**Food Connect**

Salesforce Project

**1.Object**

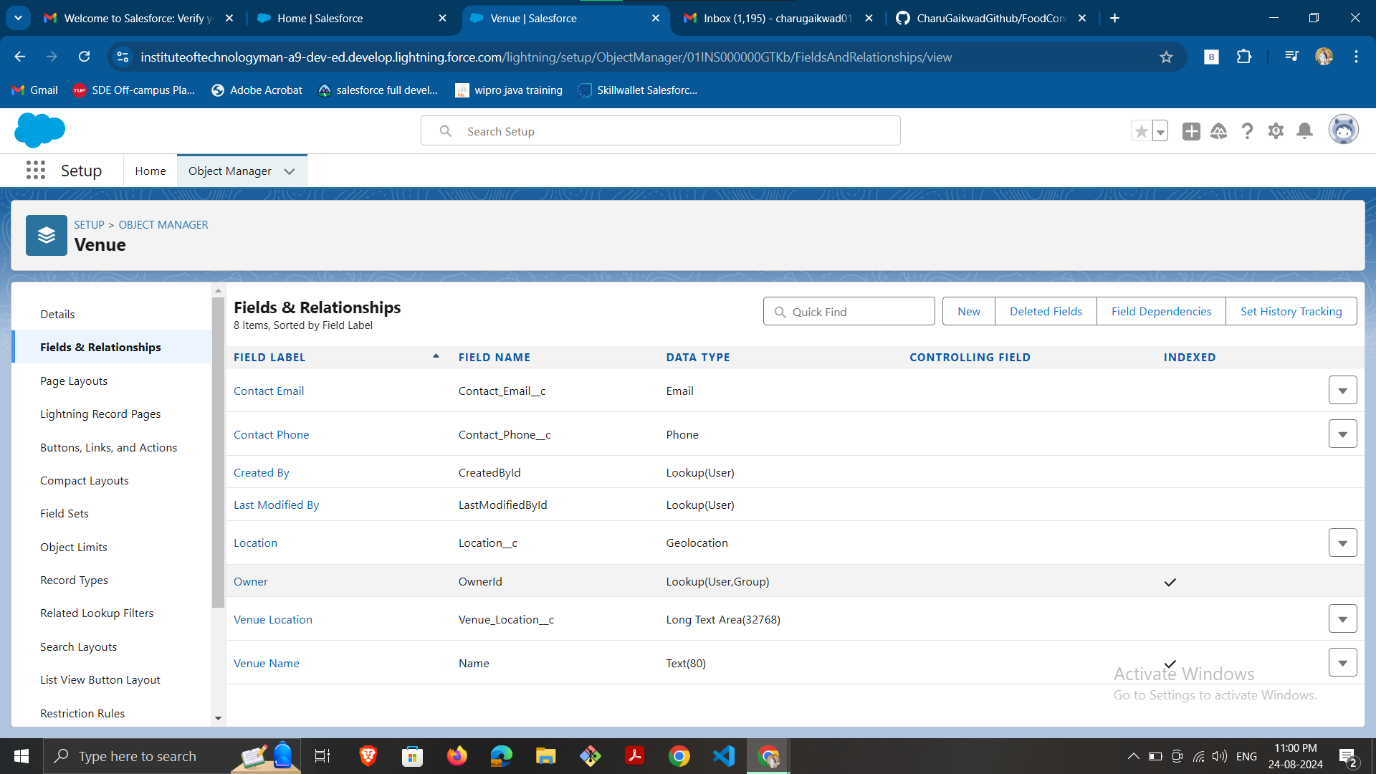
To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >>Click on Custom Object.
2. Enter the label name >> Venue
3. Plural label name >> Venues
4. Enter Record Name Label and Format

* Record Name >> Venue Name
* Data Type >> Text

1. Click on Allow reports and Track Field History,Allow Activities.
2. Allow search >> Save.

**Output:**

****

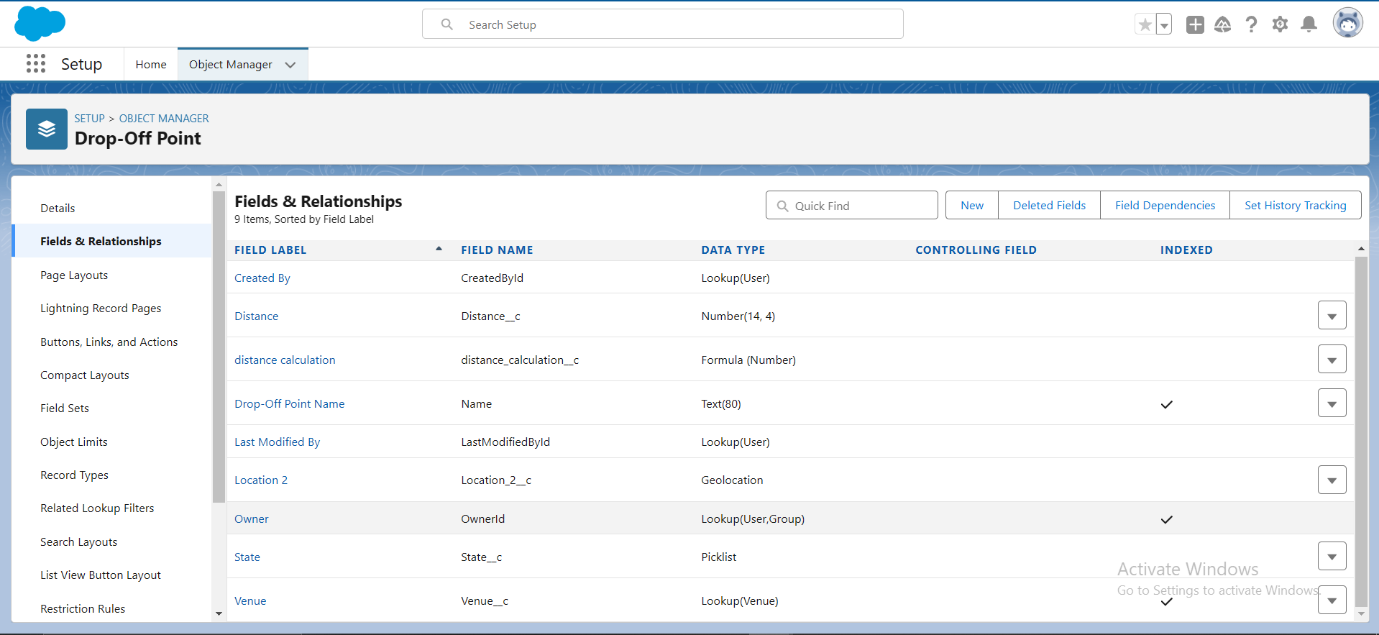
To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
2. Enter the label name >> Drop-Off Point
3. Plural label name>> Drop-Off Points
4. Enter Record Name Label and Format

