

# A CRM Application For Public Transport Management

## **Introduction: -**

The Public Transport (RTC - Regional Transport corporation) Management System is a comprehensive Salesforce application designed to streamline and manage various operational aspects of the Public Transport. This system will enable the Transport department to efficiently maintain details of employees, their roles, bus stations, buses, ticket fares, daily bus trips, passenger counts, and the total ticket fare amount. By leveraging Salesforce's robust platform, the Transport Department can improve operational efficiency, data accuracy, and reporting capabilities.

## **Project Overview:**

This project is to Consolidate all Transport-related data into a single Salesforce application to ensure easy access, management, and reporting. Enable real-time insights and reports on various operational metrics such as passenger count and revenue. Maintain detailed records of all Transport employees, including personal details, contact information. Define and assign roles and responsibilities to each employee. Schedule and manage employee shifts, especially for drivers and conductors. Maintain information on all bus stations, including location, facilities. Maintain detailed records of all buses, including model, capacity. Manage bus schedules and assign buses to specific routes and trips. Define and manage ticket fares for different routes and bus types. Track daily trips for each bus, including start and end times, routes, and driver details. Record the number of passengers on each trip. Calculate and track the total ticket fare amount collected from each trip. Provide real-time dashboards for quick insights into key metrics such as passenger count, trip efficiency, and revenue.

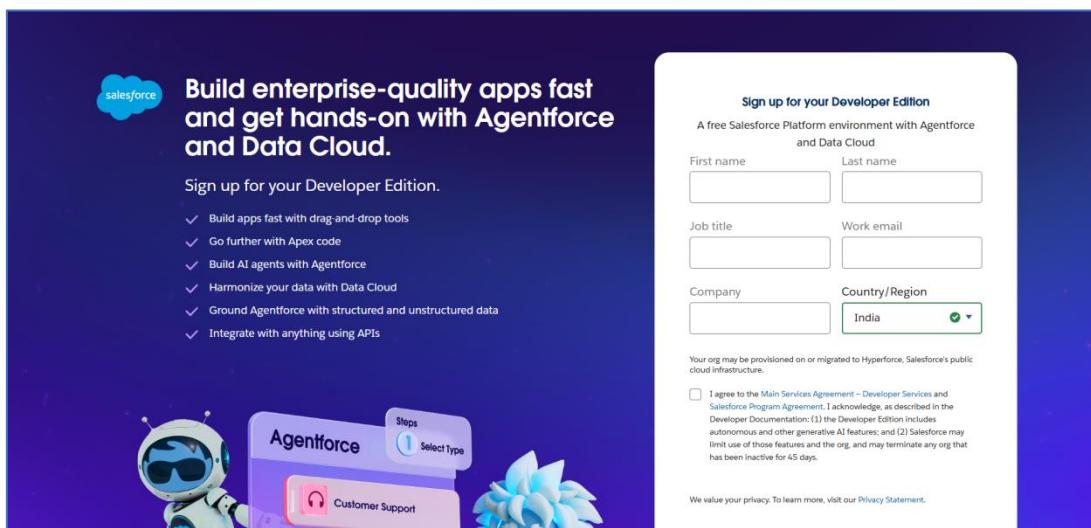
# A CRM Application For Public Transport Management

## Milestone 1 Salesforce

### Activity 1: Creating Developer Account

Creating a developer org in Salesforce

1. Go to <https://developer.salesforce.com/signup>
2. On the sign up form, enter the following details :

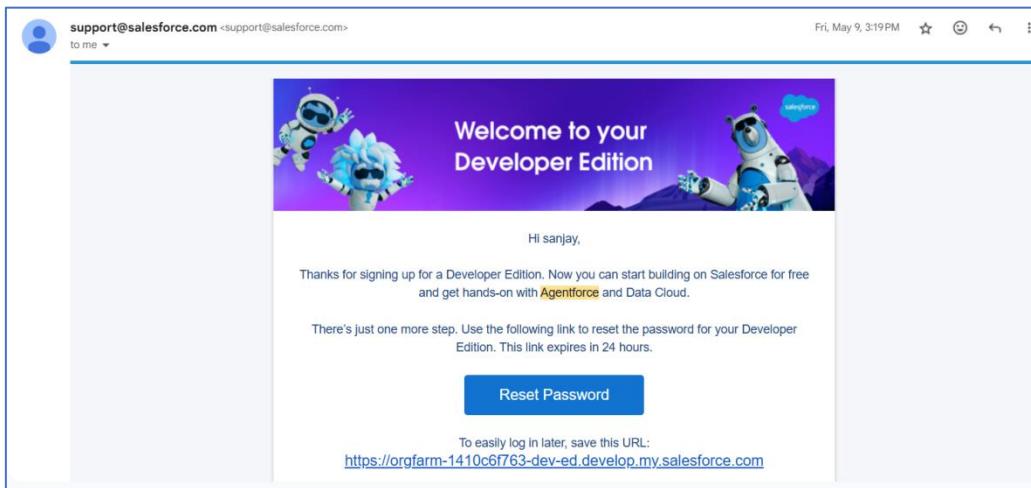


1. First name – Vaishnavi
2. Last name - Chinthamakula
3. Email – vaishnavic230404@gmail.com
4. Role : Developer
5. Company : Gayatri Degree College Tirupati
6. County : India

# A CRM Application For Public Transport Management

## Activity 2: Account Activation

1. Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.

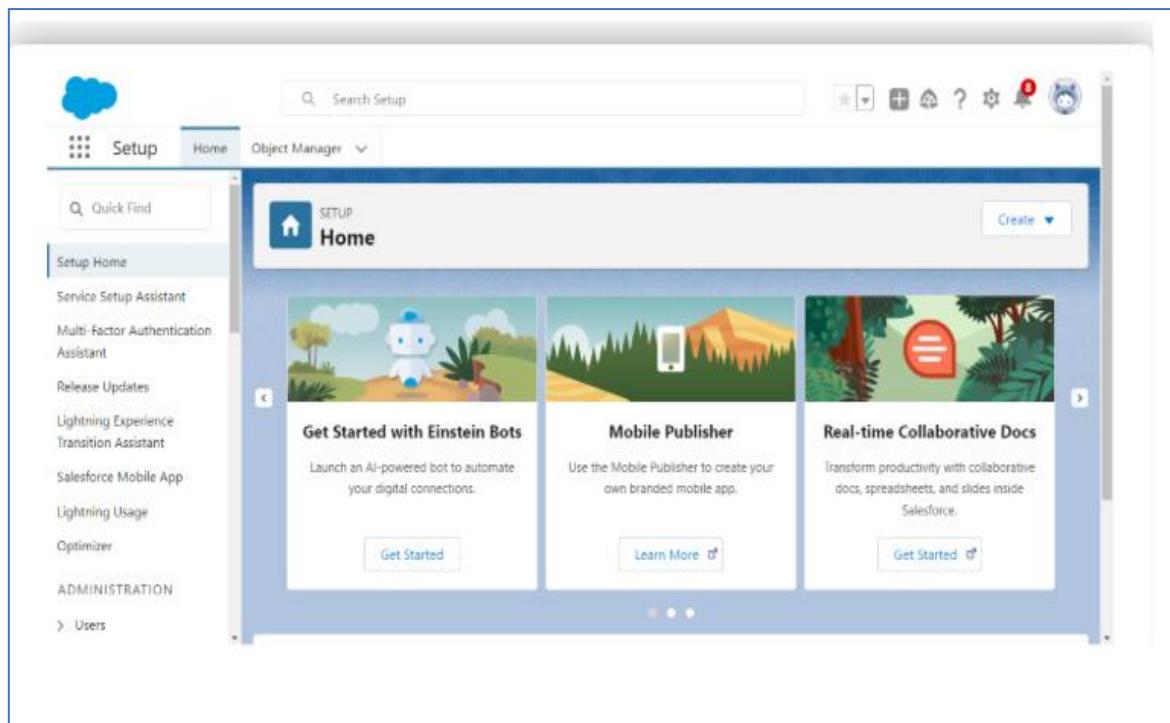


2. Click on Verify Account
3. Give a password and answer a security question and click on change password.

A screenshot of the "Change Your Password" page. It asks for a new password and a confirmation. Below that, it asks for a security question and an answer. A red box highlights the "New Password", "Confirm New Password", "Answer", and "Change Password" fields.

4. Then you will redirect to your Salesforce setup page.

# A CRM Application For Public Transport Management



# A CRM Application For Public Transport Management

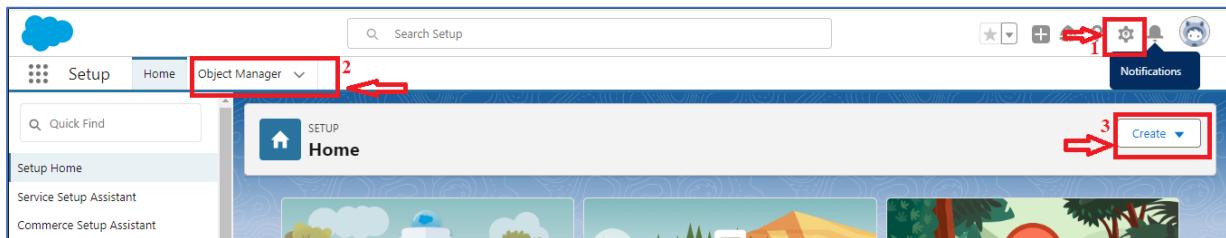
## Milestone 2 Objects

### Activity 1: Creating a Bus Station Object

The purpose of creating a Bus Station custom object is to store and manage information about Bus Stops.

To create an object:

1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.



4. Enter the label name as Bus Station
5. Enter Plural label name as Bus Stations
6. Enter Record Name as Bus Station Name

This screenshot shows the 'Custom Object Definition Edit' screen. It has a header with 'SETUP' and 'New Custom Object'. Below the header is a 'Custom Object Information' section. It includes fields for 'Label' (containing 'Bus Station') and 'Plural Label' (containing 'Bus stations'), both of which are highlighted with red boxes and have red arrows pointing to them with the numbers '4' and '5' respectively. There's also a note about starting with a vowel sound and an 'Object Name' field containing 'Bus\_Station'. The bottom of the screen shows 'Context-Sensitive Help Setting' (radio buttons for standard help or Visualforce page) and a 'Content Name' dropdown set to 'None'. A watermark 'Activate Windows' is visible in the bottom right corner.

# A CRM Application For Public Transport Management

7. Select Data Type as Text.
8. Select Allow reports.

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name".

6 Record Name Bus Station Name Example: Account Name

7 Data Type Text

Optional Features

8  Allow Reports

Allow Activities  
 Track Field History  
 Allow in Chatter Groups  
 Enable Licensing

Object Classification

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more](#).

Allow Sharing  
 Allow Bulk API Access  
 Allow Streaming API Access

Deployment Status

9  In Development  
 Deployed

Activate Windows  
Go to Settings to activate Windows.

9. Select Allow search.
10. Click on Save and New

Deployment Status

In Development  
 Deployed

Search Status

When this setting is enabled, your users can find records of this object type when they search. [Learn more](#).

9  Allow Search

Object Creation Options (Available only when custom object is first created)

Add Notes and Attachments related list to default page layout  
 Launch New Custom Tab Wizard after saving this custom object

10

Activate Windows  
Go to Settings to activate Windows.

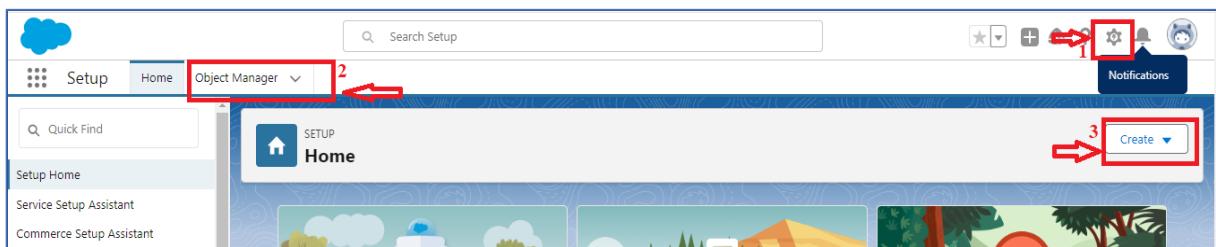
In the same way Create Bus, Trip, Ticket Fare and Employee objects.

# A CRM Application For Public Transport Management

## Activity 2: Creating a Bus Object

The purpose of creating a Bus custom object is to store and manage information about Bus Stops.  
To create an object:

1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.



4. Enter the label name as Bus
5. Enter Plural label name as Buses
6. Enter Record Name as Bus Registration Number

**Custom Object Information**

The singular and plural labels are used in tabs, page layouts, and reports.  
**Be careful when changing the name or label as it may affect existing integrations and merge templates.**

Label	<input type="text" value="Bus"/> Example: Account
Plural Label	<input type="text" value="Buses"/> Example: Accounts
Starts with vowel sound	<input type="checkbox"/>

The Object Name is used when referencing the object via the API.

Object Name	<input type="text" value="Bus"/> Example: Account
-------------	---

Description

Context-Sensitive Help Setting

<input checked="" type="radio"/> Open the standard Salesforce.com Help & Training window
<input type="radio"/> Open a window using a Visualforce page

Content Name

—None—

7. Select Data Type as Text.
8. Select Allow reports.

# A CRM Application For Public Transport Management

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name is called "Name" when referenced via the API.

Record Name:  Example: Account Name

Data Type:  Warning: If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

**Optional Features**

Allow Reports  
 Allow Activities  
 Track Field History  
 Allow in Chatter Groups  
 Enable Licensing

**Object Classification**

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more](#).

Allow Sharing  
 Allow Bulk API Access  
 Allow Streaming API Access

**Deployment Status**

In Development  
 Deployed

**Search Status**

9. Select Allow search.
10. Click on Save and New

**Deployment Status**

In Development  
 Deployed

**Search Status**

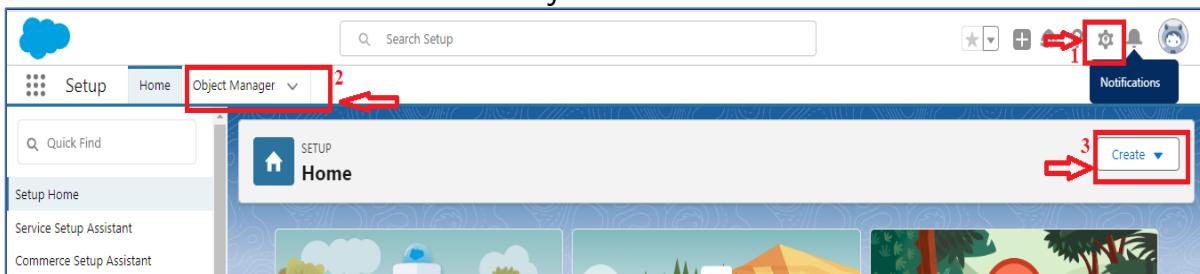
When this setting is enabled, your users can find records of this object type when they search. [Learn more](#).

Allow Search

## Activity 3: Creating a Trip Object

The purpose of creating a Bus custom object is to store and manage information about Bus Stops. To create an object:

1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.



# A CRM Application For Public Transport Management

4. Enter the label name as Trip
5. Enter Plural label name as Trips
6. Enter Record Name as Trip No

**Custom Object Definition Edit** Save Save & New Cancel

**Custom Object Information**

The singular and plural labels are used in tabs, page layouts, and reports.  
**Be careful when changing the name or label as it may affect existing integrations and merge templates.**

Label	<input type="text" value="Trip"/>	Example: Account
Plural Label	<input type="text" value="Trips"/>	Example: Accounts
Starts with vowel sound	<input type="checkbox"/>	

The Object Name is used when referencing the object via the API.

Object Name	<input type="text" value="Trip"/>	Example: Account
-------------	-----------------------------------	------------------

Description

Context-Sensitive Help Setting

<input checked="" type="radio"/> Open the standard Salesforce.com Help & Training window
<input type="radio"/> Open a window using a Visualforce page

Content Name

—None—

7. Select Data Type as Text.
8. Select Allow reports.

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name	<input type="text" value="Trip No"/>	Example: Account Name
-------------	--------------------------------------	-----------------------

Data Type  **Warning:** If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

**Optional Features**

Allow Reports  
 Allow Activities  
 Track Field History  
 Allow in Chatter Groups  
 Enable Licensing (i)

**Object Classification**

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more](#).

Allow Sharing  
 Allow Bulk API Access  
 Allow Streaming API Access

9. Select Allow search.
10. Click on Save and New

**Deployment Status**

In Development  
 Deployed

**Search Status**

When this setting is enabled, your users can find records of this object type when they search. [Learn more](#).

Allow Search

Save Save & New Cancel

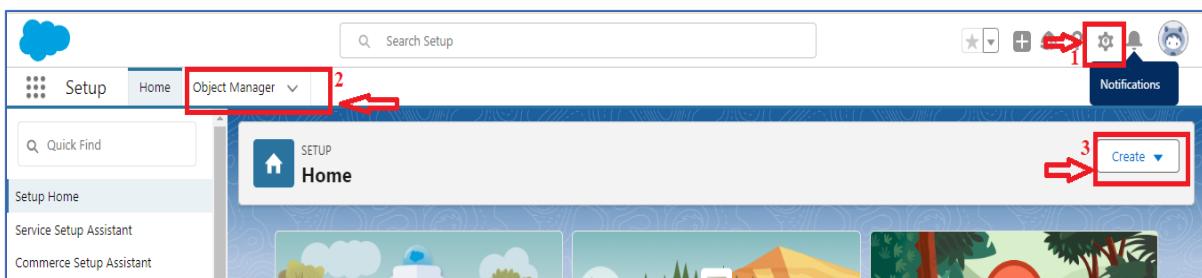
# A CRM Application For Public Transport Management

## Activity 4: Creating a Ticket Fare Object

The purpose of creating a Ticket Fare custom object is to store and manage information about Bus Stops.

To create an object:

1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.



