



Capstone Project Report On “Travel application”

Submitted by:

Name: Akshay Pusuluri

Batch WIP-SF-13

LMS Id: MGSA_510

Date: 28/02/2023

GitHub link:

Table of contents

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

Abstract

The Travel Approval App for Salesforce is a comprehensive solution designed to streamline the travel approval process for sales teams. With this app, sales representatives can quickly and easily submit travel requests, including details on travel dates, destinations, and budgets, all from within the Salesforce platform. The app automates the approval process, ensuring that requests are routed to the appropriate approvers based on pre-defined rules and workflows. The app also provides realtime visibility into travel requests and approvals, helping sales managers make informed decisions about their teams' travel plans. Overall, the Travel Approval App for Salesforce enables sales teams to save time, increase efficiency, and improve collaboration, all while ensuring that travel is managed in a compliant and cost-effective manner.

Introduction

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



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

teams' travel plans, enabling them to make informed decisions about resource allocation and budgeting.

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

Flow of the Project

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

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

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

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

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

5. Real-Time Reporting: The app provides real-time visibility into travel requests and approvals, allowing sales managers to monitor travel spend and make informed decisions about their teams' travel plans. Overall, the Travel Approval App for Salesforce automates and streamlines the travel request and approval process, providing a seamless experience for sales

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

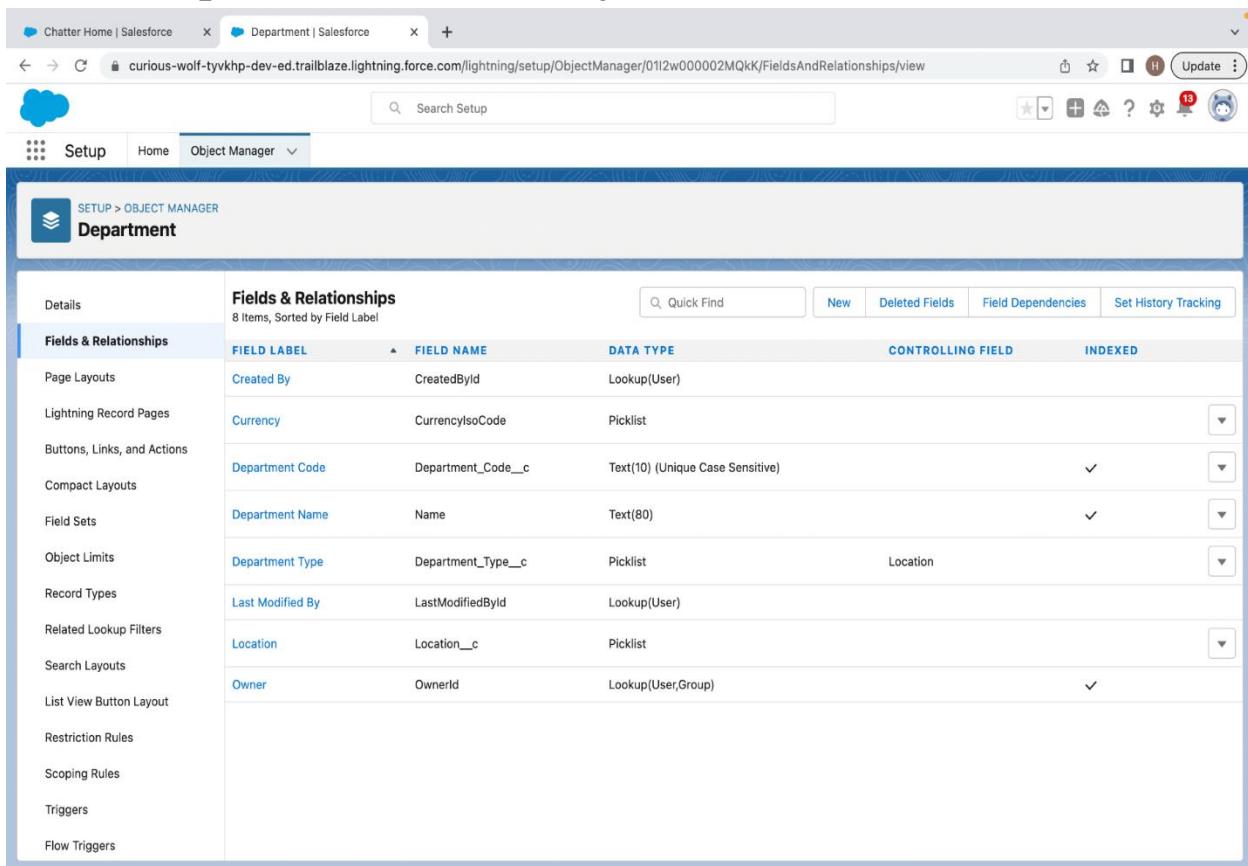
Exercise – 1

Step – 1

The screenshot shows the Salesforce Chatter interface for the 'Travel App'. The left sidebar includes links for 'What I Follow', 'To Me', 'Bookmarked', 'Company Highlights' (which is selected), 'My Drafts', 'STREAMS' (+), and 'RECENT GROUPS' (+). The main feed displays a post from user TA-00001 - Akshay Pusuluri, dated 21 February 2023 at 9:36 pm. The post asks, '@Eric Executive Which department should I associate this travel request with?'. A comment from Eric Executive (6 days ago) says, 'Technology is the correct department.' Below the comment is a placeholder for a reply: 'Write a comment...'. On the right, there's an 'Einstein Recommendations' sidebar featuring three users: 'Eric Executive Your manager' (Follow, Skip), 'Integration User Joined in the last week' (Follow, Skip), and 'Security User Joined in the last week' (Follow, Skip). The top navigation bar has links for Chatter, Reports, Dashboards, Departments, and Travel Approvals.

Step – 2&3

Create Department Custom Object with all custom fields



The screenshot shows the Salesforce Object Manager interface for the 'Department' object. The 'Fields & Relationships' tab is selected. The table displays the following fields:

| | FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
|-----------------------------|------------------|--------------------|----------------------------------|-------------------|---------|
| Page Layouts | Created By | CreatedBy | Lookup(User) | | |
| Lightning Record Pages | Currency | CurrencyIsoCode | Picklist | | |
| Buttons, Links, and Actions | Department Code | Department_Code__c | Text(10) (Unique Case Sensitive) | ✓ | |
| Compact Layouts | Department Name | Name | Text(80) | ✓ | |
| Field Sets | Department Type | Department_Type__c | Picklist | Location | |
| Object Limits | Last Modified By | LastModifiedBy | Lookup(User) | | |
| Record Types | Location | Location__c | Picklist | | |
| Related Lookup Filters | Owner | OwnerId | Lookup(User,Group) | ✓ | |
| Search Layouts | | | | | |
| List View Button Layout | | | | | |
| Restriction Rules | | | | | |
| Scoping Rules | | | | | |
| Triggers | | | | | |
| Flow Triggers | | | | | |

Step – 4:Created the required Travel Approval Object with all the required custom fileds

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

Setup > OBJECT MANAGER

Travel Approval

Details

Fields & Relationships

Description

API Name: Travel_Approval_c

Custom: ✓

Singular Label: Travel Approval

Plural Label: Travel Approvals

Enable Reports: ✓

Track Activities: ✓

Track Field History: ✓

Deployment Status: Deployed

Help Settings: Standard salesforce.com Help Window

Edit | **Delete**

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

After testing the App:

The screenshot shows a Salesforce page for a Travel Approval record named 'TA-00001'. The page has a header with the 'Travel App' logo, a search bar, and various navigation links like Chatter, Reports, Dashboards, Departments, and Travel Approvals. The main content area is divided into two sections: 'Details' and 'Activity'.

Details Section:

- Travel Approval Name: TA-00001
- Status: Draft
- Total Expenses: ₹1,320.00
- Status Indicator:
- Trip Info:**
 - Purpose of Trip: Attend Dreamforce
 - Trip Start Date: 22/02/2023
 - Trip End Date: 02/03/2023
 - Created By: Akshay Pusuluri, 21/02/2023, 8:49 pm
 - Out-of-State:
 - Destination State: CA
- Last Modified By: Akshay Pusuluri, 21/02/2023, 8:49 pm

Activity Section:

- Activity tab is selected.
- Chatter tab is available.
- Filters: All time • All activities • All types
- Buttons: Refresh, Expand All, View All.
- No activities to show. Get started by sending an email, scheduling a task, and more.
- No past activity. Past meetings and tasks marked as done show up here.

Step – 5

Imported Departments.CSV file using Data Import Wizard

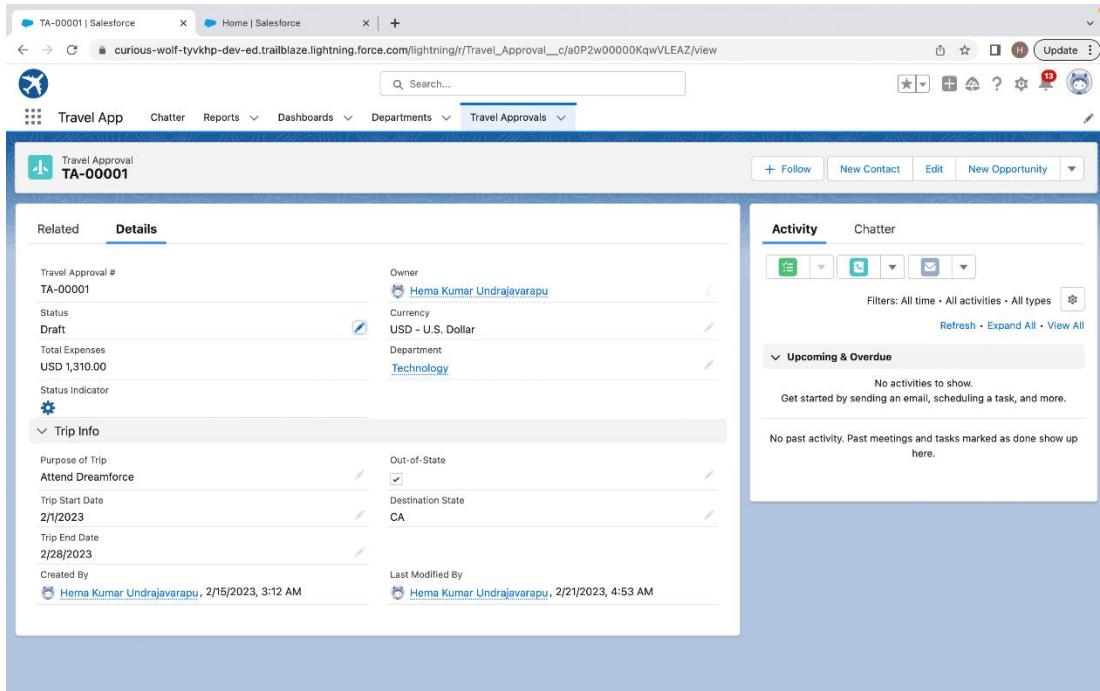
The screenshot shows the Salesforce 'Travel App' interface. The top navigation bar includes 'Chatter', 'Reports', 'Dashboards', 'Departments' (selected), and 'Travel Approvals'. A search bar is at the top right. On the left, there's a sidebar with a 'Departments' icon and a dropdown menu set to 'All'. The main content area displays a list of 16 departments, each with a checkbox and a number (1-15) to its left. The list includes: Audit Services, Contract Management, Disability Determination Bureau, Division of Aging, Division of Disability and Rehabilitative Services, Division of Family Resources, Division of Finance, Division of Mental Health and Addiction, Human Resources, Legislative Services, Office of Communications and Media, Office of Early Childhood and Out-of-School Learning, Office of General Counsel, Office of Medicaid Policy and Planning, and Quality and Compliance Office. To the right of the list is a sidebar with user information for 'Akshay Pusuluri' (wise-hawk-uw6bf0-dev-ed.trailblaze.my.sale...) and settings for 'DISPLAY DENSITY' (set to 'Comfy'), 'OPTIONS', and links to 'Switch to Salesforce Classic' and 'Add Username'. A small notification icon in the top right corner indicates 13 unread items.

| Rank | Department Name |
|------|--|
| 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 |

Exercise – 2

Step – 1

Created Travel Approval record

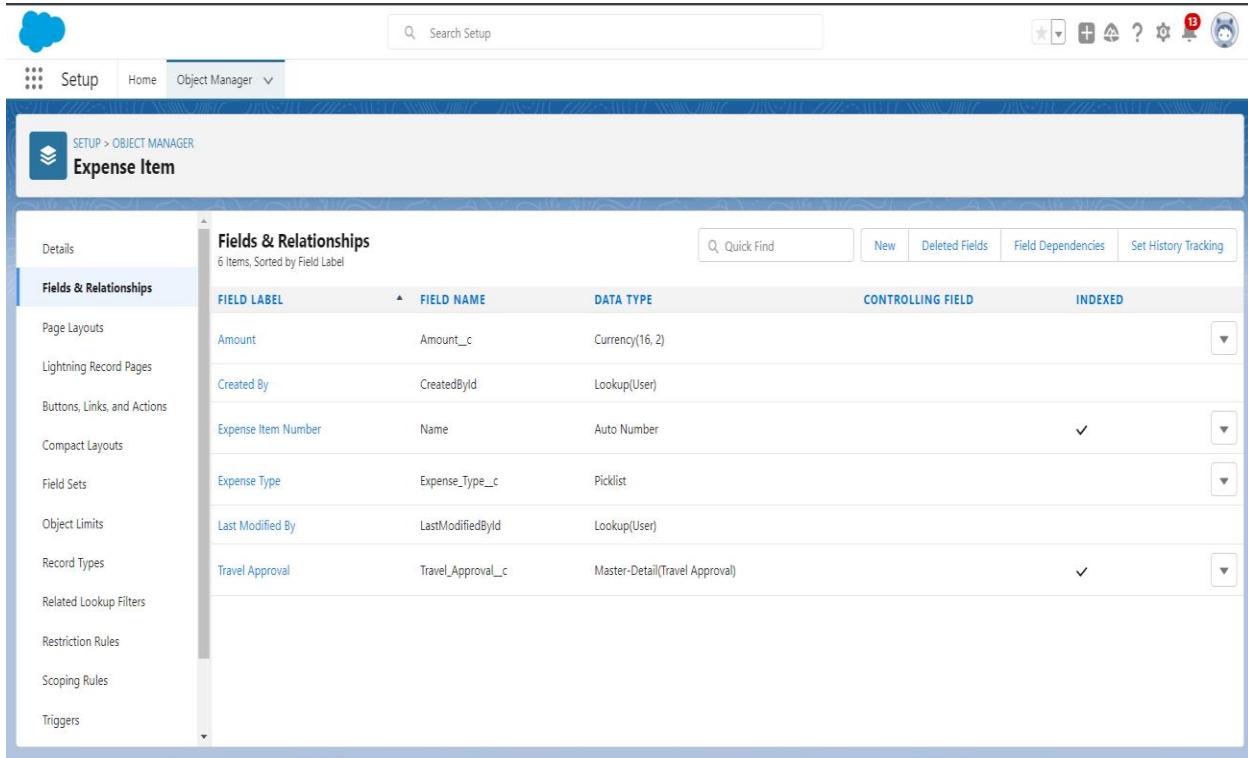


The screenshot shows a Salesforce Lightning interface for a Travel Approval record. The record ID is TA-00001. The page has a header with tabs for Travel App, Chatter, Reports, Dashboards, Departments, and Travel Approvals. The main content area is divided into two sections: 'Details' and 'Activity'. The 'Details' section contains fields for Travel Approval # (TA-00001), Status (Draft), Total Expenses (USD 1,310.00), and a Status Indicator (gear icon). It also includes a 'Trip Info' section with fields for Purpose of Trip (Attend Dreamforce), Trip Start Date (2/1/2023), Trip End Date (2/28/2023), and Out-of-State (checkbox checked). The 'Activity' section shows a summary with no activities to show and a note to start by sending an email or scheduling a task.

