



# Capstone Project Report On “Travel Approval App”

Submitted by:

**Name:** SAKAM SWETHA

**Batch:** Wipro Salesforce B13 (WIP-SF-13)

**LMS Id:** MGSA\_538

**Date:** 28/02/2023

## **Table of contents**

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

## **Abstract**

Salesforce, Inc. is an American cloud-based software company headquartered in San Francisco, California. It provides customer relationship management (CRM) software and applications focused on sales, customer service, marketing automation, analytics, and application development.

Salesforce's main technologies are tools for customer management. Other products enable customers to create apps, integrate data from other systems, visualize data, and offer training courses.

Force.com applications are built using declarative tools, backed by Lightning and Apex, a proprietary Java-like programming language for Force.com, as well as Visualforce, a framework including an XML syntax typically used to generate HTML. The Force.com platform typically receives three complete releases a year. As the platform is provided as a service to its developers, every single development instance also receives all these updates.

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

Therefore, with the help of Customer Relation Management (CRM) software services provided by Salesforce, we are making a travel approval application. It starts with building of data model where the objects and fields comes into picture follows by customizing the user interface where we Use list views and page layouts to streamline an app user's experience. Every application generally needed a business logic so in this it is achieved by validation rules , formula fields and approval process. Finally, to analyse travel approvals we add reports and dashboards. This application will help them in keeping track of the complete travel approval process.

## Introduction

Salesforce is a cloud-based software company that provides its customers with a platform to develop their own applications without following the tough steps that they used to follow in the legacy system. The software or application once created can be uploaded onto the cloud allowing the end-users to view them.

Salesforce is currently providing various software solutions and platforms for developers to create and distribute custom software/applications. Tech giants like Google, Twitter, Amazon, and Facebook are using Salesforce either in the form of SaaS or PaaS.



*Salesforce developers can make an application on the cloud and share it with multiple companies across multiple domains by using Salesforce.*

Talking about HR systems, every company across the globe has an HR team. Each HR team would require an HR application to store employee records. Almost all specifications for such an application would be common for all companies. So, as a developer, it would be very easy to create a Salesforce application for such specifications, post it onto the cloud, and provide it as a service to multiple clients at the same time. Maintenance of the same can be done altogether too. So basically, the problem of scalability gets eliminated.

## 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.

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

- 1. 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.
- 2. 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.
- 3. Real-Time Reporting:** The app provides real-time visibility into travel requests and approvals, allowing sales managers to monitor travel spend and make informed decisions about their teams' travel plans.

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

# Software Requirements

For the fastest and most stable experience, we recommend:

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

Minimum requirements are:

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

**OR**

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

# Screenshots

## Module 1

### Exercise 1 :- STEP 1:

#### Creating Lightning App: Travel App

The screenshot shows the Chatter feed within the 'Travel App' Lightning Experience. The feed displays posts from users 'sakam swetha' and 'Eric Executive'. The first post is a comment on a travel request, and the second is a question about department association. The interface includes standard Chatter controls like Like, Comment, and Share.

### STEP 2:

#### Custom Object: Department

The screenshot shows the 'Department' object details in the Salesforce Setup. The left sidebar lists various configuration options like Fields & Relationships, Page Layouts, and Record Types. The main panel displays the 'Details' tab for the 'Department' object, showing fields such as API Name (Department\_\_c), Singular Label (Department), and Plural Label (Departments). A sidebar on the right provides user-specific settings and density options.

## STEP 3:

### Creating custom fields in Department object Department Code Filed

The screenshot shows the Salesforce Object Manager interface for the 'Department' object. On the left, a sidebar lists various setup options like Details, Fields & Relationships, Page Layouts, Lightning Record Pages, etc. The 'Fields & Relationships' section is selected, displaying a table of fields. The table has columns for FIELD LABEL, FIELD NAME, DATA TYPE, and CONTR. The fields listed are: Created By (CreatedBy), Department Code (Department\_Code\_\_c), Department Name (Name), Department Type (Department\_Type\_\_c), Last Modified By (LastModifiedBy), Location (Location\_\_c), and Owner (OwnerId). The 'Department Code' field is highlighted.

## Field Dependency

The screenshot shows the 'Edit Field Dependency' page for the 'Department' object. The 'Fields & Relationships' tab is selected in the sidebar. The main area shows a grid where 'Location' is the Controlling Field and 'Department Type' is the Dependent Field. A legend indicates that yellow cells represent 'Included Value' and white cells represent 'Excluded Value'. The 'Finance' and 'IT' rows are marked as included values for the 'Banking' and 'Energy' categories respectively. Buttons for 'Save', 'Cancel', and 'Preview' are at the bottom.

## STEP 4:

### Custom Object : Travel Approval

The screenshot shows the Salesforce Setup interface with the 'Object Manager' selected. A sidebar on the left lists various configuration options like 'Fields & Relationships', 'Page Layouts', and 'Buttons, Links, and Actions'. The main panel displays the 'Details' section for the 'Travel Approval' object. It includes fields for 'Description', 'API Name' (set to 'Travel\_Approval\_\_c'), 'Custom' status (checked), 'Singular Label' ('Travel Approval'), and 'Plural Label' ('Travel Approvals'). On the right, a user profile for 'sakam swetha' is shown with display density set to 'Comfy'. A sidebar on the right contains options for 'Enable Reports', 'Track Activities', 'Track Field History', 'Deployment Status' (set to 'Deployed'), and 'Help Settings'.

### Custom Fields in Travel Approval

The screenshot shows the 'Fields & Relationships' section for the 'Travel Approval' object. It lists 13 items, sorted by Field Label. The table includes columns for Field Name, API Name, and Type. Key fields shown include 'Department' (Type: Lookup(Department)), 'Destination State' (Type: Text(2)), 'Last Modified By' (Type: Lookup(User)), 'Out-of-State' (Type: Checkbox), 'Owner' (Type: Lookup(User/Group)), 'Purpose of Trip' (Type: Text Area(255)), 'Status' (Type: Picklist), 'Status Indicator' (Type: Formula (Text)), 'Total Expenses' (Type: Roll-Up Summary (SUM Expense Item)), 'Travel Approval #' (Type: Auto Number), 'Trip End Date' (Type: Date), and 'Trip Start Date' (Type: Date). The right sidebar shows the same user profile and display density settings as the previous screenshot.

# Testing the Travel App with the given Objects and Fields

The screenshot shows the Salesforce interface for a Travel Approval record. The top navigation bar includes links for Travel App, Chatter, Dashboards, Reports, Departments, Travel Approvals, and Customers. The main content area displays a travel approval record with the ID TA-00500. The 'Details' tab is selected, showing fields such as Travel Approval # (TA-00500), Status (New), Total Expenses (\$0.00), and Created By (sakam swetha). On the right, there's a sidebar for Activity, Chatter, and Upcoming & Overdue tasks.

## STEP 5:

### Importing the Department records using the Data Import Wizard

The screenshot shows the Salesforce interface for the Department object. The top navigation bar is identical to the previous screenshot. The main content area displays a list of 16 departments, each with a checkbox and a name. The departments listed are: Audit Services, Contract Management, Disability Determination Bureau, Division of Aging, Division of Disability and Rehabilitative Services, Division of Family Resources, Division of Finance, Division of Mental Health and Addiction, Human Resources, Legislative Services, Office of Communications and Media, Office of Early Childhood and Out-of-School Learning, Office of General Counsel, Office of Medicaid Policy and Planning, Quality and Compliance Office, and Technology. A context menu is open on the right side, showing options like Comfy, Compact, and Switch to Salesforce Classic.

## Exercise 2:-

### STEP 1:

#### Creating Travel Approval Record

The screenshot shows a Salesforce page for a Travel Approval record with ID TA-00501. The page has a header with the Travel App logo, Chatter, Dashboards, Reports, Departments, Travel Approvals, and Scustomers. Below the header is a toolbar with Follow, Edit, New Contact, and New Opportunity buttons. The main area is divided into sections: 'Related' (empty), 'Details' (containing fields like Travel Approval #, Status, Total Expenses, Status Indicator, Created By, Owner, Department, Last Modified By, and Trip Info), and 'Activity' (showing no upcoming or overdue activities). The 'Details' section also includes a 'Trip Info' sub-section with fields for Purpose of Trip, Trip Start Date, Trip End Date, Out-of-State, Destination State, and a checkbox for 'Attend Dreamforce' which is checked.

### STEP 2:

#### Creating Expense Item Custom Object

The screenshot shows the Salesforce Setup interface under the Object Manager. A modal window is open for creating a new object named 'Expense Item'. The 'Details' tab is selected, showing fields for API Name (Expense\_\_Item\_\_c), Singular Label (Expense Item), and Plural Label (Expense Items). On the left, a sidebar lists various setup options like Fields & Relationships, Page Layouts, and Field Sets. On the right, a user profile for sakam swetha is visible with options to switch to Salesforce Classic or add a username. The status bar at the bottom indicates 'Switch to Salesforce Classic ⓘ' and 'Add Username'.

## STEP 3:

### Creating Custom Fields in Expense Item Object

The screenshot shows the Salesforce Setup interface under the Object Manager section for the 'Expense Item' object. On the left, a sidebar lists various setup categories like Page Layouts, Lightning Record Pages, and Field Sets. The main area displays a table titled 'Fields & Relationships' with 6 items, sorted by Field Label. The columns include FIELD LABEL, FIELD NAME, DATA TYPE, and CONTROL. The fields listed are 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)), and Travel Approval (Travel\_Approval\_\_c, Master-Detail(Travel Approval)). A context menu is open on the right side of the table, showing options like 'DISPLAY DENSITY' (set to 'Comfy'), 'OPTIONS', and links to 'Switch to Salesforce Classic' and 'Add Username'.

## STEP 4:

### Creating Records in Expense Item Object

The screenshot shows the Travel App interface. The top navigation bar includes 'Travel App', 'Chatter', 'Dashboards', 'Reports', 'Departments', 'Travel Approvals', 'Customers', and a search bar. The main content area shows a 'Travel Approval' record with ID 'TA-00500'. Below the approval details, there are sections for 'Notes & Attachments (0)' and 'Expense Items (2)'. The 'Expense Items' section lists two items: E-00003 (\$450.00, Airfare) and E-00004 (\$870.00, Hotel). A 'New' button is available to add more expense items. To the right of the main content, a sidebar displays the user profile of 'sakam swetha' and a 'DISPLAY DENSITY' dropdown set to 'Comfy'. The sidebar also includes an 'Activity' section with a note about no past activity and an 'Upcoming & Overdue' section with a note about no activities to show.