4. Enter the label name as Ticket Fare
5. Enter Plural label name as Ticket Fares
6. Enter Record Name as Route Name

This screenshot shows the 'Custom Object Definition Edit' page. At the top, there are three buttons: 'Save', 'Save & New', and 'Cancel'. The main section is titled 'Custom Object Information' with a note: 'The singular and plural labels are used in tabs, page layouts, and reports. Be careful when changing the name or label as it may affect existing integrations and merge templates.' It includes fields for 'Label' (Ticket Fare), 'Plural Label' (Ticket Fares), and 'Example' (Account). There is also a checkbox for 'Starts with vowel sound'. Below this, there is a note: 'The Object Name is used when referencing the object via the API.' It shows 'Object Name' (Ticket\_Fare) and 'Example' (Account). A 'Description' field is present with a large text input area. At the bottom, there are 'Content Name' dropdown options ('None') and 'Context-Sensitive Help Setting' radio buttons ('Open the standard Salesforce.com Help & Training window' is selected).

7. Select Data Type as Text.
8. Select Allow reports.

# A CRM Application For Public Transport Management

Enter Record Name Label and Format

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name  Example: Account Name

Data Type  **Warning:** If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

**Optional Features**

Allow Reports  
 Allow Activities  
 Track Field History  
 Allow in Chatter Groups  
 Enable Licensing [i](#)

**Object Classification**

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more](#).

Allow Sharing  
 Allow Bulk API Access  
 Allow Streaming API Access

9. Select Allow search.
10. Click on Save and New

**Deployment Status**

In Development  
 Deployed

**Search Status**

When this setting is enabled, your users can find records of this object type when they search. [Learn more](#).

Allow Search

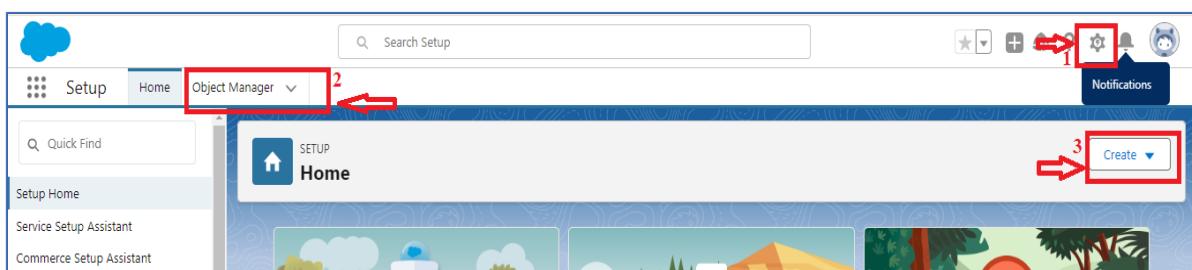
**Buttons**

## Activity 5: Creating aEmployee Object

The purpose of creating aEmployee custom object is to store and manage information about Bus Stops.

To create an object:

1. From the setup page
2. Click on Object Manager
3. Click on Create >> Click on Custom Object.



4. Enter the label name as Employee
5. Enter Plural label name as Employees
6. Enter Record Name as Employee Id

# A CRM Application For Public Transport Management

**Custom Object Information**

The singular and plural labels are used in tabs, page layouts, and reports.  
Be careful when changing the name or label as it may affect existing integrations and merge templates.

Label	<input type="text" value="Employee"/>	Example: Account
Plural Label	<input type="text" value="Employees"/>	Example: Accounts
Starts with vowel sound	<input type="checkbox"/>	

The Object Name is used when referencing the object via the API.

Object Name	<input type="text" value="Employee"/>	Example: Account
-------------	---------------------------------------	------------------

Description

Context-Sensitive Help Setting

<input checked="" type="radio"/> Open the standard Salesforce.com Help & Training window
<input type="radio"/> Open a window using a Visualforce page

Content Name

7. Select Data Type as Text.
8. Select Allow reports.

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name is called "Name" when referenced via the API.

Record Name	<input type="text" value="Employee Id"/>	Example: Account Name
-------------	--	-----------------------

Data Type  **Warning:** If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.

**Optional Features**

Allow Reports  
 Allow Activities  
 Track Field History  
 Allow in Chatter Groups  
 Enable Licensing [\[i\]](#)

**Object Classification**

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more.](#)

Allow Sharing  
 Allow Bulk API Access  
 Allow Streaming API Access

9. Select Allow search.
10. Click on Save and New

**Deployment Status**

In Development  
 Deployed

**Search Status**

When this setting is enabled, your users can find records of this object type when they search. [Learn more.](#)

Allow Search

# A CRM Application For Public Transport Management

## Milestone 3 Tabs

### Activity 1: Creating a tab for Bus Station Object

1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).

The screenshot shows the Salesforce Setup interface under the 'Tabs' section. A red box labeled '1' highlights the 'Search' bar at the top. Another red box labeled '2' highlights the 'Tabs' link in the sidebar. A third red box labeled '3' highlights the 'New' button in the 'Custom Object Tabs' section. The page displays sections for 'Custom Object Tabs', 'Web Tabs', and 'Visualforce Tabs', each with a 'New' button and a 'What Is This?' link.

4. Select Object(Bus Station) >> Select the tab style
5. Click on Next >>(Add to profiles page) keep it as default >>Click on Next (Add to Custom App) uncheck the include tab .

The screenshot shows the 'Step 1. Enter the Details' page. A red box labeled '4' highlights the 'Object' dropdown menu set to 'Bus Station'. The 'Tab Style' dropdown is set to 'Desk'. The page includes fields for 'Description' and 'Splash Page Custom Link' (set to 'None'). At the bottom right, there is an 'Activate Windows' watermark and a 'Next' button.

6. Make sure that the Append tab to user's existing personal customizations is checked

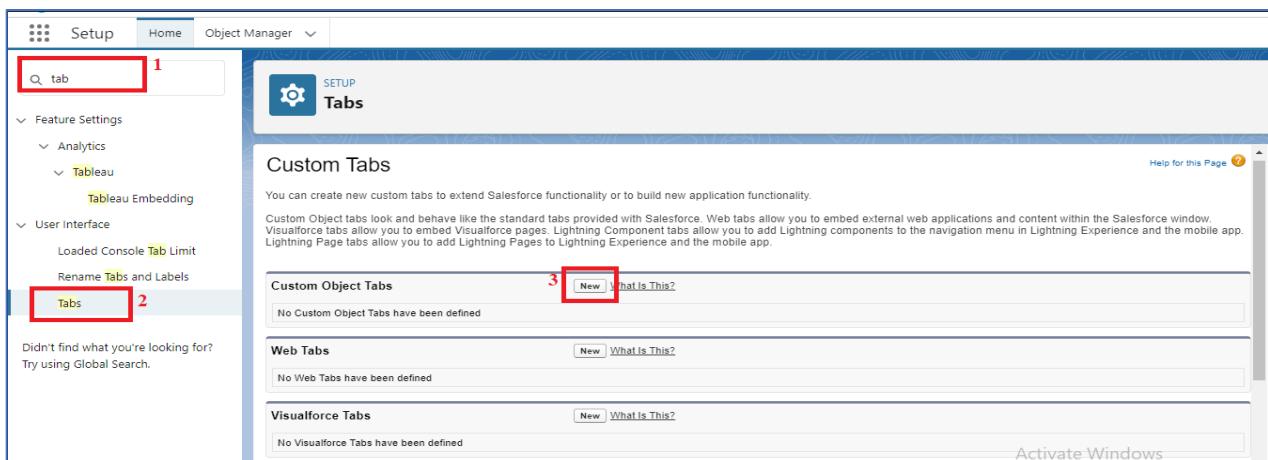
# A CRM Application For Public Transport Management

7. Click on Save.

- 1. Now create the Tabs for the remaining Objects, they are “Bus, Trip, Ticket Fare, Employee”.**
- 2. Follow the same steps as mentioned in Activity -1**

## Activity 2: Creating Tabs for Bus Object

1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).



4. Select Object(Bus ) >> Select the tab style
5. Click on Next >>(Add to profiles page) keep it as default >>Click on Next (Add to Custom App) uncheck the include tab .

Custom Object Tab Information

Tab Label: Buses  
Object: Bus  
Tab Style: Car

(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.  
Splash Page Custom Link:

Enter a short description.

Description:

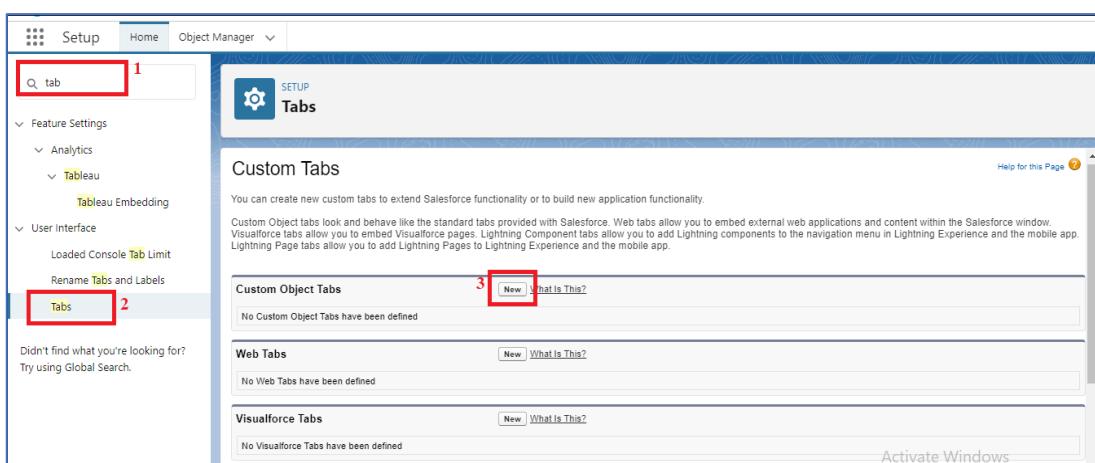
Save Cancel

# A CRM Application For Public Transport Management

6. Make sure that the Append tab to user's existing personal customizations is checked
7. Click on Save.

## Activity 3: Creating Tabs for Trip Object

1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).



4. Select Object(Trip ) >> Select the tab style
5. Click on Next >>(Add to profiles page) keep it as default >>Click on Next (Add to Custom App) uncheck the include tab .

Custom Object Tab Information

Tab Label	Trips
Object	Trip
Tab Style	<input checked="" type="radio"/> Camera <input type="radio"/>

(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.  
Splash Page Custom Link --None--

Enter a short description.

Description

Save Cancel

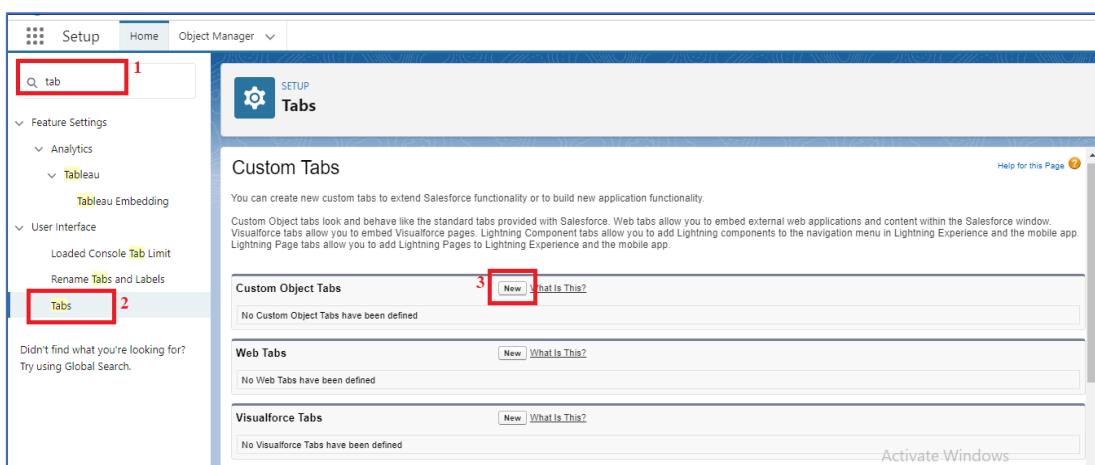
6. Make sure that the Append tab to user's existing personal customizations is checked

# A CRM Application For Public Transport Management

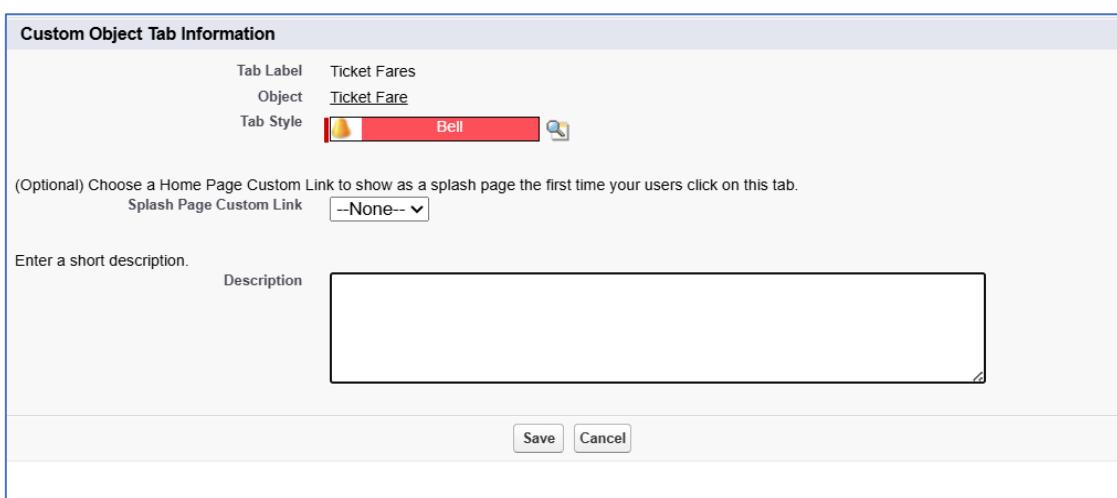
7. Click on Save.

## Activity 4: Creating Tabs for Ticket Fare Object

1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).



4. Select Object(Ticket Fare ) >> Select the tab style
5. Click on Next >>(Add to profiles page) keep it as default >>Click on Next (Add to Custom App) uncheck the include tab .



Custom Object Tab Information

Tab Label: Ticket Fares  
Object: Ticket Fare  
Tab Style: Bell

(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.  
Splash Page Custom Link:

Enter a short description.  
Description:

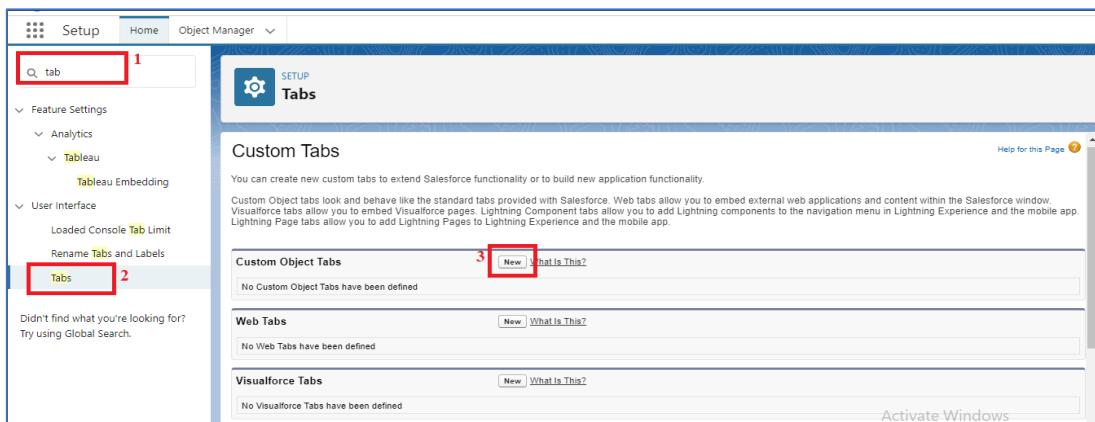
Save Cancel

6. Make sure that the Append tab to user's existing personal customizations is checked
7. Click on Save.

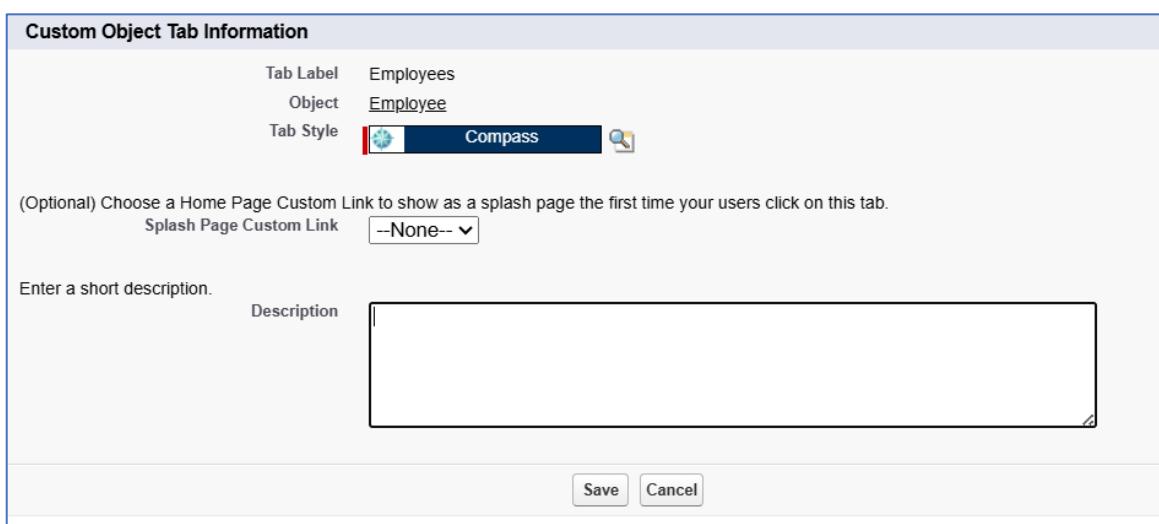
# A CRM Application For Public Transport Management

## Activity 5: Creating Tabs for Employee Object

1. Go to the setup page >> type Tabs in Quick Find bar
2. Click on tabs
3. Click on New (under custom object tab).



4. Select Object(Ticket Fare ) >> Select the tab style
5. Click on Next >>(Add to profiles page) keep it as default >>Click on Next (Add to Custom App) uncheck the include tab .