* Record Name >> Drop-Off point Name
* Data Type >> Text

1. Click on Allow reports and Track Field History,Allow Activities
2. Allow search >> Save.

**Output:**



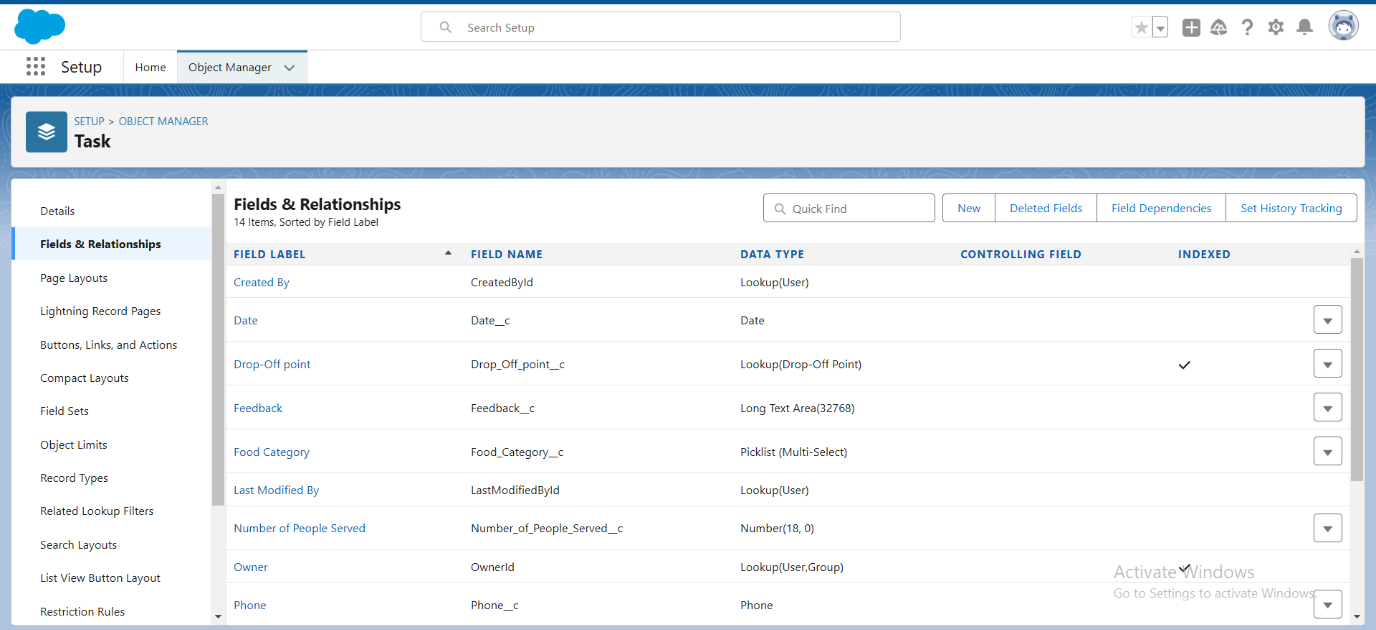
To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
2. Enter the label name>> Task
3. Plural label name>> Tasks
4. Enter Record Name Label and Format

* Record Name >> Task Name
* Data Type >> Text

1. Click on Allow reports and Track Field History,Allow Activities
2. Allow search >> Save.

**Output:**



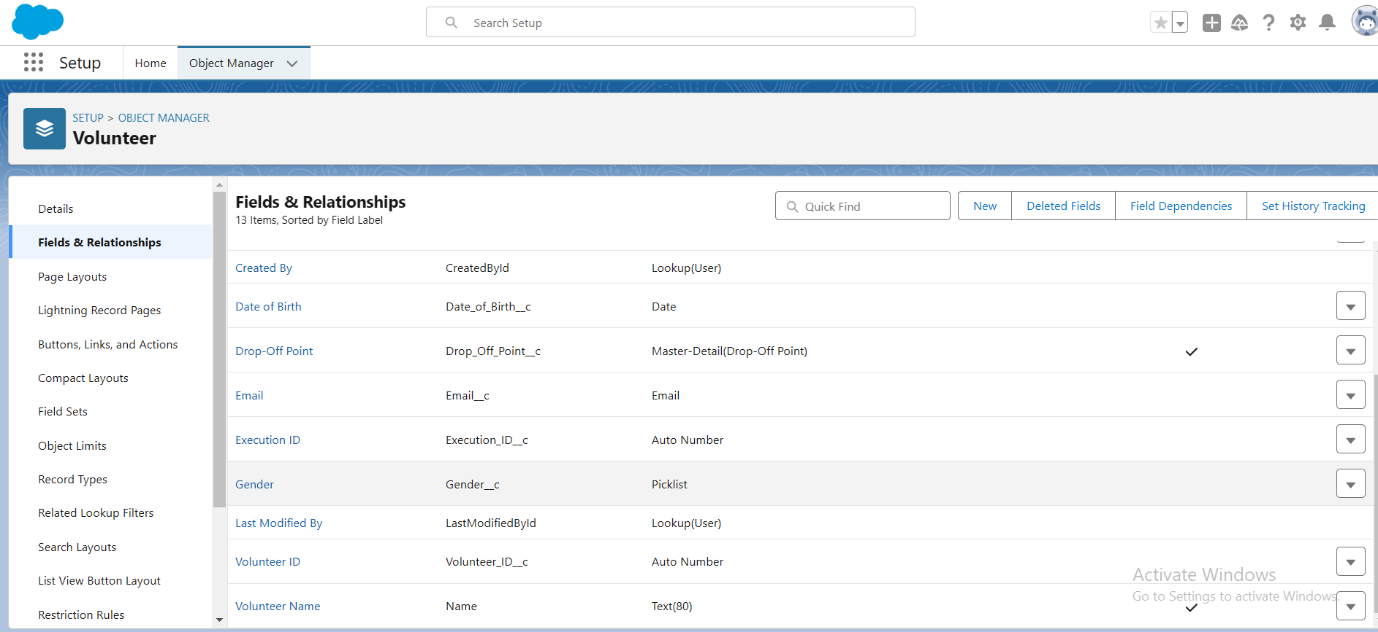
To create an object:

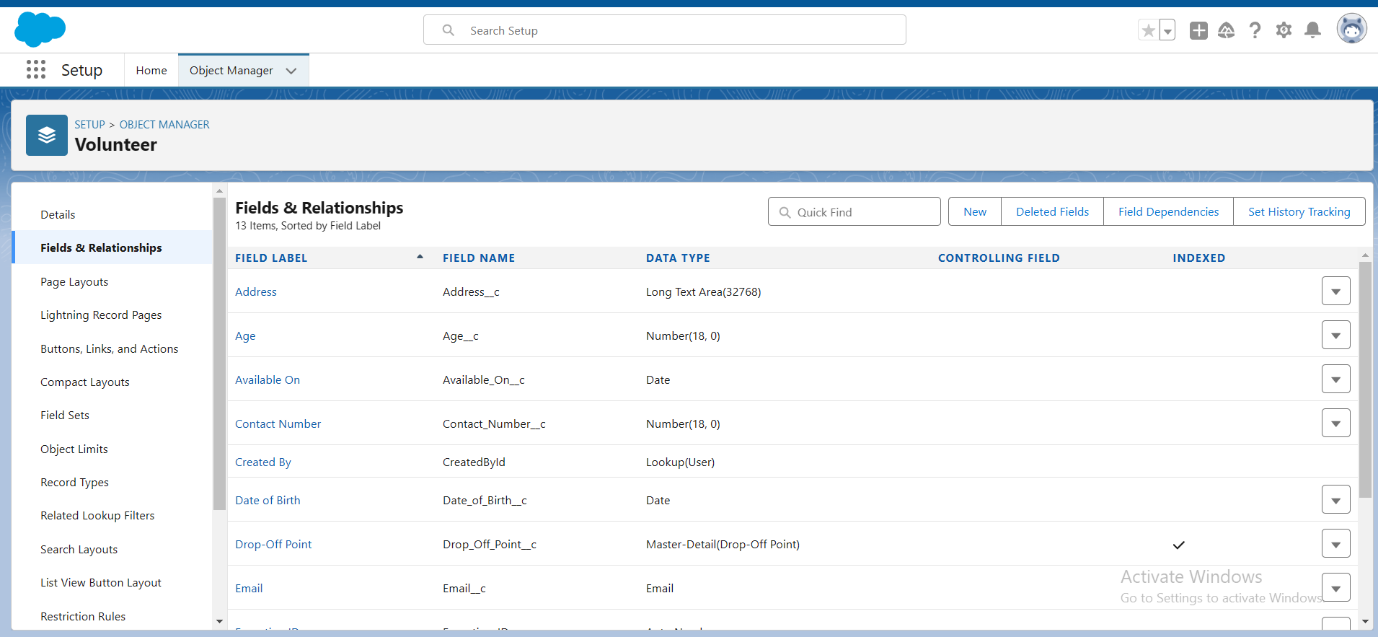
1. From the setup page >> Click on Object Manager>> Click on Create >> Click on Custom Object.
2. Enter the label name>> Volunteer
3. Plural label name>> Volunteers
4. Enter Record Name Label and Format

* Record Name >> Volunteer Name
* Data Type >> Text

1. Click on Allow reports and Track Field History, Allow Activities
2. Allow search >> Save.

**Output:**





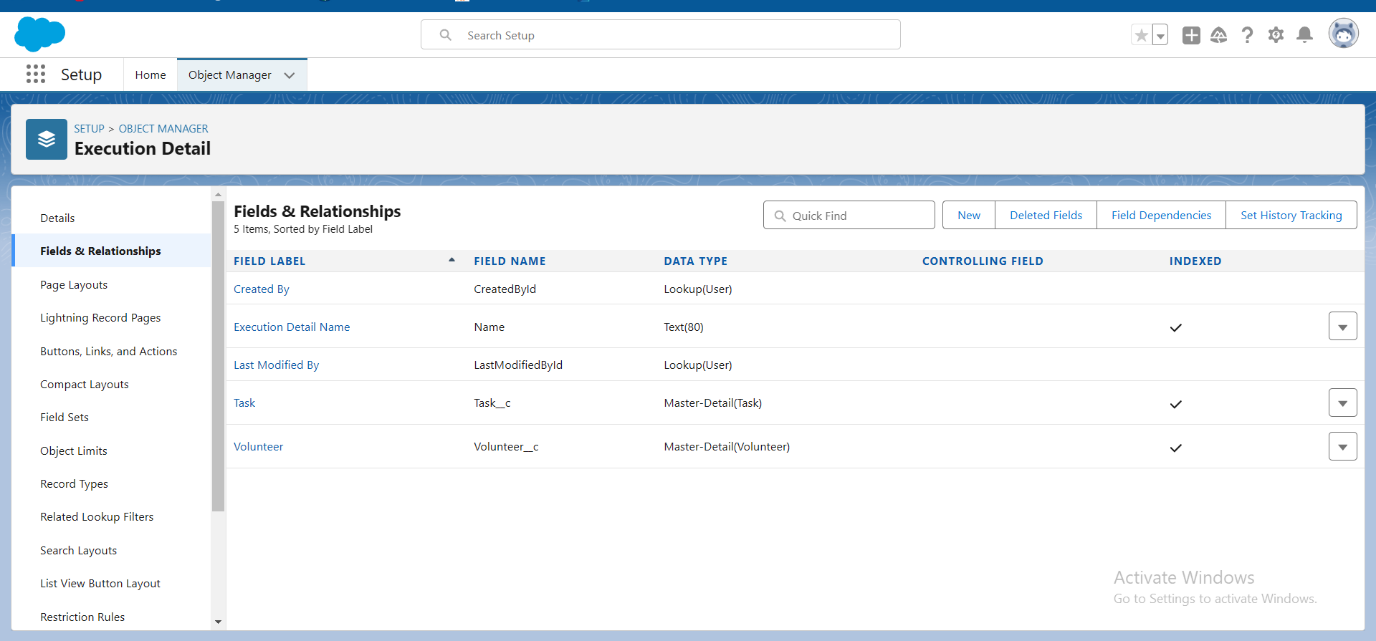
To create an object:

1. From the setup page >> Click on Object Manager >> Click on Create >> Click on Custom Object.
2. Enter the label name >> Execution Detail
3. Plural label name >> Execution Details
4. Enter Record Name Label and Format