## STEP 5:

### Creating User names Eric Executive

The screenshot shows the Salesforce Setup interface under the 'Users' section. A new user record is being created for 'Eric Executive'. The 'User Detail' tab is selected, displaying the following information:

Field	Value
Name	Eric Executive
Alias	exec
Email	sakamswetha@gmail.com
Username	eric.swetha@wipro.com
Nickname	eric.exec
Title	
Company	
Department	
Division	
Address	
Time Zone	(GMT-08:00) Pacific Standard Time (America/Los_Angeles)
Locale	English (United States)
Language	English
Delegated Approver	Only if I am an approver
Manager	
Receive Approval Request Emails	Only if I am an approver
Federation ID	
App Registration: One-Time Password Authenticator	
App Registration: Salesforce Authenticator	
Security Key (U2F or WebAuthn)	
Lightning Login	
Temporary Verification Code (Expires in 1 to 24 Hours)	[Generate]

On the right side of the screen, there are several checkboxes for various user roles and features, many of which are checked. The 'Role' field is set to 'CEO'. The 'Profile' field is set to 'System Administrator'. The 'Active' checkbox is checked. Other checked checkboxes include 'Marketing User', 'Offline User', 'Knowledge User', 'Flow User', 'Service Cloud User', 'Site.com Contributor User', 'Site.com Publisher User', 'WDC User', 'Mobile Push Registration', 'Data.com User Type', 'Accessibility Mode (Classic Only)', 'Debug Mode', 'High-Contrast Palette on Charts', 'Load Lightning Pages While Scrolling', 'Send Apex Warning Emails', 'Salesforce CRM Content User', 'Receive Salesforce CRM Content Email Alerts', 'Receive Salesforce CRM Content Alerts as Daily Digest', and 'Make Setup My Default Landing Page'.

## STEP 6:

### Add User Eric as a Manager

The screenshot shows the Salesforce Setup interface under the 'Users' section. The 'User Detail' tab is still selected, but the 'Approver Settings' section is now visible at the bottom of the page. In this section, the 'Manager' field is populated with 'Eric Executive'. The 'Receive Approval Request Emails' dropdown is set to 'Only if I am an approver'. There is also a checkbox for 'Generate new password and notify user immediately' which is unchecked. At the bottom of the page, there are three buttons: 'Save', 'Save & New', and 'Cancel'.

## STEP 7:

### Customize the Travel Approval Default Search Layout

The screenshot shows the Salesforce Setup interface with the 'Search Layouts' page open. The page title is 'Edit Search Layout' for 'Travel Approval Search Results'. On the left, there's a list of 'Available Fields' including Record ID, Out-of-State, Status Indicator, Total Expenses, Owner Alias, Owner First Name, Owner Last Name, Created By Alias, Created By, Created Date, and Last Modified By Alias. On the right, a 'Selected Fields' list contains 'Travel Approval #' and 'Purpose of Trip'. Below these lists are buttons for 'Add' and 'Remove'. To the right of the selected fields is a vertical toolbar with 'Up' and 'Down' arrows. At the bottom of the page are 'Save' and 'Cancel' buttons. A sidebar on the right displays the user profile 'sakam swetha' and the URL 'd2w00000qtbadean-dev-ed.develop.my.salesforce.com'. It also shows 'DISPLAY DENSITY' set to 'Comfy' and an 'OPTIONS' section with 'Switch to Salesforce Classic' and 'Add Username'.

## STEP 8:

### Select Fields to display in the Travel Approval “All” List View

The screenshot shows the Travel App interface with the 'Travel Approvals' list view. The list displays 50+ items sorted by Travel Approval #. A modal window titled 'Select Fields to Display' is overlaid on the list. The 'Available Fields' section contains 'Created By Alias', 'Created Date', 'Destination State', 'Last Activity Date', 'Last Modified By', and 'Last Modified By Alias'. The 'Visible Fields' section contains 'Travel Approval #', 'Department', 'Created By', 'Status', 'Trip Start Date', and 'Trip End Date'. At the bottom of the modal are 'Cancel' and 'Save' buttons, with 'Save' being highlighted.

## STEP 9:

### Create Travel approval custom List View “Open Out of State Travel Requests”

The screenshot shows the Salesforce interface for the 'Travel Approvals' list view. The top navigation bar includes 'Travel App', 'Chatter', 'Dashboards', 'Reports', 'Departments', 'Travel Approvals', and 'Scustomers'. The 'Travel Approvals' tab is selected. A search bar at the top right contains the placeholder 'Search...'. Below the header, a message indicates '2 items • Sorted by Travel Approval #' and 'Filtered by All travel approvals - Status, Out-of-State - Updated a few seconds ago'. The main table displays two rows of travel approvals:

	Travel Approval #	Department	Created By	Status	Trip Start Date	Trip End Date
1	TA-00500	Contract Management	sakam swetha	New	2/22/2023	2/28/2023
2	TA-00501	Technology	sakam swetha	Draft	2/21/2023	3/4/2023

To the right of the table is a 'Filters' sidebar. It contains a section for 'Filter by Owner' with the condition 'All travel approvals'. Below it, under 'Matching all of these filters', are two conditions: 'Out-of-State equals True' and 'Status not equal to Approved, Rejected'. Buttons for 'Add Filter' and 'Remove All' are available, along with a link to 'Add Filter Logic'.

## STEP 10:

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

The screenshot shows the 'Select Fields to Display' dialog box overlaid on the travel approval list view. The title of the dialog is 'Open Out of State Travel Requests'. The dialog has two main sections: 'Available Fields' on the left and 'Visible Fields' on the right. The 'Available Fields' section lists: 'Created By Alias', 'Created Date', 'Last Activity Date', 'Last Modified By', 'Last Modified By Alias', and 'Last Modified Date'. The 'Visible Fields' section lists: 'Travel Approval #', 'Trip End Date', 'Trip Start Date', 'Department', 'Created By', and 'Destination State'. At the bottom of the dialog are 'Cancel' and 'Save' buttons.

## STEP 11:

# Customizing Page Layout

The screenshot shows the Lightning App Builder interface for creating a "Travel Approval Record Page". The main area displays a preview of the page layout with various sections and fields. On the left, a sidebar lists components and fields. A detailed description of the page layout is provided on the right, along with tips for customizing record detail sections and fields.

The screenshot shows the Salesforce Setup interface for managing page layouts. The "Page Layouts" section is selected for the "Travel Approval" object. The layout editor allows customization of fields such as Buttons, Quick Actions, and Mobile & Lightning Actions. A sidebar on the right provides user-specific information and setup options.

## STEP 12:

### Customize the Expense Item Related List under the Travel Approval page layout

The screenshot shows the Salesforce interface for a Travel Approval record (TA-00500). The main area displays the 'Related' section with three items: 'Notes & Attachments (0)', 'Expense Items (2)', and 'Approval History (0)'. The 'Expense Items' section lists two entries:

Expense Item Number	Amount	Expense Type
E-00003	\$450.00	Airfare
E-00004	\$870.00	Hotel

The right sidebar shows the user profile (sakam swetha) and various settings like 'DISPLAY DENSITY' (set to 'Comfy'), 'Activity' (empty), and 'Upcoming & Overdue' (empty).

## STEP 13:

### Enable “Feed Tracking” for Travel Approval Object

The screenshot shows the Salesforce Setup interface under 'Feed Tracking'. The left sidebar shows 'Email' and 'Feature Settings' sections. The main area is titled 'Feed Tracking' and shows the configuration for the 'Travel Approval' object. Under 'Fields in travel approvals', several fields are listed with checkboxes for 'Enable Feed Tracking': Department, Out-of-State, Purpose of Trip, Travel Approval #, Trip Start Date, Destination State, Owner, Status, and Trip End Date. A note states 'You can also display feed activity for related objects.' and a checkbox for 'All Related Objects'.

# Test Collaboration

The screenshot shows the Chatter feed within the Travel App. A user named sakam swetha posted a question asking which department to associate a travel request with. Another user, Eric Executive, responded that Technology is the correct department. The feed also shows a comment from sakam swetha and a reply from Eric Executive. On the right side, a context menu is open for the user sakam swetha, displaying options like 'DISPLAY DENSITY' (set to 'Comfy'), 'OPTIONS', and 'Switch to Salesforce Classic'. Other users listed in the menu include 'Eric Executive' and 'John Doe'.

# Testing the App

The screenshot shows the Travel Approval record page for TA-00001. The record details include the approval number, status (Approved), total expenses (\$1,320.00), and a status indicator (green). It also shows the creation date (2/14/2023) and last modified by (Eric Executive, 2/14/2023, 11:30 PM). The 'Trip Info' section lists the purpose of the trip (Attend Dreamforce), trip start date (2/6/2023), trip end date (2/14/2023), and destination state (CA). To the right, the Chatter feed shows a post from Eric Executive updating the record, changing its status from 'Draft to Approved'. Below the feed, a comment from sakam swetha is visible, along with a reply from Eric Executive.

## MODULE 2

### Exercise 1 :-

#### STEP 1:

Validation Rule that trip end date always  $\geq$  start date

The screenshot shows the Salesforce Setup interface under the Object Manager section for the 'Travel Approval' object. On the left, a sidebar lists various setup categories like Details, Fields & Relationships, Page Layouts, etc. The main content area is titled 'Travel Approval Validation Rule' and displays the details of a validation rule named 'Trip\_end\_date\_after\_start\_date'. The rule checks if 'Trip\_End\_Date\_\_c < Trip\_Start\_Date\_\_c' and triggers an error message: 'Trip end date must be greater than or equal to start date'. The rule is active and was created by 'sakam swetha' on 2/14/2023 at 9:45 PM.

#### Working on Validation Rule with error message

The screenshot shows the 'New Travel Approval' page. In the 'Information' section, the 'Status' is set to 'New' and the 'Owner' is 'sakam swetha' from the 'Contract Management' department. In the 'Trip Info' section, the 'Trip Start Date' is '2/24/2023' and the 'Trip End Date' is '2/22/2023'. A red error message box appears stating: 'We hit a snag.' and 'Review the following fields • Trip\_End Date'. The 'Trip\_End Date' field is highlighted in yellow. At the bottom of the page are 'Cancel', 'Save & New', and 'Save' buttons.

## STEP 2:

### Creating Roll-up Summary Field on Travel Approval Object

The screenshot shows the Salesforce Setup interface under the 'Object Manager' section for the 'Travel Approval' object. On the left, a sidebar lists various setup options like Page Layouts, Lightning Record Pages, and Field Sets. The main panel displays the 'Custom Field Definition Detail' for a field named 'Total Expenses'. The field information includes its label ('Total Expenses'), name ('Total\_Expenses'), and API name ('Total\_Expenses\_\_c'). It also shows the object it belongs to ('Travel Approval') and who created it ('sakam swetha'). Under 'Roll-Up Summary Options', the data type is set to 'Roll-Up Summary' with 'Expense Item' as the summarized object and 'Expense Item: Amount' as the field to aggregate. The summary type is set to 'SUM'. A note at the bottom indicates this is a custom field for the 'Travel Approval' object.

## STEP 3:

### Creating Formula Field Static Resource with name StatusImages

The screenshot shows the Salesforce Setup interface under the 'Static Resources' section. The search bar on the left shows 'static'. The main panel displays the 'Static Resource Detail' for a resource named 'StatusImages'. The details include its name ('StatusImages'), namespace prefix (''), description (''), MIME type ('application/zip'), cache control ('Public'), and size ('39,130 bytes'). It also shows the file ('View file'). The resource was created by 'sakam swetha' on 2/14/2023, 10:00 PM and last modified by 'sakam swetha' on 2/14/2023, 10:00 PM. A sidebar on the right shows the user profile ('sakam swetha'), settings, and log out options. It also includes display density settings ('Comfy' is selected) and an option to switch to Salesforce Classic.