Custom Object Tab Information

Tab Label: Employees  
Object: Employee  
Tab Style: Compass

(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.  
Splash Page Custom Link: --None--

Enter a short description.

Description:

Save Cancel

6. Make sure that the Append tab to user's existing personal customizations is checked
7. Click on Save

# A CRM Application For Public Transport Management

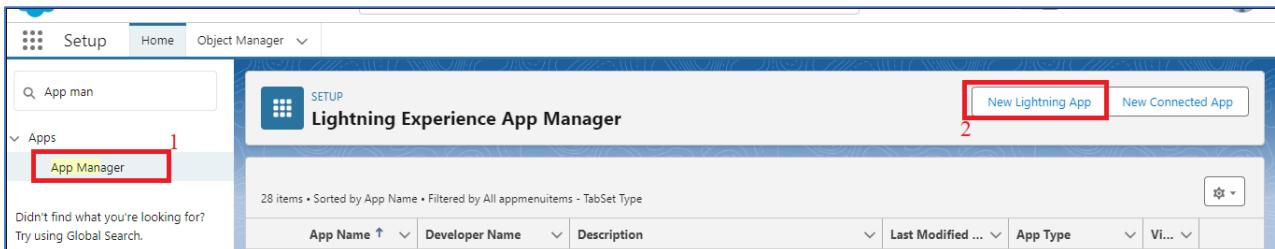
## Milestone 4

### The Lightning App

A Lightning App in Salesforce refers to an application built using the Lightning framework, which is a modern user interface framework for developing dynamic web applications for mobile and desktop devices. Lightning apps provide a more responsive and interactive user experience compared to traditional Visualforce pages.

#### Activity 1: Create a Lightning App for Banquet Hall Booking

1. From Setup, enter App Manager in the Quick Find and select App Manager.
2. Click New Lightning App.

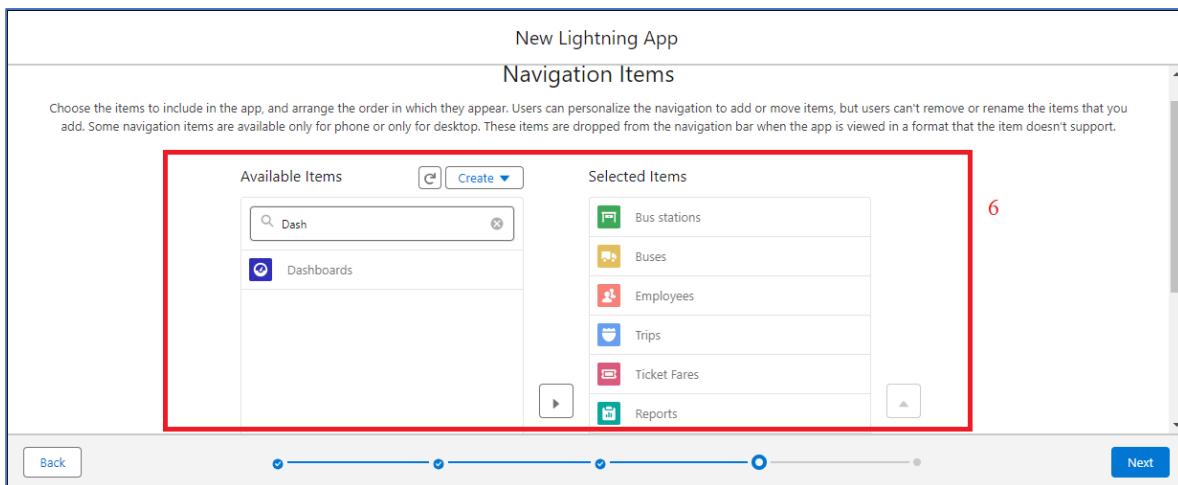


3. Enter Public Transport as the App Name, then click next

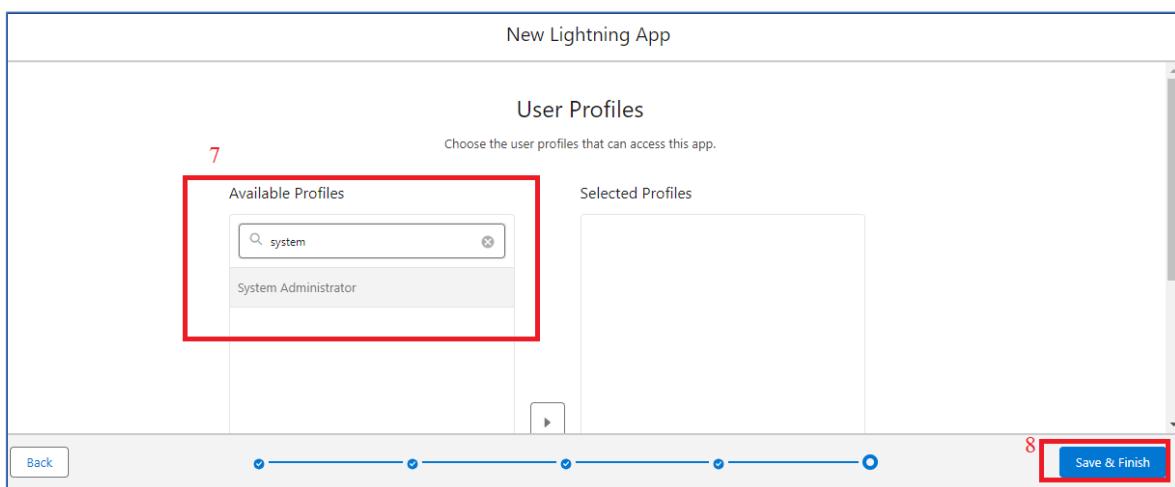
A screenshot of the 'New Lightning App' configuration page. The title is 'New Lightning App' under 'App Details & Branding'. It says 'Give your Lightning app a name and description. Upload an image and choose the highlight color for its navigation bar.' The 'App Details' section includes fields for 'App Name' (with a red box labeled '3' over it), 'Developer Name' (containing 'Public\_TransportRTC'), and 'Description' (with placeholder 'Enter a description...'). The 'App Branding' section includes 'Image' (with an 'Upload' button), 'Primary Color Hex Value' (#0070D2), and 'Org Theme Options' (checkbox). A progress bar at the bottom shows step 1 of 6 completed. A red box labeled '4' highlights the 'Next' button at the bottom right.

4. Under App Options, leave the default selections and click next.
5. Under Utility Items, leave as is and click Next.
6. From Available Items, select Bus Stations, Buses, Trips, Ticket Fares, Employees, Reports, and Dashboards and move them to Selected Item and Click Next.

# A CRM Application For Public Transport Management



7. From Available Profiles, select System Administrator and move it to Selected Profiles.
8. Click Save & Finish



# A CRM Application For Public Transport Management

## Milestone 5 Fields

Object	Field Name	Data Type
<b>Bus Station</b>	Bus Station Name(Standard)	Text
	Bus Stop Category	Picklist(Unmanaged Bus Stop, Managed Bus Stop)
	Last Updated	Formula(Date)
	Amenities	Picklist(Multi-select)
	City	Text(40)
	Street	TextArea
	State/Province	Text(25)
	Zip/PostalCode	Text(10)
	Shelter Available	Checkbox
	Bench	Checkbox
<b>Bus</b>	Bus Registration No (Standard)	Text
	Bus Station Name	Lookup(Bus Station)
	Capacity	Number(4,0)
	Category	Picklist

# A CRM Application For Public Transport Management

	Model	Picklist(Dependent on Category)
<b>Trip</b>	Trip No(Standard)	Text
	Trip Date	Date
	Bus No	Lookup(Bus)
	Route Name	Lookup(Ticket Fare)
	Arrival Time	Picklist
	Departure Time	Picklist
	Bus Starting Terminal	Text
	Destination Terminal	Text
	Driver Id	Lookup(Employee)
	Driver	Formula
	Conductor Id	Lookup(Employee)
	Conductor	Formula
	Estimated Travel Time	Number
	Frequency Per Day	Number(2,0)
	No. of Stops	Number(2,0)
	Passenger Count	Number(4,0)
	Ticket Fare	Currency(16,2)
	Total Amount	Formula

# A CRM Application For Public Transport Management

<b>Ticket Fare</b>	Route Name(Standard)	Text
	Bus Model	Picklist
	Ticket Fare	Currency(10,2)
<b>Employee</b>	Employee Id(Standard)	Text
	Bus Station Name	Lookup(Bus Station)
	Employee Name	Text
	Role	Picklist
	Date of Birth	Date
	Age	Formula(Number)
	Work Place	Text
	Salary	Currency(18,0)
	Phone	Phone
	Date of Joining	Date
	Date of Retirement	Formula(Date)
	Experience	Formula(Number)
	Street	TextArea
	City	Text
	State/Province	Text
	Country	Text

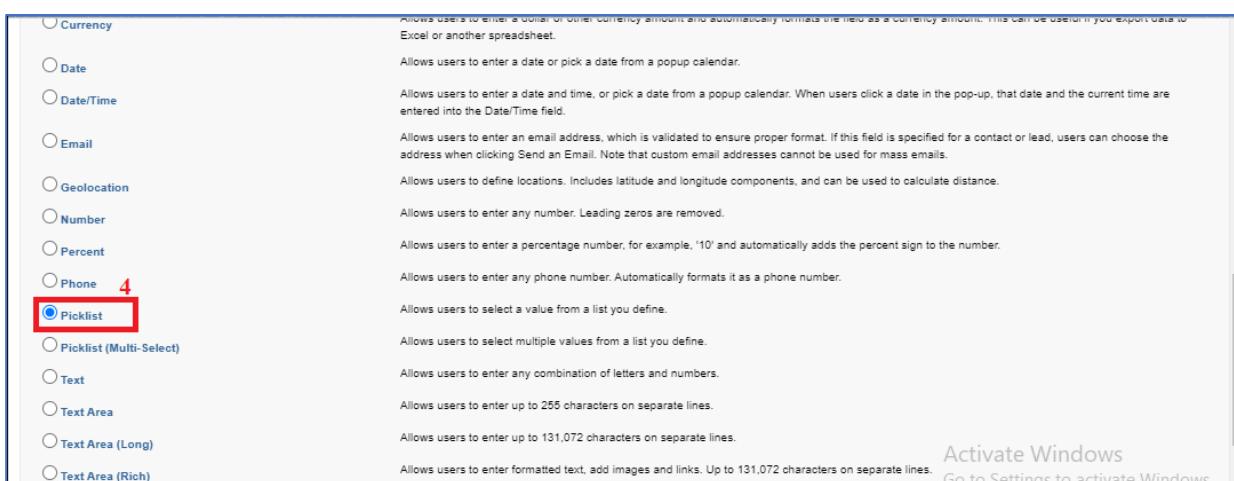
# A CRM Application For Public Transport Management

	Zip/PostalCode	Text
--	----------------	------

## Activity 1: Creating a Role Picklist Field in Employee Object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Employee) in quick find box>> click on the Employee Object.
2. Now click on "Fields & Relationships" .
3. Click on New.
4. Select Data type as "Picklist" and click Next.



5. Enter Field Label as "Role".
6. In values select "Enter values, with each value separated by a new line" and enter values as shown below.

Administrative Assistant

Cleaner

Conductor

Customer Service Representative

Driver

# A CRM Application For Public Transport Management

Safety Officer

Station Manager

Supervisor

Ticket Inspector

7. Select Display values alphabetically, not in the order entered .

Step 2. Enter the details Step 2 of 4

5 Field Label Role

Values  Use global picklist value set  Enter values, with each value separated by a new line

6 Station Manager  
Administrative Assistant  
Customer Service Representative  
Safety Officer  
Supervisor  
Cleaner

7  Display values alphabetically, not in the order entered  Use first value as default value  Restrict picklist to the values defined in the value set

Field Name Role

Description

Activate Windows Go to Settings to activate Windows.

8. Select Required, Always require a value in this field in order to save a record .

9. Click on Next, Next and Save.

7 Required  Always require a value in this field in order to save a record  Add this field to existing custom report types that contain this entity

Auto add to custom report type  Add this field to existing custom report types that contain this entity

Default Value Show Formula Editor

Use formula syntax: Enclose text and picklist value API names in double quotes: ("the\_text"), include numbers without quotes: (25), show percentages as decimals: (0.10), and express date calculations in the standard format: (Today() + 7). To reference a field from a Custom Metadata type record use: \$CustomMetadataType\_\_mdt:RecordAPIName Field\_\_c

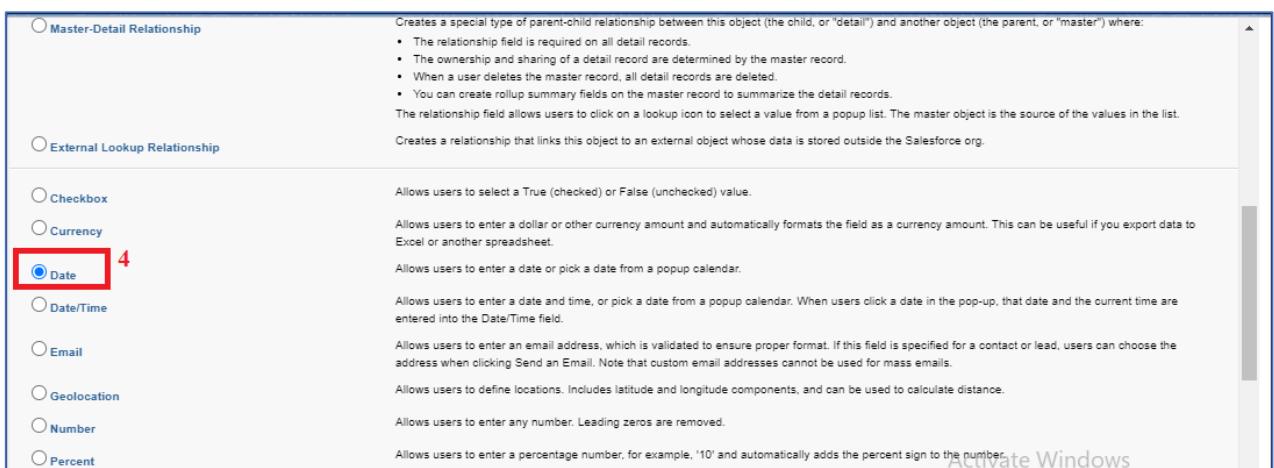
Activate Windows 8 Go to Settings to activate Windows

# A CRM Application For Public Transport Management

## Activity 2: Creating a Trip Date Field in Trip object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Trip) in quick find box>> click on the Trip object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Date” and click Next.



5. Enter Field Label as “Trip Date”.
6. Select Required, Always require a value in this field in order to save a record .
7. Click on Next, Next and Save.

The screenshot shows the 'New Field' configuration page. The 'Field Label' is set to 'Trip Date' (highlighted with a red box and number 5). The 'Field Name' is 'Trip\_Date'. Under 'General Options', the 'Required' checkbox is checked (highlighted with a red box and number 6), and the 'Always require a value in this field in order to save a record' option is selected. Other configuration options like Data Owner, Field Usage, Data Sensitivity Level, and Compliance Categorization are also visible.

# A CRM Application For Public Transport Management

## Activity 3: Creating a Number Field in Bus object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Bus) in quick find box >> click on the Bus object.
2. Now click on "Fields & Relationships"
3. Click on New.
4. Select Data type as "Number" and click Next.
5. Enter Field Label as " Capacity".
6. Length - 4, Decimal Places - 0.
7. Select Required, Always require a value in this field in order to save a record .
8. Click on Next, Next and Save.

## Activity 4 : Creating Lookup Relationship

A Lookup relationship is a type of relationship in Salesforce that connects two objects together based on a field known as the Lookup field. It establishes a relationship between a child object and a parent object, allowing the child object to reference the parent object.

To Create a relationship from Employee to Bus Station .

1. Go to the Setup page >> click on Object manager >> type object name(Employee) in the quick find bar >> click on the Employee object.
2. Click on fields & relationship
3. Click on New.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Billing No	Name	Auto Number		
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		

4. Select "Lookup relationship" as data type and click Next.

# A CRM Application For Public Transport Management

**Data Type**

Select one of the data types below.

None Selected

Auto Number      A system-generated sequence number that uses a display format you define. The number is automatically incremented for each new record.

Formula      A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.

Roll-Up Summary      A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.

**Lookup Relationship**      Creates a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other object is the source of the values in the list.

Master-Detail Relationship      Creates a special type of parent-child relationship between this object (the child, or "detail") and another object (the parent, or "master") where:

- The relationship field is required on all detail records.
- The ownership and sharing of a detail record are determined by the master record.
- When a user deletes the master record, all detail records are deleted.
- You can create rollup summary fields on the master record to summarize the detail records.

Activate Windows  
Go to Settings to activate Windows

5. Select the related object “ Bus Station”.

Employee  
New Relationship

Step 2. Choose the related object

Select the other object to which this object is related.

Related To:

Step 2

Previous Next Cancel

6. Click on Next

7. Give Field Label as “Bus Station Name” .

Employee  
New Relationship

Step 3. Enter the label and name for the lookup field

Field Label:

Field Name:

Description:

Help Text:

Step 3 of 6

Previous Next Cancel

Child Relationship Name:

Required:  Always require a value in this field in order to save a record  
 Clear the value of this field. You can't choose this option if you make this field required.

Activate Windows  
Go to Settings to activate Windows

8. Click on Next , Next, Next , Save.

# A CRM Application For Public Transport Management

## Activity 5: Creating a Checkbox Field in Bus Station object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Bus Station) in quick find box>> click on the Bus Station object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Checkbox” and click Next.
5. Enter Field Label as “ Shelter available”.