| Field | Value |
|-------------------|--|
| Travel Approval # | TA-00001 |
| Status | Draft |
| Total Expenses | USD 1,310.00 |
| Status Indicator | (gear icon) |
| Purpose of Trip | Attend Dreamforce |
| Trip Start Date | 2/1/2023 |
| Trip End Date | 2/28/2023 |
| Out-of-State | (checkbox checked) |
| Destination State | CA |
| Created By | Hema Kumar Undrajavarapu, 2/15/2023, 3:12 AM |
| Last Modified By | Hema Kumar Undrajavarapu, 2/21/2023, 4:53 AM |

Step – 2&3

Created Expense Item Custom Object with custom fields



The screenshot shows the Salesforce Object Manager interface for the 'Expense Item' custom object. The left sidebar lists various configuration tabs: Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Restriction Rules, Scoping Rules, and Triggers. The main content area displays the 'Fields & Relationships' table with the following data:

| FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
|---------------------|-------------------|--------------------------------|-------------------|---------|
| Amount | Amount_c | Currency(16, 2) | | |
| Created By | CreatedById | Lookup(User) | | |
| Expense Item Number | Name | Auto Number | | ✓ |
| Expense Type | Expense_Type_c | Picklist | | |
| Last Modified By | LastModifiedById | Lookup(User) | | |
| Travel Approval | Travel_Approval_c | Master-Detail(Travel Approval) | | ✓ |

Step – 4

Created Records

The screenshot shows a Salesforce interface for a 'Travel Approval' record. The record ID is TA-00001. The 'Details' tab is selected. The record contains the following fields:

| Field | Value |
|----------------------|------------------|
| Travel Approval Name | TA-00001 |
| Status | Draft |
| Total Expenses | ₹1,320.00 |
| Status Indicator | (blue gear icon) |

Owner: Akshay Pusuluri
Department: Technology

Activity Chatter

No activities to show. Get started by sending an email, scheduling a task, and more.

No past activity. Past meetings and tasks marked as done show up here.

Step-5&6

Created User Eric and assigned him as manager to System Administrator

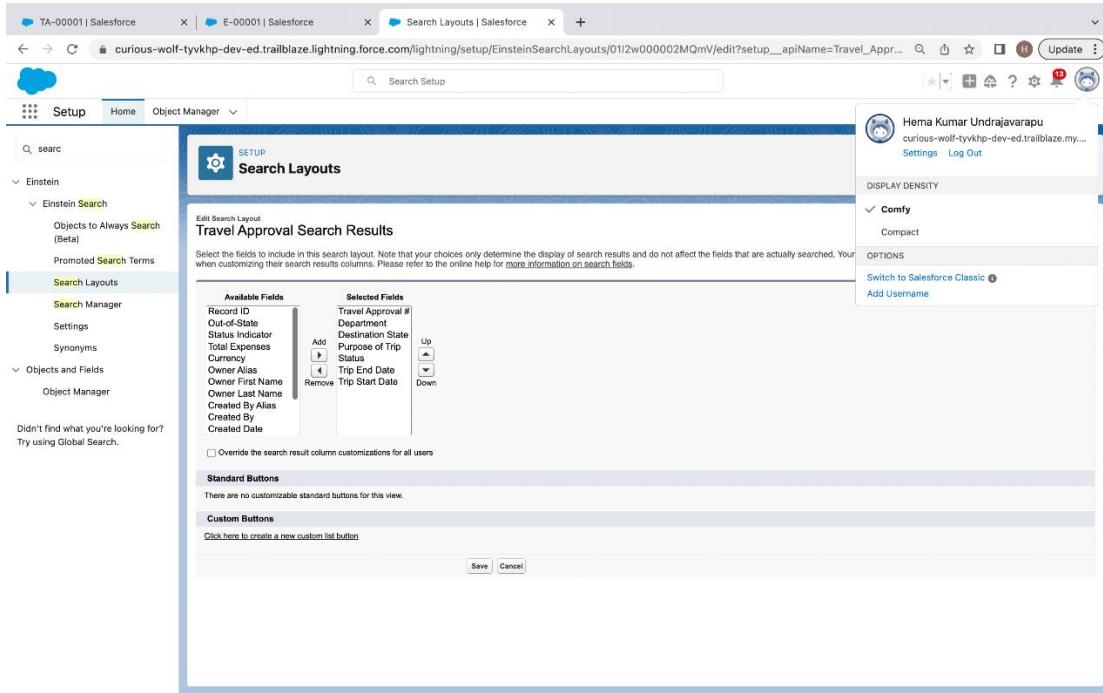
The screenshot shows the Salesforce Setup interface. On the left, the navigation sidebar is open, showing various categories like 'Setup', 'Home', 'Object Manager', and 'Users'. Under 'Users', 'User Management Settings' and 'Users' are selected. The main content area displays the 'Users' setup page for a user named 'Eric Executive'. The 'User Detail' section shows the following information:

| Field | Value | Role |
|---------------------------------|--|-----------------------------------|
| Name | Eric Executive | CEO |
| Alias | eexec | User License |
| Email | pusuluri.akshay2@wipro.com | Profile |
| Username | eric.pusuluri@wipro.com | Active |
| Nickname | eric.exec | Marketing User |
| Title | | Offline User |
| Company | | Knowledge User |
| Department | | Flow User |
| Division | | Service Cloud User |
| Address | | Site.com Contributor User |
| Time Zone | (GMT+05:30) India Standard Time (Asia/Kolkata) | Site.com Publisher User |
| Locale | English (India) | WDC User |
| Language | English | Mobile Push Registrations |
| Delegated Approver | | Data.com User Type |
| Manager | | Accessibility Mode (Classic Only) |
| Receive Approval Request Emails | Only if I am an approver | Debug Mode |
| Federation ID | | High-Contrast Palette on Charts |

At the top right of the main content area, there is a link 'User Profile Help for this Page' with a question mark icon.

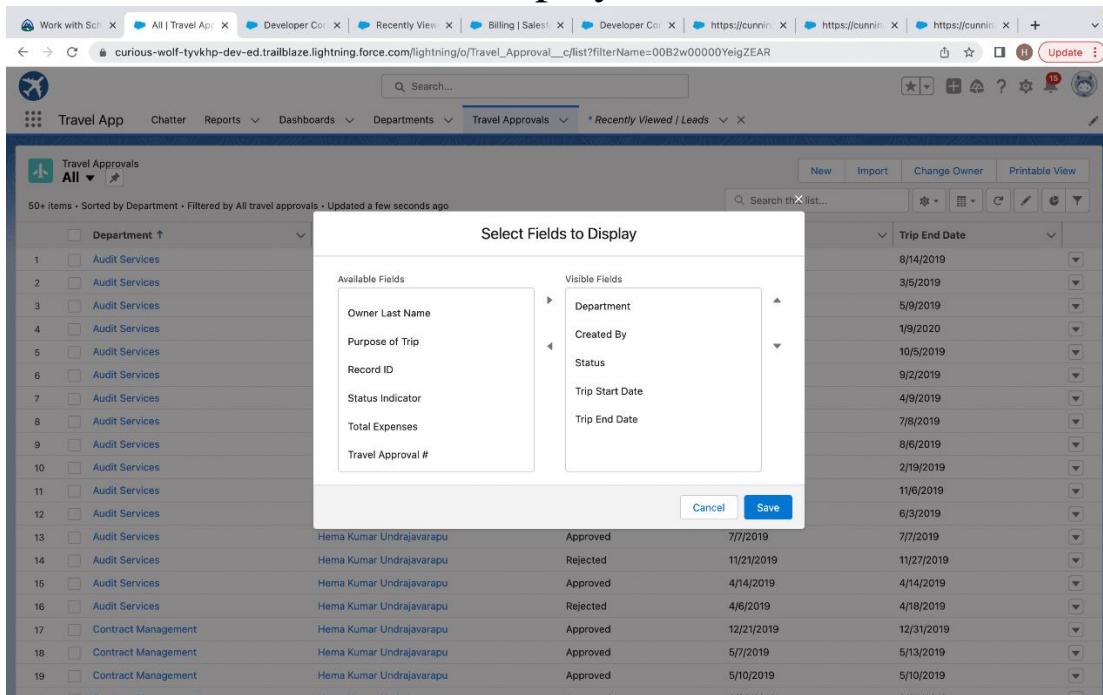
Step – 7

Customized the Travel Approval Default Search



Step – 8

Selected all the fields to display



Step – 9

Created Travel Approval Custom List View Open Out Of State Travel Request

The screenshot shows a Salesforce Lightning interface with the following details:

- Tab Bar:** TA-00001 | Salesforce, Open Out of State Travel Requests, Search Layouts | Salesforce.
- Header:** Travel App, Chatter, Reports, Dashboards, Departments, Travel Approvals.
- Page Title:** Travel Approvals - Open Out of State Travel Requests.
- Table:** Shows one item: TA-00001, Department: Technology, Created By: Hema Kumar Undrajavarapu, Destination S...: CA, Trip Start Date: 2/1/2023, Trip End Date: 2/28/2023.
- Filters Sidebar:** Displays two filters:
 - Filter by Owner: My travel approvals
 - Matching all of these filters:
 - Out-of-State equals True
 - Status not equal to Approved, Rejected
- Buttons:** New, Import, Change Owner, Printable View.

Step – 10

Customized Travel Approval Page Layout

The screenshot shows the Salesforce Object Manager interface for the 'Travel Approval' object. The left sidebar has 'Page Layouts' selected. The main area displays a table titled 'Page Layouts' with one item: 'Travel Approval Layout'. The table includes columns for 'PAGE LAYOUT NAME', 'CREATED BY', and 'MODIFIED BY'. The 'CREATED BY' and 'MODIFIED BY' columns show 'Akshay Pusuluri' with the timestamp '21/02/2023, 8:34 pm' and '21/02/2023, 10:05 pm' respectively. On the right side, there is a user profile for 'Akshay Pusuluri' and a sidebar with sections for 'USERNAMES', 'DISPLAY DENSITY', and 'OPTIONS'.

The screenshot shows the Salesforce Page Layout editor for the 'Travel Approval Layout'. The left sidebar has 'Page Layouts' selected. The main area shows the layout configuration for the 'Travel Approval' object. It includes sections for 'Fields' (with a table showing fields like 'Section', 'Destination State', 'Purpose of Trip', etc.), 'Highlights Panel' (with a note about customizing the highlights panel), 'Quick Actions in the Salesforce Classic Publisher' (with a note about overriding the global publisher layout), and 'Salesforce Mobile and Lightning Experience Actions' (with a note about overriding predefined actions). The bottom section is labeled 'Travel Approval Detail' with standard buttons like 'Edit', 'Delete', 'Clone', etc. A sidebar on the right shows the user profile for 'Akshay Pusuluri' and a sidebar with sections for 'USERNAMES', 'DISPLAY DENSITY', and 'OPTIONS'.

Step – 12

Customized the Expense Item Related List User Travel Approval Page Layout

The screenshot shows a Salesforce page for a travel approval request. The top navigation bar includes links for Travel App, Chatter, Reports, Dashboards, Departments, and Travel Approvals. The main header displays "Travel Approval TA-00001". The page has tabs for "Related" and "Details", with "Related" selected. Below the tabs are three sections:

- Notes & Attachments (0)**: Includes a file upload section with "Upload Files" and "Or drop files" buttons.
- Expense Items (2)**: A table showing two items:

| Expense Item Number | Expense Type | Amount |
|---------------------|--------------|---------|
| E-00001 | Airfare | ₹450.00 |
| E-00002 | Hotel | ₹870.00 |

Buttons for "New" and "View All" are present.
- Approval History (2)**: A table showing two steps:

| Step Name | Date | Status | Assigned To |
|----------------------------|----------------------|-----------|----------------|
| Step 1 | 21/02/2023, 10:12 pm | Pending | Eric Executive |
| Approval Request Submitted | 21/02/2023, 10:12 pm | Submitted | Nehau Daulati |

Buttons for "Approve" and "Reject" are shown.

To the right of these sections is a sidebar titled "Activity" which includes a "Chatter" tab and a "Upcoming & Overdue" section indicating no activities.

