

Field service workorder optimization

INTRODUCTION

The Field Service Work Order Optimization System streamlines operations for a company providing installations and repairs. Utilizing a robust database, the system efficiently matches work orders with skilled technicians based on technician's location, availability, and skills. The system employs a prioritization algorithm, focusing on assigning tasks to technician. Automated communication keeps technicians informed, while analytics offer insights for continuous improvement. Overall, this solution maximizes efficiency, reduces operational costs, and improves customer satisfaction in the dynamic realm of field service operations.

Field service workorder optimization

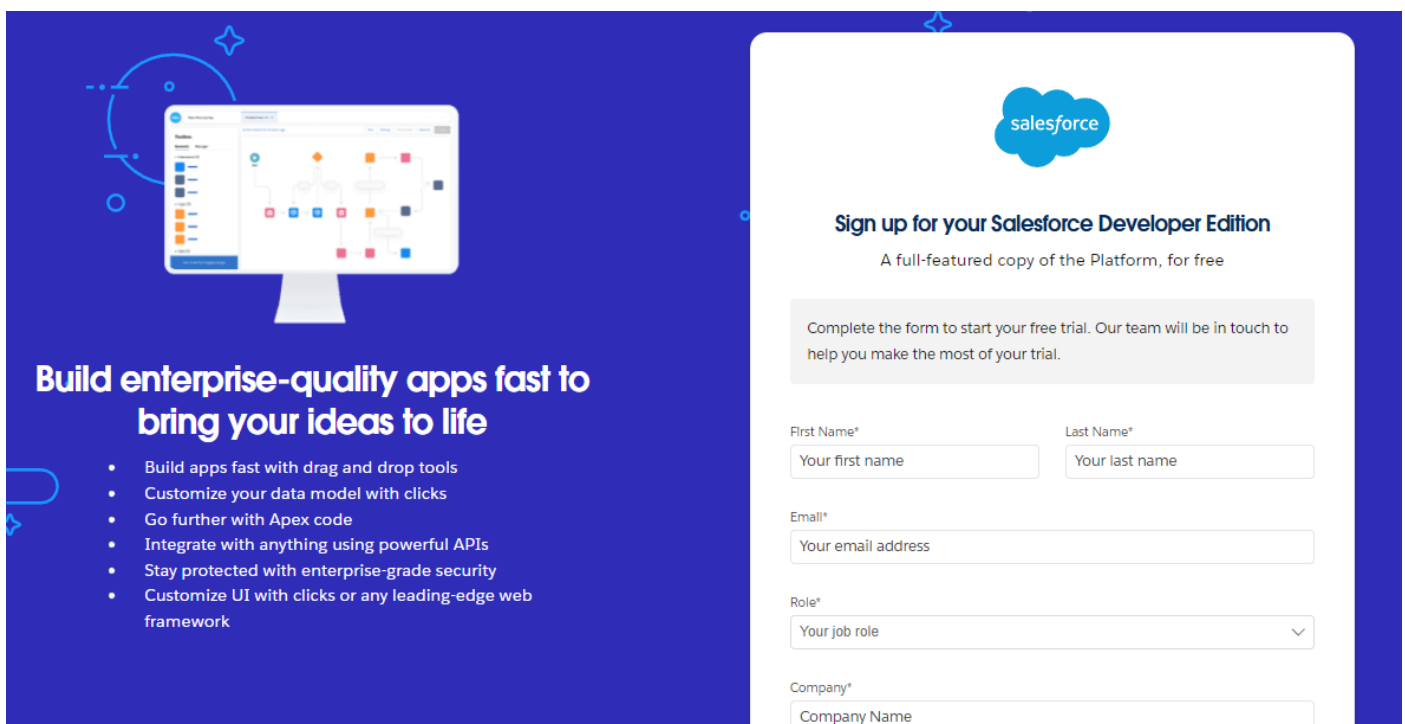
Sales force

Creating a Sales force Developer Edition org allows developers to experiment, innovate, and build customized solutions within a controlled environment. With access to Sales force's powerful development tools and features, developers can prototype, test, and refine their applications, empowering them to deliver robust and tailored solutions to meet unique business requirements. As a Sales force Developer for an organization you must have a Sales force developer edition org in order to do all the required works.

ACTIVITY 1: Creating Developer Account

Creating a developer org in sales force.

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



Build enterprise-quality apps fast to bring your ideas to life

- Build apps fast with drag and drop tools
- Customize your data model with clicks
- Go further with Apex code
- Integrate with anything using powerful APIs
- Stay protected with enterprise-grade security
- Customize UI with clicks or any leading-edge web framework

Sign up for your Salesforce Developer Edition
A full-featured copy of the Platform, for free

Complete the form to start your free trial. Our team will be in touch to help you make the most of your trial.

First Name*

Last Name*

Email*

Role*

Company*

Sign Up

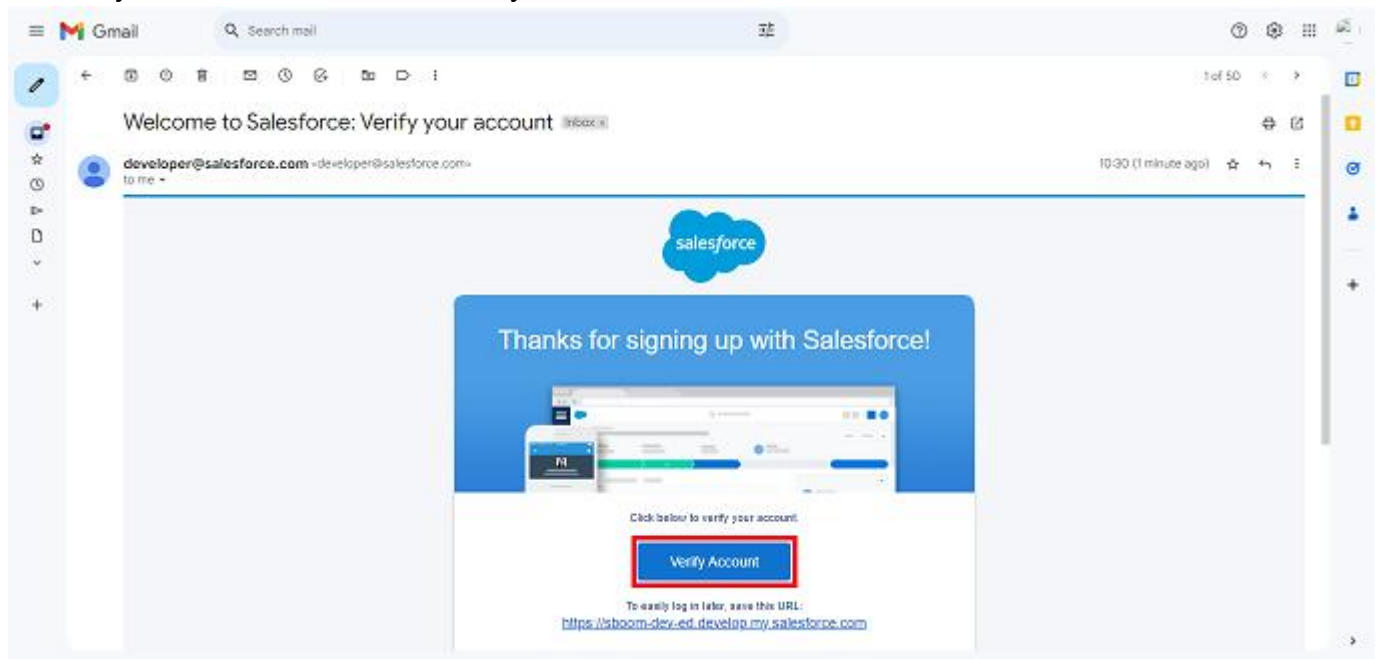
1. First name - BALU
2. Last name - PEDDAPALEM
3. Email – balub0115@gmail.com
4. Role : Developer
5. Company : Gayatri Degree College Tirupati
6. County : India