Bus Station  
New Custom Field

Step 2. Enter the details Step 2 of 4

Field Label: Shelter Available

Default Value:  Unchecked

Field Name: Shelter\_Available

Description: Indicates whether the bus stop has a shelter or covered waiting area.

Help Text:

Auto add to custom report type  Add this field to existing custom report types that contain this entity

Activate Windows  
Go to Settings to activate Windows.

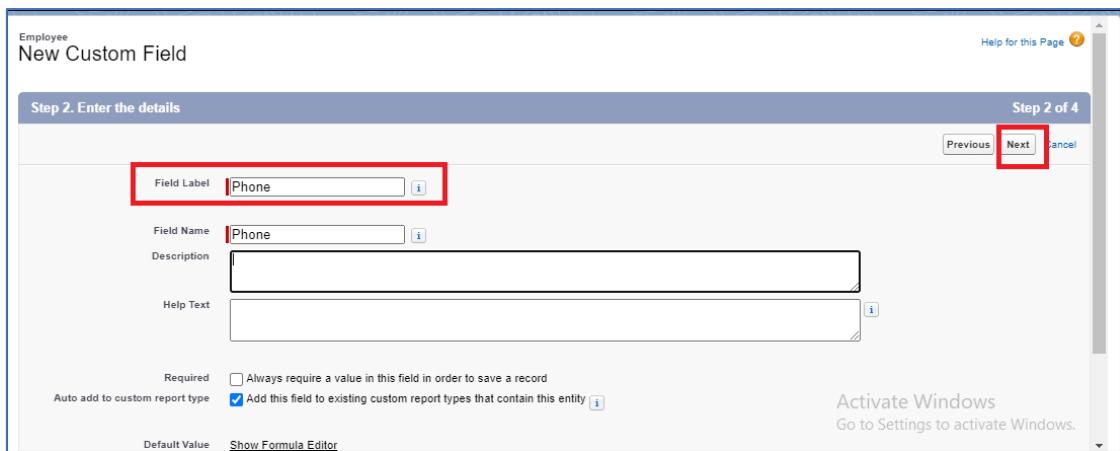
6. Select Default value : Unchecked .
7. Click on Next, Next and Save.

## Activity 6: Creating a Phone Field in Employee object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Employee) in quick find box>> click on the Employee object.
2. Now click on “Fields & Relationships”
3. Click on New.
4. Select Data type as “Phone” and click Next.
5. Enter Field Label as “ Phone No”.

# A CRM Application For Public Transport Management



The screenshot shows the 'Employee New Custom Field' setup page, Step 2 of 4. The 'Field Label' field is set to 'Phone'. The 'Next' button is highlighted with a red box.

6. Click on Next, Next and Save.

## Activity 7: Creating a Last Updated Formula Field in Bus Station object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Bus Station) in quick find box >> click on the Bus Station object.
2. Now click on "Fields & Relationships"
3. Click on New.
4. Select Data type as "Formula" and click Next.
5. Enter field label Last Updated



The screenshot shows the 'Field Information' setup page. The 'Field Label' field is set to 'Last Updated'. The 'Next' button is highlighted with a red box.

6. Select formula return type Date, Click Next
7. Create and insert Advance formula: TODAY()

# A CRM Application For Public Transport Management

Simple Formula | Advanced Formula |

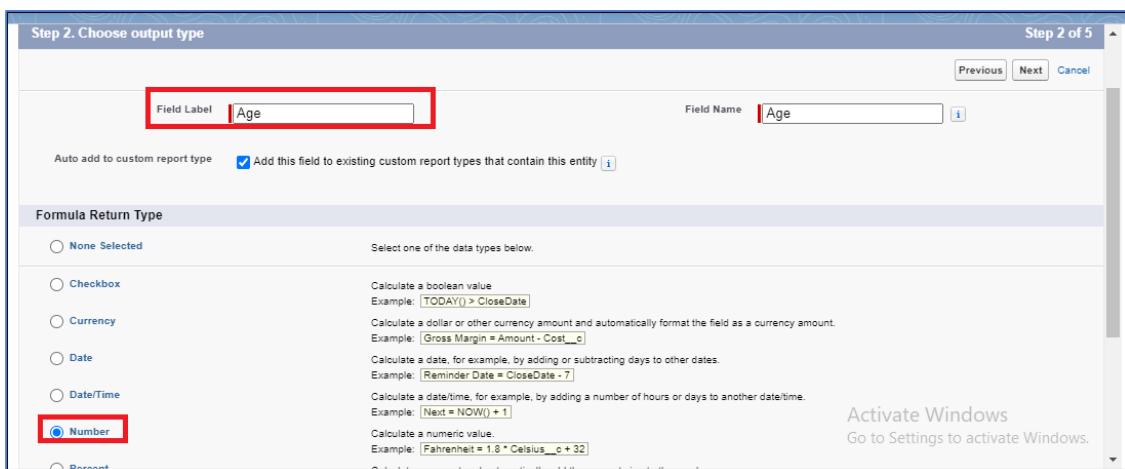
Last Updated (Date) =

8. Click Next, Next, then Save.

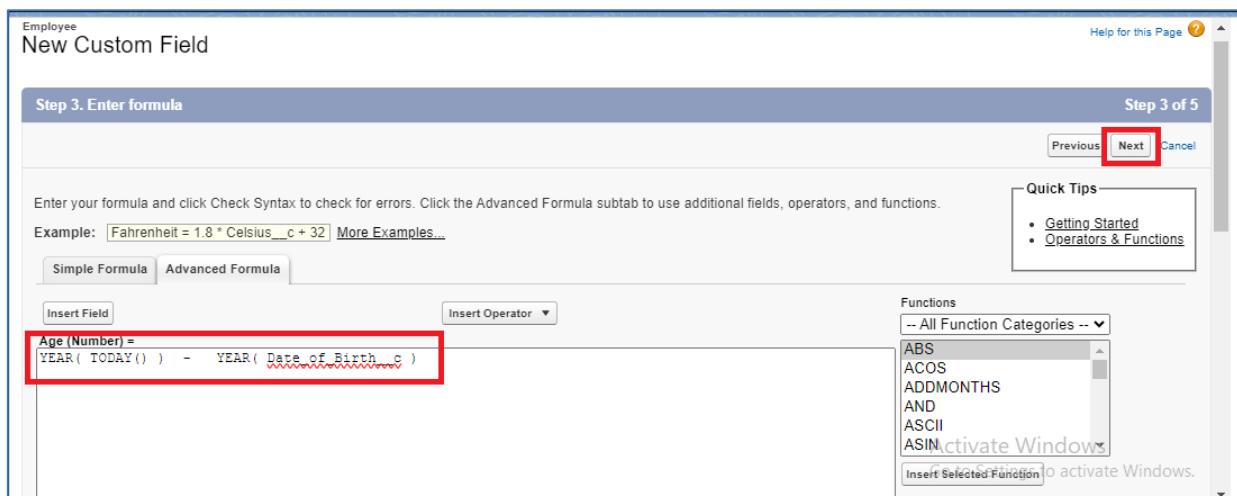
# A CRM Application For Public Transport Management

## Activity 8: Creating a Age Formula Field in Employee object

1. Go to setup >> click on Object Manager >> type object name(Employee) in quick find box >> click on the Employee object.
2. Now click on "Fields & Relationships"
3. Click on New.
4. Select Data type as "Formula" and click Next.
5. Enter field label Age



6. Select formula return type Number, Click Next
7. Create and insert Advance formula: YEAR( TODAY() ) - YEAR( Date\_of\_Birth\_c )



8. Click Next, Next, then Save.

# A CRM Application For Public Transport Management

## Activity 9: Creating a Date of Retirement Formula Field in Employee object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Employee) in quick find box >> click on the Employee object.
2. Now click on "Fields & Relationships"
3. Click on New.
4. Select Data type as "Formula" and click Next.
5. Enter field label Date of Retirement
6. Select formula return type Date, Click Next
7. Create and insert Advance formula: DATE(YEAR( Date\_of\_Birth\_c ) + 55, MONTH(Date\_of\_Birth\_c) , DAY(Date\_of\_Birth\_c) )
8. Click Next, Next, then Save.

The screenshot shows the 'Custom Field Definition Detail' page for a custom field named 'Date of Retirement'. The page includes sections for 'Field Information', 'Formula Options', and navigation links like 'Edit', 'Set Field-Level Security', 'View Field Accessibility', and 'Where is this used?'. The 'Field Information' section displays details such as Field Label (Date of Retirement), Field Name (Date\_of\_Retirement), API Name (Date\_of\_Retirement\_c), Object Name (Employee), and various other configuration options. The 'Formula Options' section shows the advance formula: DATE(YEAR( Date\_of\_Birth\_c ) + 55, MONTH(Date\_of\_Birth\_c) , DAY(Date\_of\_Birth\_c) ).

## Activity 10: Creating Experience Formula Field in Employee object

To create fields in an object:

1. Go to setup >> click on Object Manager type object name(Employee) in quick find box >> click on the Employee object.
2. Now click on "Fields & Relationships"

# A CRM Application For Public Transport Management

3. Click on New.
4. Select Data type as "Formula" and click Next.
5. Enter field label Experience

Field Information

Field Label	Experience
Field Name	Experience
Description	
Help Text	

6. Select formula return type Number, Click Next
7. Create and insert Advance formula: YEAR( TODAY() ) - YEAR( Date\_of\_joining\_c )

Experience (Number) = YEAR( TODAY() ) - YEAR( Date\_of\_joining\_c )

8. Click Next, Next, then Save.

## Activity 11: Creating a Total Amount Formula Field in Trip object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Trip) in quick find box >> click on the Trip object.
2. Now click on "Fields & Relationships"
3. Click on New.
4. Select Data type as "Formula" and click Next.
5. Enter field label Total Amount.

Field Information

Field Label	Total Amount
Field Name	Total_Amount
Description	
Help Text	

# A CRM Application For Public Transport Management

6. Select formula return type Currency, Click Next.
7. Create and insert Advance formula: Passenger\_Count\_c \* Ticket\_Fare\_c

The screenshot shows a formula editor interface. At the top, there are two tabs: "Simple Formula" and "Advanced Formula", with "Advanced Formula" being selected. Below the tabs are two buttons: "Insert Field" and "Insert Operator". The main area contains the formula "Total Amount (Currency) = Passenger\_Count\_c \* Ticket\_Fare\_c".

8. Click Next, Next, then Save.

## Activity 12: Creating a Driver Name Formula Field in Trip object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Trip) in quick find box>> click on the Trip object.
2. Now click on "Fields & Relationships"
3. Click on New.
4. Select Data type as "Formula" and click Next.
5. Enter field label Driver Name.

The screenshot shows the "Field Information" section of the object setup. It includes fields for "Field Label" (containing "Driver Name"), "Field Name" (containing "Driver\_Name"), and "Description" and "Help Text" which are currently empty.

6. Select formula return type Text, Click Next.
7. Create and insert Advance formula: Driver\_Id\_\_r.Employee\_Name\_c

# A CRM Application For Public Transport Management

The screenshot shows a formula editor interface. At the top, there are tabs for "Simple Formula" and "Advanced Formula", with "Advanced Formula" selected. Below the tabs are buttons for "Insert Field" and "Insert Operator". The main area contains the formula: "Driver Name (Text) = Conductor\_Id\_\_r.Employee\_Name\_\_c". The "Conductor\_Id\_\_r" part is highlighted with a red vertical bar.

8. Click Next, Next, then Save.

## Activity 13: Creating a Conductor Name Formula Field in Trip object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Trip) in quick find box >> click on the Trip object.
2. Now click on "Fields & Relationships"
3. Click on New.
4. Select Data type as "Formula" and click Next.
5. Enter field label Conductor Name.

The screenshot shows the "Custom Field Definition Edit" screen. At the top right are buttons for "Save", "Quick Save", and "Cancel". The main area is titled "Field Information". It includes fields for "Field Label" (set to "Conductor Name"), "Field Name" (set to "Conductor\_Name"), "Description" (empty), and "Help Text" (empty). The "Field Label" field is highlighted with a blue background.

6. Select formula return type Text, Click Next.
7. Create and insert Advance formula: Conductor\_Id\_\_r.Employee\_Name\_\_c

The screenshot shows the formula editor again. The formula "Conductor Name (Text) = Conductor\_Id\_\_r.Employee\_Name\_\_c" is entered. The "Conductor\_Id\_\_r" part is highlighted with a red vertical bar.

# A CRM Application For Public Transport Management

8. Click Next, Next, then Save.

## Activity 14: Create a Global Value Set

1.

1. From the Setup menu, enter "Picklist Value Sets" in the Quick Find box and select it.
2. Click on "New" to create a new global value set.



3. Enter the label Bus Time.
4. In values select "Enter values, with each value separated by a new line" and enter values as shown below.

6:00 AM

7:00 AM

8:00 AM

9:00 AM

10:00 AM

11:00 AM

12:00 PM

1:00 PM

2:00 PM

3:00 PM

4:00 PM

# A CRM Application For Public Transport Management

5:00 PM

6:00 PM

7:00 PM

8:00 PM

9:00 PM

10:00 PM

11:00 Pm

The screenshot shows the 'Picklist Value Sets' page in the Salesforce Setup. A new global value set is being created with the following details:

- Label:** Bus Time
- Name:** Bus\_Time
- Description:** Time to select for Departure or Arrival time of Bus
- Values:** A list of times from 9:00 AM to 11:00 PM, each separated by a new line.

- Click "Save" to create the global value set

## Activity 15: Create a Global Value Set for bus model

- From the Setup menu, enter "Picklist Value Sets" in the Quick Find box and select it.
- Click on "New" to create a new global value set.

The screenshot shows the Salesforce Setup interface with the following steps highlighted:

- The 'Quick Find' search bar at the top left is highlighted with a red box, containing the text 'picklist'.
- The 'Data' section in the sidebar is expanded, showing 'Picklist Settings', 'State and Country/Territory', and 'Picklists'. The 'Picklist Value Sets' item is highlighted with a red box.
- The main content area shows the 'Picklist Value Sets' page. A 'New' button is highlighted with a red box.

- Enter the label Bus Model.

# A CRM Application For Public Transport Management

Information	
Label	Bus Mode
Name	Bus_Model <span style="border: 1px solid #ccc; padding: 2px;">i</span>
Description	  

4. In values select "Enter values, with each value separated by a new line" and enter values as shown below

Regular

Metro

A/C

Express

Deluxe

Super Deluxe

Semi Sleeper

Sleeper

## Activity 16: Creating a Picklist Field using global picklist value set in Trip Object

To create fields in an object:

1. Go to setup >> click on Object Manager >> type object name(Employee) in quick find box>> click on the Employee Object.
2. Now click on "Fields & Relationships" .
3. Click on New.
4. Select Data type as "Picklist" and click Next.
5. Enter Field Label as "Arrival Time".

# A CRM Application For Public Transport Management

Field Information	
Field Label	Arrival Time
Field Name	Arrival_Time
Description	
Help Text	

6. Select "Use global picklist value set" and choose the global value set "Bus Time".
7. Click on Next, Next and Save.

## Activity 17: Creating a Controlled and Dependent Picklists in Bus object

Creating Controlling picklist field in Bus object:

1. Go to setup >> click on Object Manager >> type object name(Bus) in quick find box >> click on the Bus Object.
2. Now click on "Fields & Relationships" .
3. Click on New.
4. Select Data type as "Picklist" and click Next.
5. Enter Field Label as "Category".

Field Information	
Field Label	Category
Field Name	Category
Description	
Help Text	

6. In values select "Enter values, with each value separated by a new line" and enter values as shown below.  
Local  
Intercity  
InterState
7. Select Required, Always require a value in this field in order to save a record .

# A CRM Application For Public Transport Management

**General Options**

Required	<input checked="" type="checkbox"/> Always require a value in this field in order to save a record
Unique	<input type="checkbox"/> Do not allow duplicate values
External ID	<input type="checkbox"/> Set this field as the unique record identifier from an external system
AI Prediction	<input type="checkbox"/> Use this field to store AI prediction scores
Default Value	Show Formula Editor <input type="text"/> Use formula syntax: Enclose text and picklist value API names in double quotes : ("the_text"), include numbers without quotes : (25), show percentages as decimals: (0.10), and express date calculations in the standard format: (Today() + 7). To reference a field from a Custom Metadata type record use: \$CustomMetadata.Type__mdt.RecordAPIName.Field__c

8. Click on Next, Next and Save.

Creating Dependent pickliest field in Bus object:

1. Go to setup >> click on Object Manager >> type object name(Bus) in quick find box >> click on the Bus Object.
2. Now click on "Fields & Relationships" .
3. Click on New.
4. Select Data type as "Picklist" and click Next.
5. Enter Field Label as "Category".

**Field Information**

Field Label	<input type="text" value="Category1"/>	Data Type	Picklist
Field Name	<input type="text" value="Category1"/>		
Description	<input type="text"/>		
Help Text	<input type="text"/>		

6. Select "Use global picklist value set" and choose the global value set "Bus Time".
7. Select Required, Always require a value in this field in order to save a record .

**General Options**

Required	<input checked="" type="checkbox"/> Always require a value in this field in order to save a record
Unique	<input type="checkbox"/> Do not allow duplicate values
External ID	<input type="checkbox"/> Set this field as the unique record identifier from an external system
AI Prediction	<input type="checkbox"/> Use this field to store AI prediction scores
Default Value	Show Formula Editor <input type="text"/> Use formula syntax: Enclose text and picklist value API names in double quotes : ("the_text"), include numbers without quotes : (25), show percentages as decimals: (0.10), and express date calculations in the standard format: (Today() + 7). To reference a field from a Custom Metadata type record use: \$CustomMetadata.Type__mdt.RecordAPIName.Field__c

8. Click on Next, Next and Save.

# A CRM Application For Public Transport Management

## Establish the Dependency

1. Go to setup >> click on Object Manager >> type object name(Bus) in quick find box >> click on the Bus Object
2. Now click on "Fields & Relationships" .
3. Now click on "Field Dependencies" .
4. Click on New.