Step – 13

Enabled Feed Tracking

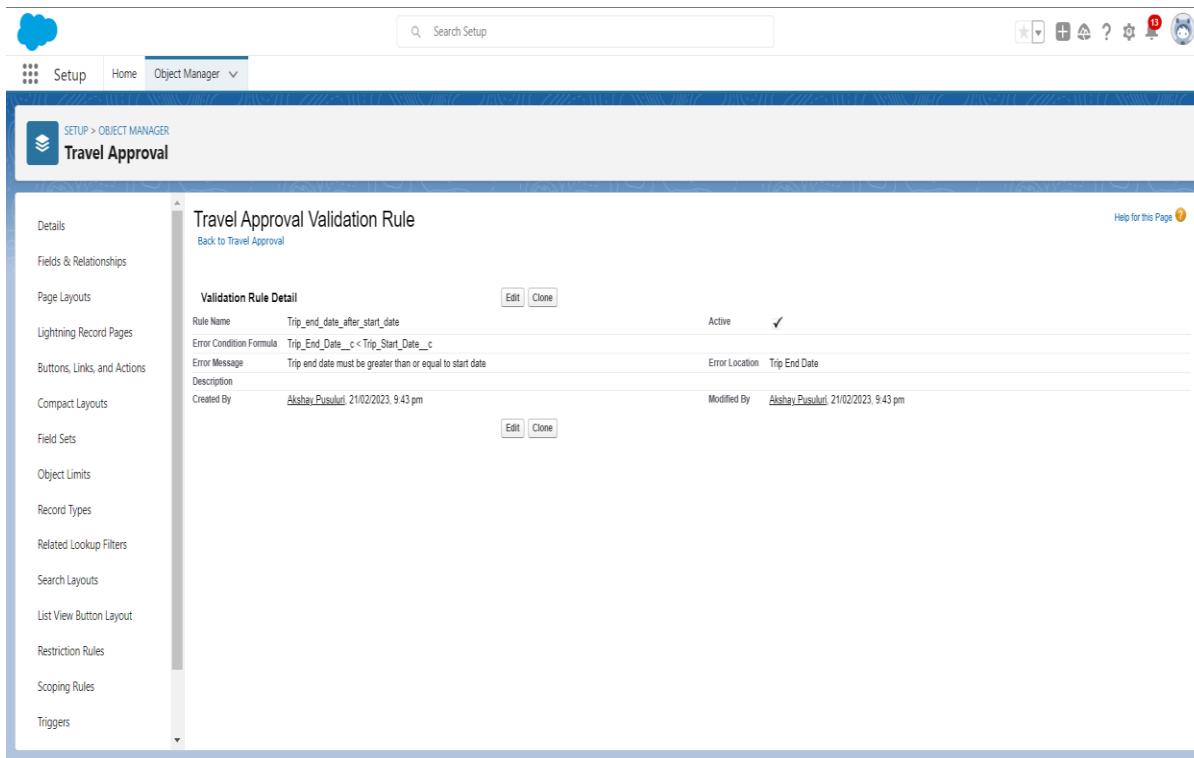
The screenshot shows the Salesforce Setup interface for Feed Tracking. The left sidebar navigation includes categories like Feature Settings, Chatter, Feed Item, Feed Tracking (which is selected), Triggers, Feed Comment Triggers, Feeditem Triggers, Service, and Feed Filters. A search bar at the top is set to 'feed'. The main content area is titled 'Feed Tracking' and contains a sub-section titled 'Fields in travel approvals'. It lists various fields such as Department, Out-of-State, Purpose of Trip, Travel Approval Name, Trip Start Date, Destination State, Owner, Status, and Trip End Date. Each field has a checkbox next to it. A note says 'You can select up to 9 fields.' Below this, another note says 'You can also display feed activity for related objects.' with a checkbox for 'All Related Objects'. At the bottom of the section are 'Save' and 'Cancel' buttons, a 'Enable Feed Tracking' checkbox, and a 'Restore Defaults' button.

MODULE -2

Exercise – 1

Step – 1

Created Trip End Date After Start Date Validation Rule



The screenshot shows the Salesforce Object Manager interface for the 'Travel Approval' object. The left sidebar lists various configuration options like 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 main content area displays the 'Validation Rule Detail' for a rule named 'Trip_end_date_after_start_date'. The rule's formula is 'Trip_End_Date__c < Trip_Start_Date__c', and its message is 'Trip end date must be greater than or equal to start date'. The rule is active and located on the 'Trip End Date' field. It was created by 'Akshay Pusuluri' on 21/02/2023, 9:43 pm, and modified by the same user on the same date and time.

| Validation Rule Detail | |
|-------------------------|---|
| Rule Name | Trip_end_date_after_start_date |
| Error Condition Formula | Trip_End_Date__c < Trip_Start_Date__c |
| Error Message | Trip end date must be greater than or equal to start date |
| Description | |
| Created By | Akshay Pusuluri, 21/02/2023, 9:43 pm |
| Modified By | Akshay Pusuluri, 21/02/2023, 9:43 pm |

Step – 2

Created a Roll Up Summary Field Total Expenses

The screenshot shows the Salesforce Setup interface under the Object Manager. A custom field named 'Total Expenses' has been created for the 'Travel Approval' object. The field is of type 'Roll-Up Summary' and aggregates the 'Expense Item Amount' field from the 'Expense Item' object. The field is labeled 'Total Expenses' and has the API name 'Total_Expenses__c'. It includes a help text and a data sensitivity level.

Step – 3

Created a Formula Field Status Images

The screenshot shows the Salesforce Setup interface under the Static Resources section. A static resource named 'StatusImages' has been created. It has a namespace prefix and is set to be application/x-zip-compressed. The file size is 39,130 bytes. The resource was created by Akshay Pusuluri on 21/02/2023, 9:48 pm.

Step – 4

Created a Formula Field Status Indicator

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

- Setup > Object Manager**: The current page is "Travel Approval".
- Custom Field Definition Detail**: The field is named "Status Indicator" (Status_Indicator) with API name "Status_Indicator_c". It is a "Formula" type field.
- Formula Options**: The formula is defined as:

```
IF( ISNULL(Status__c ), IMAGE("/resource/StatusImages/draft.png"), "In-Process", 20, 20),  
IF( ISNULL(Status__c ), IMAGE("/resource/StatusImages/thumbs-down.png"), "Rejected", 20, 20),  
IMAGE("/resource/StatusImages/thumbs-up.png", "Accepted", 20, 20)
```
- Field Information**: Other details include Field Label "Status Indicator", Field Name "Status_Indicator", API Name "Status_Indicator_c", and a note that it is used on the "Travel Approval" object.
- Object Name**: Travel Approval
- Created By**: Akshay Pusuluri
- Modified By**: Akshay Pusuluri
- Display Density**: Comfy
- Options**: Compact
- Usernames**: A sidebar lists several user profiles, including Akshay Pusuluri's profile.
- Log Out**: Option to log out of the Salesforce account.

Step – 5

Created a Record – Triggered Flow Decision Block

The screenshot shows the Salesforce Flow Builder interface. At the top, there are three tabs: "Recently Viewed | Travel Approval", "Travel Approval | Salesforce", and "Out of State Travel Flag - V1". The "Out of State Travel Flag - V1" tab is active. The title bar says "Flow Builder" and "Out of State Travel Flag - V1". The top right has buttons for "Update", "Run", "Debug", "View Tests", "Deactivate", "Save As", and "Save". Below the title bar, there's a toolbar with icons for back, forward, search, and other functions.

The main area is divided into sections:

- Toolbox:** On the left, under "Elements Manager", there are two categories:
 - Logic (5):** Assignment, Decision, Loop, Collection Sort, Collection Filter.
 - Data (2):** Update Records, Get Records.
- Start Record-Triggered Flow:** A card with a play button icon. Details: Object: Travel Approval, Trigger: A record is created or updated, Optimize for: Fast Field Updates. Buttons: "Edit", "Open Flow Trigger Explorer for Travel ...".
- Edit Decision:** A modal window titled "Edit Decision" for the decision block "Is Travel Out of State? (Is_Travel_Out_of_State)".
 - Outcomes:** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.
 - OUTCOME ORDER:** Yes Out of State, In State.
 - OUTCOME DETAILS:** *Label: Yes Out of State, *Outcome API Name: Yes_Out_of_State.
 - Default Outcome:** Condition Requirements to Execute Outcome: All Conditions Are Met (AND). Resource: \$Record > Destination State, Operator: Does Not Equal, Value: TX.
 - When to Execute Outcome:**
 - If the condition requirements are met.
 - Only if the record that triggered the flow to run is updated to meet the condition requirements.
- Flow Diagram:** Shows the flow from the Start Record-Triggered Flow to the Decision block, then branching into "Yes Out of State" and "In State" paths. The "Yes Out of State" path leads to an "Update Records" step, which then leads to an "Update Rewards" step. The "In State" path leads directly to the "Update Rewards" step.

Updated Travel Approval Record (Yes out of State)

The screenshot shows the Salesforce Flow Builder interface with two windows open:

- Edit Update Records**: This window is titled "Edit Update Records" and describes updating Salesforce records using values from the flow. It shows the selected element is "Update Travel Approval Record (Update_Travel_Approval_Record)".
 - How to Find Records to Update and Set Their Values**: Options include using the travel approval record that triggered the flow or specifying conditions to identify records and set fields individually.
 - A note states: "Because this flow runs before a record is saved, you can only update the record that triggered the flow to run. To update other records, configure the trigger to run the flow after the record is saved."
- Flow Canvas**: This window shows a flow named "Out of State Travel Flag - V1".
 - Start**: Record-Triggered Flow (Object: Travel Approval, Trigger: A record is created or updated, Optimized for: Fast Field Updates).
 - Decision**: Is Travel Out of State? (Decision condition: Out_of_State__c = True).

Updated Travel Approval Record (In State)

The screenshot shows the Salesforce Flow Builder interface with two open panels:

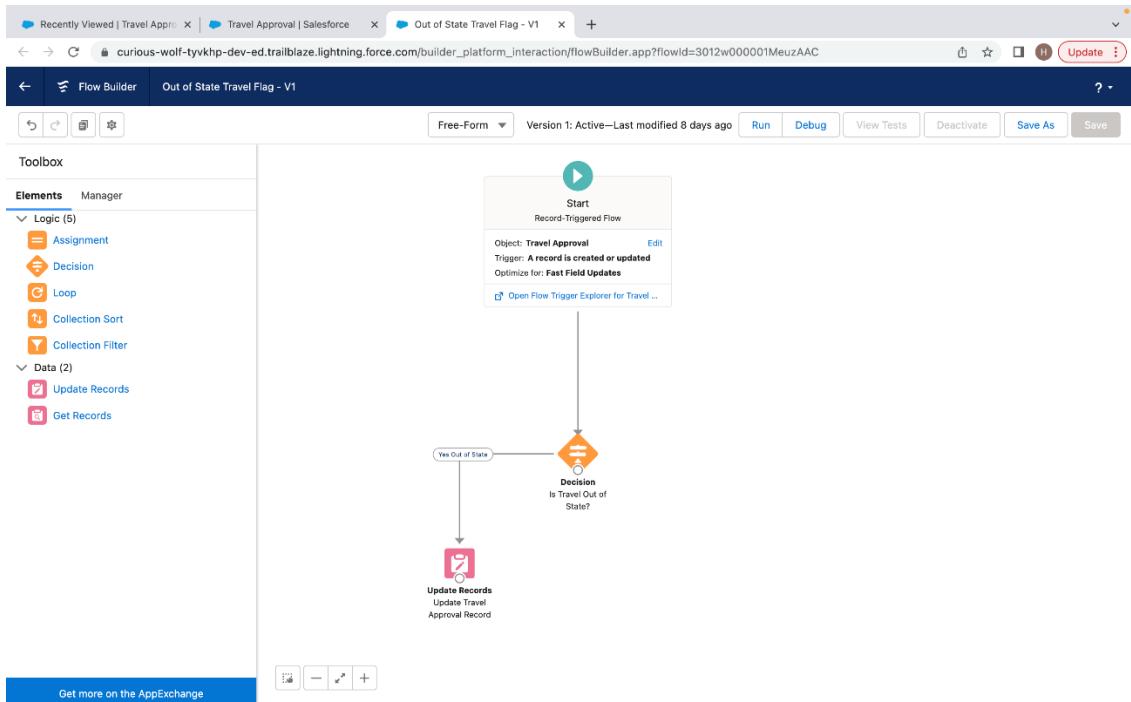
Edit Decision (Top Panel):

- Title:** Is Travel Out of State? (is_Travel_Out_of_State)
- Outcomes:** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.
- OUTCOME ORDER:** Yes Out of State, In State
- OUTCOME DETAILS:**
 - Label:** In State
 - Outcome API Name:** In_State
 - Condition Requirements to Execute Outcome:** All Conditions Are Met (AND)
 - Resource: \$Record > Destination State
 - Operator: Equals
 - Value: TX
- Default Outcome:** All Conditions Are Met (AND)
- When to Execute Outcome:**
 - If the condition requirements are met
 - Only if the record that triggered the flow is updated to meet the condition requirements

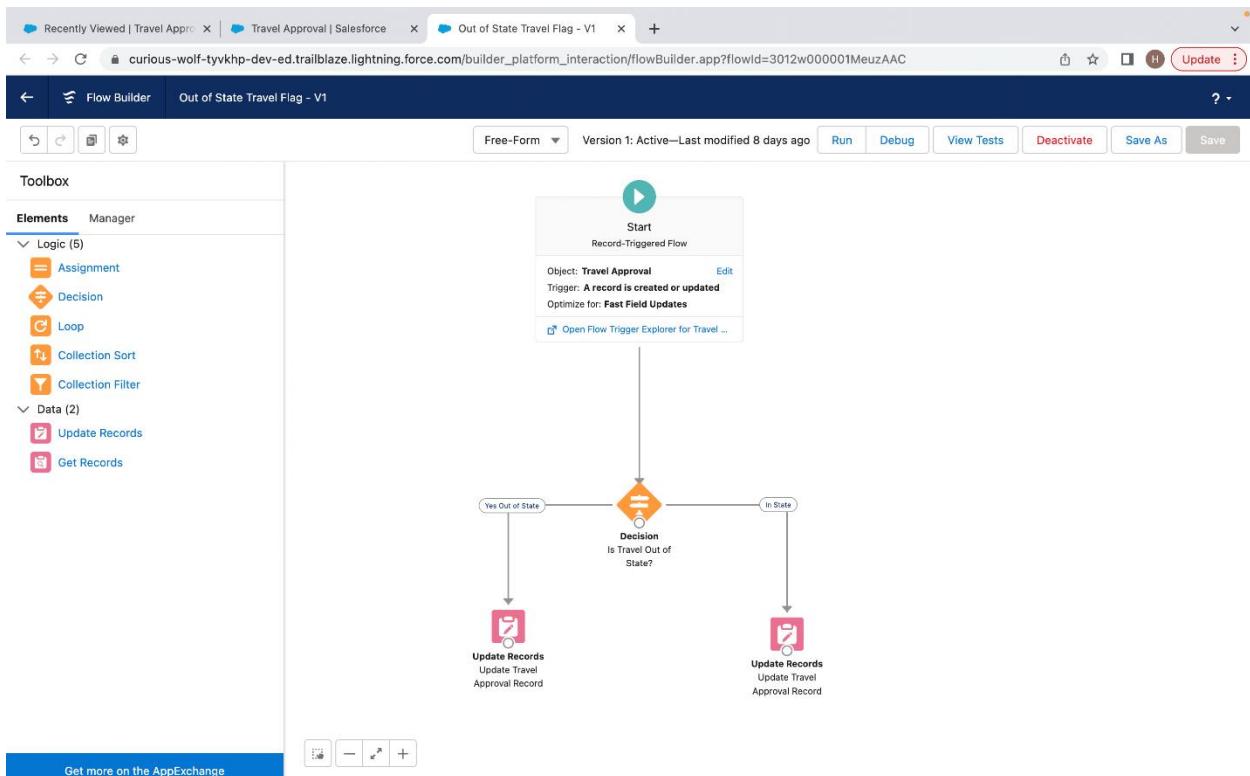
Update Travel Approval Record (Bottom Panel):

