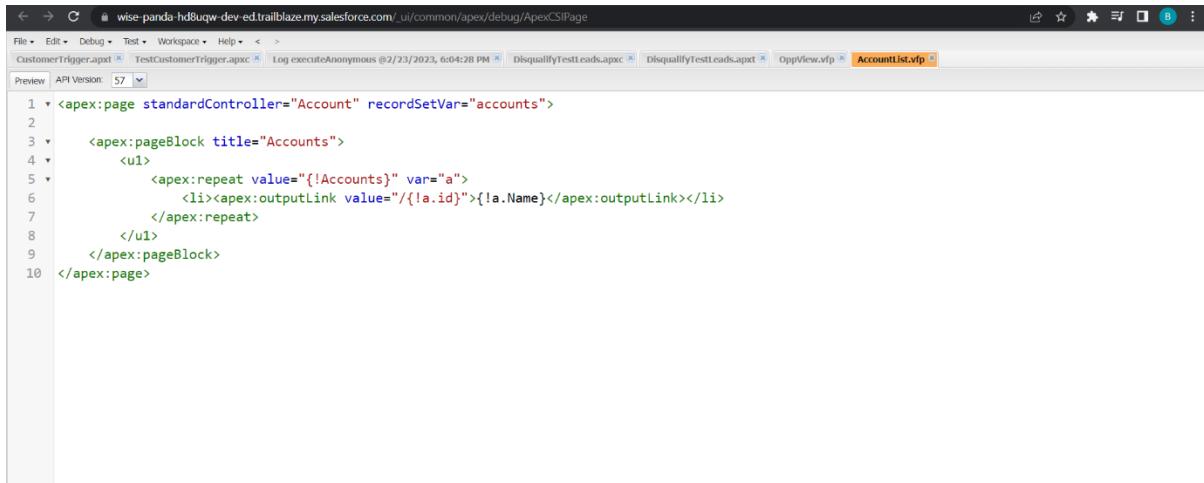


The screenshot shows the Salesforce classic interface with the URL `wise-panda-hd8uqw-dev-ed-c.trailblaze.vf.force.com/apex/OppView`. The page displays the following Visualforce output:

Opportunity Name	Amount
Close Date	Account Name

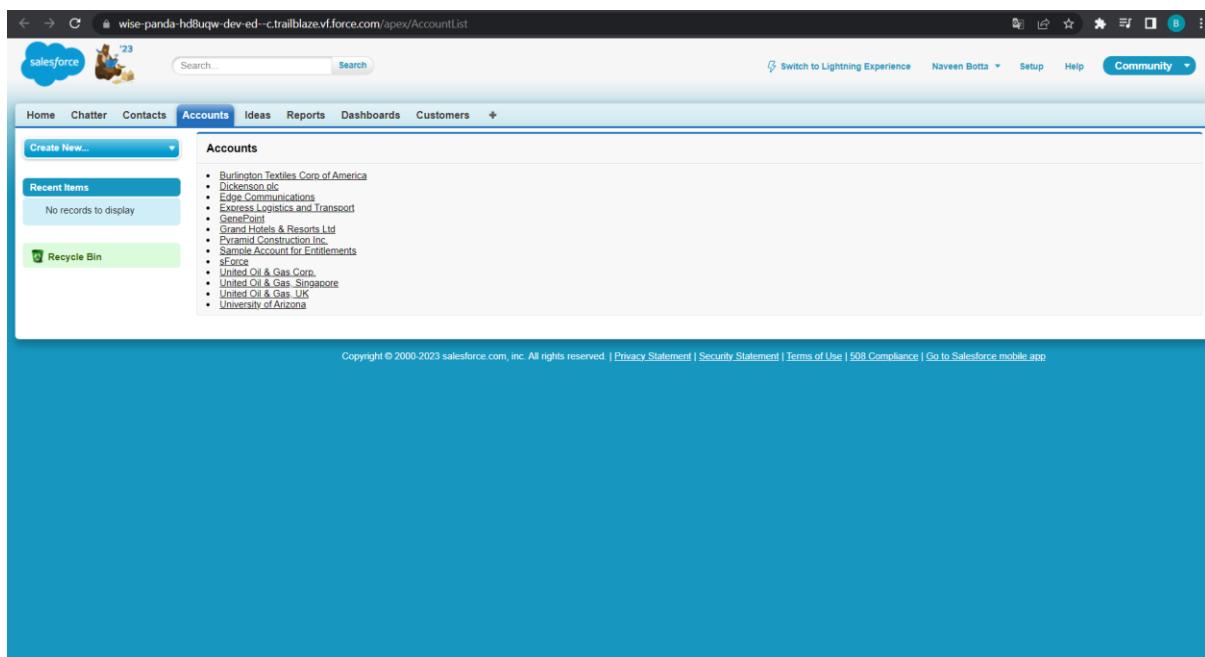
The sidebar on the left shows "Recent Items" with "No records to display". At the bottom, there is a "Recycle Bin" button.