Controlling Field: --None--

Dependent Field: --None--

Continue Cancel

5. Enter Controlling Field : Category
6. Enter Dependent field : Model
7. Click "Continue".
8. Matrix with the controlling field values on the top and the dependent field values on the side. Check the boxes to define which dependent picklist values should be available for each controlling picklist value.
9. Click Save

Click button to include or exclude selected values from the dependent picklist:

Include Values Exclude Values

Category:	Local	Intercity	Interstate
Model:	Regular Metro A/c Express Deluxe Super Deluxe Semi Sleeper Sleeper	Regular Metro A/c Express Deluxe Super Deluxe Semi Sleeper Sleeper	Regular Metro A/c Express Deluxe Super Deluxe Semi Sleeper Sleeper

Showing Columns: 1 - 3 (of 3) < Previous | Next > View All Go to

Click button to include or exclude selected values from the dependent picklist:

Include Values Exclude Values

Save Cancel Preview

# A CRM Application For Public Transport Management

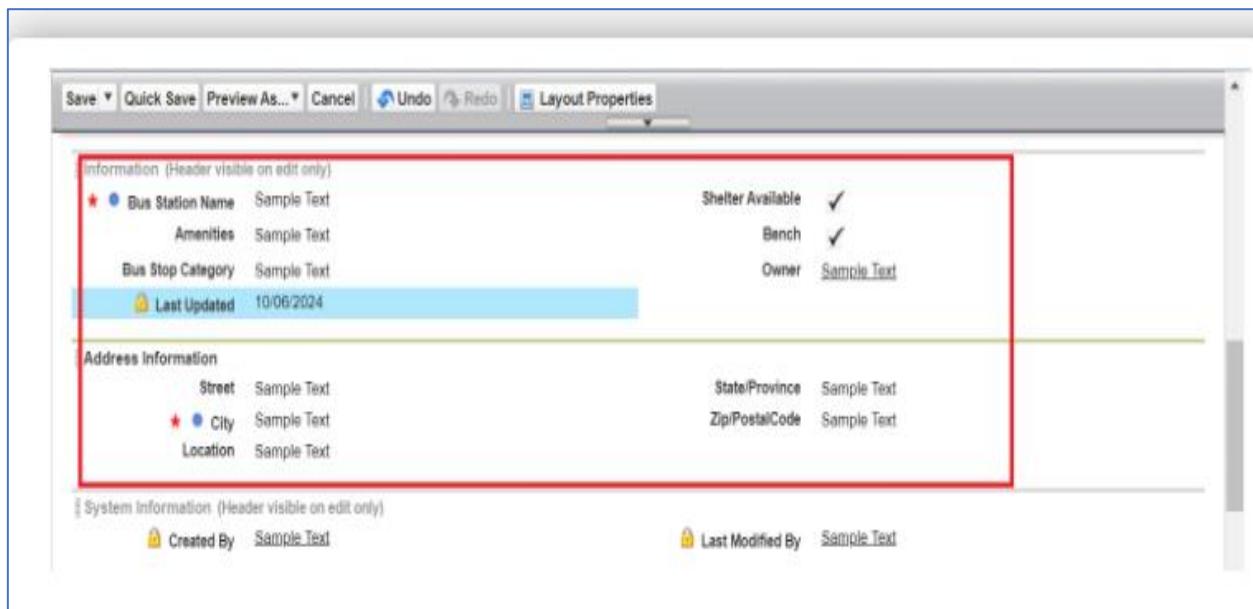
## Milestone 6

### Editing of Page Layouts

Page layouts in Salesforce control the layout and organization of fields, related lists, custom links, and other elements on a record detail or edit page. They are essential for managing how data is presented to users and can vary based on user roles and profiles.

#### Activity 1: To edit a Page Layout in Bus Station Object

1. Go to setup >> click on Object Manager >> type object name(Bus Station) in quick find box >> click on the Bus Station object >> Page Layouts .
2. Click on the Bus Station Layout.
3. Drag and Arrange the field as shown below.



4. Click on field Last Updated >> click on settings >> select Read Only and save it.
5. Click on Save.

# A CRM Application For Public Transport Management

## Activity 2: To create a Page Layout in Bus Object

1. Go to setup >> click on Object Manager >> type object name(Bus) in quick find box >> click on the Bus object >> Page Layouts.
2. Click on the Bus Layout
3. Drag and Arrange the field as shown below

The screenshot shows the 'Bus Detail' page layout configuration. At the top, there are standard buttons: Save, Quick Save, Preview As..., Cancel, Undo, Redo, and Layout Properties. Below these are custom buttons. The main area is divided into sections: 'Information' (header visible on edit only), 'System Information' (header visible on edit only), and 'Custom Links' (header visible on edit only). The 'Information' section is highlighted with a red border and contains fields: Bus Registration No (Sample Text), Bus Station Name (Sample Text), Category (Sample Text), Model (Sample Text), Capacity (4,072), Owner (Sample Text), Created By (Sample Text), and Last Modified By (Sample Text).

4. Click Save

## Activity 3: To create a Page Layout in Employee Object

1. Go to setup >> click on Object Manager >> type object name(Employee) in quick find box >> click on the Employee object >> Page Layouts.
2. Click on the Employee Layout
3. Drag and Arrange the field as shown below

The screenshot shows the 'Employee' page layout configuration. At the top, there are standard buttons: Save, Quick Save, Preview As..., Cancel, Undo, Redo, and Layout Properties. Below these are custom buttons. The main area is divided into sections: 'Information' (header visible on edit only), 'Personal Details', 'Address', and 'System Information' (header visible on edit only). The 'Information' section is highlighted with a red border and contains fields: Employee ID (Sample Text), Employee Name (Sample Text), Bus Station Name (Sample Text), Salary (₹123,45), Experience (794.17), Role (Sample Text), Date of joining (10/06/2024), Date of Retirement (10/06/2024), Owner (Sample Text), Age (88.180), and State/Province (Sample Text), Country (Sample Text), Zip/PostalCode (Sample Text). The 'System Information' section contains fields: Created By (Sample Text) and Last Modified By (Sample Text).

# A CRM Application For Public Transport Management

4. Click Save.

## Activity 4: To create a Page Layout in Trip Object

1. Go to setup >> click on Object Manager >> type object name(Trip) in quick find box >> click on the Trip object >> Page Layouts.
2. Click on the Trip Layout
3. Drag and Arrange the field as shown below

The screenshot shows the Salesforce Page Layout editor for the Trip object. The layout is structured into three main sections:

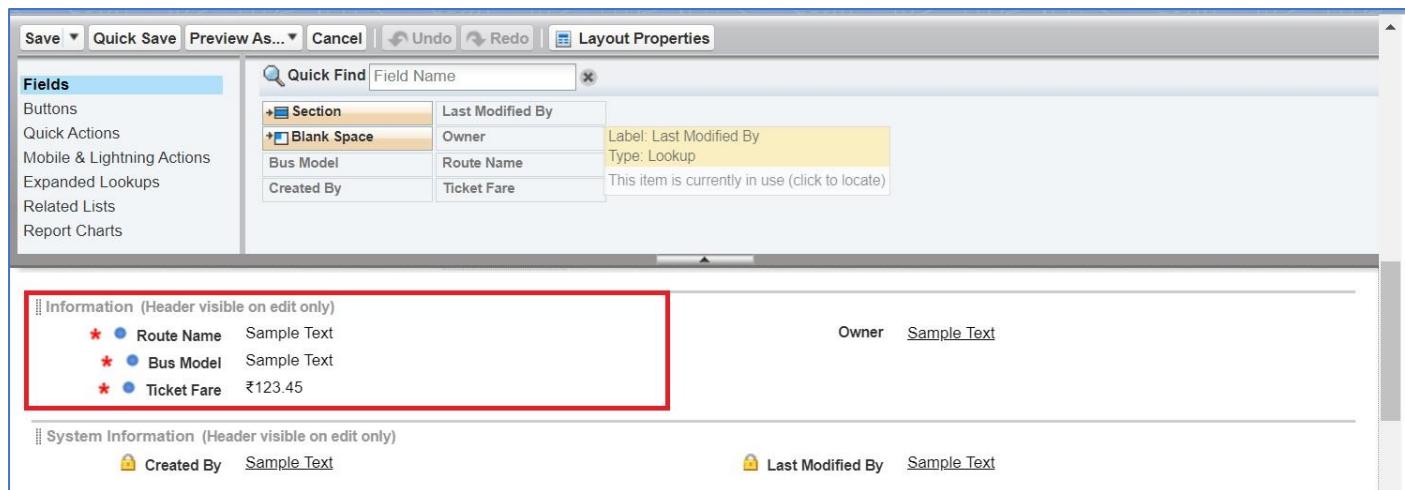
- Information (Header visible on edit only):** Contains fields for Trip No (required), Trip Date (10/06/2024), Bus No (required), Driver Id (required), Driver Name (required), Conductor Id (Sample Text), Conductor Name (Sample Text), and Owner (Sample Text).
- Bus Schedule:** Contains fields for Route Name (required), Bus Starting Terminal (required), Departure Time (Sample Text), No. of Stops (34), Estimated Travel Time (995.77), Destination Terminal (Sample Text), Arrival Time (Sample Text), and Frequency Per Day (45).
- Passenger Information:** Contains fields for Passenger Count (5,691), Ticket Fare (₹123.45), and Total Amount (₹123.45).

4. Click on field Ticket Fare >> click on settings >> select Read Only and save it.
5. Click Save.

## Activity 5: To create a Page Layout in Ticket Fare Object

1. Go to setup >> click on Object Manager >> type object name(Ticket Fare) in quick find box >> click on the Ticket Fare object >> Page Layouts.
2. Click on the Ticket Layout
3. Drag and Arrange the field as shown below

# A CRM Application For Public Transport Management



4. Click Save.

# A CRM Application For Public Transport Management

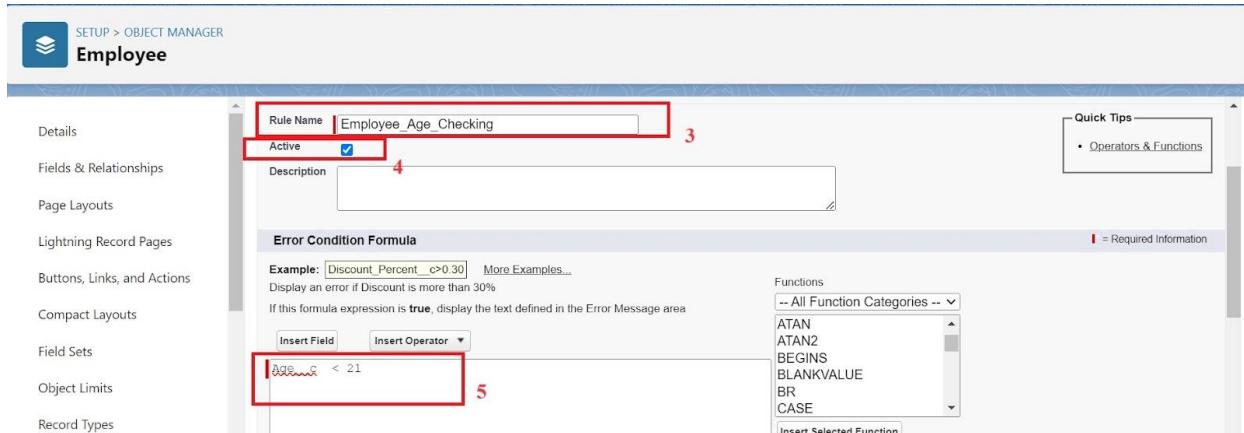
## Milestone 7 Validation Rules

Validation rules in Salesforce are used to ensure data integrity by preventing users from entering invalid data into fields. They consist of a logical formula or expression that evaluates the data in one or more fields and returns true or false. If the rule returns true, an error message is displayed, and the record is not saved.

### Activity 1: To create a validation rule to a Employee Object

1. Go to setup >> click on Object Manager >> type object name(Employee) in quick find box >> click on the Employee object
2. Click on the validation rule >> click on New.
3. Enter the Rule name as "Employee\_Age\_Checking".
4. Select Active
5. Insert the Error Condition Formula as :

Age\_c < 21



6. Enter the Error Message as "Employee Age Must be Greater than or equal to 21".
7. Select the Error location as Top of Page
8. Click Save.

# A CRM Application For Public Transport Management

## Activity 2: To create a validation rule to a Bus Station on Employee Object

1. Go to setup >> click on Object Manager >> type object name(Employee) in quick find box  
>> click on the Banquet Hall object
2. Click on the validation rule >> click on New.
3. Enter the Rule name as "Employees\_only\_for\_Managed\_Bus\_stops".

Rule Name: Employees only for Managed Bus stops (3)  
Active (4)  
Description  
Error Condition Formula  
Example: Discount\_Percent\_\_c>0.30 More Examples...  
Display an error if Discount is more than 30%  
If this formula expression is true, display the text defined in the Error Message area  
Insert Field Insert Operator (5)  
IF( ISPICKVAL( Bus\_Station\_Name\_\_r.Bus\_Stop\_Category\_\_c , "UnManaged Bus Stop") , true, false)  
Functions  
-- All Function Categories -- ABS ACOS ADDMONTHS AND ASCII ASIN Insert Selected Function ABS(number)

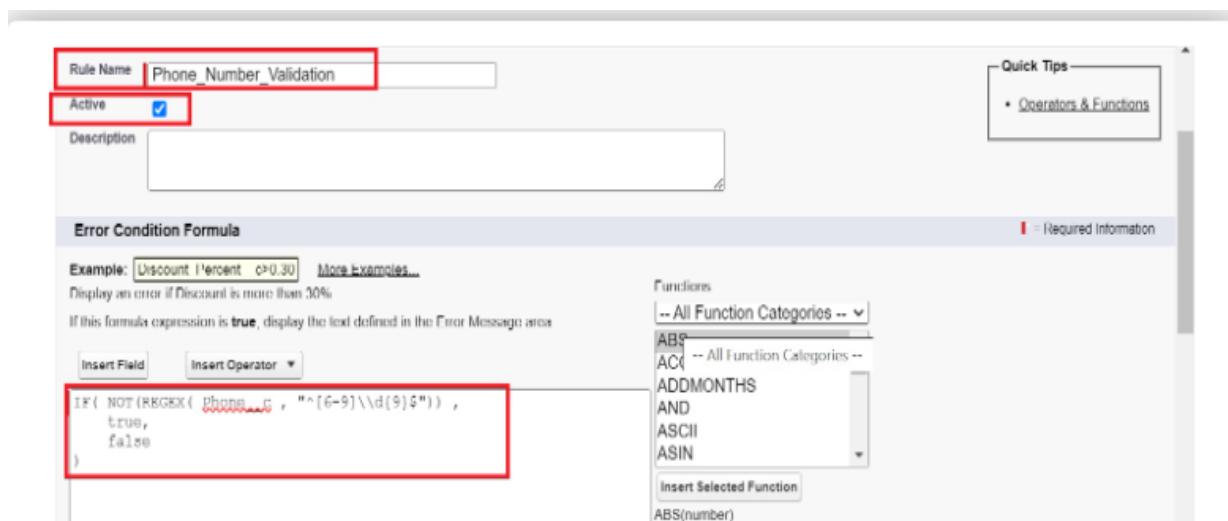
4. Select Active
5. Insert the Error Condition Formula as :  
IF( ISPICKVAL( Bus\_Station\_Name\_\_r.Bus\_Stop\_Category\_\_c , "UnManaged Bus Stop") , true, false)
6. Enter the Error Message as "The Employees must work for Managed Bus stops".
7. Select the Error location as Field and as Bus Station Name and click Save.

Error Message  
Example: Discount percent cannot exceed 30%  
This message will appear when Error Condition formula is true (6)  
Error Message: The Employees must work for Managed Bus stops  
This error message can either appear at the top of the page or below a specific field on the page  
Error Location: Top of Page (radio button) Field (radio button) Bus Station Name (dropdown) (7)  
Save Save & New Cancel

# A CRM Application For Public Transport Management

## Activity 3: To create a validation rule to a Phone No on Employee Object

1. Go to setup >> click on Object Manager >> type object name(Employee) in quick find box >> click on the Employee object
2. Click on the validation rule >> click on New.
3. Enter the Rule name as "Phone\_Number\_Validation".
4. Select Active
5. Insert the Error Condition Formula as :  
`IF( NOT(REGEX( Phone_c , "^[6-9]\d{9}$")) , true, false)`



6. Enter the Error Message as "Phone no must be 10 digits and starts with 6 or 7 or 8or9".
7. Select the Error location as Top of Page and click Save.

## Activity 4: To create a validation rule to a Trip Object

1. Go to setup >> click on Object Manager >> type object name(Trip) in quick find box >> click on the Trip object
2. Click on the validation rule >> click on New.

# A CRM Application For Public Transport Management

3. Enter the Rule name as "Departure\_and\_Arrival\_Time\_Checking".

4. Select Active

5. Insert the Error Condition Formula as :

TEXT( Departure\_Time\_c ) = TEXT( Arrival\_Time\_c )

The screenshot shows the configuration of a new rule. The 'Rule Name' field is set to 'Departure\_and\_Arrival\_Time\_Checking' (3). The 'Active' checkbox is checked (4). The 'Error Condition Formula' field contains the formula 'TEXT( Departure\_Time\_c ) = TEXT( Arrival\_Time\_c )' (5). A list of functions is visible on the right, including ABS, ACOS, ADDMONTHS, AND, ASCII, and ASIN. A 'Quick Tips' panel on the right lists 'Operators & Functions'. A note at the bottom indicates that the formula must be true to display an error message.