* Record Name >> Execution Detail Name
* Data Type >> Text

1. Click on Allow reports and Track Field History, Allow Activities
2. Allow search >> Save.

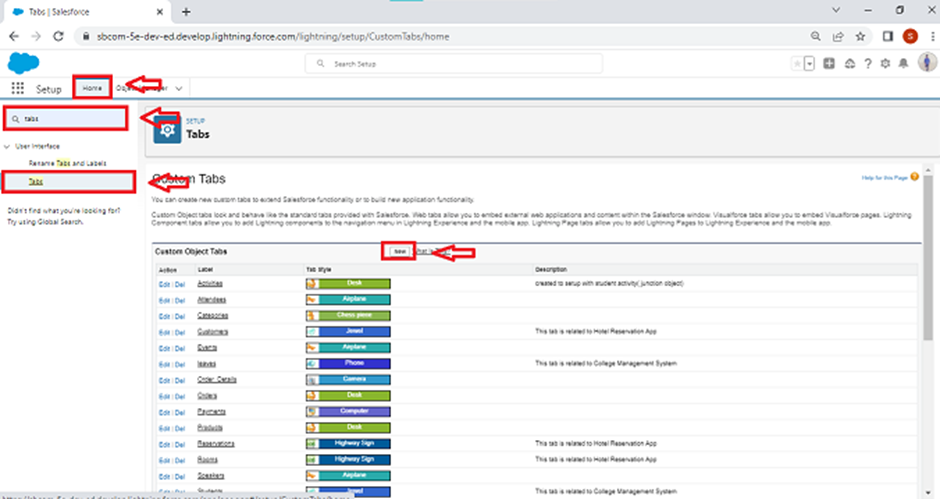
**Output:**



**2.Tabs**

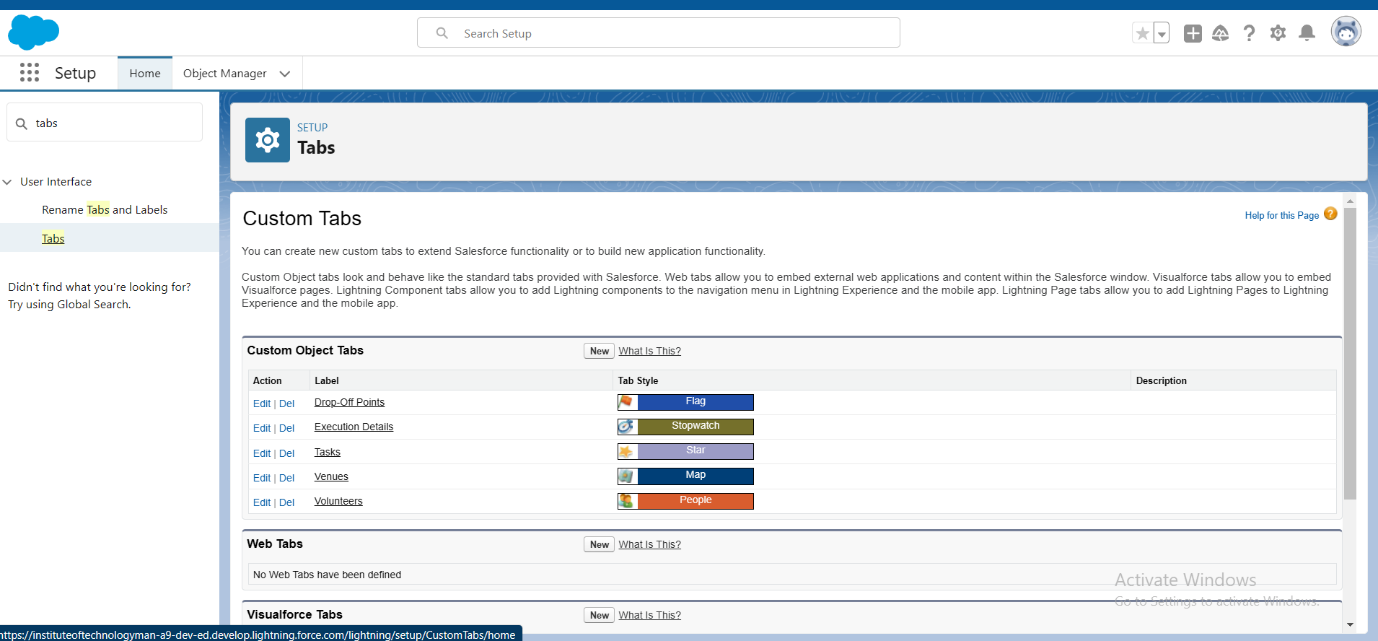
To create a Tab:(Venue)

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



1. Select Object(Venue) >> Select the tab style >> Next (Add to profiles page) keep it as default >> Next (Add to Custom App)  uncheck the include tab .
2. Make sure that the Append tab to users' existing personal customizations is checked.
3. Click save

**Output:**



**3.Flows**

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

A screenshot of a computer

Description automatically generated

1. Click on the ‘+’ icon in between start and end, and click on screen element.
2. Under the Screen Properties:

Label : Venue Details

API Name : Venue\_Details

1. Now lets add components in this flow. Click on Text Component and name it as:

Label : Venue Name

API Name : Venue\_Name

1. Click on Email Component and name it as:

Label : Email

API Name : Contact\_Email

1. Click on Phone Component and name it as:

Label : Phone

API Name : Contact\_Phone

1. Click on Text Component and name it as:

Label : Venue Location

API Name : Venue\_Location

1. Click on Number Component and name it as:

Label : Latitude

API Name : Latitude

1. Click on Number Component and name it as:

Label : longitude

API Name : longitude

1. Next click on Done. This would like below

A screenshot of a computer

Description automatically generated

1. Click on the ‘+’ icon in between Venue details and end, and click on create record element.
2. Now label it as

Label : Create Venue Record

API Name : Create\_Venue\_Record

How Many Records to Create : One

How to Set the Record Fields : Use separate resources, and literal values

Object : Venue

Set Field Values for the Venue : Click on ‘Add Field’ 5 times

Field : Value = Contact\_Email\_\_c : {!Contact\_Email.value}

Field : Value = Contact\_Phone\_\_c : {!Contact\_Phone.value}

Field : Value = Name : {!Venue\_Name}

Field : Value = Venue\_Location\_\_c : {!location}

Field : Value = Location\_\_Latitude\_\_s : {!latitude}

Field : Value = Location\_\_Longitude\_\_s : {!longitude}

1. This would look like:

A screenshot of a computer

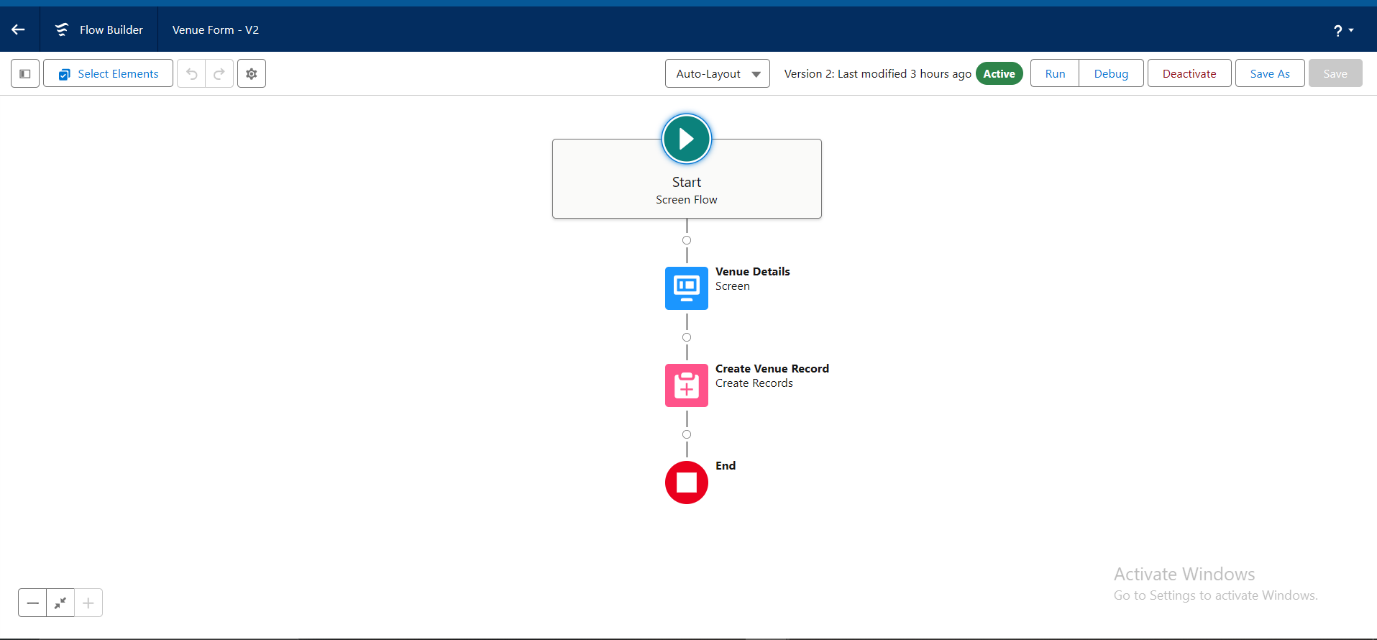
Description automatically generated

1. Click on Save as:

Flow Label : Venue Form

Flow API Name : Venue\_Form

**Output:**

****

**4.Trigger**

1. Log into the trailhead account, navigate to the gear icon in the top right corner.
2. Click on developer console and you will be navigated to a new console window.
3. Click on the File menu in the toolbar, and click on new >> Trigger.
4. Enter the trigger name and the object to be triggered.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