Exercise 17



```
1 <apex:page standardController="Account" recordSetVar="accounts">
2
3     <apex:pageBlock title="Accounts">
4         <ul>
5             <apex:repeat value="{!Accounts}" var="a">
6                 <li><apex:outputLink value="/{!a.id}">{!a.Name}</apex:outputLink></li>
7             </apex:repeat>
8         </ul>
9     </apex:pageBlock>
10 </apex:page>
```

Output in Visualforce Page

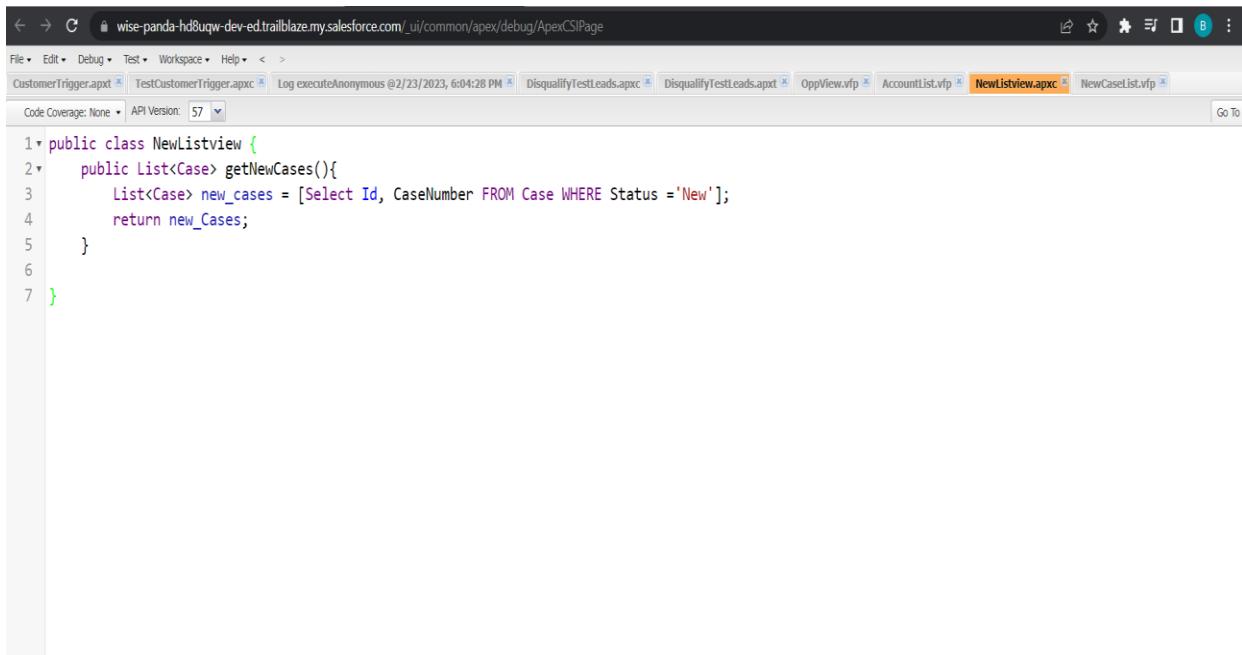


The screenshot shows the output of the Visualforce page in a browser. The page title is "wise-panda-hd8uqw-dev-ed.c.trailblaze.vf.force.com/apex/AccountList". The top navigation bar includes links for Home, Chatter, Contacts, Accounts (which is the active tab), Ideas, Reports, Dashboards, Customers, and a "Create New..." button. Below the navigation is a sidebar with "Recent Items" (empty) and a "Recycle Bin" section. The main content area is titled "Accounts" and lists the following account names:

- Burlington Textiles Corp of America
- Dickenson dlc
- Edge Communications
- Exeter Logistics and Transport
- GlobePoint
- Grand Hotels & Resorts Ltd
- Pyramid Construction Inc.
- Sainte Account for Entitlements
- Scotiabank
- United Oil & Gas Corp.
- United Oil & Gas Singapore
- United Oil & Gas UK
- University of Arizona

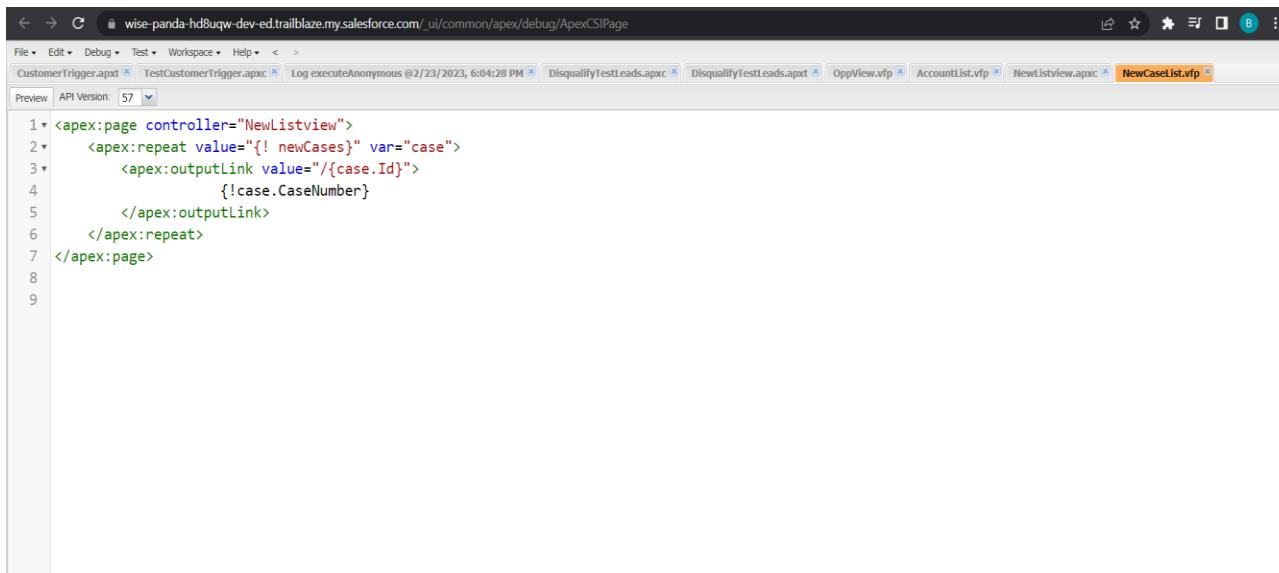
At the bottom of the page, there is a copyright notice: "Copyright © 2000-2023 salesforce.com, inc. All rights reserved. | Privacy Statement | Security Statement | Terms of Use | 508 Compliance | Go to Salesforce mobile app".