6. Enter the Error Message as "The Departure Time and Arrival Time Should not be the same".

7. Select the Error location as Top of Page and click Save.

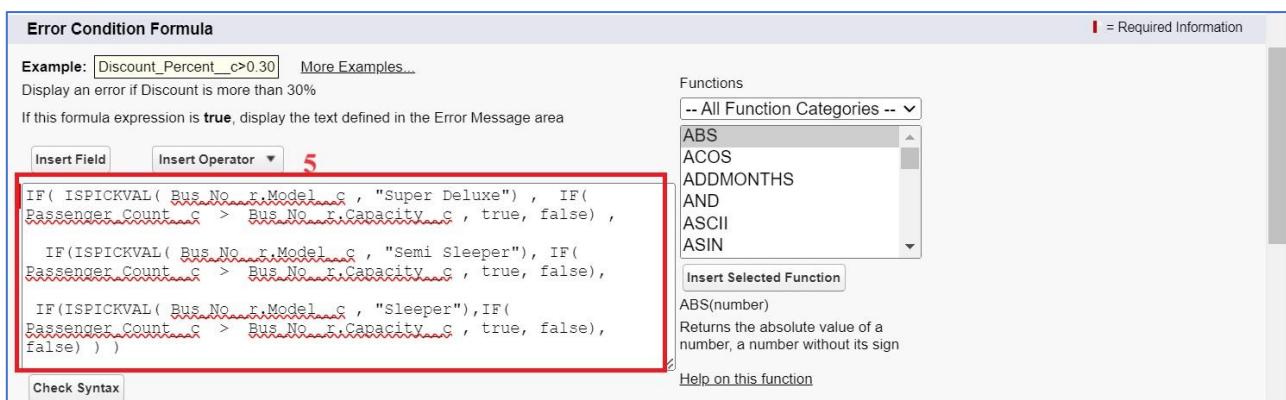
The screenshot shows the configuration of an error message. The 'Error Message' field is set to 'Booking Date must be a future Date' (6). The 'Error Location' dropdown is set to 'Top of Page' (7). The interface also includes 'Save', 'Save & New', and 'Cancel' buttons at the bottom.

# A CRM Application For Public Transport Management

## Activity 5: To create a second validation rule to a Trip Object

1. Go to setup >> click on Object Manager >> type object name(Trip) in quick find box >> click on the Trip object
2. Click on the validation rule >> click on New.
3. Enter the Rule name as "Passenger\_Count\_Checking\_for\_Few\_Buses".
4. Select Active
5. Insert the Error Condition Formula as :

```
IF( ISPICKVAL( Bus_No_r.Model_c , "Super Deluxe") , IF( Passenger_Count_c > Bus_No_r.Capacity_c , true, false) ,  
IF(ISPICKVAL( Bus_No_r.Model_c , "Semi Sleeper"), IF( Passenger_Count_c > Bus_No_r.Capacity_c , true, false),  
IF(ISPICKVAL( Bus_No_r.Model_c , "Sleeper"),IF( Passenger_Count_c > Bus_No_r.Capacity_c , true, false), false) ) )
```



6. Enter the Error Message as "For Super Deluxe, Semi Sleeper and Sleeper Buses ,the Passenger Count must be less than or equal to the Capacity of the Bus".
7. Select the Error location as Field and as Passenger Count and click Save.

# A CRM Application For Public Transport Management

## Activity 6: To create a validation rule to a Bus Object

1. Go to setup >> click on Object Manager >> type object name(Bus) in quick find box >> click on the Bus Hall object
2. Click on the validation rule >> click on New.
3. Enter the Rule name as "Bus\_Registration\_Number\_Validation".
4. Select Active
5. Insert the Error Condition Formula as :

IF( NOT( REGEX( Name , "^[A-Z]{2}\s\d{2}\s[A-Z]{1,2}\s\d{4}\$" )), true, false)

The screenshot shows the 'Validation Rule' configuration page. The 'Rule Name' field is set to 'Bus\_Registration\_Number\_Validation' (3). The 'Active' checkbox is checked (4). The 'Error Condition Formula' field contains the formula 'IF( NOT( REGEX( Name , "^[A-Z]{2}\s\d{2}\s[A-Z]{1,2}\s\d{4}\$" )), true, false)' (5). A dropdown menu titled 'Functions' is open, showing options like ABS, ACOS, ADDMONTHS, AND, ASCII, ASIN, etc. A 'Quick Tips' panel on the right lists 'Operators & Functions'. A note at the bottom indicates that the formula must be true to display an error message.

6. Enter the Error Message as "The bus Registration Number must be in the format of 2 Capital Letters(State Code), space, 2 Numbers (District Code), space 1 or 2 Capital Letters(Series), space and 4 Numbers (Number).".
7. Select the Error location as Field and as Bus Registration No and click Save.

# A CRM Application For Public Transport Management

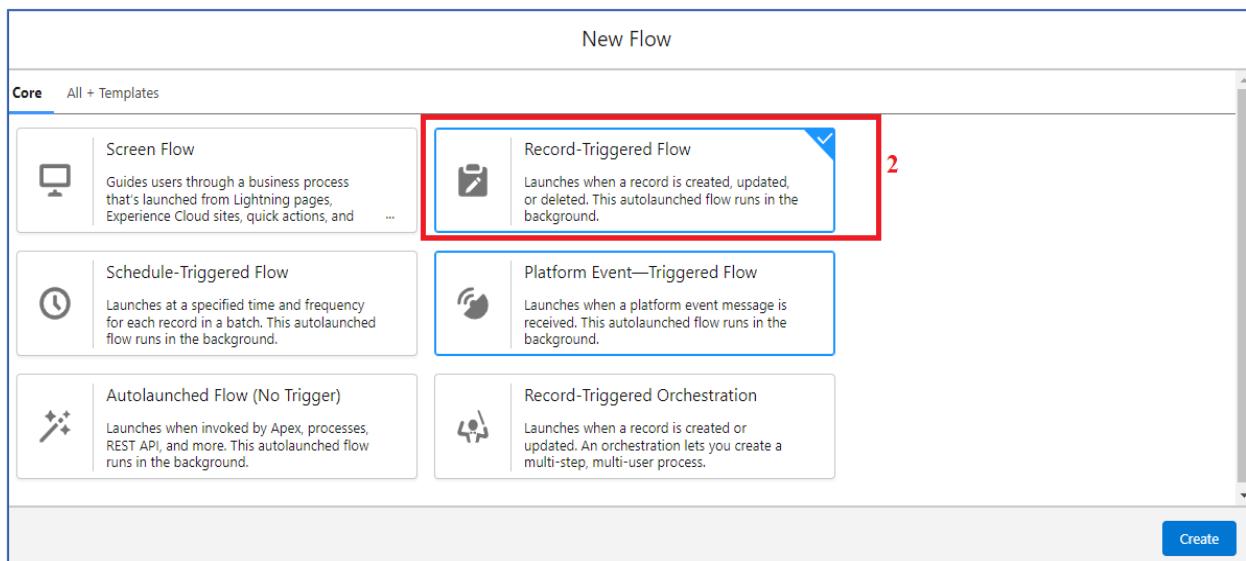
## Milestone 8 Flows

In Salesforce, a Flow is an automation tool that allows you to create complex business processes using a visual interface. Flows can be used to collect, update, delete, and create records, as well as to guide users through a series of screens to complete a process.

### Activity 1 : Create Flow to Fetch Ticket Fare for Bus.

Note: Please enter Route Name in Ticket Fare object as Hyderabad-Warangal(Express) , Hyderabad-Warangal(Deluxe), ...

1. Go to setup ?type Flow in quick find box ? Click on the Flow and Select the New Flow.
2. Select the record Triggered flow.Click on create.



3. Under Object select "Trip"
4. Select A record is created or updated

# A CRM Application For Public Transport Management

Configure Start

Select Object  
Select the object whose records trigger the flow when they're created, updated, or deleted.

\* Object Trip 3

Configure Trigger  
\* Trigger the Flow When:  
 A record is created  
 A record is updated  
 A record is created or updated 4  
 A record is deleted

5. Set Entry Conditions : None
6. Select Actions and Related Records and click on Done

Configure Start

Set Entry Conditions  
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.

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.

Condition Requirements None 5

\* Optimize the Flow for:  
Fast Field Updates  
Update fields on the record that triggers the flow to run. This high-performance flow runs *before* the record is saved to the database.  
Actions and Related Records 6  
Update any record and perform actions, like send an email. This more flexible flow runs *after* the record is saved to the database.

Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed

Cancel Done

7. Under the record trigger flow click on the "+" icon and select Get Records.
8. Enter Label as " Fetching Route Ticket Fares ".
9. For Object select Ticket Fare

Find Salesforce records and store their field values in flow variables.

Label Fetching Route Ticket Fares 8 \* API Name Fetching\_Route\_Ticket\_Fares

Description  
Getting the records from Ticket Fares based on the bus model

Get Records of This Object  
Object Ticket Fare 9

# A CRM Application For Public Transport Management

10. For Condition Requirements , select All Conditions are Met(AND)

For the first condition select as follows:

Field: Id

Operator: Equals

Value: {!\$Record.Route\_Name\_\_c}

Filter Ticket Fare Records

Condition Requirements

All Conditions Are Met (AND)

Field	Operator	Value
Id	Equals	Aa \$Record > Route Name X

+ Add Condition

11. For How many Records to store Select Only the First Record.

12. For How to Store Record Data select Choose fields and let Salesforce do the rest. Select

Field: Ticket\_Fare\_\_c. Click on Done.

How Many Records to Store

Only the first record

All records

How to Store Record Data

Automatically store all fields

Choose fields and let Salesforce do the rest

Choose fields and assign variables (advanced)

Select Ticket Fare Fields to Store in Variable

Field
Ticket_Fare__c

+ Add Field

Cancel Done

# A CRM Application For Public Transport Management

13. From the Toolbox drag and drop Decision element.

14. Enter the Decision label as "Bus Model Checking".

15. For Outcome Details:

- Label : Ticket Fare Checking
- Outcome API Name : Ticket\_Fare\_Checking
- Condition Requirements to Execute Outcome : All Conditions are Met (AND)
- Resource : {!Fetching\_Route\_Ticket\_Fares.Ticket\_Fare\_c}
- Operator : Is Null
- Value : {!\$GlobalConstant.False}

The screenshot shows the configuration of a Decision element. The 'Label' field is set to 'Bus Model Checking'. The 'API Name' field is set to 'Bus\_Model\_Checking'. The 'Description' field is empty. The 'Outcomes' section shows one outcome named 'Ticket Fare Checking' with an API name 'Ticket\_Fare\_Checking'. The 'Default Outcome' dropdown is set to 'All Conditions Are Met (AND)'. Below it, a condition is defined with a resource of 'Ticket Fare from Fetching\_Route\_Ticket\_Fares > ...', an operator of 'Is Null', and a value of 'False'. The 'Done' button at the bottom right is highlighted.

16. Click Done

17. From the Toolbox drag and drop Update Records element and connect to Decision element for Ticket Fare Fetching Output.

18. Enter the label as "Updating Trip Object Ticket Fare Field".

19. How to Find Records to Update and Set Their Values : Use the trip record that triggered the flow

20. Set Filter Conditions : None -Always Update Record

# A CRM Application For Public Transport Management

21. Field : Ticket\_Fare\_\_c

Value : {!Fetching\_Route\_Ticket\_Fares.Ticket\_Fare\_\_c}

And click Done

The screenshot shows the 'Set Field Values for the Trip Record' section of the Flow Builder. A red box highlights the 'Field' dropdown set to 'Ticket\_Fare\_\_c' and the 'Value' dropdown set to 'Ticket Fare from Fetching\_Route\_Ticket\_Fares > Tick...'. Other sections like 'How to Find Records to Update and Set Their Values' and 'Condition Requirements to Update Record' are also visible.

22. From the Toolbox drag and drop Custom Error Message element and connect to Default Outcome of Decision element..

23. Enter the label as "Route with Bus Model Does not Exists".

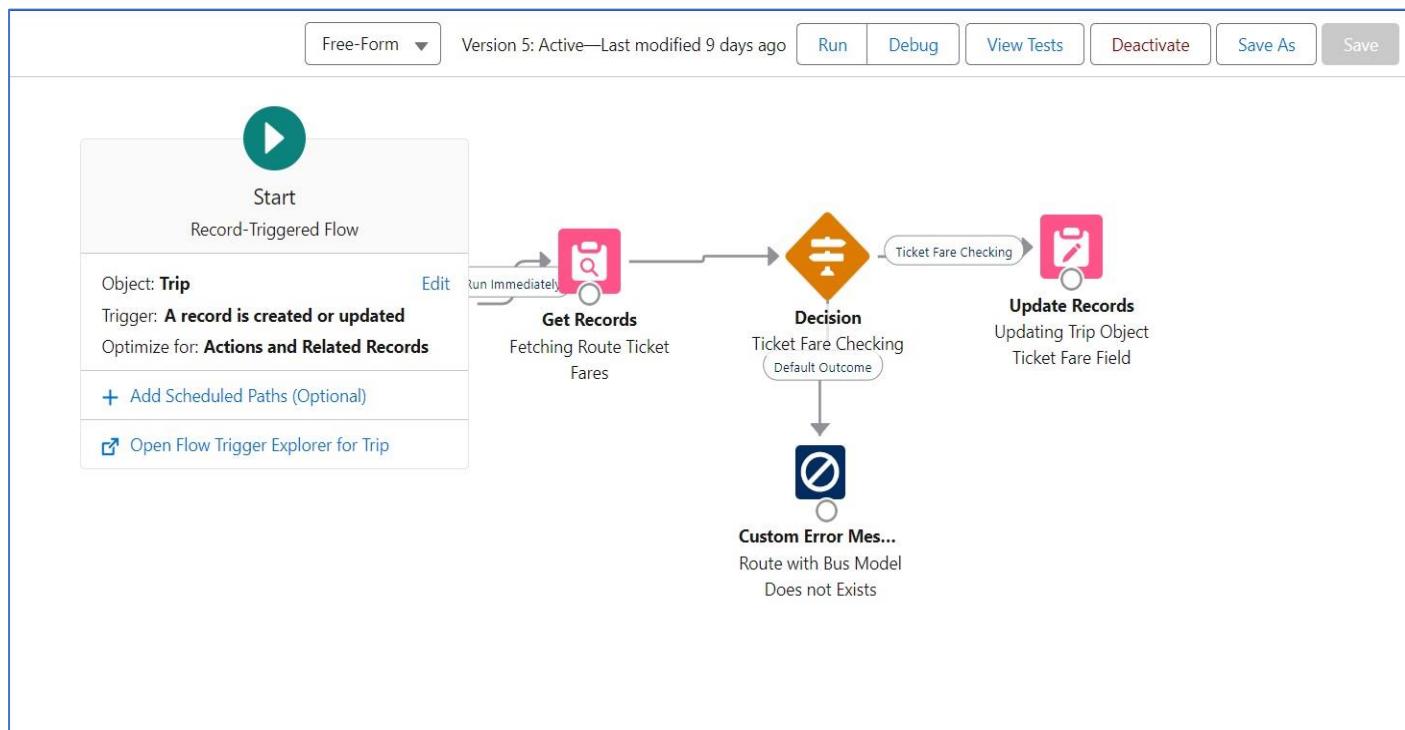
24. For Where to Show the Error Message: Select In a window on a record page

Error Message: There is no Record with the Route {!\$Record.Route\_Name\_\_r.Name} and Bus Model {!\$Record.Bus\_No\_\_r.Model\_\_c} in the Ticket Fares

The dialog box shows the 'Where to Show the Error Message' section with 'In a window on a record page' selected. The 'Error Message' section contains the text 'There is no Record with the Route {!\$Record.Route\_Name\_\_r.Name} and the Bus Model {!\$Record.Bus\_No\_\_r.Model\_\_c} in the Ticket Fares'. The 'Done' button is highlighted with a red box.

# A CRM Application For Public Transport Management

26. Click Done
27. Save the flow as "Fetching Ticket Fare For Bus"
28. Activate the flow.



# A CRM Application For Public Transport Management

## Milestone 9 Triggers

Triggers in Salesforce are pieces of Apex code that execute before or after specific database operations, such as insert, update, delete, or undelete. They allow you to perform custom actions on records in Salesforce when certain events occur. Triggers are particularly powerful for enforcing business logic and automating workflows.

### **Activity 1 : Create a Trigger to validate whether the Driver Id and Conductor Id are correct or not.**

#### **Step 1:** Login to salesforce

Log in to your Salesforce account with administrative privileges.

#### **Step 2:**

i) Navigate to Setup: Once logged in, click on the gear icon ?? (Setup) located at the top-right corner of the page. This will open the Setup menu.

ii) Click on Developer Console: Click on the "Developer Console" option from the Setup menu. This will open the Developer Console in a new browser tab or window.

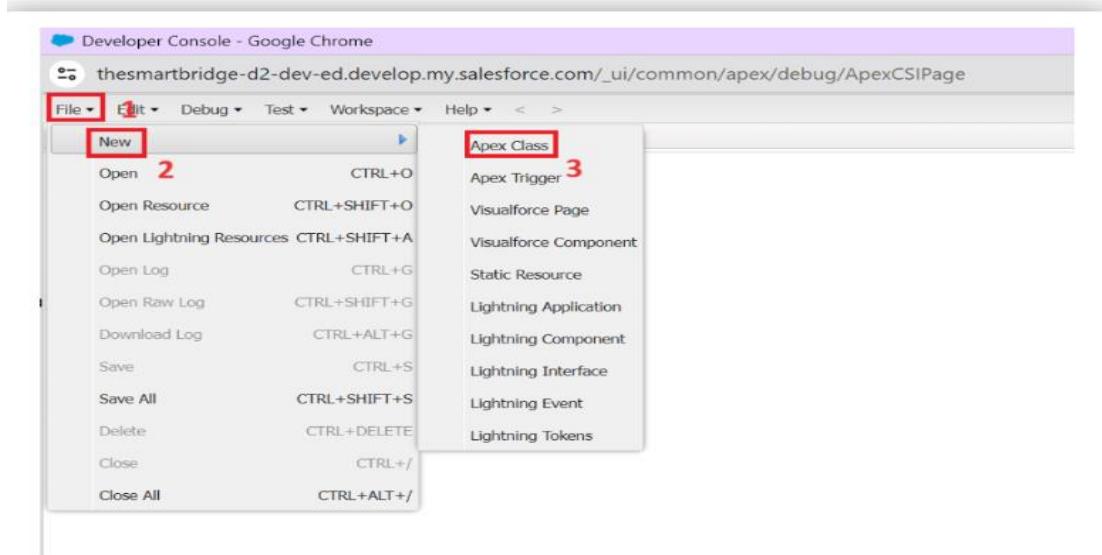
#### **Step 3:**

1) In the Developer Console window, go to the top menu and click on "File".