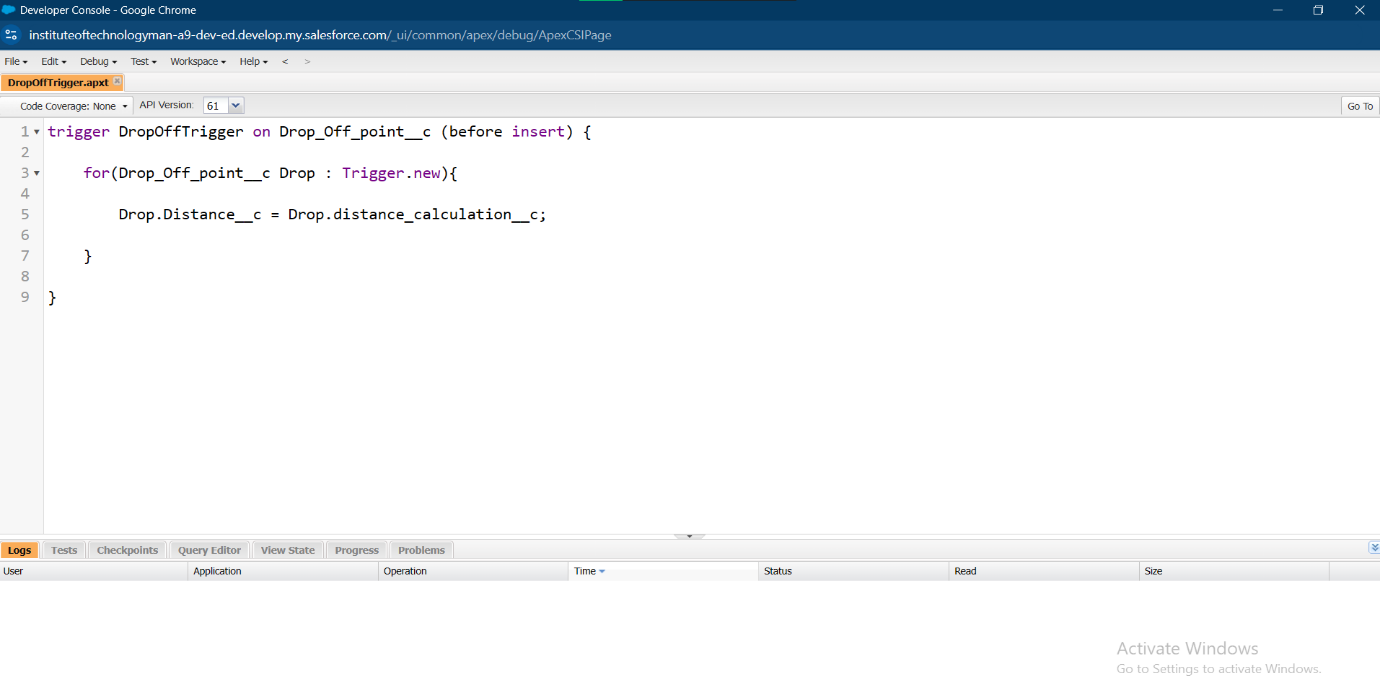
Description automatically generated

1. Enter Name : DropOffTrigger

sObject: Drop-Off Point

1. Click on Submit.

**Output:**



**5.user creation**

1. Go to setup page >> type users in Quick Find bar >> click on users>> New user.
2. In General Information give details as: (Note : create users as per your wish NGO’s)

First Name : Iksha Foundation

Last Name : Iksha\_Foundation

Alias : iiksh

Email : Give Your Email

Username :[ikshafoundation@sb.com](mailto:ikshafoundation@sb.com)(give the username different)

Nickname : Auto Populated

User License : Salesforce Platform

Profile : NGOs Profile

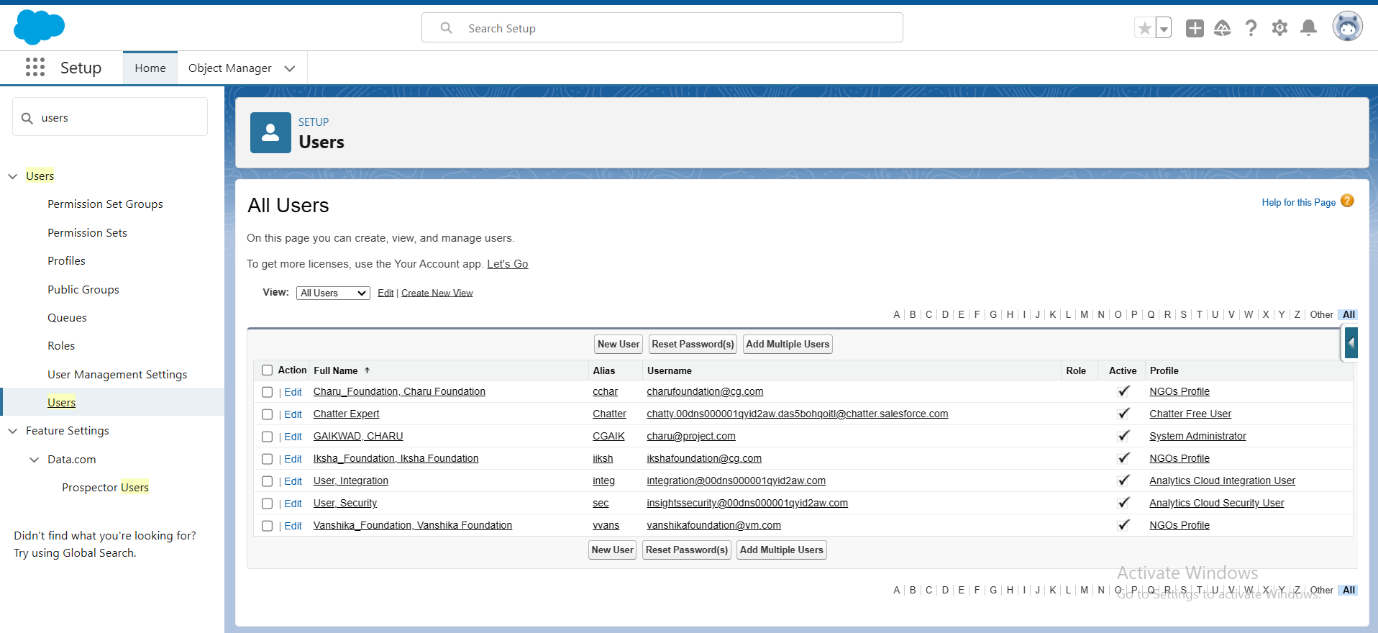
Active : Check

A screenshot of a computer

Description automatically generated

1. Click on Save

**Output:**

****

**6.Public groups**

1. Go to setup page  >>  type Public Groups in Quick Find bar  >>  click on Public Groups >>  click on New.
2. Under Group Information:

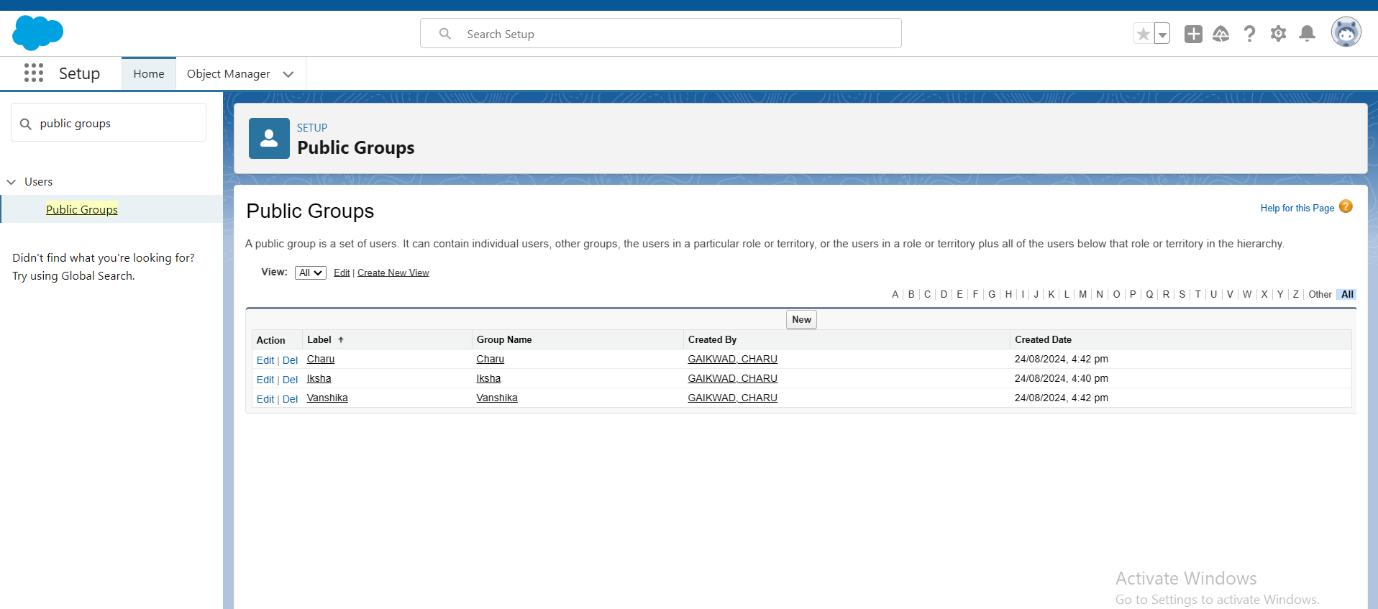
Label : Iksha

Group Name : Iksha

Grant Access Using Hierarchies : Check

1. In Search, Select Users.
2. In Selected Members Add Iksha Foundation and System Administrator

**Output:**

****

**7.Report Types**

1. Go to setup page  >>  type Report Types in Quick Find bar  >> click on Report Types  >>  click on Continue   >>  Click on New Custom Report Type.
2. In Define the Custom Report Type:

Primary Object : Select Venues

Report Type Label : Venue with DropOff with Volunteer

Report Type Name : Venue\_with\_DropOff\_with\_Volunteer

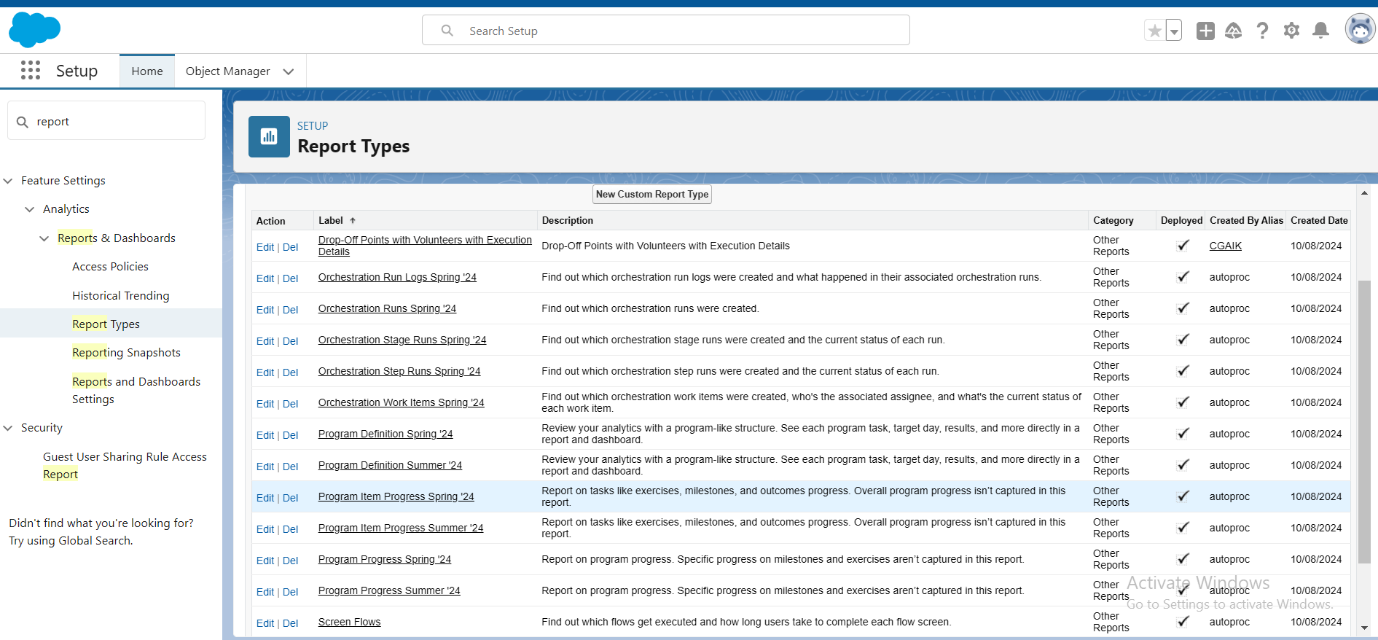
Description : Venue with DropOff with Volunteer

Store in Category : Select Other Reports

Deployment Status : Deployed

1. Click on Next
2. Near Click to relate another Object Select Drop-Off Points.
3. And also select "A" records may or may not have related "B" records.
4. Now again Near Click to relate another Object Select Volunteers.
5. Now click on Save.