- Update Records:** Update Travel Approval Record
- How to Find Records to Update and Set Their Values:**
 - Use the travel approval record that triggered the flow
 - Update records related to the travel approval record that triggered the flow
 - Use the IDs and all field values from a record or record collection
 - Specify conditions to identify records, and set fields individually
- Note:** Because this flow runs before a record is saved, you can only update the record that triggered the flow to run. To update other records, configure the trigger to run the flow after the record is saved.
- Set Filter Conditions:** Condition Requirements to Update Record: None—Always Update Record
- Set Field Values for the Travel Approval Record:**

| Field | Value |
|-----------------|-------|
| Out_of_State__c | True |



Out of State Travel Flag (Flow)



Step – 6

Created an Approval Process Travel Approval Request

The screenshot shows the Salesforce Setup interface for creating an Approval Process. The left sidebar shows categories like Data, Process Automation, and Approval Processes. The main area displays the 'Approval Processes' screen for 'Travel Approval: Travel Approval Request'. The 'Process Definition Detail' section includes fields for Process Name (Travel Approval Request), Unique Name (Travel_Approval_Request), Description, Entry Criteria (Travel Approval: Total Expenses GREATER THAN 0), Record Editability (Administrator ONLY), and Approval Assignment Email Template. It also shows Initial Submitters (Travel Approval Owner) and Created By (Akshay Pusuluri). The 'Initial Submission Actions' section contains a single entry: Action Type (Record Lock) and Description (Lock the record from being edited). The 'Approval Steps' section lists two steps: Step 1 (Show Actions) and Step 2 (Show Actions), both labeled 'Travel Coordinator Approval'. The 'Final Approval Actions' section is currently empty.

Final Approval Actions & Final Rejections Actions

The screenshot shows the Salesforce Setup interface for creating an Approval Process, similar to the previous one but with more sections visible. The left sidebar includes categories like Feature Settings, Marketing, Sales, Service, and Process Automation. The main area shows the 'Approval Processes' screen for 'Travel Approval: Travel Approval Request'. The 'Initial Submission Actions' section has one entry: Action Type (Record Lock) and Description (Lock the record from being edited). The 'Approval Steps' section shows two steps: Step 1 (Show Actions) and Step 2 (Show Actions), both labeled 'Travel Coordinator Approval'. The 'Final Approval Actions' section is visible but empty. The 'Final Rejection Actions' section is also visible but empty. A right-hand sidebar displays user information for Akshay Pusuluri, including log-in details and a switch to Salesforce Classic option.

Process Automation Settings

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

- Left Sidebar:** A navigation tree under "Q process".
 - Feature Settings
 - Marketing
 - Lead Processes
 - Sales
 - Sales Processes
 - Service
 - Entitlement Management
 - Entitlement Processes
 - Support Processes
 - Process Automation** (selected)
 - Approval Processes
 - Automation Home (Beta)
 - Flows
 - Migrate to Flow
 - Next Best Action
 - Paused And Failed Flow Interviews
 - Post Templates
 - Process Automation Settings** (selected)
 - Process Builder
- Top Bar:** Includes "Setup", "Home", "Object Manager", "Search Setup", and various icons.
- Page Content:** The "Process Automation Settings" page.
 - Default Workflow User:** Set to "Akshay Pusuluri".
 - Automated Process User Email Address:** An input field for specifying an organization-wide email address.
 - Email Approval Sender:** Set to "Approval Submitter".
 - Enable email approval response:** A checked checkbox.
 - Warning Message:** "By enabling the email approval response feature, you agree to allow Salesforce to process email approval responses, update approval requests for all active users in your organization, and update the approval object on behalf of your organization's users."
 - Let users pause flows:** A checked checkbox.
 - Let users resume shared flow interviews:** A checked checkbox.
 - Enable Lightning runtime for flows:** A checked checkbox.
 - Change required permission:** "Require the Manage Flow permission to view all Automation Home charts".
 - Note:** "This setting affects Lightning Experience only. On the Flows page in Setup, use an enhanced flow list view that includes standard flows. On a separate page, view paused flows and process scheduled actions. When disabled or using Salesforce Classic, the Flows page displays system flows instead of flows and removes scheduled actions. The concrete name isn't available."

Testing the Approval Process

The screenshot shows the Travel App interface with the following details:

- Top Bar:** Includes "Travel App", "Chatter", "Reports", "Dashboards", "Departments", "Travel Approvals", "Search", and various icons.
- Record View:** A "Travel Approval" record with ID "TA-00001".
 - Related:** Shows "Trip Info".
 - Purpose of Trip: Attend Dreamforce
 - Trip Start Date: 22/02/2023
 - Trip End Date: 02/03/2023
 - Created By: Akshay Pusuluri, 21/02/2023, 8:49 pm
 - Details:**
 - Owner: Akshay Pusuluri
 - Department: Technology
 - Total Expenses: ₹1,320.00
 - Status: Draft
 - Status Indicator: A blue gear icon.
 - Cut-of-State: An input field with a dropdown arrow.
 - Destination State: CA
 - Last Modified By: Akshay Pusuluri, 21/02/2023, 8:49 pm
- Activity & Chatter:** Shows a section for "Upcoming & Overdue" activities, which is currently empty.

Exercise – 2

Step – 1

Imported Travel Approval Records

The screenshot shows the Salesforce interface for the Travel Approvals list. The top navigation bar includes 'Travel App', 'Chatter', 'Reports', 'Dashboards', 'Departments', and 'Travel Approvals'. The main area displays a table titled 'Travel Approvals' with 50+ items. The columns are: Travel Approval Name, Department, Created By, Status, Trip Start Date, and Trip End Date. The data includes entries for various departments like Technology, Disability Determination Bureau, and Office of Communications and Media, with statuses ranging from Draft to Approved.

Step – 2

Created a Report on Travel Request by Department

The screenshot shows the Salesforce Reports interface with a report titled 'Report: Travel Approvals - Travel Requests by Department'. The report details total records (372) and total out-of-state trips (285). The table lists travel approvals by department, status, and trip details. A sidebar on the right shows user information for 'Akshay Pusuluri' and various report settings like 'DISPLAY DENSITY' (Compact selected), 'OPTIONS' (Switch to Salesforce Classic), and 'Add Username'.

| Department | Travel Approval: Travel Approval Name | Status | Out-of-State | Destination State | Trip Start Date | Trip End Date |
|---------------------|---------------------------------------|----------|-------------------------------------|-------------------|-----------------|---------------|
| Audit Services (21) | TA-00040 | Approved | <input checked="" type="checkbox"/> | FL | 08/07/2019 | 08/07/2019 |
| | TA-00044 | Approved | <input checked="" type="checkbox"/> | OK | 11/04/2019 | 11/07/2019 |
| | TA-00049 | Approved | <input checked="" type="checkbox"/> | OK | 07/08/2019 | 07/08/2019 |
| | TA-00024 | Approved | <input checked="" type="checkbox"/> | OK | 09/03/2019 | 09/03/2019 |
| | TA-00032 | Approved | <input checked="" type="checkbox"/> | OK | 07/08/2019 | 07/09/2019 |
| | TA-00142 | Approved | <input checked="" type="checkbox"/> | OK | 27/12/2019 | 10/01/2020 |
| | TA-00151 | Approved | <input checked="" type="checkbox"/> | FL | 21/09/2019 | 06/10/2019 |
| | TA-00154 | Approved | <input checked="" type="checkbox"/> | OK | 03/09/2019 | 03/09/2019 |
| | TA-00156 | Approved | <input checked="" type="checkbox"/> | CA | 20/03/2019 | 10/04/2019 |
| | TA-00191 | Approved | <input checked="" type="checkbox"/> | OK | 08/07/2019 | 09/07/2019 |
| | TA-00217 | Approved | <input checked="" type="checkbox"/> | FL | 07/08/2019 | 07/08/2019 |
| | TA-00243 | Approved | <input checked="" type="checkbox"/> | FL | 18/02/2019 | 20/02/2019 |
| | TA-00247 | Approved | <input checked="" type="checkbox"/> | OK | 04/11/2019 | 07/11/2019 |
| | TA-00264 | Rejected | <input checked="" type="checkbox"/> | FL | 31/05/2019 | 04/06/2019 |
| | TA-00272 | Approved | <input checked="" type="checkbox"/> | OK | 08/07/2019 | 08/07/2019 |
| | TA-00277 | Rejected | <input checked="" type="checkbox"/> | CA | 22/11/2019 | 28/11/2019 |

Step – 3

Created a Travel Request by Monthly Report

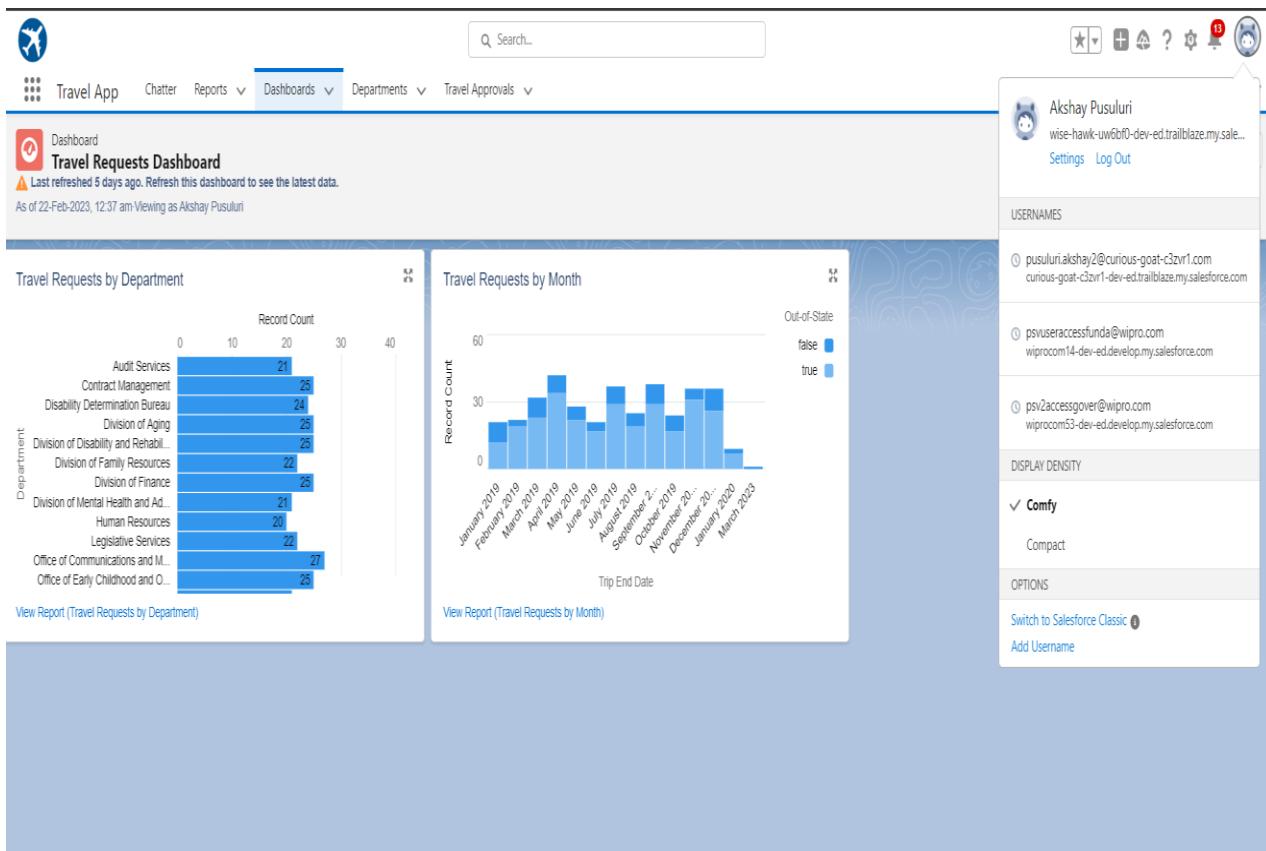
The screenshot shows a Salesforce report interface. At the top, there's a navigation bar with icons for Travel App, Chatter, Reports, Dashboards, Departments, and Travel Approvals. A search bar is also present. Below the navigation is the report title, "Report: Travel Approvals" and "Travel Requests by Month". The report displays a table of travel approvals for January 2019, with a subtotal for all approvals. The columns include Trip End Date, Out-of-State, Travel Approval Name, Department, Status, Destination State, and Trip Start Date. The data shows various travel requests from different departments like Office of General Counsel, Division of Aging, and Contract Management, with statuses ranging from Approved to Rejected.