## STEP 4:

### Create Status Formula Field on Travel Approval

The screenshot shows the 'Edit Update Records' screen in the Flow Builder. The flow is titled 'Update Travel Approval Record (Update\_Travel\_Approval\_Record)'. In the 'How to Find Records to Update and Set Their Values' section, the radio button 'Use the travel approval record that triggered the flow' is selected. A note below 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.' Under 'Set Filter Conditions', there is a dropdown menu set to 'None—Always Update Record'. In the 'Set Field Values for the Travel Approval Record' section, a field named 'Out\_of\_State\_\_c' is mapped to the value 'True'. The 'Done' button is visible at the bottom right.

## STEP 5:

### Record Trigger Flow

The screenshot shows the 'Configure Start' screen in the Flow Builder. The 'Object' dropdown is set to 'Travel Approval'. In the 'Configure Trigger' section, the radio button 'A record is created or updated' is selected. Below it, the 'Set Entry Conditions' section notes: 'Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.' It also mentions: 'If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the Only when a record is updated to meet the condition requirements option for When to Run the Flow for Updated Records.' The 'Condition Requirements' dropdown is set to 'None'. In the 'Optimize the Flow for:' section, 'Fast Field Updates' is selected. The 'Actions and Related Records' section is partially visible. The 'Done' button is visible at the bottom right.

# Creating Decisions

The screenshot shows the 'Edit Update Records' screen in the Flow Builder. The flow is titled 'Out of State Travel Flag - V1'. The main area displays instructions for finding records to update and setting their values. It highlights that the flow runs before a record is saved, so only the record that triggered the flow can be updated. A 'Set Filter Conditions' section shows a dropdown set to 'None—Always Update Record'. Below this, a 'Set Field Values for the Travel Approval Record' section shows a table with one row: 'Field' 'Out\_of\_State\_\_c' and 'Value' 'True'. At the bottom right are 'Cancel' and 'Done' buttons.

The screenshot shows the 'Edit Decision' screen in the Flow Builder. The decision is titled 'Is Travel Out of State? (Is\_Travel\_Out\_of\_State)'. The 'OUTCOME ORDER' section lists two outcomes: 'Yes Out of State' and 'In State'. The 'Default Outcome' is set to 'All Conditions Are Met (AND)'. The 'OUTCOME DETAILS' section for 'In State' includes a 'Label' 'In State' and an 'Outcome API Name' 'In\_State'. A 'Condition Requirements to Execute Outcome' dropdown is set to 'All Conditions Are Met (AND)'. Below it, a condition table shows a resource 'A\_a \$Record > Destination State X' with an operator 'Equals' and value 'TX'. A 'When to Execute Outcome' section at the bottom has a radio button selected for 'Only if the record that triggered the flow is updated to meet the condition requirements'. At the bottom right are 'Cancel' and 'Done' buttons.

Flow Builder   Out of State Travel Flag - V1

Free-Form   Version 1: Active—Last modified 8 days ago   Run   Debug   View Tests   Deactivate   Save As   Save

### Edit Decision

**Is Travel Out of State? (Is\_Travel\_Out\_of\_State)**

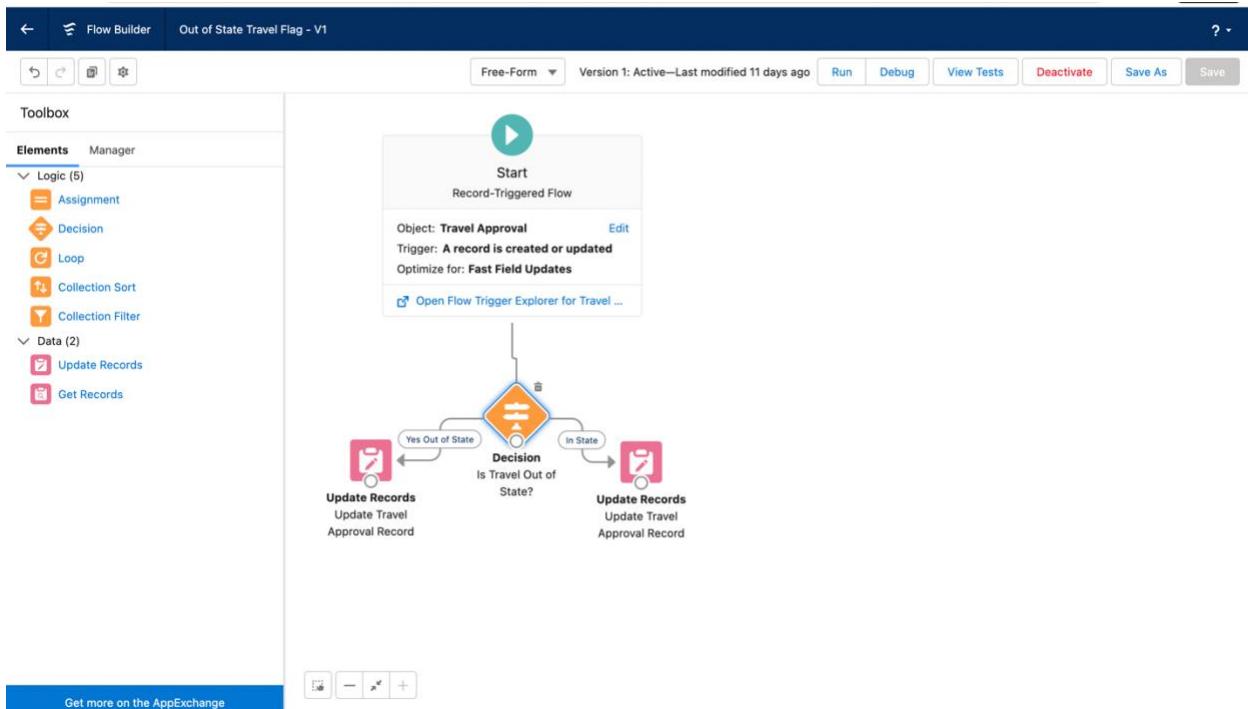
**Outcomes** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME ORDER	OUTCOME DETAILS	DELETE OUTCOME
1	<p><b>Yes Out of State</b></p> <p>* Label: Yes Out of State</p> <p>* Outcome API Name: Yes_Out_of_State</p> <p>Condition Requirements to Execute Outcome: All Conditions Are Met (AND)</p> <p>Resource: \$Record &gt; Destination State   Operator: Does Not Equal   Value: TX</p> <p>+ Add Condition</p>	Delete Outcome
2	<p><b>In State</b></p>	
3	<p><b>Default Outcome</b></p>	

**When to Execute Outcome**

- If the condition requirements are met
- Only if the record that triggered the flow to run is updated to meet the condition requirements

Cancel   Done



# STEP 6:

## Creating an Approval Process to send travel approvals to manage or travel coordinator

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

**Approval Processes**

**Travel Approval: Travel Approval Request**

**Process Definition Detail**

- Process Name: Travel Approval Request
- Unique Name: Travel\_Approval\_Request
- Description: Travel Approval: Total Expenses GREATER THAN 0
- Entry Criteria: Administrator ONLY
- Record Editability: Allow Submitters to Recall Approval Requests
- Approval Assignment Email Template: Travel Approval Owner
- Initial Submitters: Travel Approval Owner
- Created By: sakam.swetha, 2/14/2023, 10:18 PM
- Modified By: sakam.swetha, 2/14/2023, 11:25 PM

**Initial Submission Actions**

Action Type	Description
Record Lock	Lock the record from being edited

**Approval Steps**

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions   Edit	1	Step 1			Manager	Final Rejection
Show Actions   Edit	2	Travel Coordinator Approval	Travel Approval: Out-of-State EQUALS True		User:sakam.swetha	Final Rejection

**Final Approval Actions**

Action	Type	Description
Edit	Record Lock	Lock the record from being edited
Edit   Remove	Field Update	Set Status to Approved

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

**Approval Processes**

**Initial Submission Actions**

Action Type	Description
Record Lock	Lock the record from being edited

**Approval Steps**

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions   Edit	1	Step 1			Manager	Final Rejection
Show Actions   Edit	2	Travel Coordinator Approval	Travel Approval: Out-of-State EQUALS True		User:sakam.swetha	Final Rejection

**Final Approval Actions**

Action	Type	Description
Edit	Record Lock	Lock the record from being edited
Edit   Remove	Field Update	Set Status to Approved

**Final Rejection Actions**

Action	Type	Description
Edit	Record Lock	Unlock the record for editing
Edit   Remove	Field Update	Set Status to Rejected

**Recall Actions**

Action	Type	Description
	Record Lock	Unlock the record for editing

Travel Approval TA-00001

**Notes & Attachments (0)**

**Expense Items (2)**

Expense Item Number	Amount	Expense Type
E-00001	\$450.00	Airfare
E-00002	\$870.00	Hotel

**Approval History (3)**

Step Name	Date	Status	Assigned To
Travel Coordinator Approval	2/14/2023, 11:30 PM	Approved	sakam swetha
Step 1	2/14/2023, 11:29 PM	Approved	Eric Executive
Approval Request Submitted	2/14/2023, 11:18 PM	Submitted	sakam swetha

**Upcoming & Overdue**

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

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

## Final Result

**sakam swetha** d2w000000qtbadean-dev-ed.develop.my.s...

**DISPLAY DENSITY**

- ✓ **Comfy**
- Compact

**OPTIONS**

- Switch to Salesforce Classic
- Add Username

**Write a comment...**

**sakam.swetha** February 14, 2023 at 9:42 PM

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

1 comment · Seen by 1

**Like** **Comment**

**sakam.swetha** 8 days ago

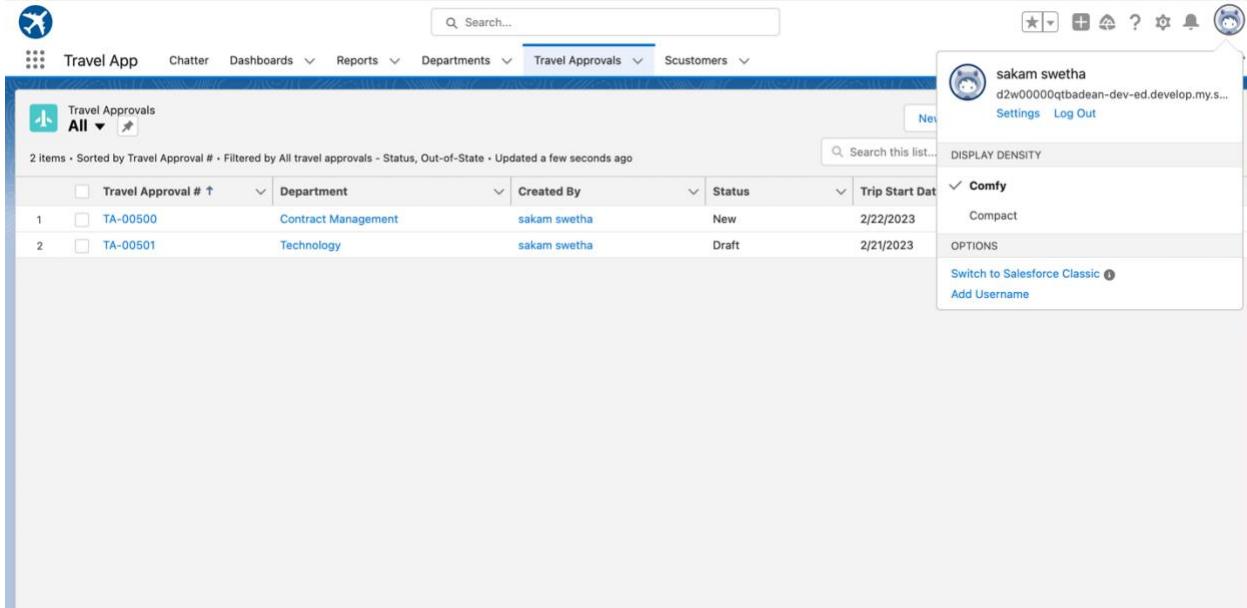
Technology is the correct department.