**Output:**

****

**8.Reports**

1. Go to the app(FoodConnect)  >>  click on the reports tab
2. Click on New Folder.

Folder Label : Custom Reports

Folder Unique Name : CustomReports

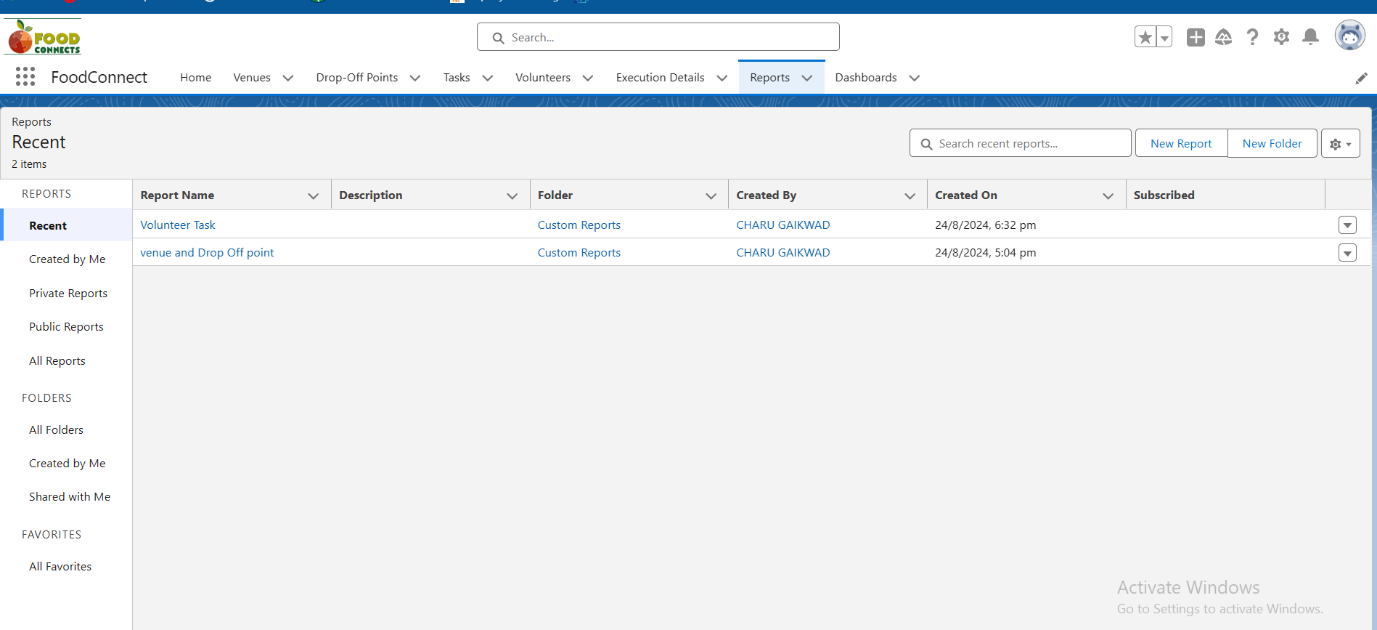
1. Open Custom Reports and click on New Report
2. Select Report Type : Venue with DropOff with Volunteer
3. Then click on Start Report.
4. In GROUP ROWS : Add Volunteer Name
5. In Columns : Add Venue Name, Drop-Off point Name, Distance.

A screenshot of a computer

Description automatically generated

1. Now click on Save & Run.
2. Give Label as :
3. Report Name : venue and Drop Off point
4. Report Unique Name : Auto Populated
5. Click on Select Folder and select Custom Report, then click on Save.

**Output:**

****

**9.Dashboards**

1. Go to the app(FoodConnect)  >>  click on the Dashboards tab.
2. Click on New Folder.

Folder Label : Custom Dashboards

Folder Unique Name : Auto Populated

1. Open Custom Dashboards and click on New Dashboards
2. Name : Organization Details
3. Click on Widget and select Chart or Table
4. In Select Report : Select venue and Drop Off point Report.
5. Then click on select
6. In Add Component:

Display As : Select Lightning Table

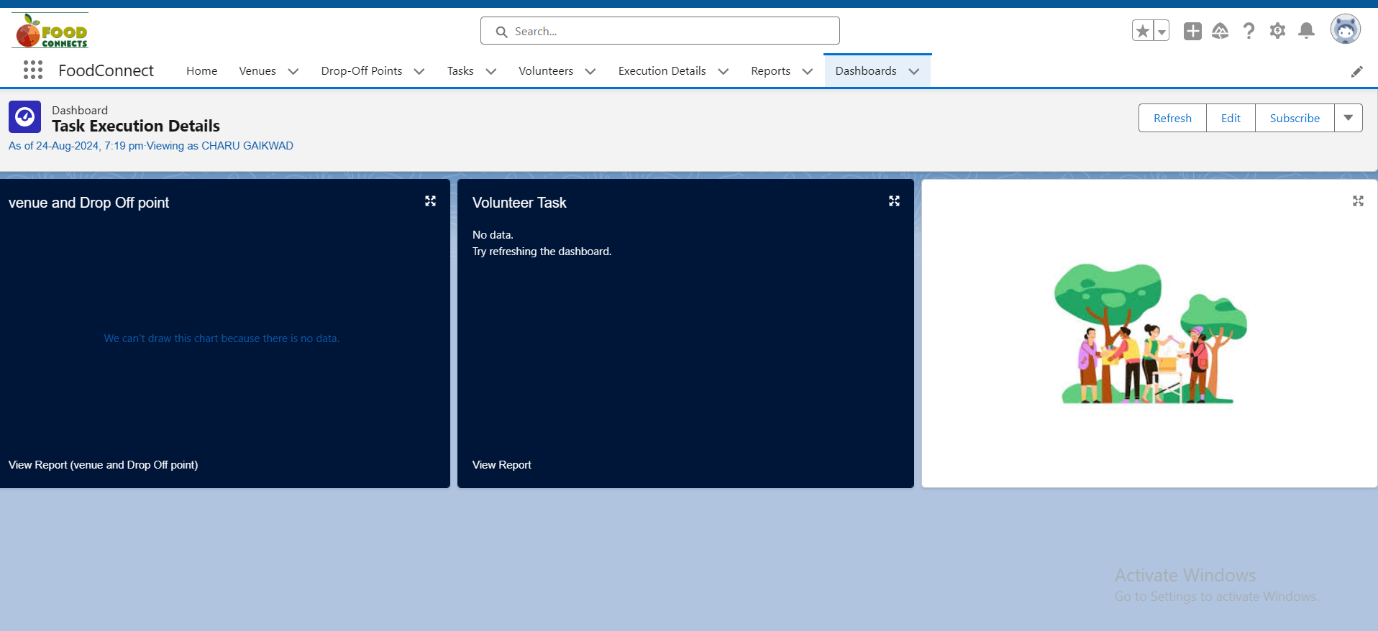
Component Theme : Select Dark (Optional)

A screenshot of a computer

Description automatically generated

1. Now click on save.

**Output:**

****

**10.Sharing Rules**

1. Go to setup  >>  type Sharing Settings in quick find box  >>  Click on the Sharing Settings.
2. Scroll down and find Drop-Off point Sharing Rules.
3. Click on new near Drop-Off point Sharing Rules and Name it as:

Label : Rule 1

Rule Name : Rule\_1

1. Select your rule type : Select Based on criteria.
2. Select which records to be shared:

Field : Operator : Value = Distance : less than : 15

1. Select the users to share with : Near Share With

Public Groups : Iksha

1. Click on Save.
2. Click on new near Drop-Off point Sharing Rules and Name it as:

Label : Rule 2

Rule Name : Rule\_2

1. Select your rule type : Select Based on criteria.
2. Select which records to be shared:

Field : Operator : Value = Distance : greater than : 15

Field : Operator : Value = Distance : less or equal : 30

1. Select the users to share with : Near Share With

Public Groups : NSS

1. Click on Save.
2. Click on new near Drop-Off point Sharing Rules and Name it as:

Label : Rule 3

Rule Name : Rule\_3

1. Select your rule type : Select Based on criteria.
2. Select which records to be shared:

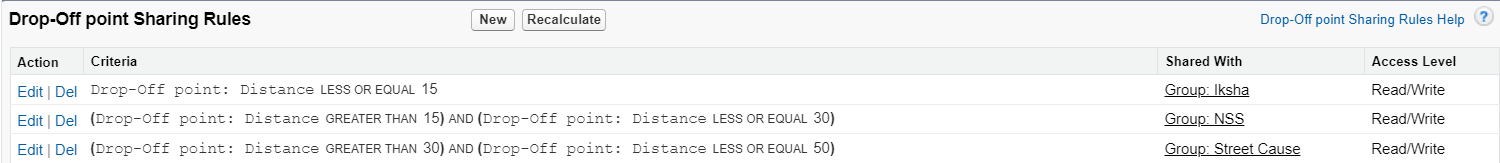
Field : Operator : Value = Distance : greater than : 30