| Trip End Date | Out-of-State | Travel Approval: Travel Approval Name | Department | Status | Destination State | Trip Start Date |
|-------------------|--------------|---------------------------------------|--|----------|-------------------|-----------------|
| January 2019 (21) | (9) | TA-00162 | Office of General Counsel | Rejected | TX | 27/01/2019 |
| | | TA-00309 | Division of Aging | Approved | TX | 22/01/2019 |
| | | TA-00139 | Legislative Services | Approved | TX | 20/01/2019 |
| | | TA-00330 | Contract Management | Approved | TX | 18/01/2019 |
| | | TA-00321 | Quality and Compliance Office | Approved | TX | 14/01/2019 |
| | | TA-00175 | Disability Determination Bureau | Approved | TX | 12/01/2019 |
| | | TA-00039 | Division of Finance | Approved | TX | 10/01/2019 |
| | | TA-00072 | Office of General Counsel | Approved | TX | 04/01/2019 |
| | | TA-00013 | Division of Family Resources | Approved | TX | 04/01/2019 |
| Subtotal | | | | | | |
| | (12) | TA-00188 | Contract Management | Approved | GA | 30/01/2019 |
| | | TA-00232 | Division of Disability and Rehabilitative Services | Approved | FL | 29/01/2019 |
| | | TA-00268 | Contract Management | Approved | FL | 29/01/2019 |
| | | TA-00117 | Division of Aging | Approved | OK | 24/01/2019 |
| | | TA-00102 | Division of Disability and Rehabilitative Services | Approved | OK | 23/01/2019 |
| | | TA-00283 | Division of Family Resources | Approved | FL | 21/01/2019 |

Row Counts: Detail Rows: Subtotals: Grand Total:

Step – 4

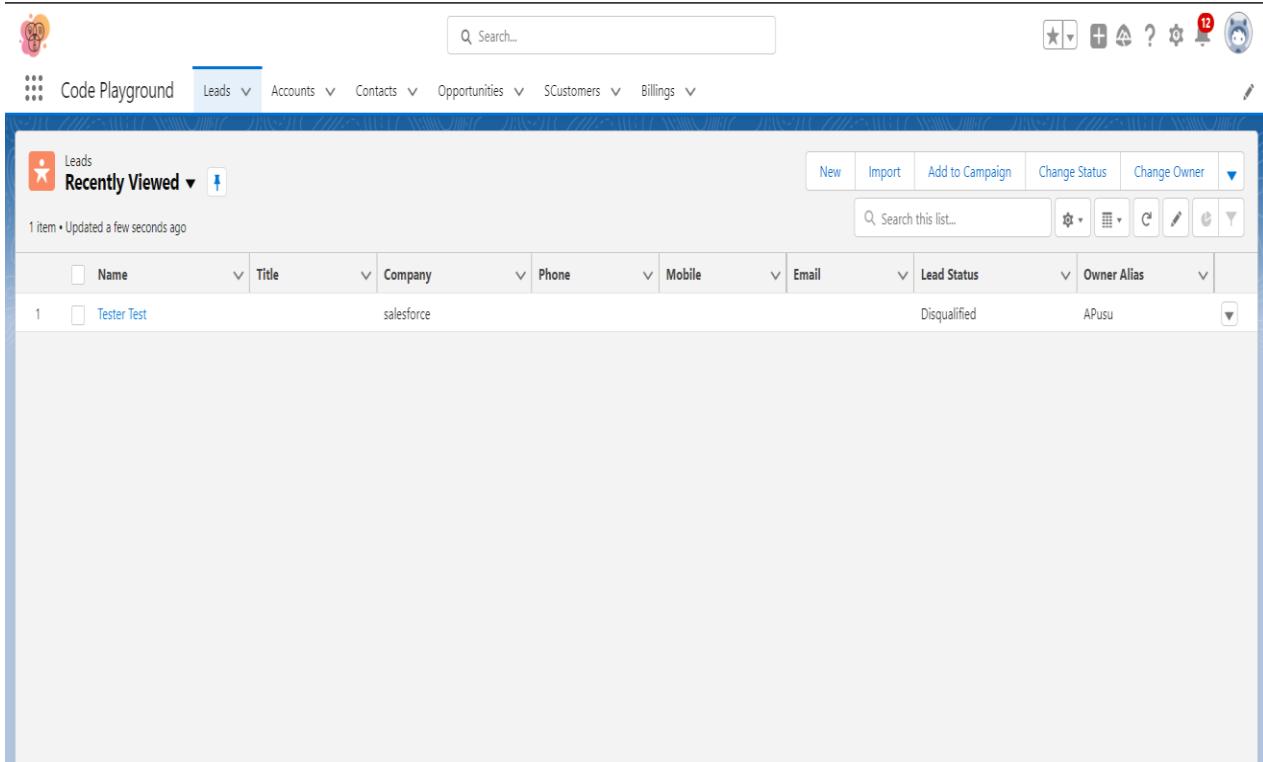
Created Dashboard for Travel Approvals



MODULE – 3

Exercise – 1

Created Code Playground App



The screenshot shows a Salesforce application interface for managing leads. At the top, there's a navigation bar with icons for user profile, search, and various modules like Leads, Accounts, Contacts, Opportunities, Customers, and Billings. Below the navigation is a toolbar with buttons for New, Import, Add to Campaign, Change Status, Change Owner, and a dropdown menu. A secondary search bar labeled "Search this list..." is also present. The main content area displays a table titled "Recently Viewed" under the "Leads" tab. The table has columns for Name, Title, Company, Phone, Mobile, Email, Lead Status, and Owner Alias. One record is listed: "Tester Test" from "salesforce" with a status of "Disqualified" and owned by "APusu". A note at the top left of the table says "1 item • Updated a few seconds ago".

| | Name | Title | Company | Phone | Mobile | Email | Lead Status | Owner Alias |
|---|-------------|-------|------------|-------|--------|-------|--------------|-------------|
| 1 | Tester Test | | salesforce | | | | Disqualified | APusu |

Created the Custom Object Customer as (SCustomer) as there is already a Standard Object Named Customer

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

- Description:** SCustomer
- API Name:** SCustomer_c
- Custom:** ✓
- Singular Label:** SCustomer
- Plural Label:** SCustomers
- Enable Reports:** ✓
- Track Activities:** ✓
- Track Field History:** ✓
- Deployment Status:** Deployed
- Help Settings:** Standard salesforce.com Help Window

The left sidebar lists various object settings: 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, and Restriction Rules.

Created all the Custom Fields

The screenshot shows the 'Fields & Relationships' section for the 'SCustomer' object. The table displays the following fields:

| FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD |
|------------------|------------------|--------------------|-------------------|
| Active | Active__c | Checkbox | |
| Created By | CreatedBy | Lookup(User) | |
| Customer Type | Customer_Type__c | Picklist | |
| Description | Description__c | Text Area(255) | |
| Last Modified By | LastModifiedBy | Lookup(User) | |
| Owner | OwnerId | Lookup(User,Group) | |
| SCustomer Name | Name | Text(80) | |

A sidebar on the right shows user information for 'Akshay Pusuluri' and options for 'DISPLAY DENSITY' (Compact), 'OPTIONS' (Switch to Salesforce Classic, Add Username), and a 'Comfy' checkbox.

The left sidebar lists other object settings: 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, and Restriction Rules.

Created a Custom Object named Billing and created custom fields

The screenshot shows the Salesforce Setup interface for the Billing object. The left sidebar includes links for Details, Fields & Relationships (which is selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, and Restriction Rules. The main content area displays the 'Fields & Relationships' table with the following data:

| FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
|------------------|------------------|--------------------------|-------------------|---------|
| Amount Paid | Amount_Paid__c | Currency(18, 0) | | |
| Bill Number | Name | Auto Number | | ✓ |
| Created By | CreatedById | Lookup(User) | | |
| Customer | Customer__c | Lookup(Customer) | | ✓ |
| Customer Type | Customer_Type__c | Picklist | | |
| Last Modified By | LastModifiedById | Lookup(User) | | |
| SCustomer | SCustomer__c | Master-Detail(SCustomer) | | ✓ |
| Status | Status__c | Picklist | | |

Code Playground App

The screenshot shows the Code Playground App interface with the 'Accounts' tab selected. The top navigation bar includes links for Leads, Accounts (selected), Contacts, Opportunities, SCustomers, and Billings. The main content area displays the 'All Accounts' list with the following data:

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

Exercise – 2

String Variable Using endsWith() method

The screenshot shows the Salesforce IDE interface. At the top, there's a menu bar with File, Edit, Debug, Test, Workspace, Help, and a search bar. Below the menu is a tab bar with DiscountClass.apxc, Opportunity, Opportunity@3:24 PM, and Log executeAnonymous @2/27/2023, 7:54:38 PM. The main area has a title bar "Execution Log" with columns for Timestamp, Event, and Details. A timestamp row shows 19:54:38:002 USER_DEBUG [2]DEBUG|true. To the right is an "Enter Apex Code" window containing the following code:

```
1 String test= 'Akshay Pusuluri';
2 System.debug(test.endsWith('i'));
3
```

Below the code editor are buttons for Open Log, Execute, and Execute Highlighted. At the bottom of the interface, there are checkboxes for This Frame, Executable, Debug Only, Filter, and a link to Click here to filter the log. A status bar at the very bottom shows Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems.

Used Data Method and Displayed the Date

The screenshot shows the Salesforce IDE interface. The top navigation and tabs are identical to the previous screenshot. The main area has a title bar "Execution Log" with columns for Timestamp, Event, and Details. Two timestamp rows are visible: 19:58:10:008 USER_DEBUG [2]DEBUG|2023-02-27 00:00:00 and 19:58:10:008 USER_DEBUG [4]DEBUG|2023-03-29 00:00:00. To the right is an "Enter Apex Code" window containing the following code:

```
1 date date1=Date.today();
2 System.debug(date1);
3 date date2=date1.addDays(30);
4 System.debug(date2);
5
```

Below the code editor are buttons for Open Log, Execute, and Execute Highlighted. At the bottom of the interface, there are checkboxes for This Frame, Executable, Debug Only, Filter, and a link to Click here to filter the log. A status bar at the very bottom shows Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems.

Converted the String to Integer and Added 20 to it

The screenshot shows the Salesforce Developer Console interface. At the top, there are tabs for File, Edit, Debug, Test, Workspace, Help, and several log entries. Below the tabs is the 'Execution Log' header with columns for Timestamp, Event, and Details. Two log entries are visible:

| Timestamp | Event | Details |
|--------------|------------|-------------|
| 20:00:32:003 | USER_DEBUG | [3]DEBUG 10 |
| 20:00:32:003 | USER_DEBUG | [4]DEBUG 30 |

In the center, an 'Enter Apex Code' window displays the following Apex code:

```
1 String a='10';
2 Integer b=Integer.valueOf(a);
3 System.debug(b);
4 System.debug(b+20);
5
```

At the bottom of the code editor are buttons for Open Log, Execute, and Execute Highlighted. Below the code editor is a toolbar with filters for This Frame, Executable, Debug Only, Filter, and Click here to filter the log. A status bar at the bottom shows the user's name, application, operation, time, status, read count, and size.

Used Length method to display String length

The screenshot shows the Salesforce Developer Console interface. At the top, there are tabs for File, Edit, Debug, Test, Workspace, Help, and several log entries. Below the tabs is the 'Execution Log' header with columns for Timestamp, Event, and Details. One log entry is visible:

| Timestamp | Event | Details |
|--------------|------------|-------------|
| 20:03:32:003 | USER_DEBUG | [3]DEBUG 15 |

In the center, an 'Enter Apex Code' window displays the following Apex code:

```
1 String a='Akshay Pusuluri';
2 Integer b=a.length();
3 System.debug(b);
4
```

At the bottom of the code editor are buttons for Open Log, Execute, and Execute Highlighted. Below the code editor is a toolbar with filters for This Frame, Executable, Debug Only, Filter, and Click here to filter the log. A status bar at the bottom shows the user's name, application, operation, time, status, read count, and size.

Defined a list and displayed the output using different methods

The screenshot shows the Salesforce IDE's Execution Log interface. At the top, there are five tabs: executeAnonymous @2/27/2023, 7:54:38 PM, Log executeAnonymous @2/27/2023, 7:58:10 PM, Log executeAnonymous @2/27/2023, 8:00:32 PM, Log executeAnonymous @2/27/2023, 8:03:32 PM, and Log executeAnonymous @2/27/2023, 8:08:51 PM. The last tab is highlighted in orange. Below the tabs is a toolbar with buttons for File, Edit, Debug, Test, Workspace, Help, and several others. A sub-menu bar is visible above the main menu.

The main area is titled "Execution Log" and contains a table with columns: Timestamp, Event, and Details. The table shows the following log entries:

| Timestamp | Event | Details |
|--------------|------------|--|
| 20:08:51:003 | USER_DEBUG | [5] DEBUG List After Adding elements[10, 20, 30] |
| 20:08:51:003 | USER_DEBUG | [6] DEBUG List Using Get Method[30] |
| 20:08:51:003 | USER_DEBUG | [8] DEBUG List after set method[10, 40, 30] |
| 20:08:51:004 | USER_DEBUG | [10] DEBUG List after clear method[] |

Below the log table is a large text area titled "Enter Apex Code" containing the following Apex code:

```
1 List<Integer> li =new List<Integer>();
2 li.add(10);
3 li.add(20);
4 li.add(30);
5 System.debug('List After Adding elements'+li);
6 System.debug('List Using Get Method'+li.get(2));
7 li.set(1,40);
8 System.debug('List after set method'+li);
9 li.clear();
10 System.debug('List after clear method'+li);
11
```

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

Below the code editor is a toolbar with buttons for This Frame, Executable, Debug Only, Filter, and Click here to filter the log. The "Logs" tab is selected.

At the very bottom is a table showing the execution details:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|------|---------|
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:08:51 PM | Success | | 4.07 KB |

Printed numbers between 0 to 9

The screenshot shows the Salesforce IDE's Execution Log interface. At the top, there are five tabs: executeAnonymous @2/27/2023, 7:58:10 PM, Log executeAnonymous @2/27/2023, 8:00:32 PM, Log executeAnonymous @2/27/2023, 8:03:32 PM, Log executeAnonymous @2/27/2023, 8:08:51 PM, and Log executeAnonymous @2/27/2023, 8:11:04 PM. The last tab is highlighted in orange. Below the tabs is a toolbar with buttons for File, Edit, Debug, Test, Workspace, Help, and several others. A sub-menu bar is visible above the main menu.