**Like**

**Write a comment...**

## Exercise 2:-

### STEP 1: Using Data Import Wizard Importing Travel Approval Records

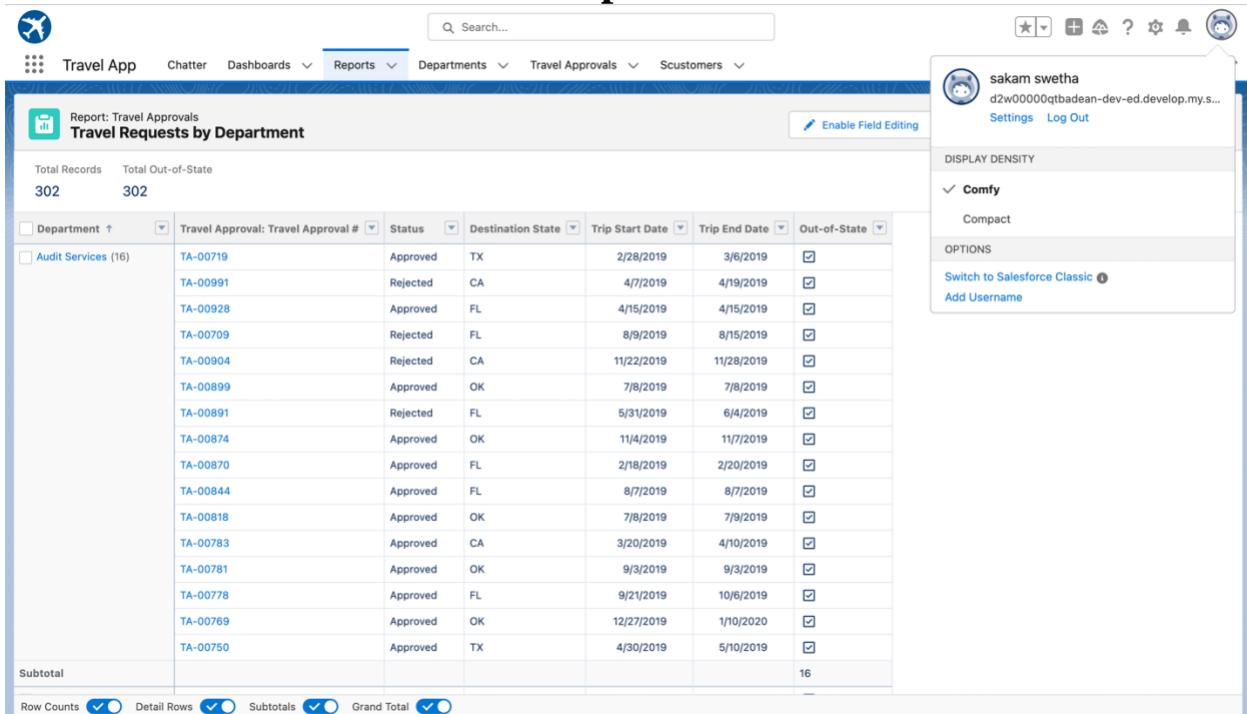


The screenshot shows the Salesforce interface for the Travel App. The top navigation bar includes links for Chatter, Dashboards, Reports, Departments, Travel Approvals, and Scustomers. The 'Travel Approvals' tab is selected. A search bar at the top right contains the placeholder 'Search...'. On the left, a sidebar shows a 'Travel Approvals' section with a 'All' button and a refresh icon. The main content area displays a table with two rows of travel approval records:

	Travel Approval #	Department	Created By	Status	Trip Start Date
1	TA-00500	Contract Management	sakam swetha	New	2/22/2023
2	TA-00501	Technology	sakam swetha	Draft	2/21/2023

To the right of the table is a user profile for 'sakam swetha' and a sidebar titled 'DISPLAY DENSITY' with options 'Comfy' (selected) and 'Compact'. Below these are 'OPTIONS' buttons for 'Switch to Salesforce Classic' and 'Add Username'.

### STEP 2: Creating Travel Request by Department report and add chart Properties



The screenshot shows the Salesforce interface for Reports. The top navigation bar includes links for Chatter, Dashboards, Reports, Departments, Travel Approvals, and Scustomers. The 'Reports' tab is selected. A search bar at the top right contains the placeholder 'Search...'. The main content area displays a report titled 'Report: Travel Approvals' with the subtitle 'Travel Requests by Department'. The report table shows travel approval details grouped by department:

Department	Travel Approval #	Status	Destination State	Trip Start Date	Trip End Date	Out-of-State
Audit Services (16)	TA-00719	Approved	TX	2/28/2019	3/6/2019	<input checked="" type="checkbox"/>
	TA-00991	Rejected	CA	4/7/2019	4/19/2019	<input checked="" type="checkbox"/>
	TA-00928	Approved	FL	4/15/2019	4/15/2019	<input checked="" type="checkbox"/>
	TA-00709	Rejected	FL	8/9/2019	8/15/2019	<input checked="" type="checkbox"/>
	TA-00904	Rejected	CA	11/22/2019	11/28/2019	<input checked="" type="checkbox"/>
	TA-00899	Approved	OK	7/8/2019	7/8/2019	<input checked="" type="checkbox"/>
	TA-00891	Rejected	FL	5/31/2019	6/4/2019	<input checked="" type="checkbox"/>
	TA-00874	Approved	OK	11/4/2019	11/7/2019	<input checked="" type="checkbox"/>
	TA-00870	Approved	FL	2/18/2019	2/20/2019	<input checked="" type="checkbox"/>
	TA-00844	Approved	FL	8/7/2019	8/7/2019	<input checked="" type="checkbox"/>
	TA-00818	Approved	OK	7/8/2019	7/9/2019	<input checked="" type="checkbox"/>
	TA-00783	Approved	CA	3/20/2019	4/10/2019	<input checked="" type="checkbox"/>
	TA-00781	Approved	OK	9/3/2019	9/3/2019	<input checked="" type="checkbox"/>
	TA-00778	Approved	FL	9/21/2019	10/6/2019	<input checked="" type="checkbox"/>
	TA-00769	Approved	OK	12/27/2019	1/10/2020	<input checked="" type="checkbox"/>
	TA-00750	Approved	TX	4/30/2019	5/10/2019	<input checked="" type="checkbox"/>

The report also includes a 'Subtotal' row and summary statistics: Total Records (302) and Total Out-of-State (302). At the bottom, there are checkboxes for Row Counts, Detail Rows, Subtotals, and Grand Total.

Travel App Chatter Dashboards Reports Departments Travel Approvals Scustomers

Report: Travel Approvals  
Travel Requests by Department

Department	Travel Approval: Travel Approval #	Status	Destination State	Trip Start Date	Trip End Date	Out-of-State
Contract Management (21)	TA-00500	New	jk	2/22/2023	2/28/2023	<input checked="" type="checkbox"/>
	TA-00725	Approved	OK	5/11/2019	5/11/2019	<input checked="" type="checkbox"/>
	TA-00999	Approved	GA	10/30/2019	11/2/2019	<input checked="" type="checkbox"/>
	TA-00996	Approved	CA	6/17/2019	6/20/2019	<input checked="" type="checkbox"/>
	TA-00987	Approved	FL	4/14/2019	5/5/2019	<input checked="" type="checkbox"/>
	TA-00979	Approved	FL	6/20/2019	6/23/2019	<input checked="" type="checkbox"/>
	TA-00971	Approved	TX	4/4/2019	4/4/2019	<input checked="" type="checkbox"/>
	TA-00966	Rejected	TX	10/17/2019	10/20/2019	<input checked="" type="checkbox"/>
	TA-00965	Approved	CA	7/5/2019	7/8/2019	<input checked="" type="checkbox"/>
	TA-00962	Approved	TX	7/4/2019	7/4/2019	<input checked="" type="checkbox"/>
	TA-00957	Approved	TX	1/18/2019	1/18/2019	<input checked="" type="checkbox"/>
	TA-00940	Approved	CA	3/31/2019	4/2/2019	<input checked="" type="checkbox"/>
	TA-00717	Approved	FL	5/8/2019	5/14/2019	<input checked="" type="checkbox"/>
	TA-00707	Approved	TX	12/22/2019	1/1/2020	<input checked="" type="checkbox"/>
	TA-00896	Approved	FL	2/20/2019	2/24/2019	<input checked="" type="checkbox"/>
	TA-00895	Approved	FL	1/29/2019	1/29/2019	<input checked="" type="checkbox"/> true
	TA-00821	Approved	GA	2/9/2019	2/9/2019	<input checked="" type="checkbox"/>
	TA-00815	Approved	GA	1/30/2019	1/30/2019	<input checked="" type="checkbox"/>
	TA-00736	Rejected	CA	3/20/2019	4/1/2019	<input checked="" type="checkbox"/>

Row Counts Detail Rows Subtotals Grand Total

sakam swetha d2w000000qtbadean-dev-ed.develop.my.s... Settings Log Out

DISPLAY DENSITY Comfy Compact OPTIONS Switch to Salesforce Classic Add Username

## Final Report

Travel App Chatter Dashboards Reports Departments Travel Approvals Scustomers

REPORT Travel Requests by Department Travel Approvals

Report "Travel Requests by Department" was saved

Fields > Groups: Department

Record Count

Department

Department	Travel Approval: Travel Approval #	Status	Destination State	Trip Start Date	Trip End Date	Out-of-State
Audit Services (1)	TA-00750	Approved	TX	4/30/2019	5/10/2019	<input checked="" type="checkbox"/>
Subtotal						1
Disability Determination Bureau (1)	TA-00736	Approved	FL	10/20/2019	10/22/2019	<input checked="" type="checkbox"/>
Subtotal						1
Division of Aging (3)	TA-00737	Approved	FL	2/17/2019	2/20/2019	<input checked="" type="checkbox"/>
	TA-00743	Approved	GA	3/5/2019	3/14/2019	<input checked="" type="checkbox"/>
	TA-00744	Approved	OK	1/24/2019	1/24/2019	<input checked="" type="checkbox"/>
Subtotal						3
Division of Disability and Rehabilitative Services (2)	TA-00732	Approved	GA	7/7/2019	7/7/2019	<input checked="" type="checkbox"/>
Row Counts Detail Rows Subtotals Grand Total	Conditional Formatting					

## STEP 3:

### Creating Travel Request by Month Report

The screenshot shows the Salesforce interface with the 'Travel App' tab selected. The page title is 'Report: Travel Approvals - Travel Requests by Month'. The report displays 302 total records for January 2019. The columns include Trip End Date, Out-of-State, Travel Approval: Travel Approval #, Department, Status, Destination State, and Trip Start Date. The data shows various travel approvals from different departments like Contract Management, Quality and Compliance Office, and Office of General Counsel, with statuses ranging from Approved to Rejected.