2) Select New: From the dropdown menu under "File", select "New".

3) Choose Apex Class: In the submenu that appears, select "Apex Class". This will open a new Apex Class editor tab.

# A CRM Application For Public Transport Management



Give Class Name : TripTriggerHandlerClass

Create an Apex Class:

```
Public class TripTriggerHandlerClass {  
    // Checking whether the entered Driver Id belongs to a Driver or not  
    public Static void driverValidation(List<Trip_c> tripList){  
        List<Employee_c> driverList = [SELECT Id, Name FROM Employee_c WHERE Role_c ='Driver'];  
        If(driverList != null){  
            Map<Id, String> driverMap = new Map<Id, String>();  
            for(Employee_c emp : driverList ){  
                driverMap.put(emp.Id, emp.Name);  
            }  
            for(Trip_c trip : tripList ){  
                If(trip.Driver_Id_c!=null){  
                    Boolean hasDriverId = driverMap.containsKey(trip.Driver_Id_c); // hasDriverId will be true  
                    If(hasDriverId == false){  
                        tripaddError('The assigned person is not a Driver.');
```

# A CRM Application For Public Transport Management

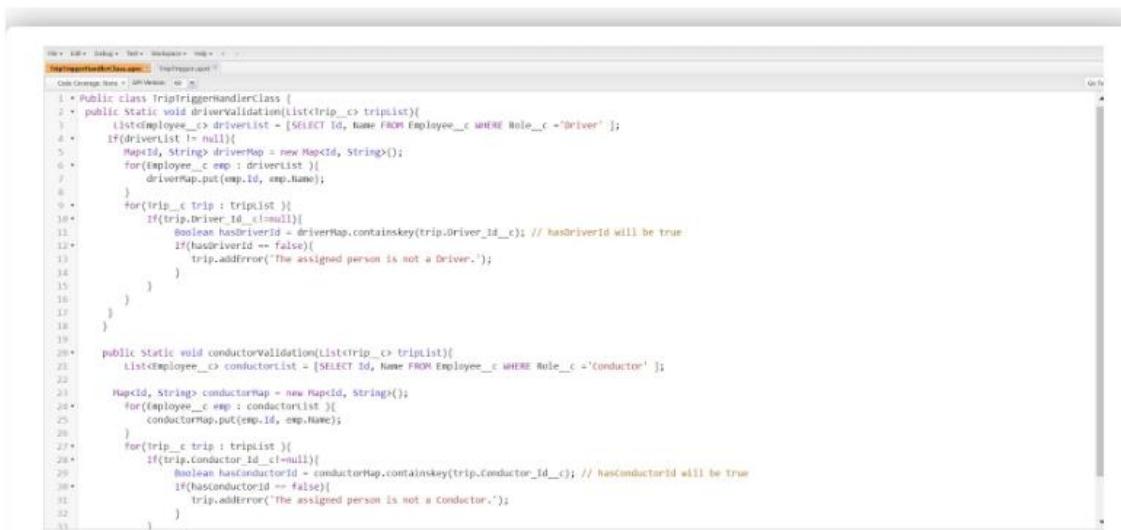
```
// Checking whether the entered conductor Id belongs to a Conductor or not

public Static void conductorValidation(List<Trip_c> tripList){

    List<Employee_c> conductorList = [SELECT Id, Name FROM Employee_c WHERE Role_c
='Conductor'];

    Map<Id, String> conductorMap = new Map<Id, String>();
    for(Employee_c emp : conductorList ){
        conductorMap.put(emp.Id, emp.Name);
    }
    for(Trip_c trip : tripList ){
        If(trip.Conductor_Id_c!=null){
            Boolean hasConductorId = conductorMap.containsKey(trip.Conductor_Id_c); // hasConductorId will be true
            If(hasConductorId == false){
                tripaddError('The assigned person is not a Conductor.');
            }
        }
    }
}
```

# A CRM Application For Public Transport Management



The screenshot shows the Developer Console window with the code for the TripTriggerHandlerClass. The code includes methods for driver validation and conductor validation, both utilizing maps to check if assigned IDs belong to their respective roles.

```
1 * Public class TripTriggerHandlerClass {
2 *     public static void driverValidation(List<Trip_c> tripList){
3 *         List<Employee_c> driverList = [SELECT Id, Name FROM Employee_c WHERE Role__c = 'Driver' ];
4 *         if(driverList != null){
5 *             Map<Id, String> driverMap = new Map<Id, String>();
6 *             for(Employee_c emp : driverList){
7 *                 driverMap.put(emp.Id, emp.Name);
8 *             }
9 *             for(Trip_c trip : tripList){
10 *                 if(trip.Driver_Id__c!=null){
11 *                     Boolean hasDriverId = driverMap.containsKey(trip.Driver_Id__c); // hasDriverId will be true
12 *                     if(hasDriverId == false){
13 *                         trip.addError('The assigned person is not a Driver.');
14 *                     }
15 *                 }
16 *             }
17 *         }
18 *     }
19 *
20 *     public static void conductorValidation(List<Trip_c> tripList){
21 *         List<Employee_c> conductorList = [SELECT Id, Name FROM Employee_c WHERE Role__c = 'Conductor' ];
22 *
23 *         Map<Id, String> conductorMap = new Map<Id, String>();
24 *         for(Employee_c emp : conductorList){
25 *             conductorMap.put(emp.Id, emp.Name);
26 *         }
27 *         for(Trip_c trip : tripList){
28 *             if(trip.Conductor_Id__c!=null){
29 *                 Boolean hasConductorId = conductorMap.containsKey(trip.Conductor_Id__c); // hasConductorId will be true
30 *                 if(hasConductorId == false){
31 *                     trip.addError('The assigned person is not a Conductor.');
32 *                 }
33 *             }
34 *         }
35 *     }
36 * }
```

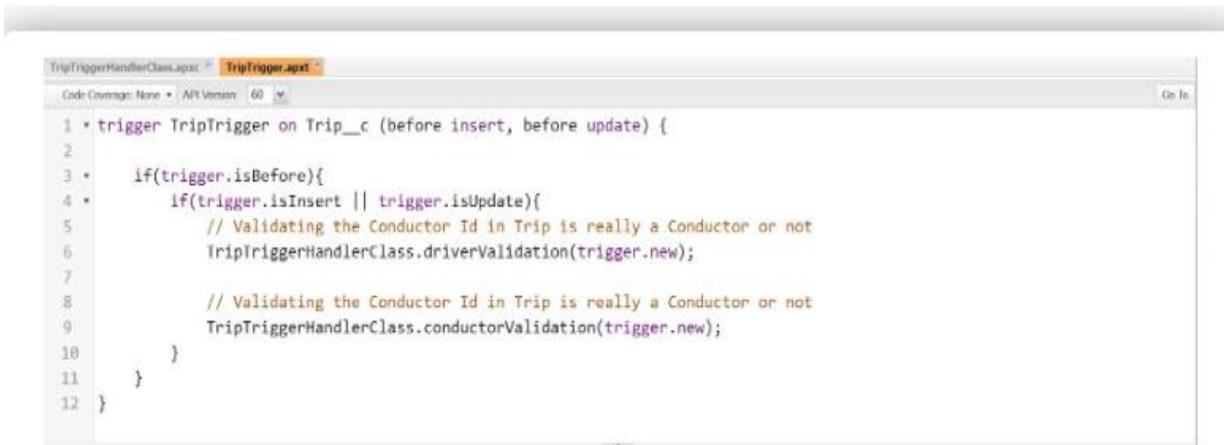
## Step 4:

- i) In the Developer Console window, go to the top menu and click on "File".
- ii) Select New: From the dropdown menu under "File", select "New".
- iii) Choose Apex Class: In the submenu that appears, select "Apex Trigger". This will open a new Apex Trigger editor tab.

Create an Apex Trigger:

```
trigger TripTrigger on Trip_c (before insert, before update) {
    if(trigger.isBefore){
        if(trigger.isInsert || trigger.isUpdate){
            // Validating the Conductor Id in Trip is really a Conductor or not
            TripTriggerHandlerClass.driverValidation(trigger.new);
            // Validating the Conductor Id in Trip is really a Conductor or not
            TripTriggerHandlerClass.conductorValidation(trigger.new);
        }
    }
}
```

# A CRM Application For Public Transport Management



The screenshot shows a code editor window with the title "TripTriggerHandlerClass.apxc" and the tab "TripTrigger.apxt" highlighted. The code is written in Apex and defines a trigger for the "Trip\_\_c" object. The trigger logic checks if the trigger is before insert or update, then validates the conductor and driver IDs.

```
1 * trigger TripTrigger on Trip__c (before insert, before update) {
2
3 *     if(trigger.isBefore){
4 *         if(trigger.isInsert || trigger.isUpdate){
5 *             // Validating the Conductor Id in Trip is really a Conductor or not
6 *             TripTriggerHandlerClass.driverValidation(trigger.new);
7
8 *             // Validating the Conductor Id in Trip is really a Conductor or not
9 *             TripTriggerHandlerClass.conductorValidation(trigger.new);
10    }
11 }
12 }
```

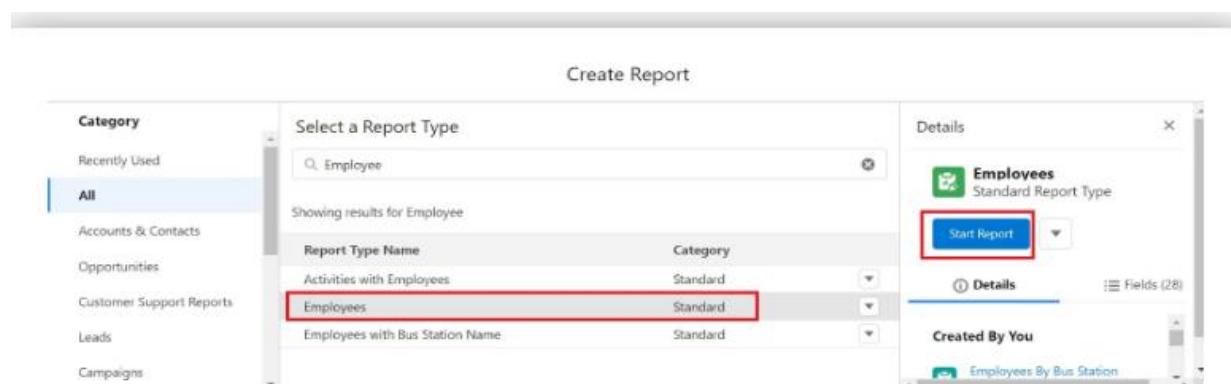
# A CRM Application For Public Transport Management

## Milestone 10 Reports

Reports in Salesforce are tools that allow you to analyze and present your Salesforce data in a structured format. They help you understand and monitor key metrics and trends, providing insights into your business operations. Salesforce reports are highly customizable and can be tailored to meet specific business requirements.

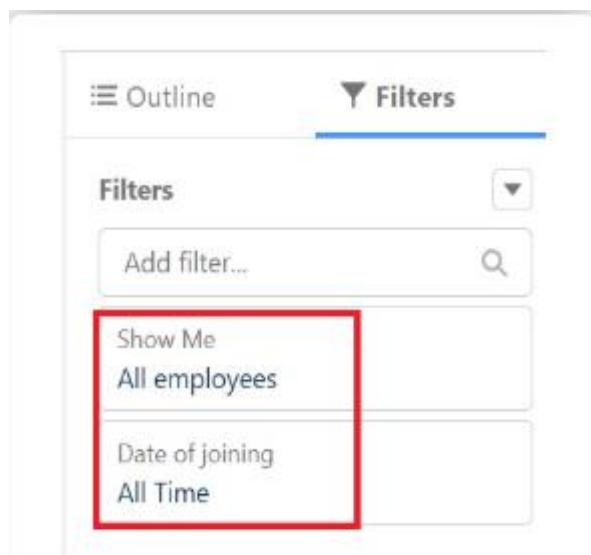
### Activity 1: Create a Employees By Bus Station(Summary) Report

1. Click App Launcher
2. Select Public Transport(RTC) App
3. Click on Reports tab
4. Click on New Report.
5. Click the report type as Employees Click Start report.



# A CRM Application For Public Transport Management

6. Click on Filters and select as follows and click on Apply



7. Customize your report, in group rows select – Bus Station Name, for columns Employee ID, Employee Name, Role (In this way we are making a Summary Report).
8. Click save and run
9. Give report name – Employees By Bus Station
10. Click Save

NOTE: In this report you can see your all record of the object you selected for reporting  
(What you selects in "Select a report type option")

# A CRM Application For Public Transport Management

Bus Station Name	Employee: Employee ID	Employee Name	Role
Dwarka Bus Station (Visakhapatnam) (1)	EMP-0002	Radha	Station Manager
Subtotal			
JBS (3)	EMP-0005	Sravan	Driver
	EMP-0007	Venu	Driver
	EMP-0001	Krishna	Supervisor
Subtotal			
Kempegowda Bus Station(Bangalore) (1)	EMP-0006	Manohar	Driver
Subtotal			
Mahatma Gandhi Bus Station(MGBS) (2)	EMP-0008	Karthik	Driver
	EMP-0009	Madhu	Conductor
Subtotal			
Srinivasa Bus Station (Tirupathi) (2)	EMP-0003	Karthik	Customer Service Representative

## View Report

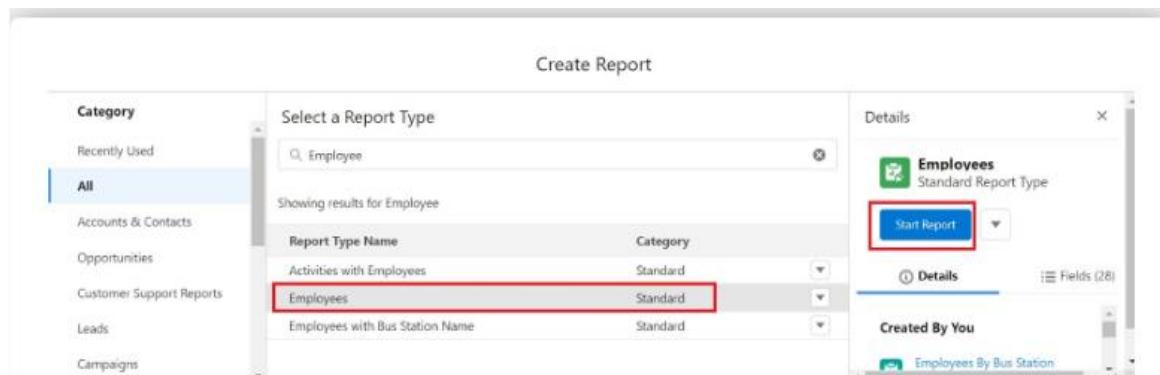
1. Click on App Launcher on the left side of the screen.
2. Search Public Transport(RTC) App & click on it.
3. Click on Reports Tab.
4. Click on Employees By Bus Station and see records.

Bus Station Name	Employee: Employee ID	Employee Name	Role
Dwarka Bus Station (Visakhapatnam) (1)	EMP-0002	Radha	Station Manager
Subtotal			
JBS (3)	EMP-0005	Sravan	Driver
	EMP-0007	Venu	Driver
	EMP-0001	Krishna	Supervisor
Subtotal			
Kempegowda Bus Station(Bangalore) (1)	EMP-0006	Manohar	Driver
Subtotal			
Mahatma Gandhi Bus Station(MGBS) (2)	EMP-0008	Madhu	Conductor
	EMP-0009	Karthik	Driver
Subtotal			
Srinivasa Bus Station (Tirupathi) (2)	EMP-0003	Karthik	Customer Service Representative

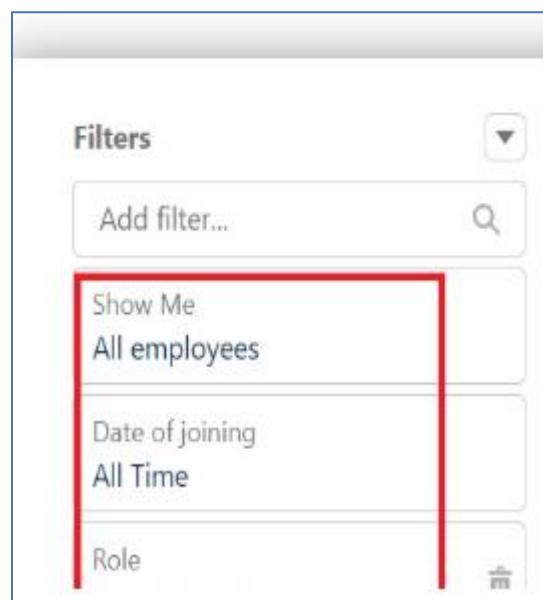
# A CRM Application For Public Transport Management

## Activity 2: Create a Drivers and Conductors Information Report

1. Click App Launcher
2. Select Public Transport(RTC) App
3. Click on Reports tab
4. Click on New Report.
5. Click the report type as Employees Click Start report.



6. Click on Filters and select as follows and click on Apply



# A CRM Application For Public Transport Management

7. Customize your report, in group rows select – Bus Station Name, for columns Employee ID, Employee Name, Role (In this way we are making a Summary Report).