Field : Operator : Value = Distance : less or equal : 50

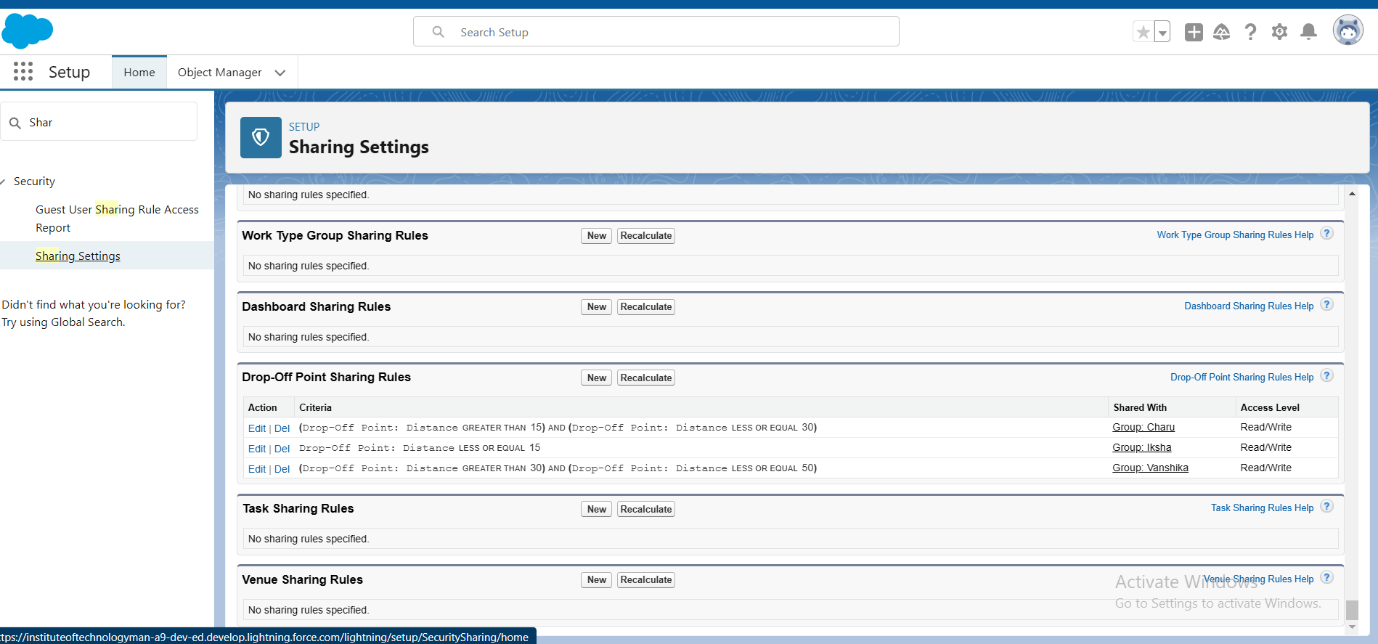
1. Select the users to share with : Near Share With

Public Groups : Street Cause

1. Click on Save.



**Output:**

****

**11.Home Page**

1. Go to setup >> type Lightning App Builder in quick find box  >>  Click on the Lightning App Builder and Select the New.
2. Select Home Page and give Label as HOME Page.
3. Select Standard Home Page.
4. Near Components search for Flow and Drag and Drop in Right Side Section..
5. On the right hand side:

Flow : Venue Flow

1. Near Components search for Dashboard, then Drag and Drop it in first Section.

A screenshot of a computer

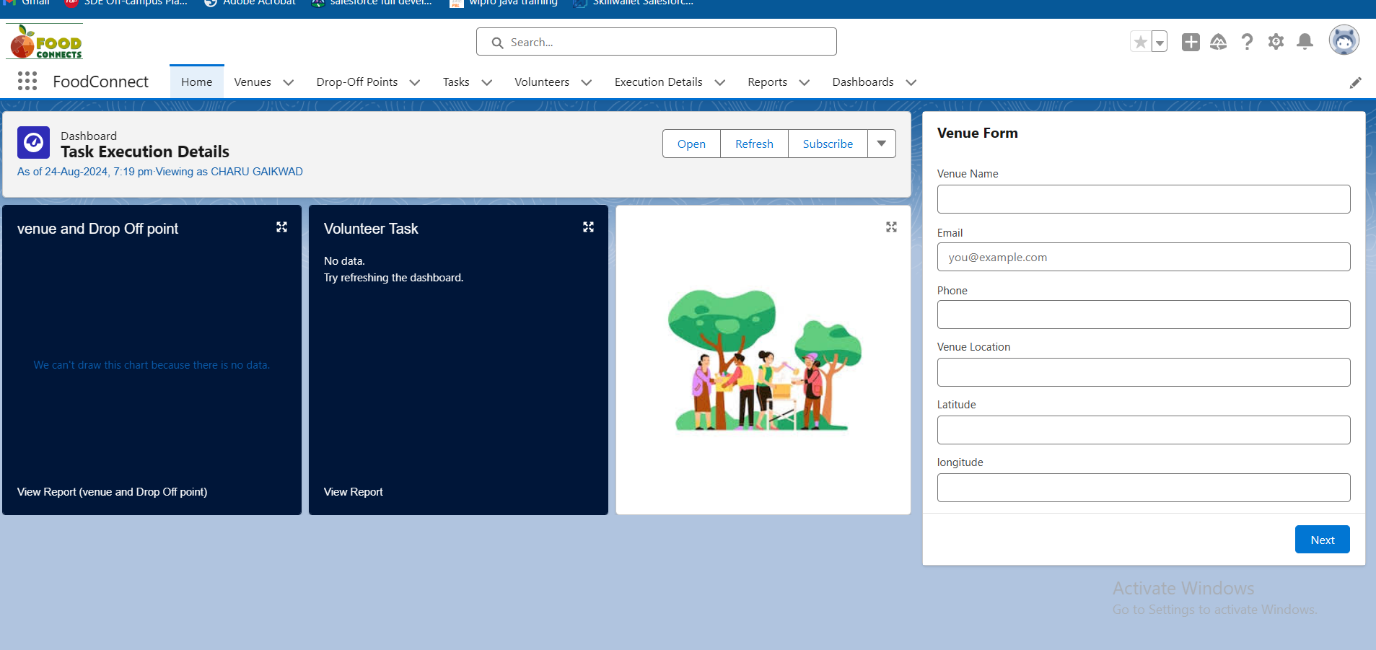
Description automatically generated

1. Click on Save and Activation, then click on App Default, then Add Assignments.
2. Add FoodConnect App and then Save.
3. FoodConnect Home Page would Look Like this.

A screenshot of a computer

Description automatically generated

**Output:**

****