Trip End Date	Out-of-State	Travel Approval: Travel Approval #	Department	Status	Destination State	Trip Start Date
January 2019 (17)	(17)	TA-00957	Contract Management	Approved	TX	1/18/2019
		TA-00895	Contract Management	Approved	FL	1/29/2019
		TA-00815	Contract Management	Approved	GA	1/30/2019
		TA-00948	Quality and Compliance Office	Approved	TX	1/14/2019
		TA-00802	Disability Determination Bureau	Approved	TX	1/12/2019
		TA-00766	Legislative Services	Approved	TX	1/20/2019
		TA-00723	Office of Communications and Media	Rejected	FL	1/12/2019
		TA-00789	Office of General Counsel	Rejected	TX	1/27/2019
		TA-00936	Division of Aging	Approved	TX	1/22/2019
		TA-00804	Division of Aging	Approved	FL	1/3/2019
		TA-00744	Division of Aging	Approved	OK	1/24/2019
		TA-00729	Division of Disability and Rehabilitative Services	Approved	OK	1/23/2019
		TA-00859	Division of Disability and Rehabilitative Services	Approved	FL	1/29/2019
		TA-00910	Division of Family Resources	Approved	FL	1/21/2019
		TA-00814	Division of Mental Health and Addiction	Approved	OK	1/3/2019
		TA-00913	Office of Early Childhood and Out-of-School Learning	Rejected	OK	1/4/2019
		TA-00908	Office of Early Childhood and Out-of-School Learning	Approved	GA	1/18/2019

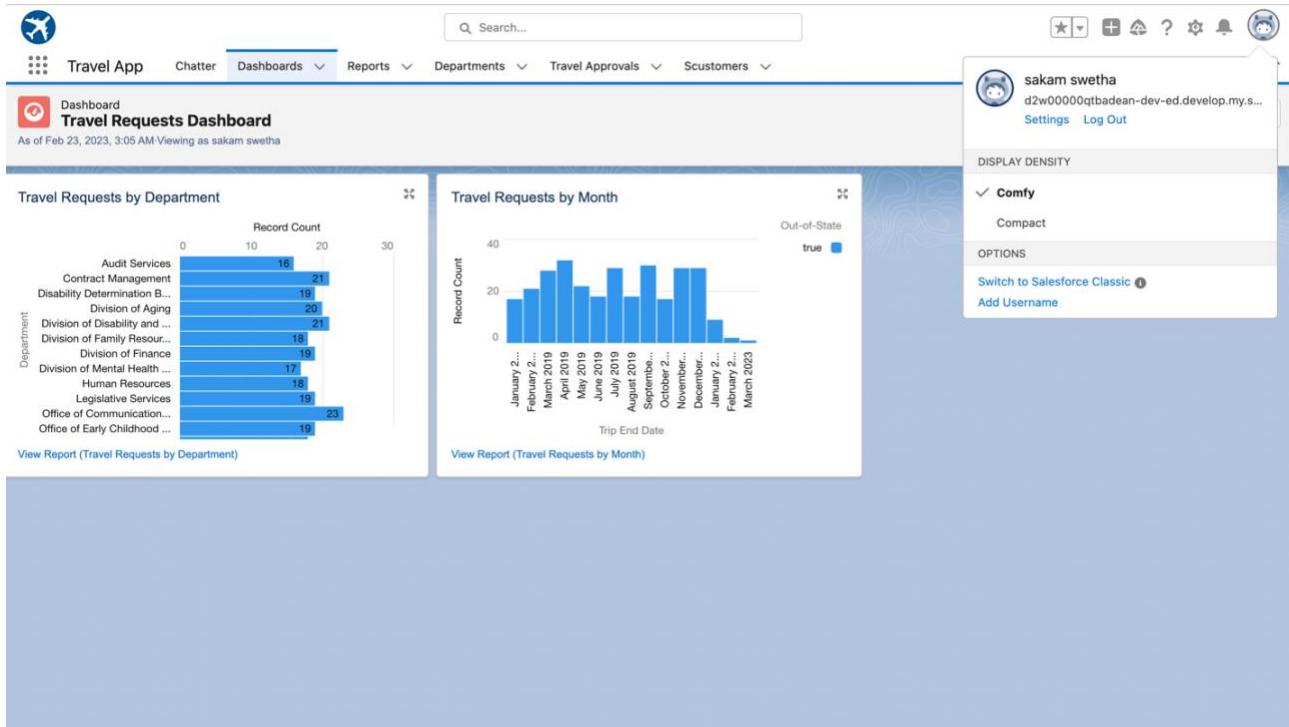
## Final Report

The screenshot shows the Salesforce interface with the 'Travel App' tab selected. The page title is 'Report: Travel Approvals - Travel Requests by Month'. The report displays 302 total records across three months: January 2020 (9), February 2023 (2), and March 2023 (1). The columns include Trip End Date, Out-of-State, Travel Approval: Travel Approval #, Department, Status, Destination State, and Trip Start Date. The data shows various travel approvals from different departments like Audit Services, Contract Management, and Technology, with statuses ranging from Approved to Draft.

Trip End Date	Out-of-State	Travel Approval: Travel Approval #	Department	Status	Destination State	Trip Start Date
Subtotal						
January 2020 (9)	(9)	TA-00769	Audit Services	Approved	OK	12/27/2019
		TA-00707	Contract Management	Approved	TX	12/22/2019
		TA-00931	Disability Determination Bureau	Approved	CA	12/30/2019
		TA-00981	Human Resources	Approved	FL	12/29/2019
		TA-00864	Office of Communications and Media	Rejected	CA	12/22/2019
		TA-00855	Office of General Counsel	Rejected	TX	12/23/2019
		TA-00837	Division of Aging	Rejected	GA	12/24/2019
		TA-00967	Division of Mental Health and Addiction	Approved	CA	12/13/2019
		TA-00954	Division of Mental Health and Addiction	Approved	GA	12/24/2019
Subtotal						
Subtotal						
Subtotal						
February 2023 (2)	(2)	TA-00500	Contract Management	New	jk	2/22/2023
		TA-00001	Technology	Approved	CA	2/6/2023
Subtotal						
March 2023 (1)	(1)	TA-00501	Technology	Draft	CA	2/21/2023
Subtotal						
Total (302)						

## STEP 4:

### Final Dashboard



# MODULE 3

## Exercise 1:- STEP 1:

### Creating Code Playground Lightning App

The screenshot shows the Salesforce Code Playground Lightning App interface. At the top, there's a navigation bar with tabs for Leads, Accounts, Opportunities, Contacts, and Cases. Below the navigation is a search bar labeled "Search...". The main area displays a table titled "All Open Leads" with 15 items. The columns include Name, Company, Status, Email, Lead Status, and Created Date. The table is sorted by Name. A context menu is open on the right side of the screen, showing options like "New", "Import", "Add to Case", "Switch to Salesforce Classic", and "Add Username". The user profile at the top right is "sakam swetha" with the ID "d2w00000qtbadean-dev-ed.develop.my.s...".

### Creating Custom Object : Customer

The screenshot shows the Salesforce Setup page for creating a new object. The left sidebar lists various setup categories: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules, Triggers, and Flow Triggers. The main content area shows the "Customer" object details. It includes fields for Description, API Name (set to "Customer\_\_c"), Singular Label ("Customer"), and Plural Label ("Customers"). Other settings shown include Enable Reports (checked), Track Activities, Track Field History, Deployment Status (set to "Deployed"), and Help Settings (set to "Standard salesforce.com Help Window"). A context menu is open on the right, identical to the one in the previous screenshot, showing options like "New", "Import", "Add to Case", "Switch to Salesforce Classic", and "Add Username". The user profile at the top right is "sakam swetha" with the ID "d2w00000qtbadean-dev-ed.develop.my.s...".

## Creating Custom Fields on Customer Object

The screenshot shows the Salesforce Setup interface for the Customer object. The left sidebar lists various setup categories like Details, Fields & Relationships, Page Layouts, etc. The main content area is titled 'Fields & Relationships' and displays a table of fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Active	Active__c	Checkbox		
Created By	CreatedBy	Lookup(User)		
Customer	Customer__c	Master-Detail(Customer)		
Customer Name	Name	Text(80)		
Customer Type	Customer_Type__c	Picklist		
Description	Description__c	Text Area(255)		
Last Modified By	LastModifiedBy	Lookup(User)		

## Creating Custom Object : Billing

The screenshot shows the Salesforce Setup interface for the Billing custom object. The left sidebar lists various setup categories. The main content area is titled 'Details' and shows the following configuration:

API Name	Description
Billing__c	Description

On the right side, there is a user profile sidebar and a 'DISPLAY DENSITY' dropdown menu set to 'Comfy'. Other options in the menu include 'Compact', 'OPTIONS', 'Switch to Salesforce Classic', and 'Add Username'.

# Creating Custom Fields on Billing Object

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

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING
Amount Paid	Amount_Paid__c	Currency(18, 0)	
Billing Number	Name	Auto Number	
Created By	CreatedById	Lookup(User)	
Customer Type	Customer_Type__c	Picklist	
Last Modified By	LastModifiedById	Lookup(User)	
Owner	OwnerId	Lookup(User,Group)	✓
Status	Status__c	Picklist	

The right side of the screen shows the user profile for sakam swetha and various display density and options settings.

# Testing the App

The screenshot shows the Salesforce Code Playground interface with the Accounts tab selected. The top navigation bar includes Leads, Accounts, Opportunities, Contacts, Cases, Customers, and Billings. The main area displays a table titled "All Accounts" with 13 items. The columns are Account Name, Account Site, Billing State/Province, Phone, and Type. The data includes:

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

The right side of the screen shows the user profile for sakam swetha and various display density and options settings.