8. Click save and run

9. Give report name – Drivers And Conductors Information

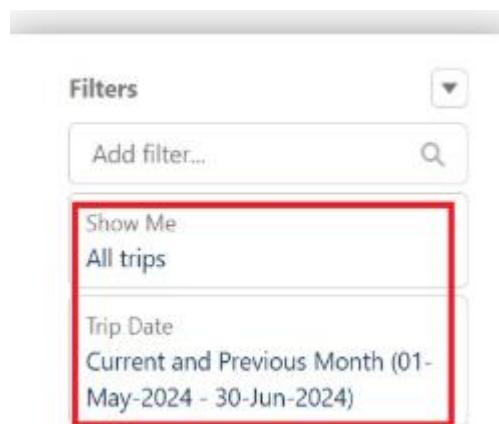
10. Click Save

Bus Station Name	Employee: Employee ID	Employee Name	Role
JBS (2)	EMP-0007 EMP-0005	Venu Sravan	Driver
Subtotal			
Kempegowda Bus Station(Bangalore) (1)	EMP-0006	Manohar	Driver
Subtotal			
Mahatma Gandhi Bus Station(MGBS) (2)	EMP-0008 EMP-0009	Karthik Madhu	Driver Conductor
Subtotal			
Srinivasa Bus Station (Tirupathi) (1)	EMP-0004	Bhuvan	Driver
Subtotal			
Total (6)			

## Activity 3: Create a Previous and Current Month Trip Details Report

1. Click App Launcher
2. Select Public Transport(RTC) App
3. Click on Reports tab
4. Click on New Report.
5. Click the report type as Trips Click Start report.
6. Click on Filters and select as follows and click on Apply

# A CRM Application For Public Transport Management



7. Customize your report, in group rows select – Trip Date, Bus No , for columns Trip No, Route Name, Passenger Count, Total Amount (In this way we are making a Summary Report).

8. Click save and run

9. Give report name – Previous And Current Month Trips Details

10. Click Save

Trip Data	Bus No	Trip No	Route Name	Passenger Count	Total Amount
05/05/2024	IN BU AC 0729	1	Ecll X Roads - Gakkar-M	90	₹12,600.00
		1	Ecll X Roads - Gakkar-M	100	₹14,000.00
		1	Ecll X Roads - Gakkar-M	200	₹30,000.00
06/06/2024	TS 02 AC 0229	1	Ecll X Roads - Gakkar-M	100	₹14,000.00
		1	Ecll X Roads - Gakkar-M	150	₹16,000.00
		1	Ecll X Roads - Gakkar-M	250	₹30,000.00
		Total		950	₹38,000.00

# A CRM Application For Public Transport Management

## Milestone 11 Dashboards

Dashboards in Salesforce are visual representations of your reports and key metrics, providing a consolidated view of your data. They allow you to monitor performance, track progress, and make informed decisions at a glance. Dashboards are composed of various components such as charts, tables, gauges, and metrics, each displaying data from one or more reports.

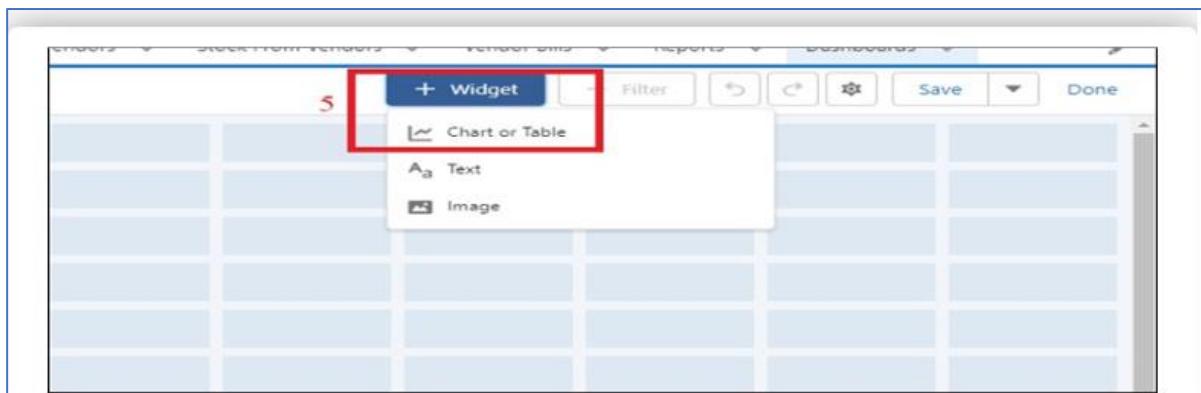
### Activity 1: - Create Dashboard

1. Click on the Dashboards tab from the Public Transport(RTC) application.
2. Click on the new dashboard.

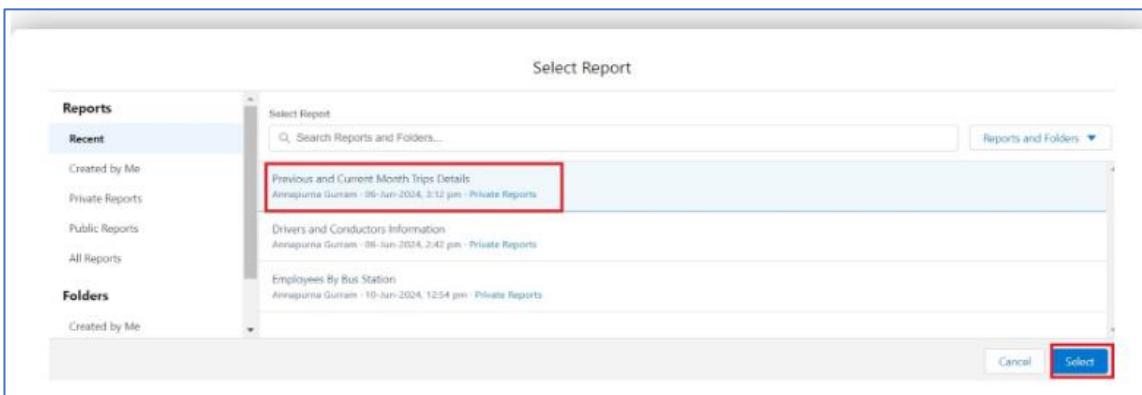
The screenshot shows a 'New Dashboard' dialog box. At the top, it says 'New Dashboard'. Below that is a 'Name' field containing 'Public Transport DashBoard', which is enclosed in a red rectangular box. There is also a 'Description' field and a 'Folder' section with 'Private Dashboards' selected. To the right of the folder dropdown is a 'Select Folder' button. At the bottom right of the dialog are two buttons: 'Cancel' and 'Create', with the 'Create' button also enclosed in a red rectangular box.

3. Give name - Public Transport DashBoard
4. Click create
5. Click on +widget
6. Select the Previous and Current Month Trips Details Report
7. For the data visualization select any of the charts, tables etc. as per your choice/requirement
8. Click add.
9. Click on +widget

# A CRM Application For Public Transport Management



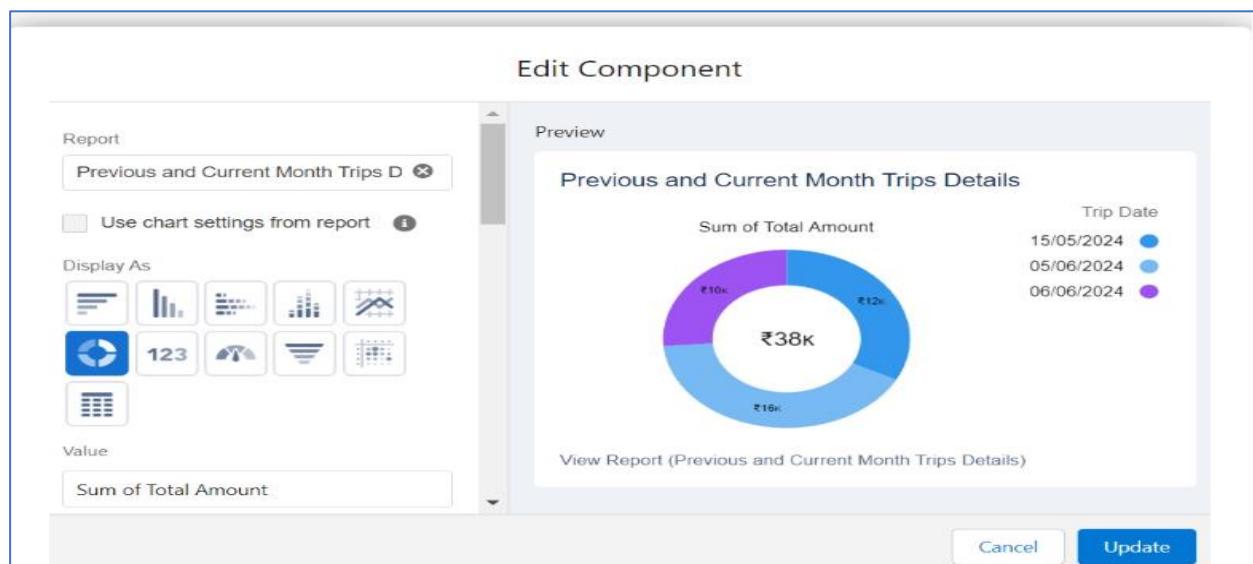
10. Select the Drivers and Conductors Information Report



11. For the data visualization select any of the charts, tables etc. as per your choice/requirement

12. Click add.

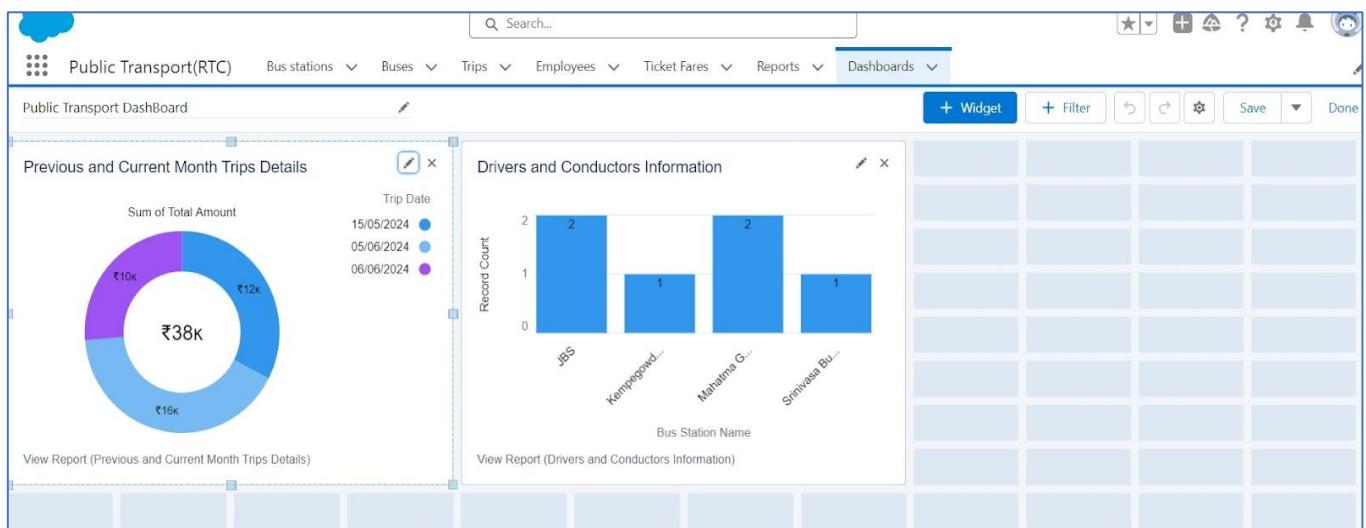
13. Click save.



# A CRM Application For Public Transport Management

## Activity 2: View Dashboard

1. Click on App Launcher on the left side of the screen.
2. Search Public Transport(RTC) & click on it.
3. Click on Dashboard Tab.
4. Click on Public Transport DashBoard see graph view of records



# A CRM Application For Public Transport Management

## OUT PUTS:

### Bus station:-

New Bus Station

\* = Required Information

**Information**

\* Bus Station Name

**Amenities**

Please fill out this field.

Available	Chosen
Accessibility	
Waiting Area	
Information and S...	
Food and Drink	

Shelter available

Bench

Bus Stop Category

--None--

Owner

 sanjay polisetty

- Address Information:

Address Information

Street

State/Province

\* City

Zip/PostalCode

Location

**Cancel** **Save & New** **Save**

# A CRM Application For Public Transport Management

## Buses:

**New Bus**

\* = Required Information

<b>Information</b>	
Bus Station Name <input type="text" value="Search Bus Stations..."/> 	Owner  sanjay polisetty
* Category <input type="text" value="--None--"/> 	
<a href="#">View all dependencies</a>	
Model <input type="text" value="--None--"/> 	
<a href="#">View all dependencies</a>	
* Capacity <input type="text"/>	
* Bus Registration No <input type="text"/>	

## Employees:

- Information:

**New Employee**

\* = Required Information

<b>Information</b>	
* Employee Id <input type="text"/>	Work Place <input type="text"/>
Bus Station Name <input type="text" value="Search Bus Stations..."/> 	* Role <input type="text" value="--None--"/> 
Salary <input type="text"/>	Date of Joining <input type="text"/> 
Owner  sanjay polisetty	

# A CRM Application For Public Transport Management

- Personal Details:

Personal Details

Date of Birth

Phone

- Address:

Address

Street

State/Province

City

Country

Zip/PostalCode

[Cancel](#) [Save & New](#) [Save](#)

# A CRM Application For Public Transport Management

## Ticket fares:

New Ticket Fare

\* = Required Information

Information	
* Route Name	Owner
<input type="text"/>	sanjay polisetty
Ticket Fare	
<input type="text"/>	
Bus Model	
--None--	
<input type="button" value="Cancel"/> <input type="button" value="Save &amp; New"/> <input type="button" value="Save"/>	

- Information:

*Trip Date	Conductor Id
<input type="text"/>	<input type="text"/> Search Employees...
Format: 12/31/2024	
*Trip No	Owner
<input type="text"/>	sanjay polisetty
*Bus No	
<input type="text"/> Search Buses...	
Driver Id	
<input type="text"/> Search Employees...	

- Bus Schedule:

Bus Schedule

*Route Name	Estimated Travel Time
<input type="text"/> Search Ticket Fares...	<input type="text"/>
*Bus Starting Terminal	*Destination Terminal
<input type="text"/>	<input type="text"/>
Departure Time	Arrival Time
--None--	--None--
No. of Stops	Frequency Per Day
<input type="text"/>	<input type="text"/>

# A CRM Application For Public Transport Management

- Passenger Information:

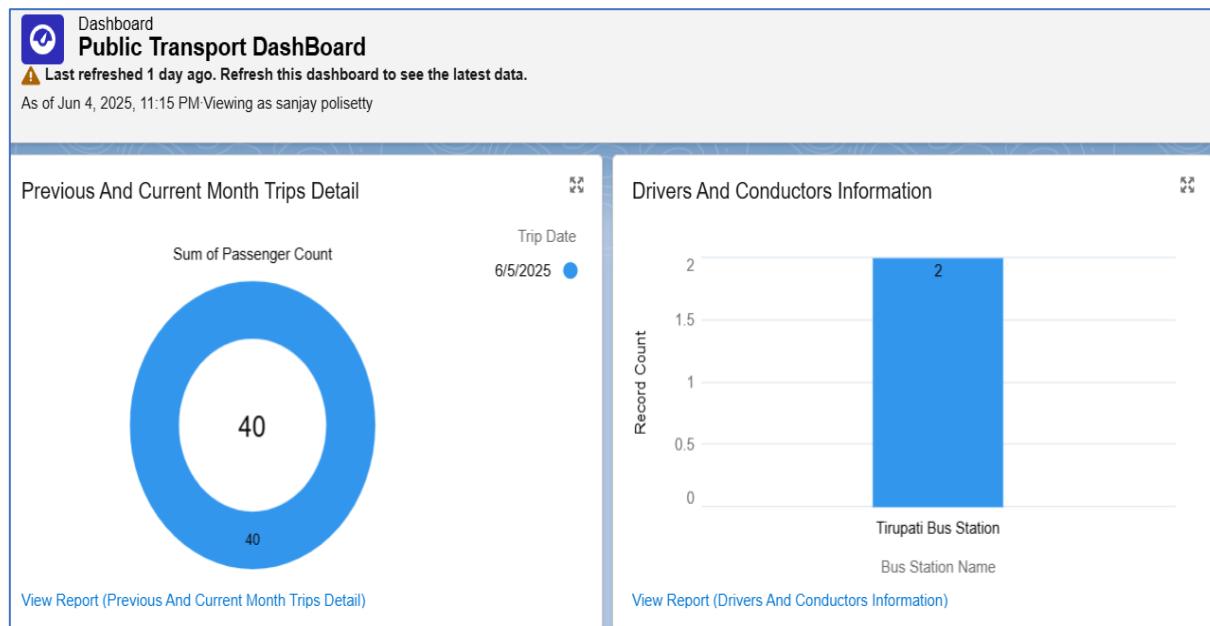
Passenger Information

* Passenger Count	Ticket Fare
<input type="text"/>	<input type="text"/>
<input type="button" value="Cancel"/> <input type="button" value="Save &amp; New"/> <input type="button" value="Save"/>	

## Reports:

Reports						
Recent						
4 items						
REPORTS	Report Name	Description	Folder	Created By	Created On	Subscribed
Recent	Previous And Current Month Trips Detail		Private Reports	sanjay polisetty	6/4/2025, 9:13 PM	<input type="checkbox"/>
Created by Me	Employees By Bus Station		Private Reports	sanjay polisetty	6/3/2025, 11:42 PM	<input type="checkbox"/>
Private Reports	Drivers And Conductors Information		Private Reports	sanjay polisetty	6/4/2025, 9:06 PM	<input type="checkbox"/>
Public Reports	Which flows run, what's the status of each interview, and how long do users take to complete the screens?					
All Reports	Sample Flow Report: Screen Flows	Public Reports	Automated Process	4/30/2025, 12:24 AM		<input type="checkbox"/>
FOLDERS						

## Dashboards:



# A CRM Application For Public Transport Management

## **Milestone 12**

### **Conclusion**

By implementing this Salesforce-based Public Transport( RTC) Management System, the RTC department can significantly improve its operational efficiency, data management, and overall service quality to passengers. Data-driven decision-making capabilities for management. Accurate and up-to-date records of all operational data.