Field service workorder optimization

This need not be an actual email id, you can give anything in the format : username@organization.com
Click on sign me up after filling these.

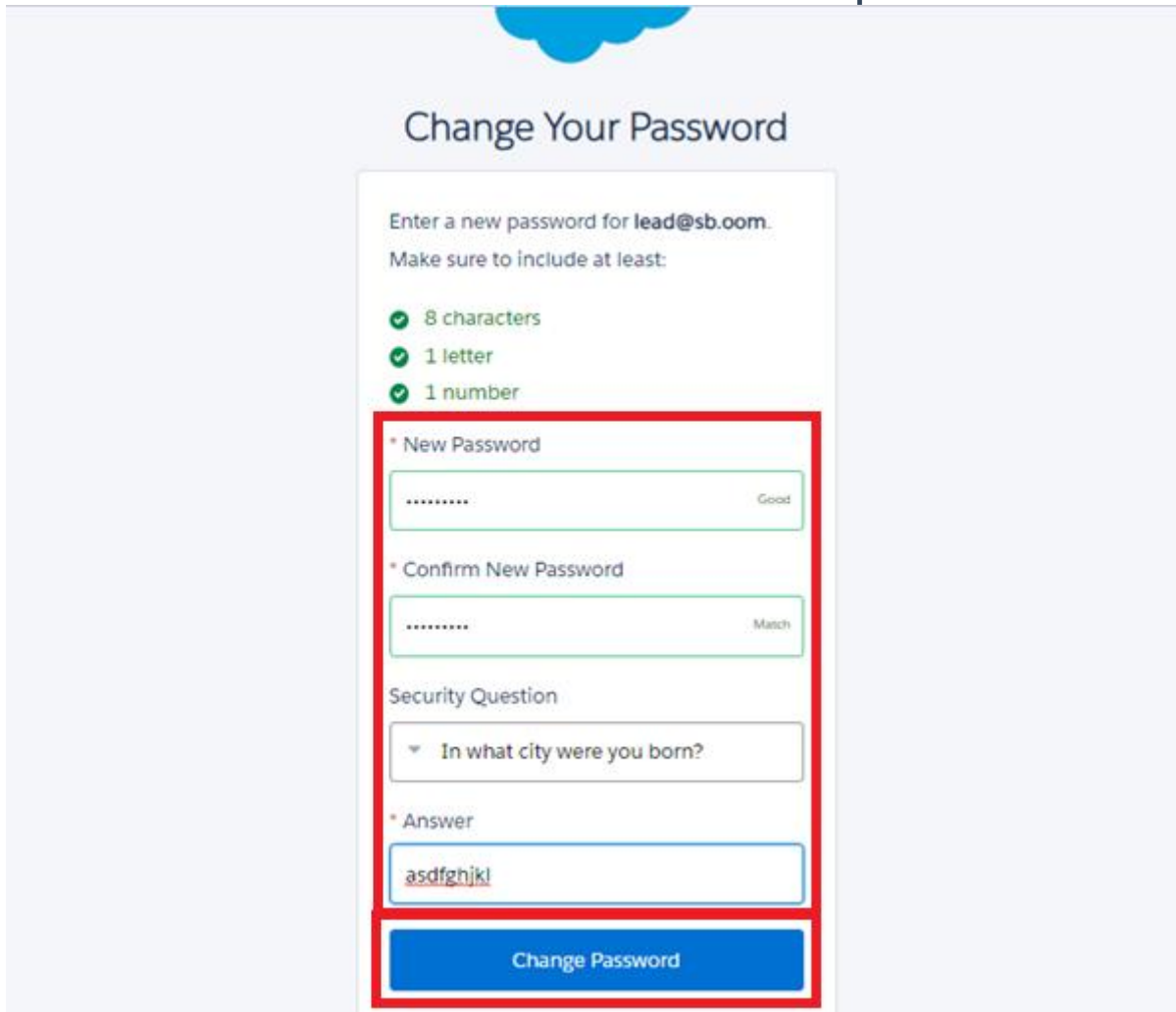
ACTIVITY 2: Account Activation

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.

Field service workorder optimization



Change Your Password

Enter a new password for **lead@sb.oom**.
Make sure to include at least:

- ✓ 8 characters
- ✓ 1 letter
- ✓ 1 number

* New Password

Good

* Confirm New Password

Match

Security Question

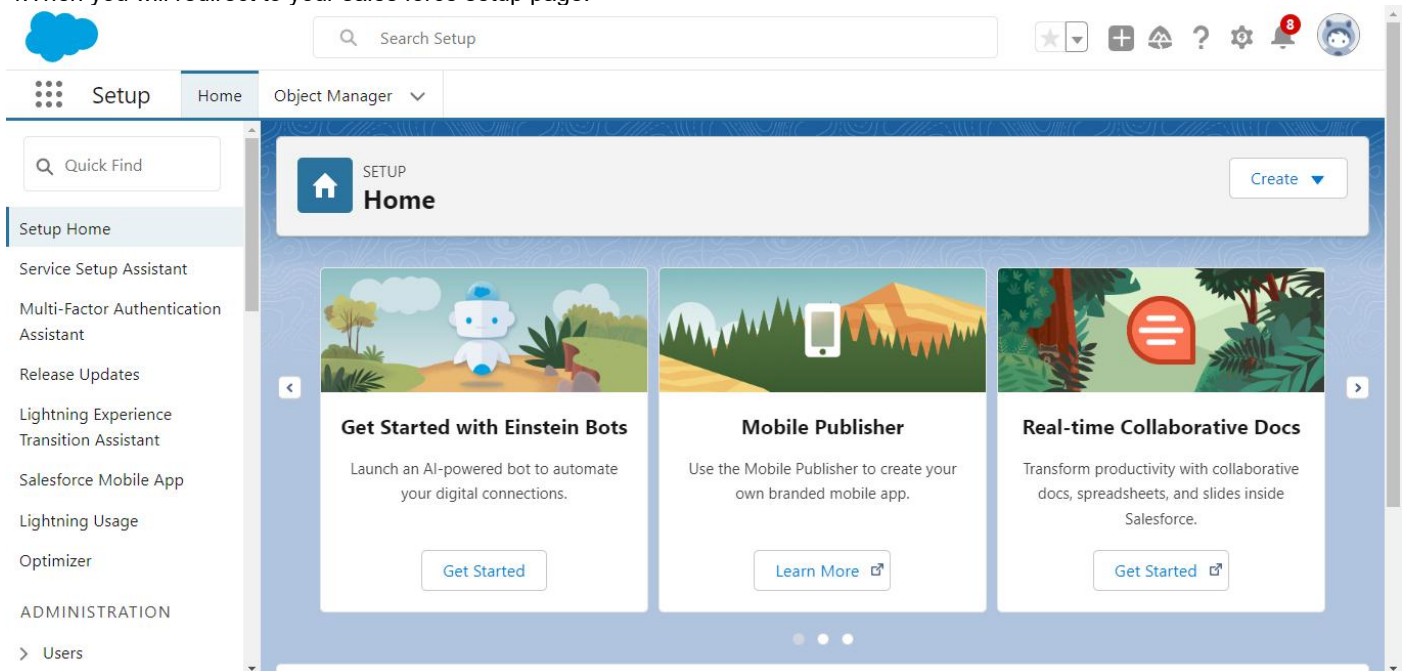
▼ In what city were you born?

* Answer

asdfghjkl

Change Password

4. Then you will redirect to your sales force setup page.



Setup Home

Quick Find

Service Setup Assistant

Multi-Factor Authentication Assistant

Release Updates

Lightning Experience Transition Assistant

Salesforce Mobile App

Lightning Usage

Optimizer

ADMINISTRATION

> Users

SETUP Home

Create

Get Started with Einstein Bots

Launch an AI-powered bot to automate your digital connections.

Get Started

Mobile Publisher

Use the Mobile Publisher to create your own branded mobile app.

Learn More

Real-time Collaborative Docs

Transform productivity with collaborative docs, spreadsheets, and slides inside Salesforce.

Get Started

Field service workorder optimization

Object

To store the data as per business requirement.

ACTIVITY 1: Create Technician Object

1. Download and open [this spreadsheet](#), edit the email column (provide your email for at least one or two records) and save it as Technician.csv.
2. Log into your salesforce account, click on Gear icon, then select Setup.
3. Click the Object Manager tab.



4. Click Create.
5. Select Custom Object from Spreadsheet.



6. Click Login With Salesforce.
7. Enter your Salesforce account username and password. (which you have created in the Milestone 1, Activity 1)
8. Click Log In.
9. Click Allow.
10. Click Upload.
11. Navigate to the Technician.csv file you downloaded and upload it. Salesforce automatically detects the fields and populates all its record data. Choose Technician ID as the Record Name field and make sure all fields are with the proper datatypes as below as they are.

Field service workorder optimization

CSV File Details

Encoding Format ¹ Values Separated By Field Label Source ☐ Enter manually ☒ Detect from row * Field Labels Row Import 5 rows of Data? ¹ ☐ No, skip import ☒ Yes, import data Record Name Field ¹

Fields 7 of 7 to import ☐ Hide mapped fields

| IMPORT FILE FIELD NAME | | SALESFORCE FIELD NAME | SALESFORCE FIELD TYPE | ADD TO LAYOUTS ¹ | FIELD PREVIEW |
|------------------------|---|-----------------------|---------------------------------------|-------------------------------------|----------------------|
| ✓ Technician ID | × | Technician ID | <input type="text" value="Text"/> | <input checked="" type="checkbox"/> | T-0001 |
| ✓ Name | × | Name | <input type="text" value="Text"/> | <input checked="" type="checkbox"/> | Raghu |
| ✓ Phone | × | Phone | <input type="text" value="Phone"/> | <input checked="" type="checkbox"/> | 7892341560 |
| ✓ Email | × | Email | <input type="text" value="Email"/> | <input checked="" type="checkbox"/> | example@gmail.com |
| ✓ Location | × | Location | <input type="text" value="Picklist"/> | <input checked="" type="checkbox"/> | Hyderabad |
| ✓ Availability | × | Availability | <input type="text" value="Picklist"/> | <input checked="" type="checkbox"/> | Available |
| ✓ Skills | × | Skills | <input type="text" value="Picklist"/> | <input checked="" type="checkbox"/> | Machine Installation |

12. Click Next and enter the following settings.

13. Click Finish. The Technician object is successfully created and data imported, all within minutes.

ACTIVITY 2: Create WorkOrder Object

Create WorkOrder object, just as we have created Technician Object using [this spreadsheet](#) :

Note: Make sure you do field mapping with proper field type as shown below.

CSV File Details

Encoding Format ¹ Values Separated By Field Label Source ☐ Enter manually ☒ Detect from row * Field Labels Row Import 0 rows of Data? ¹ ☒ No, skip import ☐ Yes, import data Record Name Field ¹

Fields 7 of 7 to import ☐ Hide mapped fields

| IMPORT FILE FIELD NAME | | SALESFORCE FIELD NAME | SALESFORCE FIELD TYPE | ADD TO LAYOUTS ¹ | FIELD PREVIEW |
|------------------------|---|-----------------------|---|-------------------------------------|---------------|
| ✓ WorkOrder ID | × | WorkOrder ID | <input type="text" value="Text"/> | <input checked="" type="checkbox"/> | |
| ✓ Email | × | Email | <input type="text" value="Email"/> | <input checked="" type="checkbox"/> | |
| ✓ Service Type | × | Service Type | <input type="text" value="Picklist"/> | <input checked="" type="checkbox"/> | |
| ✓ Description | × | Description | <input type="text" value="Text Area (Long)"/> | <input checked="" type="checkbox"/> | |
| ✓ Location | × | Location | <input type="text" value="Picklist"/> | <input checked="" type="checkbox"/> | |
| ✓ Priority | × | Priority | <input type="text" value="Picklist"/> | <input checked="" type="checkbox"/> | |
| ✓ Status | × | Status | <input type="text" value="Picklist"/> | <input checked="" type="checkbox"/> | |

ACTIVITY 3: Create Assignment Object

To create an object:

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

Field service workorder optimization



1. Enter the label name --> Assignment
2. Plural label name --> Assignments

Custom Object Definition Edit

Save Save & New Cancel

Custom Object Information

The singular and plural labels are used in tabs, page layouts, and reports.