Exercise 18



The screenshot shows the Salesforce Apex code editor interface. The title bar indicates the URL is `wise-panda-hd8ugw-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage`. The tabs at the top show various files like CustomerTrigger.apxt, TestCustomerTrigger.apxc, Log executeAnonymous @2/23/2023, 6:04:28 PM, DisqualifyTestLeads.apxc, OppView.vfp, AccountList.vfp, NewListview.apxc (which is the active tab), and NewCaseList.vfp. The code editor displays the following Apex class:

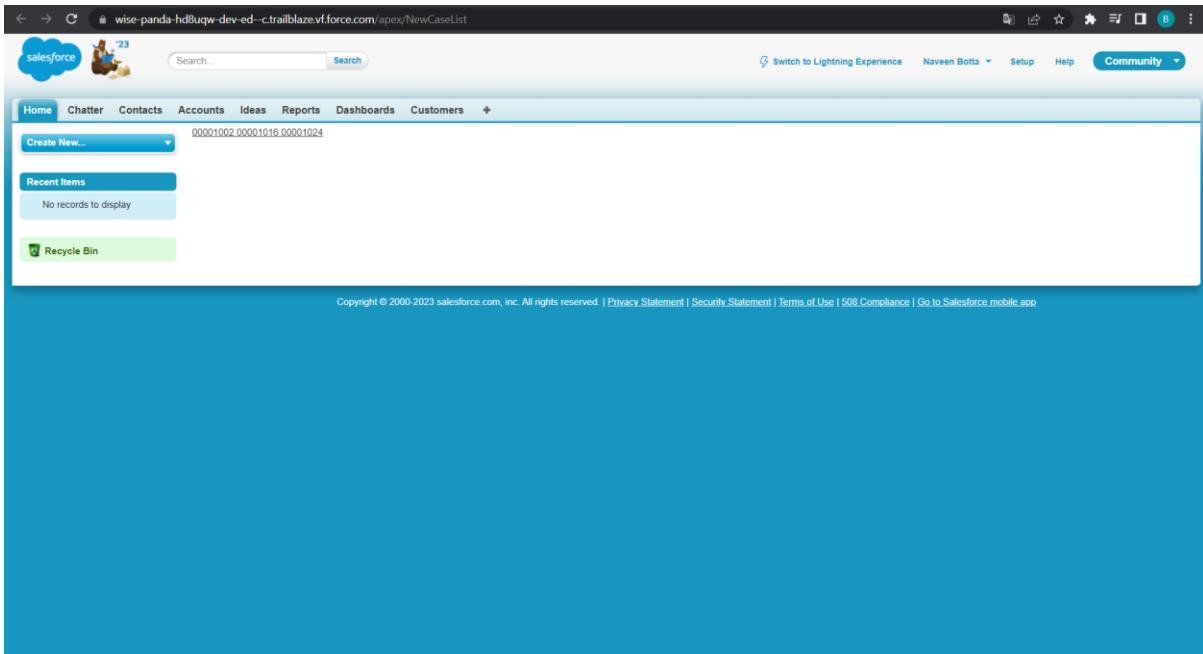
```
1 public class NewListview {
2     public List<Case> getNewCases(){
3         List<Case> new_cases = [Select Id, CaseNumber FROM Case WHERE Status ='New'];
4         return new_cases;
5     }
6 }
7 }
```



The screenshot shows the Salesforce Visualforce code editor interface. The title bar indicates the URL is `wise-panda-hd8ugw-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage`. The tabs at the top show various files like CustomerTrigger.apxt, TestCustomerTrigger.apxc, Log executeAnonymous @2/23/2023, 6:04:28 PM, DisqualifyTestLeads.apxc, OppView.vfp, AccountList.vfp, NewListview.apxc, and NewCaseList.vfp (which is the active tab). The code editor displays the following Visualforce page:

```
1 <apex:page controller="NewListview">
2     <apex:repeat value="{! newCases}" var="case">
3         <apex:outputLink value="/{case.Id}">
4             {!case.CaseNumber}
5         </apex:outputLink>
6     </apex:repeat>
7 </apex:page>
8
9
```

Output



References

1. Trailhead: <https://trailhead.salesforce.com/>
2. Developer Guide: [Salesforce Developers](#)

References

1. [Manage sales - Salesforce IN](#)
2. [Salesforce - ADX201 Administrative Essentials for New Admins in Lightning Experience \(SFADX201\) \(qa.com\)](#)
3. [Understand the Salesforce Architecture Unit | Salesforce Trailhead](#)