## Exercise 2

### 1. Define a String Variable & use string method ‘endsWith’ to display the output.

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 the 'Execution Log' tab, which is active. It displays a log entry for 'Log executeAnonymous @21/02/2023, 11:55:29' with the message '[1]DEBUG/true'. Underneath the log is a table with columns: Timestamp, Event, and Details. The first row shows '11:56:45:004' and 'USER\_DEBUG [2]DEBUG/true'. Below the log is the 'Enter Apex Code' window, which contains the following code:

```
1 String name='sakam swetha';
2 System.debug(name.endsWith('a'));
```

At the bottom of the interface, there's a toolbar with buttons for This Frame, Executable, Debug Only, Filter, Click Here, Open Log, Execute, and Execute Highlighted. Below the toolbar is a 'Logs' tab in a navigation bar, followed by Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The main area shows a table of logs with columns: User, Application, Operation, Time, Status, Read, and Size. The logs list five entries from 'sakam swetha' with various URLs and success status.

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

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 the 'Execution Log' tab, which is active. It displays a log entry for 'Log executeAnonymous @21/02/2023, 11:55:29' with the message '[2]DEBUG/2023-02-20 00:00:00' and '[4]DEBUG/2023-03-22 00:00:00'. Underneath the log is a table with columns: Timestamp, Event, and Details. The first two rows show '11:55:29:002' and 'USER\_DEBUG [2]DEBUG/2023-02-20 00:00:00'. The last two rows show '11:55:29:002' and 'USER\_DEBUG [4]DEBUG/2023-03-22 00:00:00'. Below the log is the 'Enter Apex Code' window, which contains the following code:

```
1 Date day1=Date.today();
2 System.debug(day1);
3 Date day2=day1.addDays(30);
4 System.debug(day2);
```

At the bottom of the interface, there's a toolbar with buttons for This Frame, Executable, Debug Only, Filter, Click Here, Open Log, Execute, and Execute Highlighted. Below the toolbar is a 'Logs' tab in a navigation bar, followed by Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The main area shows a table of logs with columns: User, Application, Operation, Time, Status, Read, and Size. The logs list four entries from 'sakam swetha' with various URLs and success status.

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

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 toolbar with icons for New, Open, Save, and others. A title bar says "Log executeAnonymous @21/02/2023, 12:00:12".

The main area has two tabs: "Execution Log" and "Logs". The "Execution Log" tab shows the following entries:

Timestamp	Event	Details
12:00:12:002	USER_DEBUG	[3]  DEBUG 10
12:00:12:002	USER_DEBUG	[4]  DEBUG 10
12:00:12:003	USER_DEBUG	[5]  DEBUG 30

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

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

At the bottom of the interface are several buttons: This Frame, Executable, Debug Only (which is checked), Filter, Click here, Open Log, Execute, and Execute Highlighted.

The "Logs" tab displays a table of logs:

User	Application	Operation	Time	Status	Read	Size
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:00:12	Success		2.54 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:59:30	Success		2.52 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:56:45	Success		2.14 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:55:29	Success		2.52 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:53:06	Success		2.14 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:52:35	Success		2.14 KB

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

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 toolbar with icons for New, Open, Save, and others. A title bar says "Log executeAnonymous @21/02/2023, 12:00:12" and "Log executeAnonymous @21/02/2023, 12:01:40".

The main area has two tabs: "Execution Log" and "Logs". The "Execution Log" tab shows the following entry:

Timestamp	Event	Details
12:01:40:003	USER_DEBUG	[2]  DEBUG Length:16

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

```
1 String str='length of string';
2 System.debug('Length:'+str.length());
```

At the bottom of the interface are several buttons: This Frame, Executable, Debug Only (which is checked), Filter, Click here, Open Log, Execute, and Execute Highlighted.

The "Logs" tab displays a table of logs:

User	Application	Operation	Time	Status	Read	Size
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:01:40	Success		2.25 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:00:12	Success		2.54 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:59:30	Success		2.52 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:56:45	Success		2.14 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:55:29	Success		2.52 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:53:06	Success		2.14 KB

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

The screenshot shows the Salesforce IDE interface. At the top, there's a menu bar with File, Edit, Debug, Test, Workspace, Help, and a back arrow. Below the menu is a toolbar with Log executeAnonymous @21/02/2023, 12:20:34. The main area has tabs for Execution Log, Enter Apex Code, Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Execution Log tab is active, showing a table with columns: Timestamp, Event, and Details. The details column contains debug logs: [6] DEBUG Using ADD: (1, 2, 3, 4), [7] DEBUG Using GET method: 1, [9] DEBUG Using SET method: (1, 2, 5, 4), and [11] DEBUG Using CLEAR method: (). The Enter Apex Code tab shows the following code:

```

1 List<Integer> l=new List<Integer>();
2 l.add(1);
3 l.add(2);
4 l.add(3);
5 l.add(4);
6 System.debug('Using ADD: '+l);
7 System.debug('Using GET method: '+l.get(0));
8 l.set(2,5);
9 System.debug('Using SET method: '+l);
10 l.clear();
11 System.debug('Using CLEAR method: '+l);

```

Below the code editor is a toolbar with checkboxes for This Frame, Executable, Debug Only, Filter, Click, Open Log, Execute, and Execute Highlighted. The Logs tab is selected in the navigation bar, showing a table of log entries from a user named sakam swetha. The table includes columns for User, Application, Operation, Time, Status, Read, and Size. All entries show Success status and various file sizes.

User	Application	Operation	Time	Status	Read	Size
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:20:34	Success		4.15 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:07:37	Success		4.14 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:01:40	Success		2.25 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:00:12	Success		2.54 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:59:30	Success		2.52 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 11:56:45	Success		2.14 KB

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

The screenshot shows the Salesforce IDE interface. At the top, there's a menu bar with File, Edit, Debug, Test, Workspace, Help, and a back arrow. Below the menu is a toolbar with Log executeAnonymous @21/02/2023, 12:23:16. The main area has tabs for Execution Log, Enter Apex Code, Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Execution Log tab is active, showing a table with columns: Timestamp, Event, and Details. The details column contains debug logs for each iteration of a loop: [4] DEBUG Value of x is: 0, [4] DEBUG Value of x is: 1, [4] DEBUG Value of x is: 2, [4] DEBUG Value of x is: 3, [4] DEBUG Value of x is: 4, [4] DEBUG Value of x is: 5, [4] DEBUG Value of x is: 6, [4] DEBUG Value of x is: 7, [4] DEBUG Value of x is: 8, and [4] DEBUG Value of x is: 9. The Enter Apex Code tab shows the following code:

```

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

```

Below the code editor is a toolbar with checkboxes for This Frame, Executable, Debug Only, Filter, Click, Open Log, Execute, and Execute Highlighted. The Logs tab is selected in the navigation bar, showing a table of log entries from a user named sakam swetha. The table includes columns for User, Application, Operation, Time, Status, Read, and Size. All entries show Success status and various file sizes.

User	Application	Operation	Time	Status	Read	Size
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:23:38	Success		5.61 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:23:16	Success		2.01 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:20:34	Success		4.15 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:07:37	Success		4.14 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:01:40	Success		2.25 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:00:12	Success		2.54 KB

## Exercise 3

Integer myluckyNumber = 15; Integer myunluckyNumber = 7;  
myluckyNumber != myunluckyNumber + 8

The screenshot shows the Salesforce IDE interface. At the top, there's a navigation bar with File, Edit, Debug, Test, Workspace, Help, and a timestamp Log executeAnonymous @21/02/2023, 12:25:52. Below it is an 'Execution Log' table:

Timestamp	Event	Details
12:25:52:005	USER_DEBUG	[4]DEBUG false

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

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

At the bottom of the interface is a 'Logs' tabbed panel with tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The 'Logs' tab is selected, showing a list of operations:

User	Application	Operation	Time	Status	Read	Size
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:25:52	Success		2.62 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:23:38	Success		5.61 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:23:16	Success		2.01 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:20:34	Success		4.15 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:07:37	Success		4.14 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:01:40	Success		2.25 KB

## Exercise 4

Answer the following in True Or False:

isTrue || isFalse Boolean isFalse = false; Boolean isTrue = True;

The screenshot shows the Salesforce IDE interface. At the top, there's a navigation bar with File, Edit, Debug, Test, Workspace, Help, and a timestamp Log executeAnonymous @21/02/2023, 12:25:52 and Log executeAnonymous @21/02/2023, 12:28:14. Below it is an 'Execution Log' table:

Timestamp	Event	Details
12:28:14:004	USER_DEBUG	[3]DEBUG true

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

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

At the bottom of the interface is a 'Logs' tabbed panel with tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The 'Logs' tab is selected, showing a list of operations:

User	Application	Operation	Time	Status	Read	Size
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:28:14	Success		2.35 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:25:52	Success		2.62 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:23:38	Success		5.61 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:23:16	Success		2.01 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:20:34	Success		4.15 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:07:37	Success		4.14 KB

## Exercise 5

today != tomorrow

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

Date today = Date.today();

The screenshot shows the Salesforce IDE interface. At the top is the navigation bar with File, Edit, Debug, Test, Workspace, and Help. Below it is the 'Log executeAnonymous @21/02/2023, 12:29:00' section of the Execution Log. The log table has columns for Timestamp, Event, and Details. A single entry is shown: 12:29:00:002 USER\_DEBUG [4] DEBUG|true. Below the log is the 'Enter Apex Code' window containing the following code:

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

At the bottom of the screen is the 'Logs' tab of the execution log, which lists several successful operations by user 'sakam swetha' with various file paths and sizes.

## Exercise 6

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

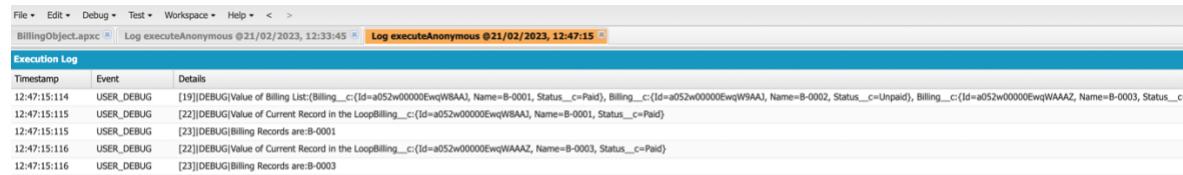
The screenshot shows the Salesforce IDE interface. At the top is the navigation bar with File, Edit, Debug, Test, Workspace, and Help. Below it is the 'Log executeAnonymous @21/02/2023, 12:33:45' section of the Execution Log. The log table has columns for Timestamp, Event, and Details. A single entry is shown: 12:33:45:003 USER\_DEBUG [9] DEBUG|Grade: B. Below the log is the 'Enter Apex Code' window containing the following code:

```
1 Integer Score=80;
2 if(Score==100){
3     System.debug('Grade: A+');
4 }
5 else if (Score>=90){
6     System.debug ('Grade: A');
7 }
8 else if(Score>=80){
9     System.debug ('Grade: B');
10}
11 else{
12     System.debug ('Grade: Failed');
13 }
```

At the bottom of the screen is the 'Logs' tab of the execution log, which lists several successful operations by user 'sakam swetha' with various file paths and sizes.