Label Example: Account

Plural Label Example: Accounts

Starts with vowel sound ☐

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

Object Name Example: account

Description

Context-Sensitive Help Setting ☒ Open the standard Salesforce.com Help & Training window
☐ Open a window using a Visualforce page

Content Name

3. Enter Record Name Label and Format

- Record Name --> Assignment ID
- Data Type --> Auto Number
- Display Format --> A-{0000}
- Starting Number --> 1

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 the Record Name for Opportunity is "Opportunity Name".

Record Name Example: Account Name

Data Type

Display Format Example: A-{0000} [What Is This?](#)

Starting Number

Field service workorder optimization

2. Click on Allow reports,
Allow search --> Save.

Tabs

A tab is like a user interface that is used to build records for objects and to view the records in the objects.

ACTIVITY 1: Creating a Custom Tab

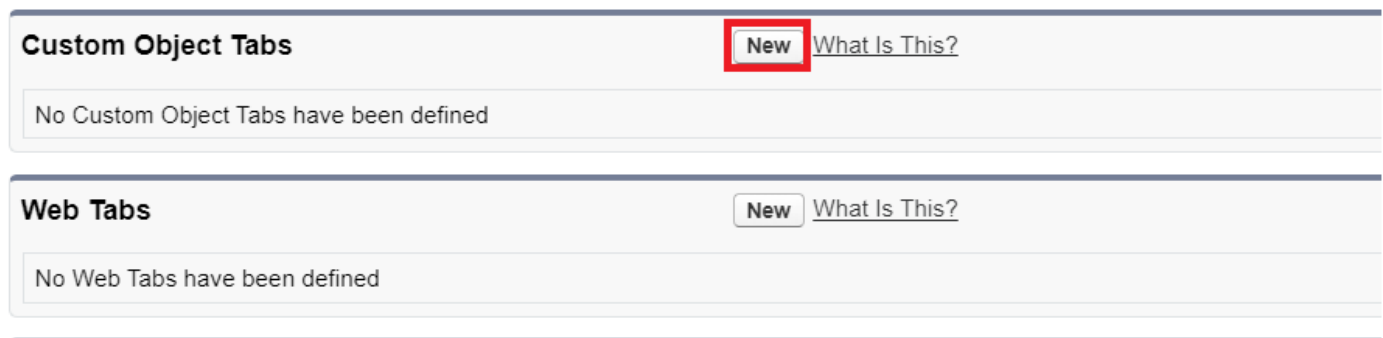
To create a Tab:(Assignment)

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

Custom Tabs

You can create new custom tabs to extend Salesforce functionality or to build new application functionality.

Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external content or allow you to embed Visualforce pages. Lightning Component tabs allow you to add Lightning components to the navigation bar. Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.



The screenshot shows the Salesforce Setup page for creating tabs. It is divided into two main sections: 'Custom Object Tabs' and 'Web Tabs'. Each section has a 'New' button and a 'What Is This?' link. The 'Custom Object Tabs' section is currently selected, and the 'New' button is highlighted with a red box. Below each section, there is a message stating 'No Custom Object Tabs have been defined' and 'No Web Tabs have been defined' respectively.

2. Select Object(Assignment) --> Select any tab style --> Next (Add to profiles page) keep it as default --> Next (Add to Custom App) keep it as default --> Save.

Field service workorder optimization

New Custom Object Tab

Step 1. Enter the Details

Choose the custom object for this new custom tab. Fill in other details.

Select an existing custom object or [create a new custom object now](#).

| | |
|-----------|------------------------|
| Object | Assignment ▼ |
| Tab Style | --None-- Assignment |

(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-- ▼ |
|-------------------------|------------|

Note: Tabs for Workorder & Technician objects do get created automatically. We do not need to create tabs for those objects.

The Lightning App

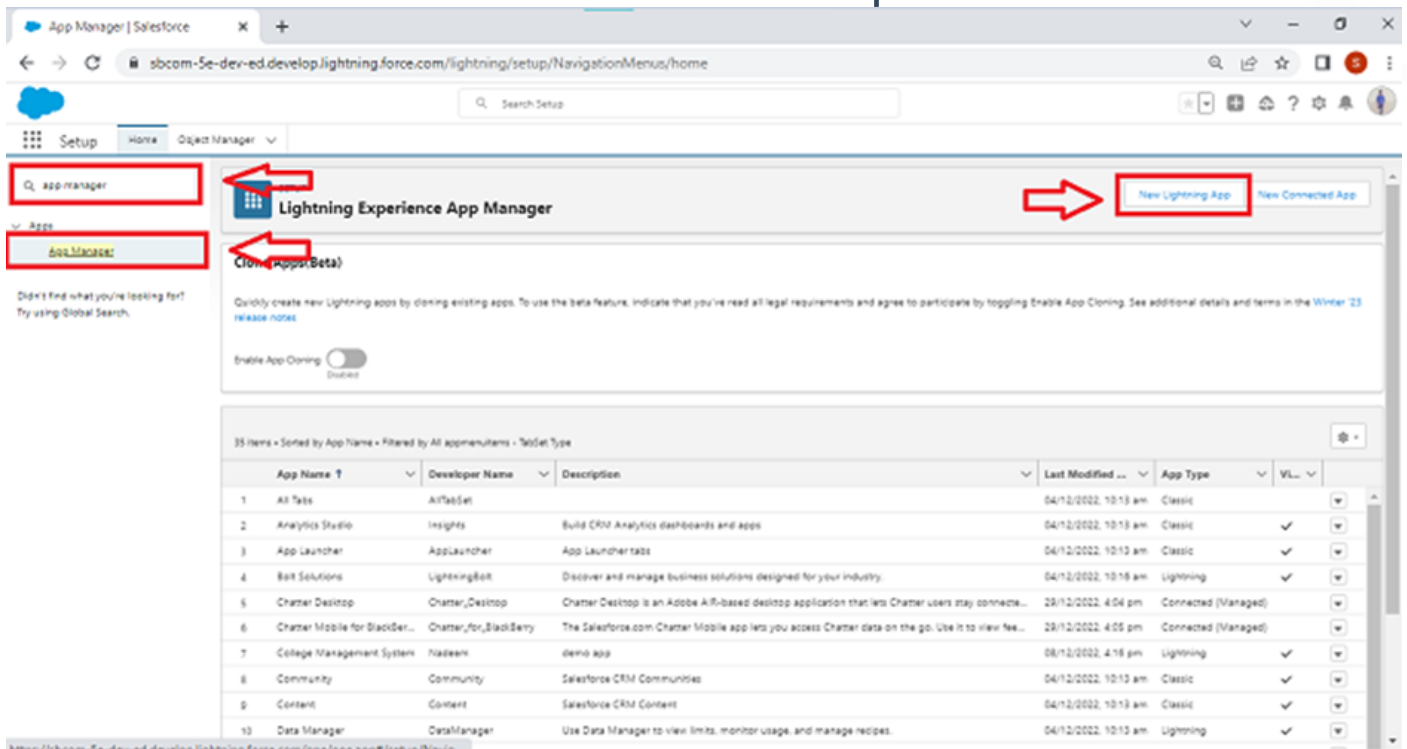
Well done you have reached close to your organizational requirement by creating the objects to store the organization's data. Making a database for an organization is just not enough to reach out the requirements, the task is how the users at the organization can access the objects you have created for them