The main area is titled "Execution Log" and contains a table with columns: Timestamp, Event, and Details. The table shows the following log entries:

| Timestamp | Event | Details |
|--------------|------------|-------------------------------|
| 20:11:04:003 | USER_DEBUG | [2] DEBUG the value of x is:0 |
| 20:11:04:003 | USER_DEBUG | [2] DEBUG the value of x is:1 |
| 20:11:04:003 | USER_DEBUG | [2] DEBUG the value of x is:2 |
| 20:11:04:003 | USER_DEBUG | [2] DEBUG the value of x is:3 |
| 20:11:04:003 | USER_DEBUG | [2] DEBUG the value of x is:4 |
| 20:11:04:003 | USER_DEBUG | [2] DEBUG the value of x is:5 |
| 20:11:04:003 | USER_DEBUG | [2] DEBUG the value of x is:6 |
| 20:11:04:003 | USER_DEBUG | [2] DEBUG the value of x is:7 |
| 20:11:04:003 | USER_DEBUG | [2] DEBUG the value of x is:8 |
| 20:11:04:004 | USER_DEBUG | [2] DEBUG the value of x is:9 |

Below the log table is a large text area titled "Enter Apex Code" containing the following Apex code:

```
1 for(integer x=0;x<10;x++){
2     System.debug('the value of x is:'+x);
3 }
```

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

Below the code editor is a toolbar with buttons for This Frame, Executable, Debug Only, Filter, and Click here to filter the log. The "Logs" tab is selected.

At the very bottom is a table showing the execution details:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|------|---------|
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:11:04 PM | Success | | 5.38 KB |

Exercise – 3

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

executeAnonymous @2/27/2023, 8:00:32 PM Log executeAnonymous @2/27/2023, 8:03:32 PM Log executeAnonymous @2/27/2023, 8:08:51 PM Log executeAnonymous @2/27/2023, 8:11:04 PM Log executeAnonymous @2/27/2023, 8:13:57 PM

Execution Log

| Timestamp | Event | Details |
|--------------|------------|----------------|
| 20:13:57:002 | USER_DEBUG | [3]DEBUG false |

Enter Apex Code

```
1 Integer myLuckyNumber=15;
2 Integer myUnLuckyNumber=7;
3 System.debug(myLuckyNumber!=myUnLuckyNumber+8);
```

Open Log

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

Logs Tests Checkpoints Query Editor View State Progress Problems

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|------|---------|
| Aishay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:13:57 PM | Success | | 2.39 KB |

Exercise – 4

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

executeAnonymous @2/27/2023, 8:03:32 PM Log executeAnonymous @2/27/2023, 8:08:51 PM Log executeAnonymous @2/27/2023, 8:11:04 PM Log executeAnonymous @2/27/2023, 8:13:57 PM Log executeAnonymous @2/27/2023, 8:15:58 PM

Execution Log

| Timestamp | Event | Details |
|--------------|------------|---------------|
| 20:15:58:003 | USER_DEBUG | [3]DEBUG true |

Enter Apex Code

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

Open Log

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

Logs Tests Checkpoints Query Editor View State Progress Problems

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|------|---------|
| Aishay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:15:58 PM | Success | | 2.34 KB |

Exercise – 5

The screenshot shows the Salesforce Dev Console interface. At the top, there are several tabs: executeAnonymous @2/27/2023, 8:08:51 PM, Log executeAnonymous @2/27/2023, 8:11:04 PM, Log executeAnonymous @2/27/2023, 8:13:57 PM, Log executeAnonymous @2/27/2023, 8:15:58 PM, and Log executeAnonymous @2/27/2023, 8:18:33 PM. The current tab is Log executeAnonymous @2/27/2023, 8:18:33 PM. Below the tabs is a toolbar with File, Edit, Debug, Test, Workspace, and Help buttons.

The main area is titled "Execution Log". It displays a table with columns: Timestamp, Event, and Details. A single entry is shown: 20:18:33:003, USER_DEBUG, [3]DEBUG|true. To the right of the log table is a modal window titled "Enter Apex Code". Inside the modal, the following Apex code is displayed:

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

Below the modal are three buttons: Open Log, Execute, and Execute Highlighted. At the bottom of the Dev Console, there is a toolbar with filters: This Frame, Executable, Debug Only, Filter, and Click here to filter the log. The Logs tab is selected in the navigation bar.

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|------|---------|
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:18:33 PM | Success | | 2.45 KB |

Exercise – 6

Program for Finding the Grade Based on Score

The screenshot shows the Salesforce Dev Console interface. At the top, there are several tabs: DiscountClass.apxc, Opportunity, Opportunity@3:24 PM, and StudentGrade.apxc. The current tab is StudentGrade.apxc. Below the tabs is a toolbar with Code Coverage: None, API Version: 57, and Go To button.

The main area is a code editor showing the following Apex class:

```
1 public class StudentGrade {
2     public static void gradeAssigning() {
3         Integer Score = 100;
4         System.debug('Score is: '+Score);
5         if(Score == 100) {
6             System.debug('Grade: A+');
7         }
8         else if(Score >= 90) {
9             System.debug('Grade: A');
10        }
11        else if(Score >= 80) {
12            System.debug('Grade: B');
13        }
14        else {
15            System.debug('Grade: Failed');
16        }
17    }
18 }
```

At the bottom of the code editor is a toolbar with Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Logs tab is selected.

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|-----------------|-----------------------|---------|--------|----------|
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:22:02 PM | Success | Unread | 27.71 KB |

Screenshot of the Salesforce IDE showing the Execution Log and Logs tabs.

Execution Log:

| Timestamp | Event | Details |
|--------------|------------|------------------------|
| 20:23:32:013 | USER_DEBUG | [4]DEBUG Score is: 100 |
| 20:23:32:013 | USER_DEBUG | [5]DEBUG Grade: A+ |

Logs:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|--------|-----------|
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:23:32 PM | Success | | 3.09 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:22:02 PM | Success | Unread | 8.43 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:22:02 PM | Success | Unread | 989 bytes |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:22:02 PM | Success | Unread | 27.71 KB |
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:18:33 PM | Success | | 2.45 KB |
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:15:58 PM | Success | | 2.34 KB |

Exercise - 7

Program to execute Apex – for loop

Screenshot of the Salesforce IDE showing the Code Coverage tab and Logs tab.

Code Coverage: None • API Version 57

```

1 public class Billingobj {
2     public static void createrecords(){
3         Billing_c b1 = new Billing_c();
4         b1.Amount_Paid__c=15000;
5         b1.Status__c='Paid';
6         Billing_c b2 =new Billing_c();
7         b2.Amount_Paid__c=15000;
8         b2.Status__c='Paid';
9         Billing_c b3 =new Billing_c();
10        b3.Amount_Paid__c=15000;
11        b3.Status__c='Paid';
12        List<Billing_c> bills= new List<Billing_c>();
13        bills.add(b1);
14        bills.add(b2);
15        bills.add(b3);
16        try{
17            insert bills;
18        }
    
```

Logs:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|--------|-----------|
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:26:08 PM | Success | Unread | 27.72 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:26:08 PM | Success | Unread | 8.43 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:26:08 PM | Success | Unread | 989 bytes |
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:23:32 PM | Success | | 3.09 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:22:02 PM | Success | Unread | 8.43 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:22:02 PM | Success | Unread | 27.71 KB |

The screenshot shows the Salesforce Developer Console interface. At the top, there are tabs for various Apex classes and logs. A modal window titled "Enter Apex Code" is open, containing the following code:

```
1 Billingobj.createRecords();
```

Below the modal, the execution log table shows the following entries:

| Timestamp | Event | Details |
|--------------|------------|---|
| 20:28:03:027 | USER_DEBUG | [20]DEBUG[Invalid Data] |
| 20:28:03:047 | USER_DEBUG | [23]DEBUG[Value of Billing List: Billing__c:[{Id=a012w000018f5yQAS, Name=B - 0006, Status__c=Paid}] |
| 20:28:03:048 | USER_DEBUG | [26]DEBUG[Value of current record over which current record is iterating is: Billing__c:[{Id=a012w000018f5yQAS, Name=B - 0006, Status__c=Paid}] |
| 20:28:03:048 | USER_DEBUG | [27]DEBUG[Billing Records are: B - 0006] |

At the bottom, there are buttons for "Open Log", "Execute", and "Execute highlighted".

Exercise – 8

Created a Class to demonstrate constants in Apex

The screenshot shows the Salesforce Developer Console interface. The code editor contains the following Apex class:

```
1 public class DiscountClass {
2     Decimal regularDiscount = 1.56;
3     Decimal finalPrice;
4     public Decimal calculateDiscount(Integer price){
5         finalPrice = price - price*regularDiscount;
6         System.debug('finalPrice :'+finalPrice);
7         return finalPrice;
8     }
9 }
10 }
```

Below the code editor, the execution log table shows the following entries:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|------|--------|
| Aishay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:28:03 PM | Success | | 7.7 KB |

The screenshot shows the Salesforce IDE interface. At the top, there are several tabs: File, Edit, Debug, Test, Workspace, Help, and a few others. Below the tabs, the title bar includes the file names: StudentGrade.apxc, Log executeAnonymous @2/27/2023, 8:23:32 PM, Billingobj.apxc, Log executeAnonymous @2/27/2023, 8:28:03 PM, Log executeAnonymous @2/27/2023, 8:31:45 PM, and Log executeAnonymous @2/27/2023, 8:33:06 PM.

The main area has a header "Execution Log" with columns: Timestamp, Event, and Details. A single log entry is shown: 20:33:06:015 USER_DEBUG [6] DEBUG finalPrice :80.0.

A modal window titled "Enter Apex Code" contains the following code:

```

1 DiscountClass dc=new DiscountClass();
2 dc.calculateDiscount(400);

```

Below the modal are three buttons: Open Log, Execute, and Execute Highlighted.

At the bottom, there's a navigation bar with links: This Frame, Executable, Debug Only, Filter, and Click here to filter the log. The "Logs" tab is selected. The logs table has columns: User, Application, Operation, Time, Status, Read, and Size. It lists two entries:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|--------|---------|
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:33:11 PM | Success | Unread | 8.43 KB |
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:33:06 PM | Success | | 4.45 KB |

Exercise – 9

The screenshot shows the Salesforce IDE interface. At the top, there are several tabs: File, Edit, Debug, Test, Workspace, Help, and a few others. Below the tabs, the title bar includes the file names: DiscountClass.apxc, Opportunity, Opportunity@3:24 PM, StudentGrade.apxc, Log executeAnonymous @2/27/2023, 8:33:06 PM, and InterfaceExample.apxc.

The main area displays the Apex code for the InterfaceExample class:

```

1 public interface InterfaceExample {
2     Double percentageDiscountTobeApplied();
3 }

```

At the bottom, there's a navigation bar with links: Code Coverage: None, API Version: 57, and Go To.

At the very bottom, there's a "Logs" section with a table showing log entries. The table has columns: User, Application, Operation, Time, Status, Read, and Size. It lists multiple entries from the same user, all successful and unread, with sizes ranging from 988 bytes to 27.67 KB.

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|--------|-----------|
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:35:20 PM | Success | Unread | 988 bytes |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:35:17 PM | Success | Unread | 8.43 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:35:14 PM | Success | Unread | 27.68 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:33:11 PM | Success | Unread | 8.43 KB |
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:33:06 PM | Success | | 4.45 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:33:06 PM | Success | Unread | 27.67 KB |

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

DiscountClass.apxc Opportunity Opportunity@8:24 PM StudentGrade.apxc Log executeAnonymous @2/27/2023, 8:33:06 PM InterfaceExample.apxc PremiumCustomer.apxc

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

```

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

```

Logs Tests Checkpoints Query Editor View State Progress Problems

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|-----------------|-----------------------|---------|--------|-----------|
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:36:20 PM | Success | Unread | 8.47 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:36:20 PM | Success | Unread | 27.67 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:36:18 PM | Success | Unread | 988 bytes |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:35:20 PM | Success | Unread | 988 bytes |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:35:17 PM | Success | Unread | 8.43 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:35:14 PM | Success | Unread | 27.68 KB |

Filter Click here to filter the log list

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

DiscountClass.apxc Opportunity Opportunity@8:24 PM StudentGrade.apxc Log executeAnonymous @2/27/2023, 8:33:06 PM InterfaceExample.apxc PremiumCustomer.apxc Log executeAnonymous @2/27/2023, 8:39:21 PM

Execution Log

| Timestamp | Event | Details |
|--------------|------------|---------------|
| 20:39:21:017 | USER_DEBUG | [3] DEBUG 0.. |

Enter Apex Code

```

1 PremiumCustomer pc=new premiumCustomer();
2 Double test=pc.percentageDiscountToBeApplied();
3 System.debug(test);

```

Open Log Execute Execute highlighted

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 |
|-----------------|-------------|--|-----------------------|---------|--------|-----------|
| Aishay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:39:21 PM | Success | Unread | 3.87 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:36:20 PM | Success | Unread | 27.67 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:36:20 PM | Success | Unread | 8.47 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:36:18 PM | Success | Unread | 988 bytes |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:35:20 PM | Success | Unread | 988 bytes |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:35:17 PM | Success | Unread | 8.43 KB |

Filter Click here to filter the log list

Screencast 1: A screenshot of the Salesforce IDE interface showing the code editor and logs.

The code editor displays the following Apex class:

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

The logs panel shows the following log entries:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-----------------|--|-----------------------|---------|--------|-----------|
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:41:33 PM | Success | | 3.86 KB |
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:39:21 PM | Success | | 3.87 KB |
| Akshay Pusuluri | ApexTestHandler | | 2/27/2023, 8:36:20 PM | Success | Unread | 8.47 KB |
| Akshay Pusuluri | ApexTestHandler | | 2/27/2023, 8:36:20 PM | Success | Unread | 27.67 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:36:18 PM | Success | Unread | 988 bytes |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 8:35:20 PM | Success | Unread | 988 bytes |

Screencast 2: A screenshot of the Salesforce IDE interface showing the execution log and logs.