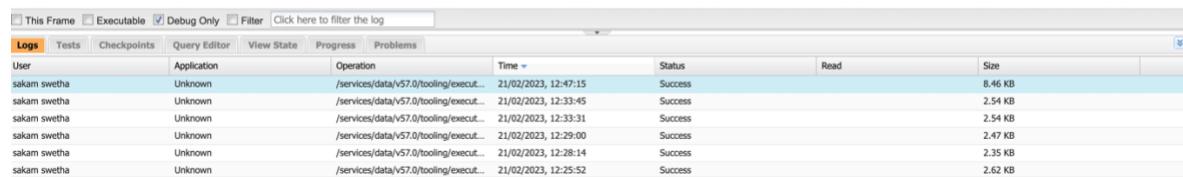
## Exercise 7

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



The screenshot shows the Salesforce IDE's Execution Log window. The title bar indicates the file is BillingObject.apxc and the log entry is "Log executeAnonymous @21/02/2023, 12:47:15". The log table has columns for Timestamp, Event, and Details. The details column contains several DEBUG logs related to a Billing list and current records.

Timestamp	Event	Details
12:47:15:114	USER_DEBUG	[19]DEBUGValue of Billing List:(Billing__c:{Id=a052w00000EwqW8AAJ, Name=B-0001, Status__c=Paid}, Billing__c:{Id=a052w00000EwqW9AAJ, Name=B-0002, Status__c=Unpaid}, Billing__c:{Id=a052w00000EwqWAQAZ, Name=B-0003, Status__c=Unpaid})
12:47:15:115	USER_DEBUG	[22]DEBUGValue of Current Record in the LoopBilling__c:{Id=a052w00000EwqW8AAJ, Name=B-0001, Status__c=Paid}
12:47:15:115	USER_DEBUG	[23]DEBUGBilling Records are:B-0001
12:47:15:116	USER_DEBUG	[22]DEBUGValue of Current Record in the LoopBilling__c:{Id=a052w00000EwqWAQAZ, Name=B-0003, Status__c=Paid}
12:47:15:116	USER_DEBUG	[23]DEBUGBilling Records are:B-0003

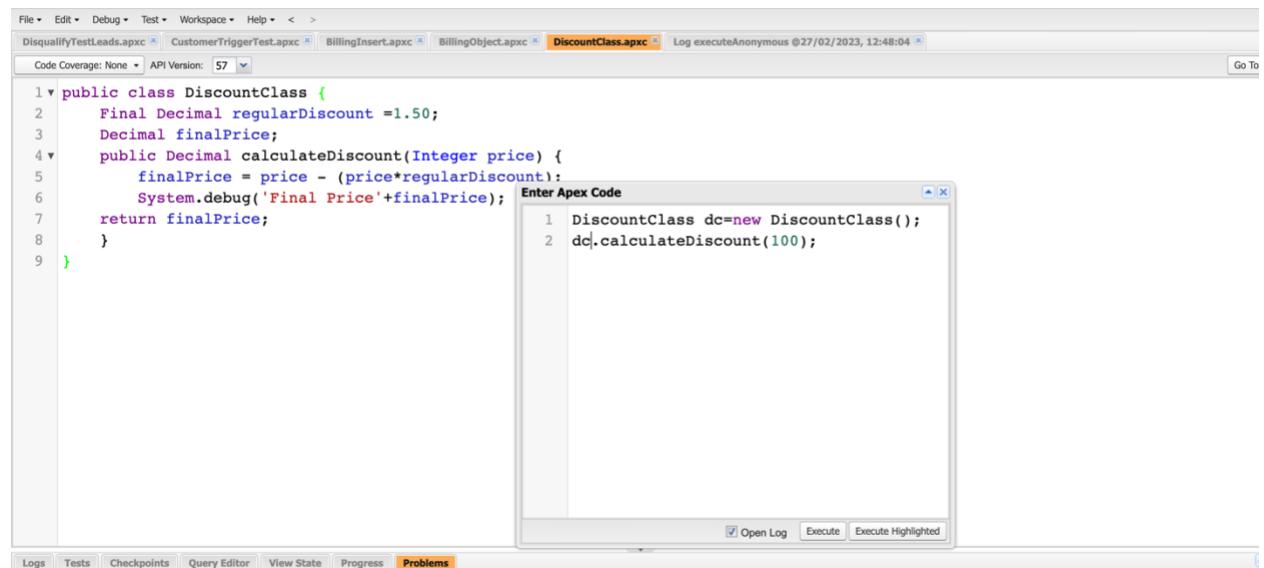


The screenshot shows the Salesforce IDE's Log tab. The title bar indicates the file is BillingObject.apxc and the log entry is "Log executeAnonymous @21/02/2023, 12:47:15". The log table has columns for User, Application, Operation, Time, Status, Read, and Size. The log entries show multiple successful executions of the /services/data/v57.0/tooling/execut... endpoint by a user named "sakam swetha".

User	Application	Operation	Time	Status	Read	Size
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:47:15	Success		8.46 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:33:45	Success		2.54 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:33:31	Success		2.54 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:29:00	Success		2.47 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:28:14	Success		2.35 KB
sakam swetha	Unknown	/services/data/v57.0/tooling/execut...	21/02/2023, 12:25:52	Success		2.62 KB

## Exercise 8

Write a Class to demo the use of Constants in Apex



The screenshot shows the Salesforce IDE with an Apex class named DiscountClass. The class has a static final decimal regularDiscount set to 1.50. It also has a calculateDiscount method that subtracts regularDiscount from the input price. An "Enter Apex Code" dialog box is open, showing a sample call to the constructor and the calculateDiscount method with a value of 100.

```
1 v public class DiscountClass {
2   Final Decimal regularDiscount =1.50;
3   Decimal finalPrice;
4 v   public Decimal calculateDiscount(Integer price) {
5     finalPrice = price - (price*regularDiscount);
6     System.debug('Final Price'+finalPrice);
7     return finalPrice;
8   }
9 }
```

Enter Apex Code

```
1 DiscountClass dc=new DiscountClass();
2 dc.calculateDiscount(100);
```

Open Log Execute Execute Highlighted

## Exercise 9

### Write a Class to demo the use of Interface in Apex

The screenshot shows the Salesforce IDE interface. At the top, there are tabs for 'File', 'Edit', 'Debug', 'Test', 'Workspace', and 'Help'. Below the tabs, the 'Execution Log' window is open, showing a log entry for 'executeAnonymous @21/02/2023, 12:59:42'. The log details show a timestamp of '12:59:42:025', an event of 'USER\_DEBUG', and a message '[3]DEBUG|0.1'. Below the log is the 'Enter Apex Code' window, which contains the following code:

```
1 normalCustomer nc=new normalCustomer();
2 Double d=nc.percentageDiscountToBeApplied();
3 System.debug(d);
```

At the bottom of the interface, there are several tabs: 'Logs', 'Tests', 'Checkpoints', 'Query Editor', 'View State', 'Name', 'Line', and 'Problem'. There are also buttons for 'Open Log', 'Execute', and 'Execute Highlighted'.

## Exercise 10

### Demo on DML Insert Operation Using Database methods

The screenshot shows the Salesforce IDE interface. At the top, there are tabs for 'File', 'Edit', 'Debug', 'Test', 'Workspace', and 'Help'. Below the tabs, the 'Execution Log' window is open, showing a log entry for 'executeAnonymous @22/02/2023, 10:59:55'. The log details show a timestamp of '10:59:55:071', an event of 'USER\_DEBUG', and a message '[16]DEBUG|Record inserted. Billing ID: a052w00000EwsFBAAJ'. Below the log is the 'Enter Apex Code' window, which contains the following code:

```
1 BillingInsert.recordInsert();
```

At the bottom of the interface, there are several tabs: 'Logs', 'Tests', 'Checkpoints', 'Query Editor', 'View State', 'Progress', and 'Problems'. There are also buttons for 'Open Log', 'Execute', and 'Execute Highlighted'.

## Exercise 11

### Write and execute SOQL queries from Developer Console

The screenshot shows the Salesforce Developer Console interface. At the top, there's a navigation bar with links like File, Edit, Debug, Test, Workspace, Help, and a back/forward button. Below the navigation is a toolbar with a dropdown for 'Opportunity' (set to 'Opportunity @ 2:43 PM') and a search bar. The main area contains a query editor window with the following SOQL query:

```
SELECT AccountId, Amount, StageName, Account.Industry, Account.website From Opportunity Where Account.Industry='Energy' AND Account.AnnualRevenue>5000
```

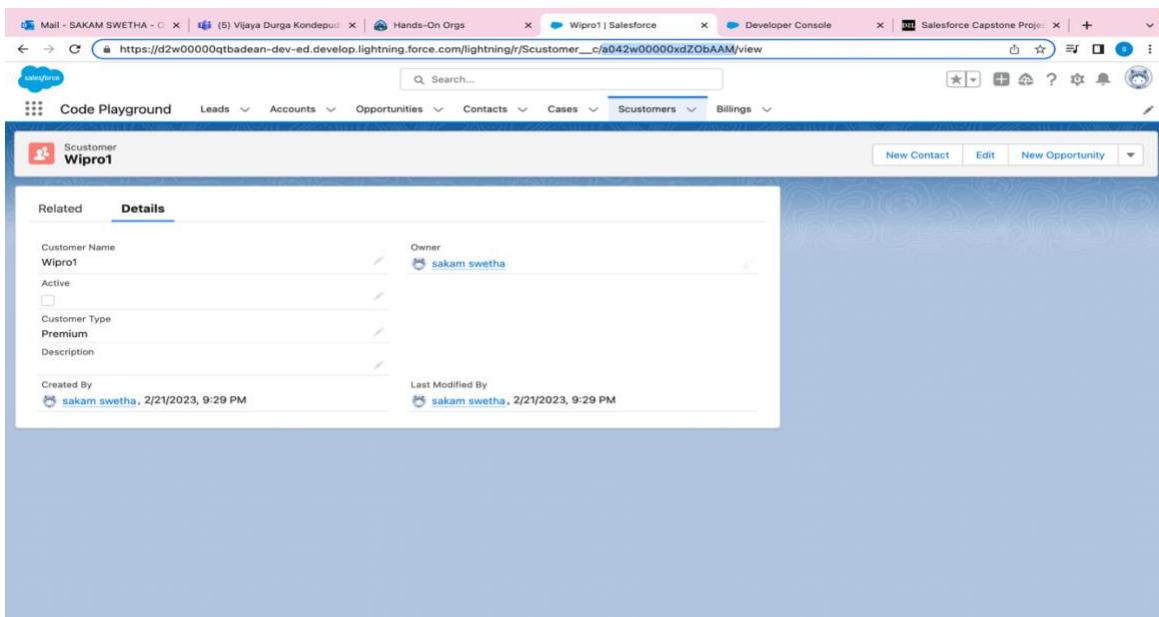
Below the query is a 'Query Results - Total Rows: 10' table with the following data:

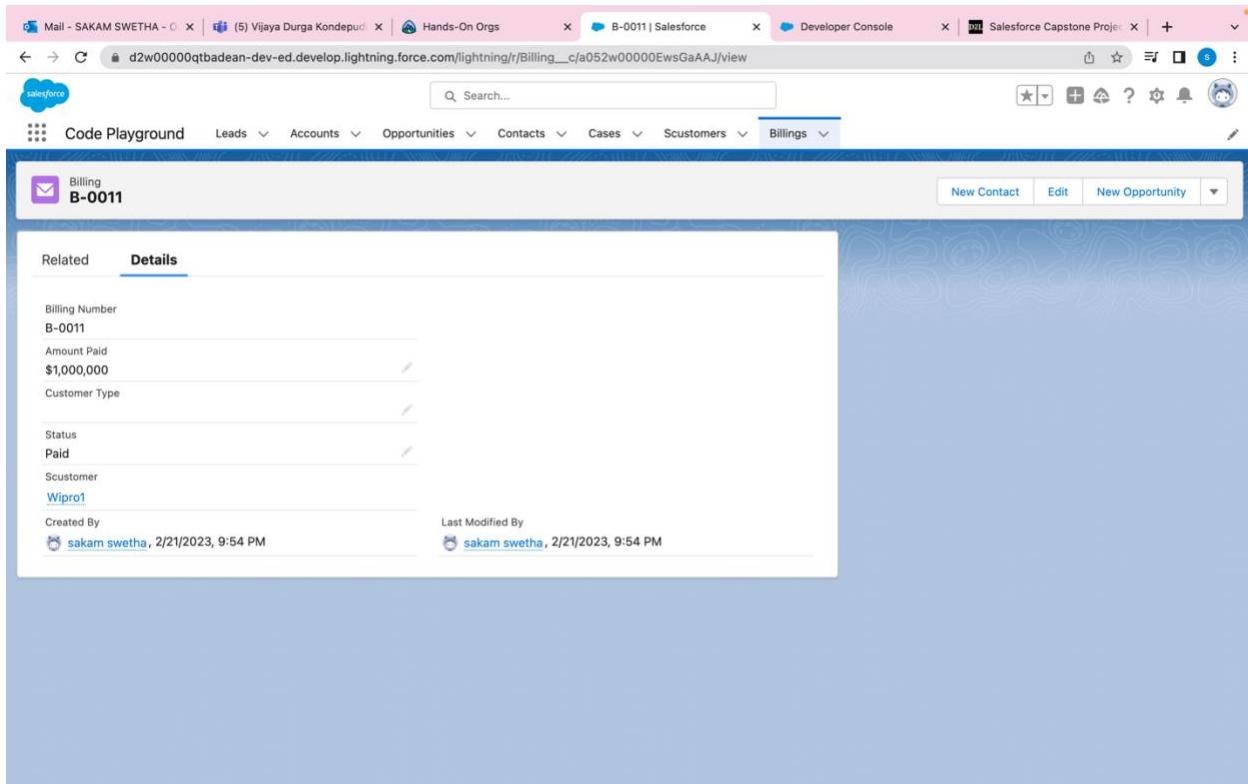
AccountId	Amount	StageName	Account.Industry	Account.Website
0012w00001jxka8AB	125000	Negotiation/Review	Energy	http://www.uos.com
0012w00001jxka8AB	270000	Proposal/Price Quote	Energy	http://www.uos.com
0012w00001jxka8AB	120000	Closed Won	Energy	http://www.uos.com
0012w00001jxka8AB	270000	Negotiation/Review	Energy	http://www.uos.com
0012w00001jxka8AB	270000	Closed Won	Energy	http://www.uos.com
0012w00001jxka8AB	915000	Closed Won	Energy	http://www.uos.com
0012w00001jxka8AB	440000	Closed Won	Energy	http://www.uos.com
0012w00001jxka8AB	235000	Closed Won	Energy	http://www.uos.com
0012w00001jxka8AB	120000	Closed Won	Energy	http://www.uos.com
0012w00001jxka8AB	675000	Needs Analysis	Energy	http://www.uos.com

At the bottom of the query editor, there are buttons for 'Save Rows', 'Insert Row', 'Delete Row', and 'Refresh Grid'. To the right, there are buttons for 'Create New', 'Open Detail Page', and 'Edit Page'. Below the query editor, tabs include Logs, Tests, Checkpoints, Query Editor (which is selected), View State, Progress, and Problems. A 'History' section on the right shows the query was executed. The status bar at the bottom left shows 'Execute' and 'Use Tooling API'.

## Exercise 12

### Write an Apex Trigger, Name = CustomerTrigger.





## Exercise 13

### Write a Test Class for CustomerTrigger

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

CustomerTriggerTest.apxc ▪ BillingInsert.apxc

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

```

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

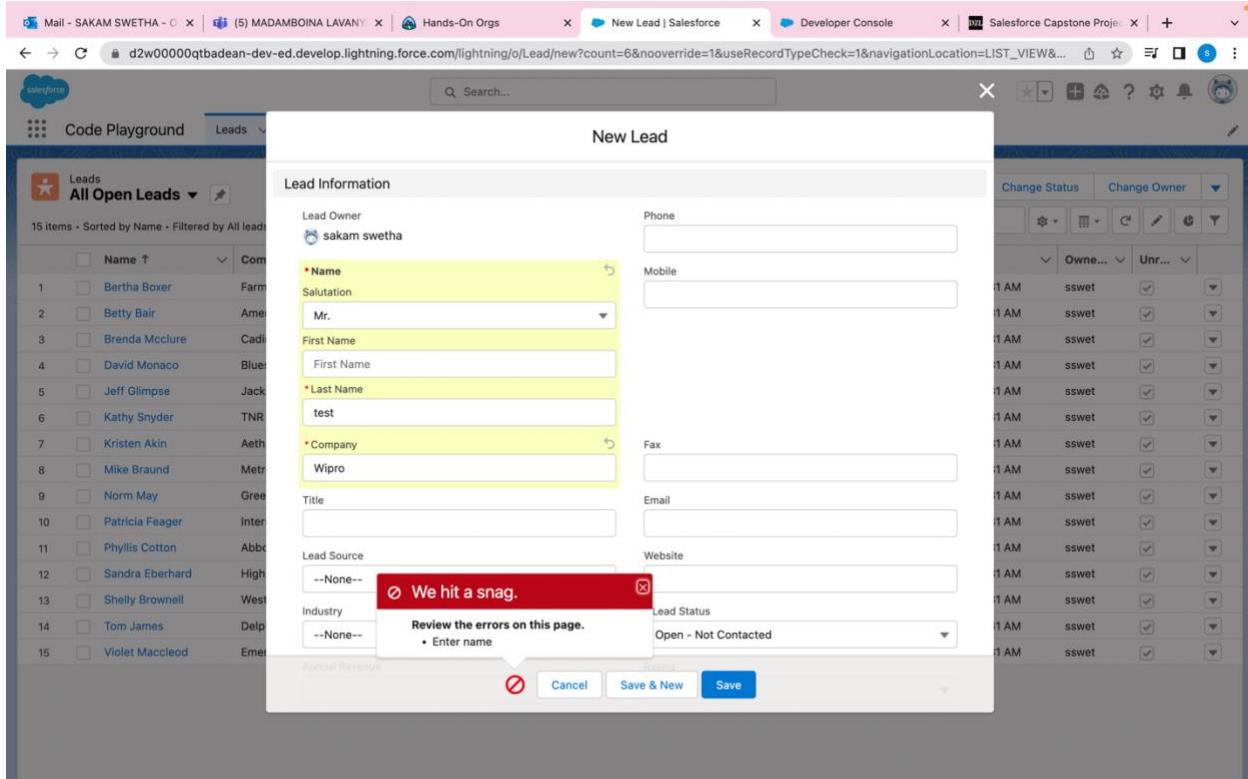
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Status	Test Run	Enqueued Time	Duration	Failures	Total	Overall Code Coverage																		
✓	TestRun @ 2:23:58 pm			0	1	<table border="1"> <thead> <tr> <th>Class</th> <th>Percent</th> <th>Lines</th> </tr> </thead> <tbody> <tr> <td>BillingInsert</td> <td>0%</td> <td>0/14</td> </tr> <tr> <td>BillingObject</td> <td>0%</td> <td>0/20</td> </tr> <tr> <td>CustomerTrigger</td> <td>100%</td> <td>9/9</td> </tr> <tr> <td>DiscountClass</td> <td>0%</td> <td>0/4</td> </tr> <tr> <td>DisqualifyTestLeads</td> <td>0%</td> <td>0/8</td> </tr> </tbody> </table>	Class	Percent	Lines	BillingInsert	0%	0/14	BillingObject	0%	0/20	CustomerTrigger	100%	9/9	DiscountClass	0%	0/4	DisqualifyTestLeads	0%	0/8
Class	Percent	Lines																						
BillingInsert	0%	0/14																						
BillingObject	0%	0/20																						
CustomerTrigger	100%	9/9																						
DiscountClass	0%	0/4																						
DisqualifyTestLeads	0%	0/8																						

## Exercise 14

### Write an Apex Trigger, Name = DisqualifyTestLeads.



## Exercise 15

### Write a Test Class for DisqualifyTestLeads.

```
File • Edit • Debug • Test • Workspace • Help • < >
BillingInsert.apxc • DisqualifyTestLeads.apxc • DisqualifyTestLeadsTest.apxc TestRun @ 12:33:13 pm Log executeAnonymous @22/02/2023, 12:33:14
Code Coverage: None • API Version: 57 • Run Test Go To
```

```
1  @isTest
2  public class DisqualifyTestLeadsTest {
3      @isTest
4      private static void insertData(){
5          List<Lead> leadlist = new List<Lead>();
6          Lead l = new Lead();
7          l.FirstName ='Test';
8          l.LastName ='Test';
9          l.Company ='Test';
10         leadlist.add(l);
11         Test.startTest();
12         insert leadlist;
13         Test.stopTest();
14         Lead updateLead =[Select FirstName,LastName,Company From Lead order by CreatedDate];
15         system.Assert(updateLead.LastName!= Null,'Record cannot be created');
16         Database.SaveResult res = Database.insert(l,false);
17         System.assertEquals('Will be Disqualified', res.getErrors()[0].getMessage());
18     }
19 }
20 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
sakam swetha	Unknown	/services/data/v57.0/tooling/runTest...	22/02/2023, 12:33:14	Insert failed. First exception on row ...	Unread	15.29 KB
sakam swetha	Browser	/aura	22/02/2023, 12:05:11	Success	Unread	4.61 KB
sakam swetha	Browser	/aura	22/02/2023, 12:04:58	Success	Unread	1.27 KB
sakam swetha	Browser	/aura	22/02/2023, 12:04:57	Success	Unread	4.61 KB
sakam swetha	Browser	/aura	22/02/2023, 12:04:52	Success	Unread	1.27 KB
sakam swetha	Browser	/aura	22/02/2023, 12:04:45	Success	Unread	1.27 KB

Filter Click here to filter the log list

## Exercise 16

Create a Visualforce page which displays Opportunity fields as output fields.



## Exercise 17

Create a Visualforce page which shows a list of Accounts linked to their record pages



## Exercise 18

Create a Visualforce page that uses a custom controller to display a list of cases with the status of 'New'. The page must be named NewCaseList.



## 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](#)