ACTIVITY 1: Create a Lightning App

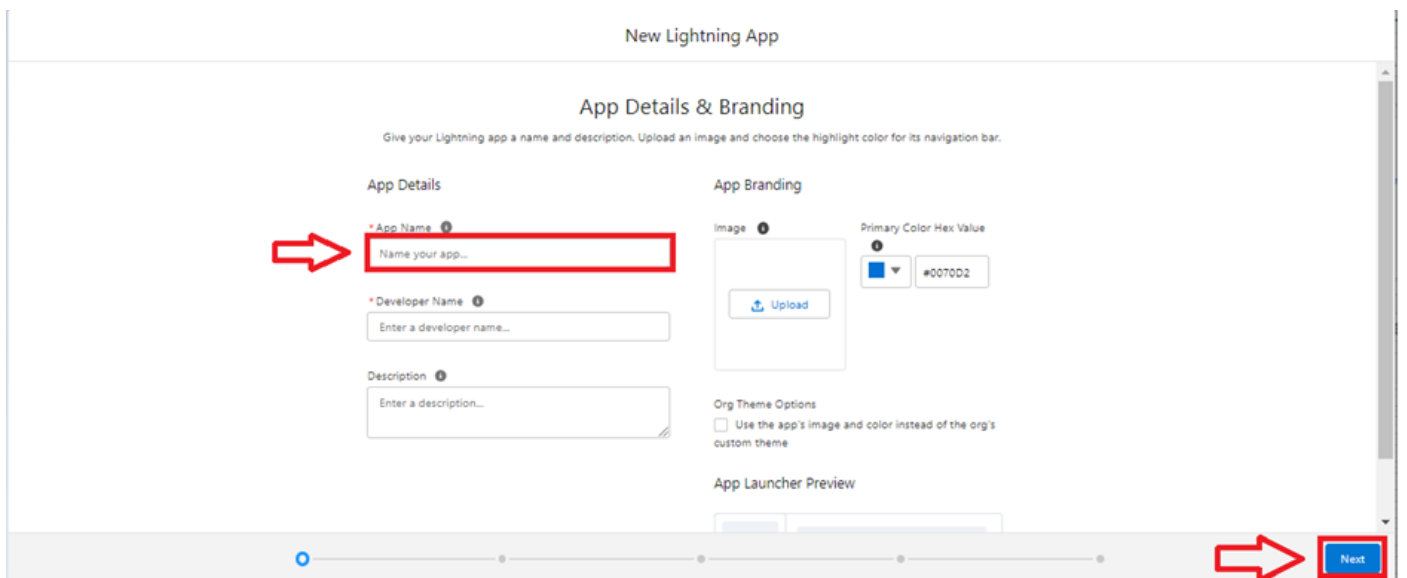
To create a lightning app page:

1. Go to setup page --> search "app manager" in quick find --> select "app manager" --> click on New lightning App.

Field service workorder optimization

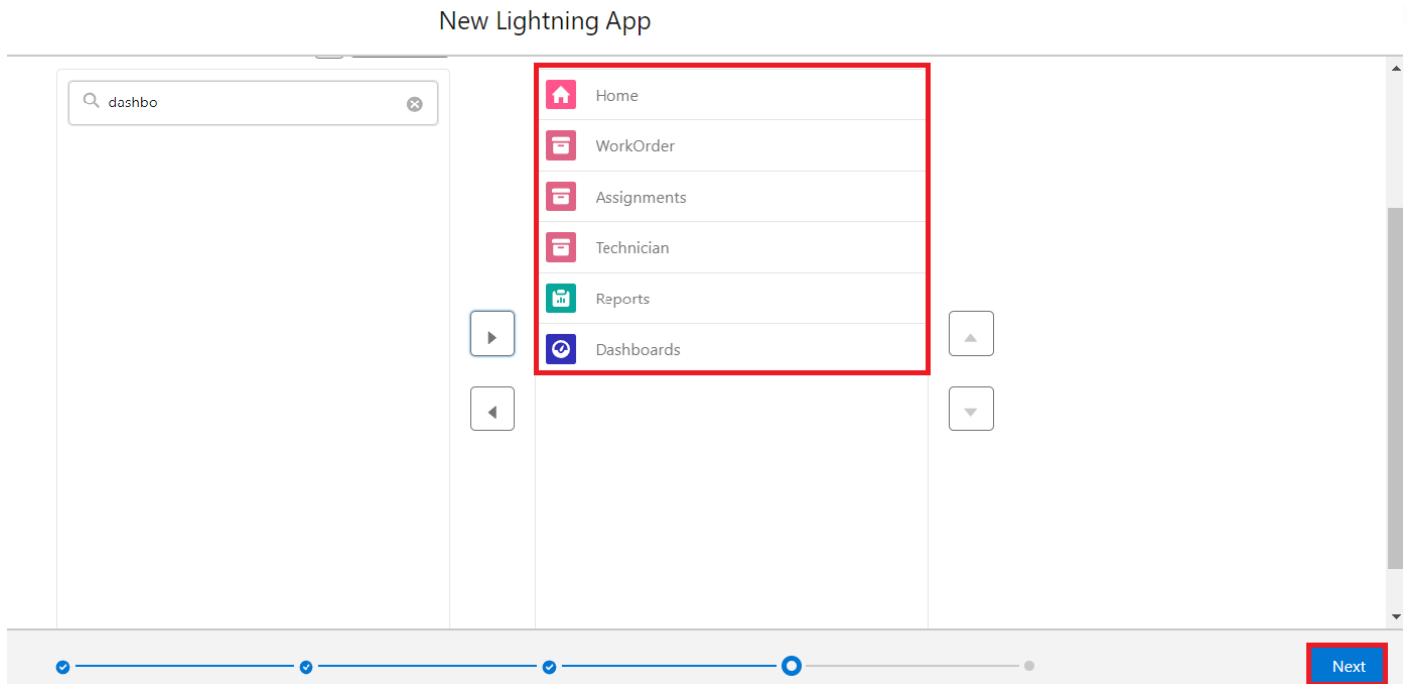


- Fill the app name in app details and branding as follow
 App Name : Field Service WorkOrder Optimization
 Developer Name : this will auto populated
 Description : Give a meaningful description
 Image : optional (if you want to give any image you can otherwise not mandatory)
 Primary color hex value : keep this default
- Then click Next --> (App option page) keep it as default --> Next --> (Utility Items) keep it as default --> Next.