The execution log shows the following entry:

| Timestamp | Event | Details |
|--------------|------------|--------------|
| 20:41:33:013 | USER_DEBUG | [3]DEBUG 0.1 |

An apex code editor window is open with the following code:

```
1 NormalCustomer nc=new NormalCustomer();
2 Double test=nc.percentageDiscountTobeApplied();
3 System.debug(test);
```

The logs panel shows the following log entry:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|------|---------|
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/27/2023, 8:41:33 PM | Success | | 3.86 KB |

Exercise – 10

Demo on DML Insert Operation Using Database Methods

The screenshot shows the Salesforce IDE interface. The top navigation bar includes File, Edit, Debug, Test, Workspace, Help, and tabs for StudentGrade.apxc, Log executeAnonymous @2/24/2023, 1:47:35 PM, Billingobj.apxc, DiscountClass.apxc, and BillingRecordInsertion.apxc. A dropdown for Code Coverage and API Version is set to 57. On the right, there's an 'Enter Apex Code' panel with the following code:

```
1 BillingRecordInsertion dc= new BillingRecordInsertion();
2 dc.insertRecord();
3
```

Below the code editor is a toolbar with Open Log, Execute, and Execute Highlight buttons. At the bottom, a 'Logs' tab is selected in the navigation bar, showing a table with one log entry:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|------|---------|
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/26/2023, 4:48:53 PM | Success | | 6.52 KB |

The screenshot shows the Salesforce IDE interface with the same top navigation bar and tabs. The 'Logs' tab is selected. The 'Execution Log' section displays the following log entry:

| Timestamp | Event | Details |
|--------------|------------|--|
| 16:48:53:087 | USER_DEBUG | [14]DEBUG Id of Record Inserted:a012w000017u887AAA |

The screenshot shows the Salesforce IDE interface with the same top navigation bar and tabs. The 'Logs' tab is selected. The table below shows the log entry again:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|--|-----------------------|---------|------|---------|
| Akshay Pusuluri | Unknown | /services/data/v57.0/tooling/executeA... | 2/26/2023, 4:48:53 PM | Success | | 6.52 KB |

Billing
B - 0004

Related Details

Bill Number
B - 0004

Amount Paid
₹50,00,000

Customer Type
Customer

Status
Paid

SCustomer
[Wipro](#)

Created By
 [Akshay Pusuluri](#), 26/02/2023, 4:48 pm

Last Modified By
 [Akshay Pusuluri](#), 26/02/2023, 4:48 pm

Exercise – 11

Displaying Records According to the required Conditions

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

StudentGrade.apxc Log executeAnonymous @2/24/2023, 1:47:35 PM Billingobj.apxc DiscountClass.apxc Opportunity.obj Account@5:13 PM Opportunity@5:18 PM

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

| Query Results - Total Rows: 10 | | | | | |
|--------------------------------|--------|----------------------|---------------------------------------|------------------|--------------------|
| AccountId | Amount | StageName | Name | Account.Industry | Account.Website |
| 0012w000011t5Z2AAJ | 125000 | Negotiation/Review | United Oil Office Portable Generators | Energy | http://www.uos.com |
| 0012w000011t5Z2AAJ | 270000 | Proposal/Price Quote | United Oil Refinery Generators | Energy | http://www.uos.com |
| 0012w000011t5Z2AAJ | 120000 | Closed Won | United Oil SLA | Energy | http://www.uos.com |
| 0012w000011t5Z2AAJ | 270000 | Negotiation/Review | United Oil Installations | Energy | http://www.uos.com |
| 0012w000011t5Z2AAJ | 270000 | Closed Won | United Oil Installations | Energy | http://www.uos.com |
| 0012w000011t5Z2AAJ | 915000 | Closed Won | United Oil Refinery Generators | Energy | http://www.uos.com |
| 0012w000011t5Z2AAJ | 235000 | Closed Won | United Oil Installations | Energy | http://www.uos.com |
| 0012w000011t5Z2AAJ | 440000 | Closed Won | United Oil Emergency Generators | Energy | http://www.uos.com |
| 0012w000011t5Z2AAJ | 120000 | Closed Won | United Oil Standby Generators | Energy | http://www.uos.com |
| 0012w000011t5Z2AAJ | 675000 | Needs Analysis | United Oil Plant Standby Generators | Energy | http://www.uos.com |

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

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

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

Any query errors will appear here...

History Executed select Industry,Website,(Select id,Amount,StageName from Opportunities).select Industry,Website,(Select id,Amount,StageName from Opportunities).Select AccountId,Amount,StageName,Name,Account.Industry,Account.Website

Execute Use Tooling API

Exercise – 12

Created a Trigger for Billing Named CustomerTrigger

The screenshot shows the Salesforce IDE interface. The top navigation bar includes File, Edit, Debug, Test, Workspace, Help, and tabs for various Apex classes like StudentGrade.apxc, Log executeAnonymous @2/24/2023, 1:47:35 PM, Billingobj.apxc, DiscountClass.apxc, Opportunity, Account@3:24 PM, Opportunity@3:24 PM, and CustomerTrigger.apxt. The CustomerTrigger.apxt tab is active. Below the tabs, the code editor contains the following Apex trigger:

```
trigger CustomerTrigger on SCustomer__c (after update) {
    if(Trigger.isAfter && Trigger.isUpdate) {
        for(SCustomer__c cs:Trigger.new){
            if(cs.Active__c == True){
                Billing__c b = new Billing__c();
                b.Status__c = 'Paid';
                b.Amount_Paid__c = 100000;
                b.SCustomer__c = 'a002w00000ZUSPIAAS';
                insert b;
            }
        }
    }
}
```

Below the code editor is the Logs panel, which displays a list of log entries:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|-----------------|-----------------------|---|--------|-----------|
| Akshay Pusuluri | Browser | /aura | 2/27/2023, 4:20:01 PM | Success | Unread | 4.03 KB |
| Akshay Pusuluri | Browser | /aura | 2/27/2023, 4:17:52 PM | Insert failed. First exception on row 0;... | Unread | 4.67 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 4:15:50 PM | Success | Unread | 988 bytes |
| Akshay Pusuluri | Browser | /aura | 2/27/2023, 4:02:00 PM | Success | Unread | 717 bytes |
| Akshay Pusuluri | Browser | /aura | 2/27/2023, 4:01:59 PM | Success | Unread | 719 bytes |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 3:59:44 PM | Success | Unread | 990 bytes |

At the bottom of the Logs panel is a filter bar with the placeholder "Click here to filter the log list".

New Record Created in Billing Object with Salesforce Customer

The screenshot shows the Salesforce Lightning Experience. At the top, there is a navigation bar with icons for Home, Search, and various objects like Leads, Accounts, Contacts, Opportunities, SCustomers, and Billings. Below the navigation bar is a header for a Billing record with the ID B - 0006.

The main content area displays the Details tab of the Billing record. The fields shown include:

- Bill Number: B - 0006
- Amount Paid: ₹10.00.000
- Customer Type: (empty)
- Status: Paid
- Customer: Wipro
- SCustomer: Wipro
- Created By: Akshay Pusuluri, 27/02/2023, 4:20 pm
- Last Modified By: Akshay Pusuluri, 27/02/2023, 4:20 pm

At the top right of the details panel, there are buttons for New Contact, Edit, and New Opportunity.

Exercise – 13

Created a Test Class for CustomTrigger

The screenshot shows the Salesforce IDE interface. The top navigation bar includes File, Edit, Debug, Test, Workspace, Help, and tabs for various Apex classes like StudentGrade.apxc, Billingobj.apxc, etc., followed by **CustomTriggerTestClass.apxc**. Below the tabs, there's a dropdown for Code Coverage and API Version.

```

1  @isTest
2  public class CustomTriggerTestClass {
3    @isTest static void testCustomer() {
4      SCustomer__c c=new SCustomer__c();
5      c.Active__c=false;
6      c.Name='Test';
7      c.Customer_Type__c='Standard';
8      insert c;
9
10     Test.startTest();
11     c.Active__c=true;
12     update c;
13     System.assertEquals(c.Active__c,true);
14     System.assertEquals(c.Customer_Type__c,'Standard');
15     System.assertEquals(c.Name, 'Test');
16     Test.stopTest();
17   }
18 }
```

Below the code editor is a table showing the status of three test runs:

| Status | Test Run | Enqueued Time | Duration | Failures | Total |
|--------|-------------------|--------------------------------|----------|----------|-------|
| ✓ | 7072wv00008MUJyK | Mon Feb 27 2023 16:38:25 GM... | | 0 | 2 |
| ✓ | 7072wv00008MUJyUT | Mon Feb 27 2023 16:15:50 GM... | | 0 | 2 |
| ✓ | 7072wv00008MUjdw | Mon Feb 27 2023 15:59:38 GM... | | 0 | 2 |

On the right side, there's a section titled "Overall Code Coverage" with the following details:

| Class | Percent | Lines |
|------------------------|------------|-------|
| Overall | 14% | |
| Billingobj | 0% | 0/20 |
| BillingRecordInsertion | 0% | 0/12 |
| CustomerTrigger | 100% | 8/8 |
| DiscountClass | 0% | 0/4 |
| NewCaseListController | 0% | 0/4 |

Exercise – 14

Create a Trigger for Leads Named DisqualifyTestLeads

The screenshot shows the Salesforce IDE interface. The top navigation bar includes File, Edit, Debug, Test, Workspace, Help, and tabs for various Apex classes like hou.s, Billingobj.apxc, etc., followed by **DisqualifyTestLeads.apxc**. Below the tabs, there's a dropdown for Code Coverage and API Version.

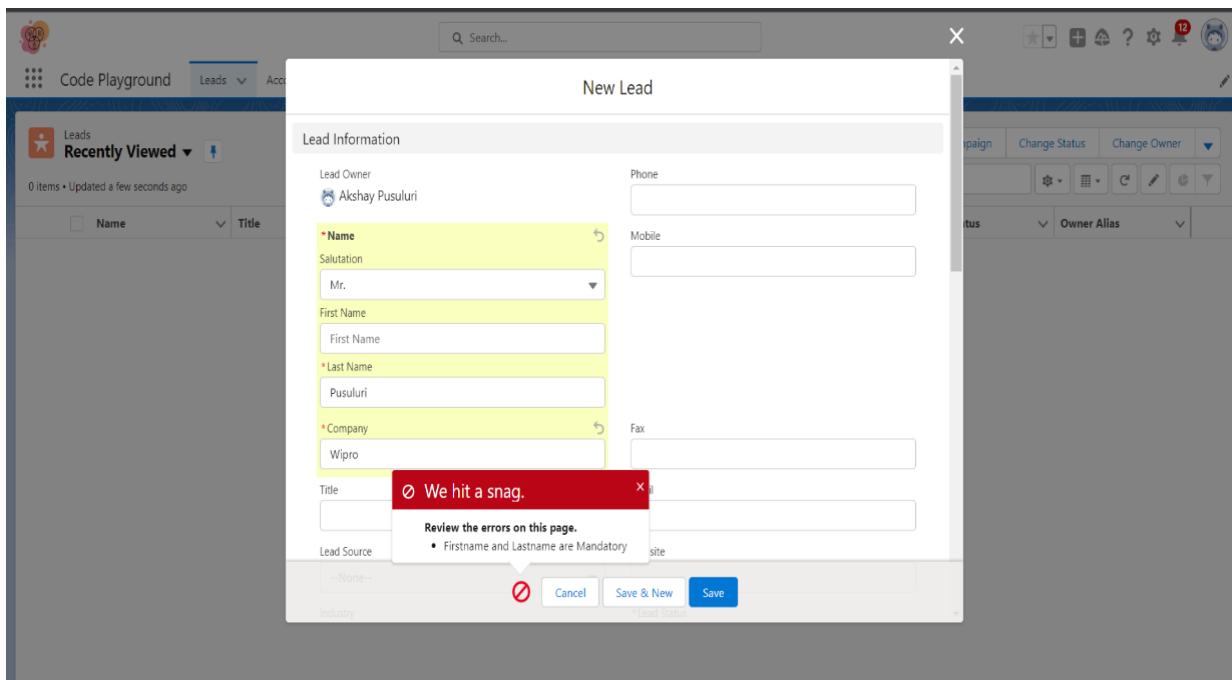
```

1 trigger DisqualifyTestLeads on Lead (after insert) {
2   List<Lead> li = new List<Lead>();
3   for(Lead l : Trigger.new){
4     if(l.FirstName=='Test' && l.LastName!=null || l.LastName=='Test' && l.FirstName!=null){
5       system.debug(l.FirstName' '+l.LastName+' Will be disqualified');
6       li.add(l);
7     }
8     list<Lead> get_leads = [select Id, Name, Status from Lead where Id IN:Trigger.new];
9     for(Lead ls:get_leads){
10       ls.Status = 'Disqualified';
11     }
12     update get_leads;
13   }else if(l.LastName!=null &&l.FirstName!=null){
14     li.add(l);
15   }
16   else{
17     l.addError('Firstname and Lastname are Mandatory');
18   }
}
```

Below the code editor is a table showing the log list:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|-----------------|-----------------------|---------|--------|-----------|
| Akshay Pusuluri | Browser | /aura | 2/27/2023, 4:54:47 PM | Success | Unread | 3.58 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 4:53:35 PM | Success | Unread | 8.45 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 4:53:34 PM | Success | Unread | 968 bytes |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 4:38:28 PM | Success | Unread | 968 bytes |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 4:38:26 PM | Success | Unread | 8.46 KB |
| Akshay Pusuluri | Browser | /aura | 2/27/2023, 4:20:01 PM | Success | Unread | 4.03 KB |

At the bottom, there's a search bar with the placeholder "Filter Click here to filter the log list".