- To Add Navigation Items:

Field service workorder optimization



Search the items in the search bar(Home, WorkOrder, Technician, Assignment, Reports, Dashboard) from the search bar and move it using the arrow button ? Next.

Note: select asset the custom object which we have created in the previous activity.

5. To Add User Profiles:

Field service workorder optimization

New Lightning App

User Profiles

Choose the user profiles that can access this app.

Available Profiles

Selected Profiles

System administrator

System Administrator

Save & Finish

Search profiles (System administrator) in the search bar --> click on the arrow button --> save & finish.

Fields & Relationship

Now it's time for you to think out of the box for your organization. You have successfully created the database objects for the organization but now all eyes turn on you as you have to define what sort of information the objects store which you have created. As a life saver of your organization you come up with the idea of creating fields to store different types of data.

ACTIVITY 1: Creating Lookup Field in Assignment Object

To create fields in an object:

1. Go to setup --> click on Object Manager --> type object name(Assignment) in quick find bar--> click on the object.

Setup Home Object Manager

Object Manager

2 Items, Sorted by Label

| LABEL | API NAME | TYPE | DESCRIPTION | LAST MODIFIED | DEPLOYED |
|---------------------------|-------------------------|-----------------|-------------|---------------|----------|
| Assignment | Assignment__c | Custom Object | | 20/11/2023 | ✓ |
| Location Group Assignment | LocationGroupAssignment | Standard Object | | | |

2. Now click on "Fields & Relationships" --> New

Field service workorder optimization

Setup Home Object Manager

SETUP > OBJECT MANAGER
Assignment

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets

Fields & Relationships
4 Items, Sorted by Field Label

Quick Find New Deleted Fields Field Dependencies Set History Tracking

| FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
|------------------|----------------|--------------------|-------------------|---------|
| Assignment ID | Name | Auto Number | | ✓ |
| Created By | CreatedBy | Lookup(User) | | |
| Last Modified By | LastModifiedBy | Lookup(User) | | |
| Owner | OwnerId | Lookup(User,Group) | | ✓ |

3. Select Data type as “Lookup”.

Step 1. Choose the field type Step 1

Next Cancel

Specify the type of information that the custom field will contain.

Data Type

☐ None Selected Select one of the data types below.

☐ 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 *i* 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.

4. Click on Next

5. For field label related to: select “WorkOrder” object and click Next.

Note: Do not select other standard object with the same name for sake of ease copy the above and paste it.

Assignment
New Relationship

[Help for this Page](#)

Step 2. Choose the related object Step 2

Previous Next Cancel

Select the other object to which this object is related.

Related To WorkOrder

Previous Next Cancel

6. Give Field Label as “WorkOrder ID” and click Next.

Field service workorder optimization

Assignment
New Relationship

Help for this Page ?

Step 3. Enter the label and name for the lookup field Step 3 of 6

[Previous](#) [Next](#) [Cancel](#)

Field Label [i](#)

Field Name [i](#)

Description

Help Text [i](#)

7. Next --> Next --> Save & New.

ACTIVITY 2: Manage your picklist values

1. From the setup page go to object manager
2. Search and Select WorkOrder object.
3. Go to fields & relationship, select Location field, scroll down to values and click "New".

| Values | | | | | |
|---|-----------|-----------|--------------------------|----------------------|--|
| <div>New Reorder Replace Printable View Chart Colors Delete Selected Deactivate Selected Replace Selected</div> | | | | | |
| <input type="checkbox"/> Action | Values | API Name | Default | Chart Colors | Modified By |
| <input type="checkbox"/> Edit Del Deactivate | Pune | Pune | <input type="checkbox"/> | Assigned dynamically | demo.project , 22/11/2023, 9:53 am |
| <input type="checkbox"/> Edit Del Deactivate | Hyderabad | Hyderabad | <input type="checkbox"/> | Assigned dynamically | demo.project , 22/11/2023, 9:53 am |

4. Add the below values:
Nasik
Warangal
Nanded
5. Click Save.

Field service workorder optimization

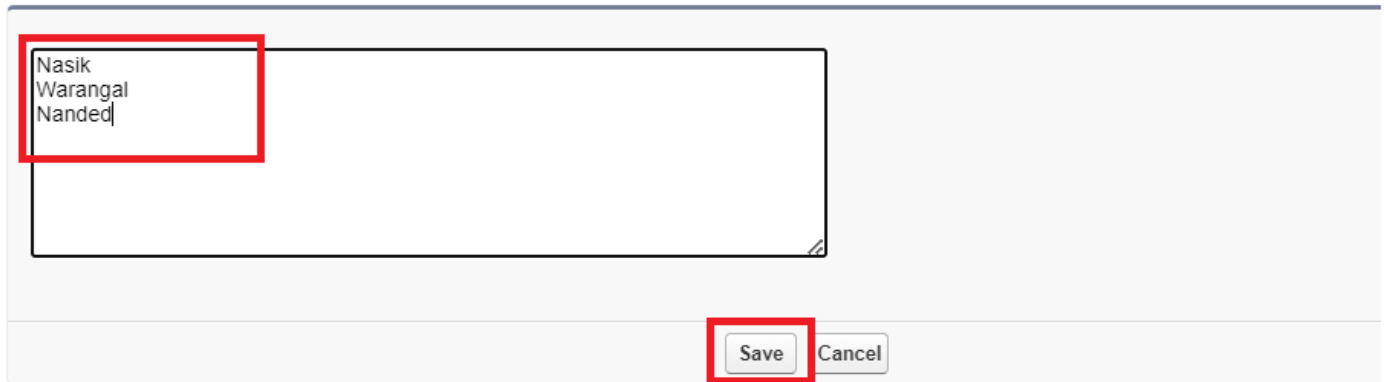
Add Picklist Values

Location

Add one or more picklist values below. Each value should be on its own line and it is used for both a value's label and API name.

If a value matches an inactive value's API name, that value is reactivated with its previous label.

If a value matches an inactive value's label but not the API name, a new value is created.



ACTIVITY 3: Manage your picklist values

Add following values to the respective fields in WorkOrder object:

| Field | Values |
|--------------|--|
| Priority | High |
| Service Type | Hardware repair Troubleshoot/Debugging Lane-Management |

ACTIVITY 4: Creating Formula Field in WorkOrder Object

1. Repeat step 1 and 2 mentioned in activity 1
2. Select Data type as "Formula" and click Next.
3. Give Field Label and Field Name as "Date" and select formula return type as "Date" and click next.

Field service workorder optimization

Step 2. Choose output type Step 2 of 5

[Previous](#) [Next](#) [Cancel](#)

Field Label Field Name

Auto add to custom report type ☒ Add this field to existing custom report types that contain this entity [i](#)

Formula Return Type

☐ None Selected Select one of the data types below.

☐ Checkbox Calculate a boolean value.
Example: `{TODAY()} > {CloseDate}`

☐ Currency Calculate a dollar or other currency amount and automatically format the field as a currency amount.
Example: `{Gross Margin = Amount - Cost}_c`

☒ Date Calculate a date, for example, by adding or subtracting days to other dates.
Example: `{Reminder Date = CloseDate - 7}`

☐ Date/Time Calculate a date/time, for example, by adding a number of hours or days to another date/time.
Example: `{Next = NOW()} + 1`

☐ Number Calculate a numeric value.
Example: `{Fahrenheit = 1.8 * Celsius}_c + 32`

☐ Percent Calculate a percent and automatically add the percent sign to the number.
Example: `{Discount = (Amount - Discounted_Amount)_c} / Amount`

☐ Text Create a text string, for example, by concatenating other text fields.
Example: `{Full Name = LastName & ", " & FirstName}`

☐ Time Calculate a time, for example, by adding a number of hours to another time.
Example: `{Next = TIMEVALUE(NOW()) + 1}`

4. Under Advanced Formula write down the formula and click “Check Syntax”

Formula: CreatedDate

Simple Formula **Advanced Formula**

[Insert Field](#)

Date (Date) =

[Check Syntax](#) No syntax errors in merge fields or functions. (Compiled size: 20 characters)

5. Next--> Next--> Save.

Field service workorder optimization

ACTIVITY 5: Creating Remaining fields for the respective objects

Now create the remaining fields using the data types mentioned in the table.

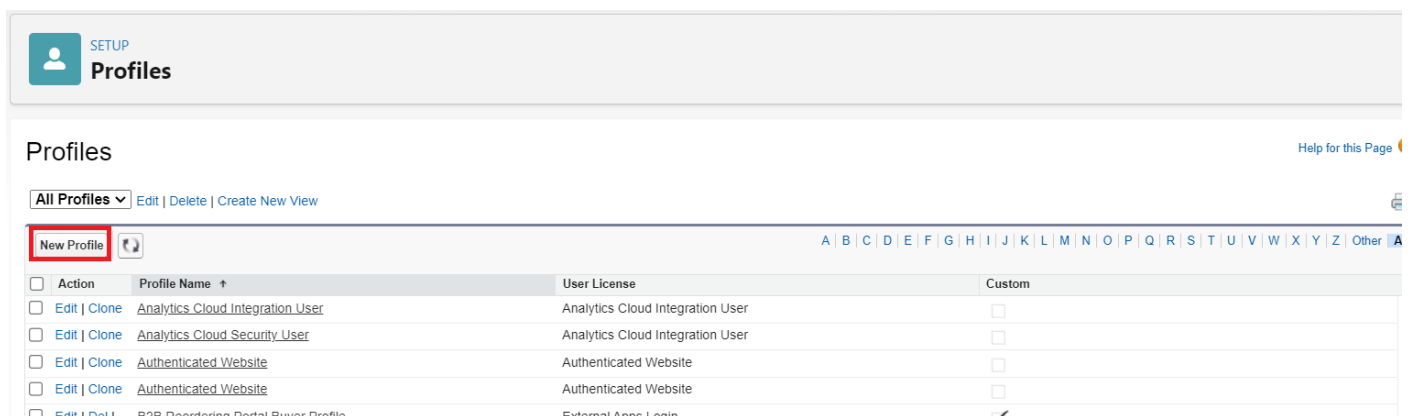
| Sl No | Object Name | Field | |
|-------|-------------|---|---|
| 1 | Assignment | | |
| | | Field Name | Datatype |
| | | <ul style="list-style-type: none">Technician IDAssignment DateCompletion Date | <p>Lookup(Technician)</p> <p>Formula: return type : Date (WorkOrder_ID__r.Date__c)</p> <p>Formula: return type : Date IF(ISPICKVAL(WorkOrder_ID__r.Status__c , 'Resolved'), WorkOrder_ID__r.LastModifiedDate , NULL)</p> |

Profiles

Profile defines what an user is able to do or see in the Salesforce Org

ACTIVITY 1: Technician Profile

1. Go to setup --> type profiles in quick find box --> click on profiles --> click on new profile.



2. . Select 'Standard Platform User' for existing profile and give 'Technician' for Profile Name and click on Save.

Field service workorder optimization

Clone Profile

Enter the name of the new profile.

You must select an existing profile to clone from.

Existing Profile

User License

Profile Name

3. While still on the profile page, then click Edit.
4. Scroll down to Custom Object Permissions and Give Read only access permissions for Technician, WorkOrder and Assignment objects and field access permission as shown below:

| Custom Object Permissions | | | | | | |
|---------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| | Basic Access | | | | Data Administration | |
| | Read | Create | Edit | Delete | View All | Modify All |
| Assets | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Asset Services | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Assignments | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Billings | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Bookings | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Candidates | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Child object | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Crews | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Customer Orders | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Employees | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Employment Websites | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Flights | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Items | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Jewel Customers | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Job Applications | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Lead Scoring Rules | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

| | Basic Access | | | | Data Administration | |
|-----------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| | Read | Create | Edit | Delete | View All | Modify All |
| Leaves | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Parent object 1 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Parent object 2 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Passengers | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Positions | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Prices | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Projects | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| ProjectTasks | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Reviews | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Sessions | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Students | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| StudentSessions | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Technician | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Trainers | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| WorkOrder | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

Field service workorder optimization

5. Scroll down and Click on Save.
6. Now from the profile detail page scroll down to custom field level security click on view next to WorkOrder object.