Exercise – 15

Created a test class for DisqualifyTestLeads Trigger

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

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

DisqualifyTestLeads.apxc [] Billingobj.apxc [] DiscountClass.apxc [] Opportunity [] Account@3:24 PM [] Opportunity@3:24 PM [] CustomerTrigger.apxc [] CustomTriggerTestClass.apxc [] DisqualifyTestLeads.apxc [] DisqualifyTestLeads.apxc []

Code Coverage: None ▾ API Version 57 ▾ Go To

```

1 @isTest
2 public class DisqualifyTestLeads {
3     @istest
4     public static void insertTest(){
5         List<Lead> test_list = new List<Lead>();
6         Lead l1=new Lead();
7         l1.FirstName='Teja';
8         l1.LastName='Test';
9         l1.Company='Wepto';
10        l1.Status='Disqualified';
11        test_list.add(l1);
12        Test.startTest();
13        insert test_list;
14        Test.stopTest();
15        system.assertEquals(l1.FirstName,'Teja');
16        system.assertEquals(l1.LastName,'Test');
17        System.assertEquals(l1.Status,'Disqualified');
18    }

```

Logs Tests Checkpoints Query Editor View State Progress Problems

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|-----------------|-----------------------|---------|--------|------------|
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 5:02:48 PM | Success | Unread | 1001 bytes |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 5:02:48 PM | Success | Unread | 27.67 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 5:02:48 PM | Success | Unread | 8.45 KB |
| Aishay Pusuluri | Browser | /aura | 2/27/2023, 4:57:21 PM | Success | Unread | 710 bytes |
| Aishay Pusuluri | Browser | /aura | 2/27/2023, 4:57:20 PM | Success | Unread | 6.03 KB |
| Aishay Pusuluri | Browser | /aura | 2/27/2023, 4:54:47 PM | Success | Unread | 3.58 KB |

Filter Click here to filter the log list

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

hours @2/24/2023, 1:47:35 PM Billingobj.apxc DiscountClass.apxc Opportunity Account@3:24 PM Opportunity@3:24 PM CustomerTrigger.apxt CustomTriggerTestClass.apxc DisqualifyTestLeads.apxc DisqualifyTestLeads.apxt

Code Coverage: None ▾ API Version 57 Go To

```

19     @isTest
20     public static void test2(){
21         List<Lead> test_list2 = new list<Lead>();
22         Lead l2 = new Lead();
23         l2.LastName='Test';
24         l2.Company='wepto';
25         l2.FirstName=null;
26         test_list2.add(l2);
27         Test.startTest();
28     try{
29         insert test_list2;
30     }catch(Exception e){
31         system.debug('Cannot insert Lead');
32     }
33     Test.stopTest();
34     system.assertEquals(l2.FirstName,null);
35     system.assertEquals(l2.LastName, 'Test');
36 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|-----------------|-----------------------|---------|--------|------------|
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 5:02:48 PM | Success | Unread | 1001 bytes |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 5:02:48 PM | Success | Unread | 27.67 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 5:02:48 PM | Success | Unread | 8.45 KB |
| Akshay Pusuluri | Browser | /aura | 2/27/2023, 4:57:21 PM | Success | Unread | 710 bytes |
| Akshay Pusuluri | Browser | /aura | 2/27/2023, 4:57:20 PM | Success | Unread | 6.03 KB |
| Akshay Pusuluri | Browser | /aura | 2/27/2023, 4:54:47 PM | Success | Unread | 3.58 KB |

Filter: Click here to filter the log list

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

hours @2/24/2023, 1:47:35 PM Billingobj.apxc DiscountClass.apxc Opportunity Account@3:24 PM Opportunity@3:24 PM CustomerTrigger.apxt CustomTriggerTestClass.apxc DisqualifyTestLeads.apxc DisqualifyTestLeads.apxt

Code Coverage: None ▾ API Version 57 Go To

```

37     @isTest
38     static void leadTest3(){
39         List<Lead> test_list3 = new list<Lead>();
40         Lead l3 = new Lead();
41         l3.FirstName='Surya';
42         l3.LastName='Darl';
43         l3.Company='wepto';
44         test_list3.add(l3);
45         Test.startTest();
46         insert test_list3;
47         Test.stopTest();
48         system.assertEquals(l3.FirstName,'Surya');
49         system.assertEquals(l3.LastName,'Darl');
50
51
52     }
53
54 }
```

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

| Status | Test Run | Enqueued Time | Duration | Failures | Total |
|--------|-----------------|--------------------------------|----------|----------|-------|
| ✓ | 7072w00008MUHKY | Mon Feb 27 2023 16:38:25 GM... | | 0 | 2 |
| ✓ | 7072w00008MULDT | Mon Feb 27 2023 16:15:50 GM... | | 0 | 2 |
| ✓ | 7072w00008MlyJw | Mon Feb 27 2023 15:59:38 GM... | | 0 | 2 |
| ✓ | 7072w00008MlQSI | Mon Feb 27 2023 16:53:34 GM... | | 0 | 1 |
| ✓ | 7072w00008MlQJ9 | Mon Feb 27 2023 17:02:48 GM... | | 0 | 6 |

Overall Code Coverage

| Class | Percent | Lines |
|------------------------|---------|-------|
| Overall | 27% | |
| Billingobj | 0% | 0/20 |
| BillingRecordInsertion | 0% | 0/12 |
| CustomerTrigger | 100% | 8/8 |
| DiscountClass | 0% | 0/4 |
| DisqualifyTestLeads | 100% | 11/11 |

Exercise – 16

Created a Visualforce Page OppView and Displayed
Four apex: outputfields

The screenshot shows the Salesforce IDE interface. At the top, there's a menu bar with File, Edit, Debug, Test, Workspace, Help, and a preview area showing the current page. Below the menu is a tab bar with StudentGrade.apxc, Log executeAnonymous @2/24/2023, 1:47:35 PM, Billingobj.apxc, DiscountClass.apxc, CustomerTrigger.apxt, and OppView.vfp. The preview area displays the following Apex code:

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

Below the code editor is a logs section with tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Logs tab shows two entries:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|-----------------|-----------------------|---------|--------|-----------|
| Aishay Pusuluri | Browser | /apex/OppView | 2/27/2023, 2:24:35 AM | Success | Unread | 1.18 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 2:20:58 AM | Success | Unread | 990 bytes |

The browser preview at the bottom shows the "Opportunity Details" page with four output fields: Opportunity Name, Amount, Close Date, and Account Name.

Exercise – 17

Created a Visualforce page that shows a list of Account linked to their record pages

The screenshot displays the Salesforce IDE interface. At the top, the tabs show various Apex classes and a Visualforce page named "AccountList.vfp". Below the tabs, the "Preview" section shows the Visualforce code:

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

Below the code editor is the "Logs" tab, which lists several log entries from the developer console:

| User | Application | Operation | Time | Status | Read | Size |
|-----------------|-------------|-------------------|-----------------------|---------|--------|-----------|
| Akshay Pusuluri | Browser | /apex/AccountList | 2/27/2023, 2:48:01 AM | Success | Unread | 1.19 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 2:47:55 AM | Success | Unread | 989 bytes |
| Akshay Pusuluri | Browser | /apex/AccountList | 2/27/2023, 2:35:51 AM | Success | Unread | 1.19 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 2:35:45 AM | Success | Unread | 989 bytes |
| Akshay Pusuluri | Browser | /apex/AccountList | 2/27/2023, 2:31:55 AM | Success | Unread | 1.19 KB |
| Akshay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 2:31:50 AM | Success | Unread | 989 bytes |

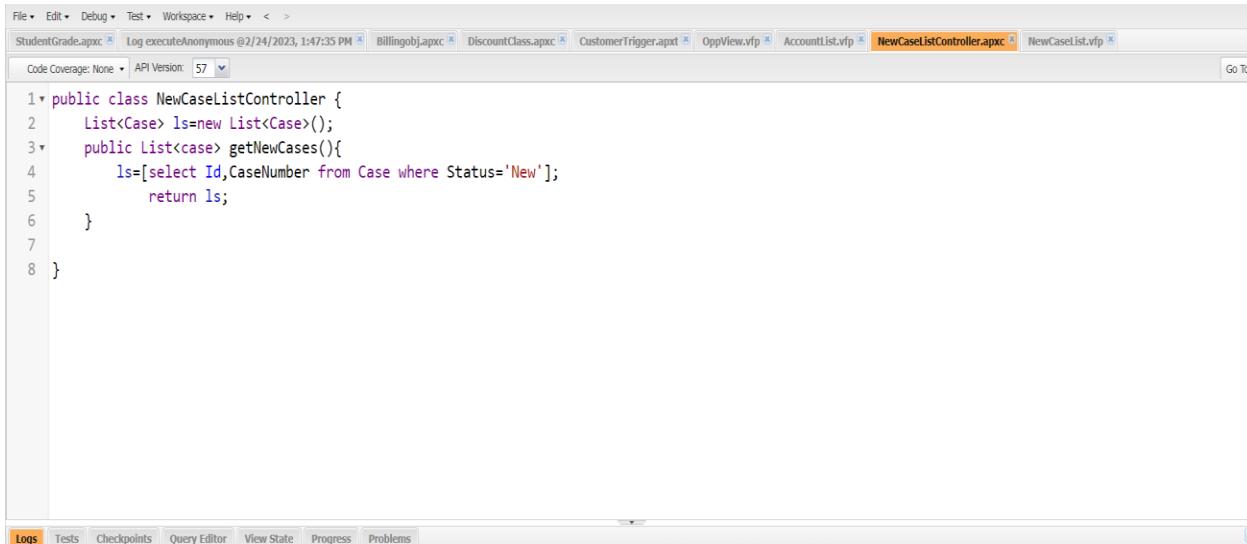
At the bottom, the browser window shows the visualforce page titled "Account Names" with a list of account names:

Account Names

- [Burlington Textiles Corp of America](#)
- [Dickenson plc](#)
- [Edge Communications](#)
- [Express Logistics and Transport](#)
- [GenePoint](#)
- [Grand Hotels & Resorts Ltd](#)
- [Pyramid Construction Inc.](#)
- [Sample Account for Entitlements](#)
- [sForce](#)
- [United Oil & Gas Corp.](#)
- [United Oil & Gas, Singapore](#)
- [United Oil & Gas, UK](#)
- [University of Arizona](#)

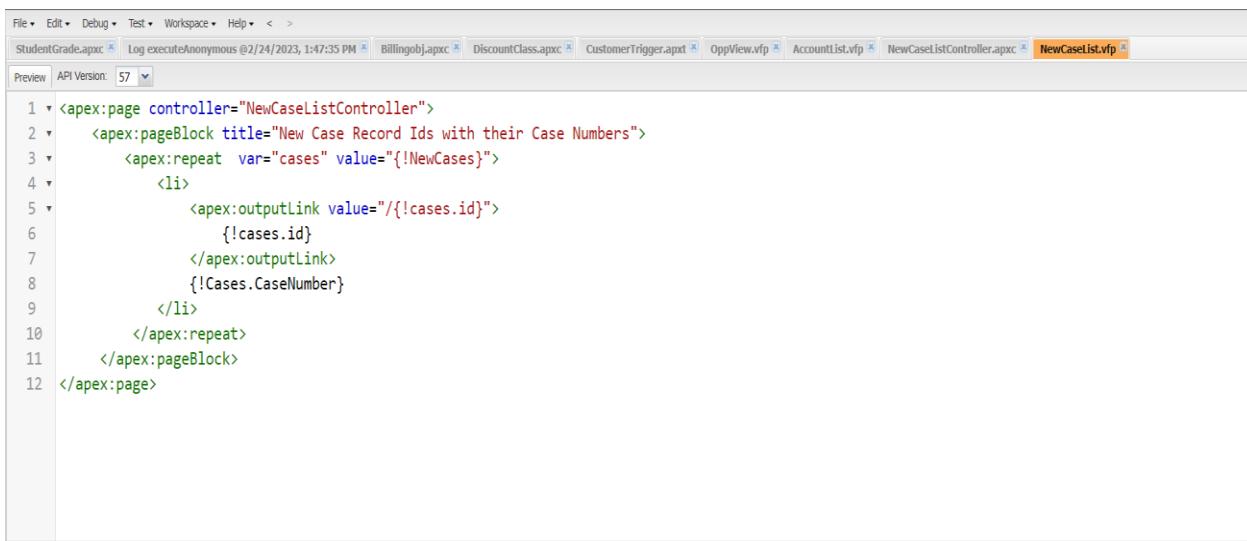
Exercise – 18

Created Vf page that uses an Apex class to Display
A list of cases with the status of ‘New’



```
1 * public class NewCaseListController {  
2     List<Case> ls=new List<Case>();  
3     public List<case> getNewCases(){  
4         ls=[select Id,CaseNumber from Case where Status='New'];  
5         return ls;  
6     }  
7 }  
8 }
```

| Logs | | | | | | |
|-----------------|-------------|-------------------|-----------------------|---------|--------|-----------|
| User | Application | Operation | Time | Status | Read | Size |
| Aishay Pusuluri | Browser | /apex/NewCaseList | 2/27/2023, 3:23:35 AM | Success | Unread | 3.44 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 3:19:57 AM | Success | Unread | 989 bytes |
| Aishay Pusuluri | Browser | /apex/NewCaseList | 2/27/2023, 3:17:37 AM | Success | Unread | 3.46 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 3:17:25 AM | Success | Unread | 989 bytes |
| Aishay Pusuluri | Browser | /apex/NewCaseList | 2/27/2023, 3:09:58 AM | Success | Unread | 3.46 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 3:09:54 AM | Success | Unread | 991 bytes |



```
1 <apex:page controller="NewCaseListController">  
2     <apex:pageBlock title="New Case Record Ids with their Case Numbers">  
3         <apex:repeat var="cases" value="{!NewCases}">  
4             <li>  
5                 <apex:outputLink value="/{!cases.id}">  
6                     {!cases.id}  
7                 </apex:outputLink>  
8                 {!Cases.CaseNumber}  
9             </li>  
10        </apex:repeat>  
11    </apex:pageBlock>  
12 </apex:page>
```

| Logs | | | | | | |
|-----------------|-------------|-------------------|-----------------------|---------|--------|-----------|
| User | Application | Operation | Time | Status | Read | Size |
| Aishay Pusuluri | Browser | /apex/NewCaseList | 2/27/2023, 3:23:35 AM | Success | Unread | 3.44 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 3:19:57 AM | Success | Unread | 989 bytes |
| Aishay Pusuluri | Browser | /apex/NewCaseList | 2/27/2023, 3:17:37 AM | Success | Unread | 3.46 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 3:17:25 AM | Success | Unread | 989 bytes |
| Aishay Pusuluri | Browser | /apex/NewCaseList | 2/27/2023, 3:09:58 AM | Success | Unread | 3.46 KB |
| Aishay Pusuluri | Unknown | ApexTestHandler | 2/27/2023, 3:09:54 AM | Success | Unread | 991 bytes |

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "playful-wolf-om1zom-dev-ed--c.trailblaze.vf.force.com/apex/NewCaseList?core.apexpages.request.devconsole=1". The page content is a list titled "New Case Record Ids with their Case Numbers" containing three items:

- [5002w00000g4VndAAE](#) 00001002
- [5002w00000g4VnrAAE](#) 00001016
- [5002w00000g4VnzAAE](#) 00001024

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](#)