| | |
|-----------------------------|----------------------|
| Custom Field-Level Security | |
| Asset | View |
| Asset Service | View |
| Assignment | View |
| Billing | View |
| Booking | View |
| Candidate | View |
| Child object | View |
| Crew | View |
| Customer Order | View |
| Employee | View |
| Employment Website | View |
| Flight | View |
| Item | View |
| Jewel Customer | View |
| Job Application | View |
| Knowledge | View |
| Lead Scoring Rule | View |
| Leave | View |
| Parent object 1 | View |
| Parent object 2 | View |
| Passenger | View |
| Position | View |
| Price | View |
| Project | View |
| ProjectTask | View |
| Review | View |
| Sessions | View |
| Student | View |
| StudentSession | View |
| Technician | View |
| Trainer | View |
| WorkOrder | View |

7. Click on Save.

Field service workorder optimization

Users

Users are defined as the employees of your organization

ACTIVITY 1: Create User

1. Go to setup --> type users in quick find box --> select users --> click New user.
2. Fill in the fields
 1. First Name : Elina
 2. Last Name : Gilbert
 3. Alias : Give a Alias Name
 4. Email id : Give your Personal Email id
 5. Username : Username should be in this form: text@text.text
 6. Nick Name : Give a Nickname
 7. Role :
 8. User license : Salesforce Platform
 9. Profiles : Technician

New User

User Edit Save Save & New Cancel

General Information = Required Information

| | | | |
|------------|--------------------------|---------------------------|-------------------------------------|
| First Name | Elina | Role | <None Specified> |
| Last Name | Gilbert | User License | Salesforce Platform |
| Alias | elina | Profile | Technician |
| Email | nadeem@thesmartbridge.co | Active | <input checked="" type="checkbox"/> |
| Username | elina@smart.com | Marketing User | <input type="checkbox"/> |
| Nickname | elina | Offline User | <input type="checkbox"/> |
| Title | | Knowledge User | <input type="checkbox"/> |
| Company | | Flow User | <input type="checkbox"/> |
| Department | | Service Cloud User | <input type="checkbox"/> |
| Division | | Site.com Contributor User | <input type="checkbox"/> |
| | | Site.com Publisher User | <input type="checkbox"/> |
| | | WDC User | <input type="checkbox"/> |

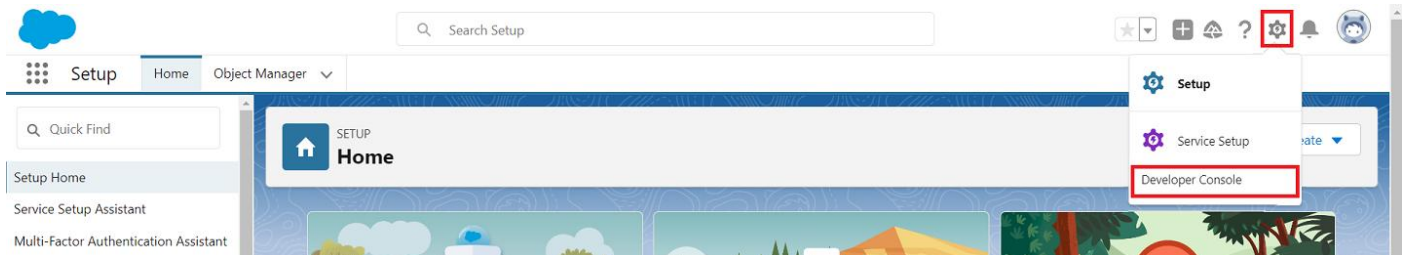
3. Save

Field service workorder optimization

Apex Trigger

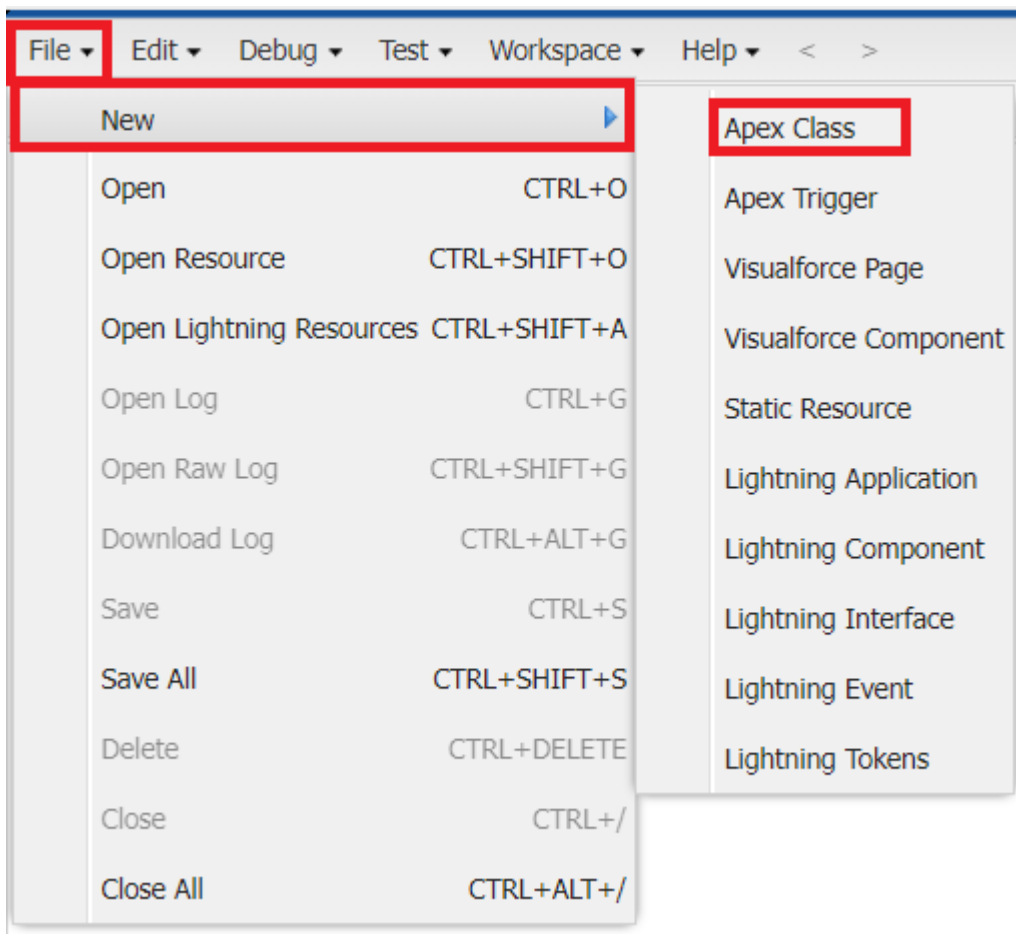
ACTIVITY 1: Create an Apex Class

1. Go to Setup --> Click on the gear icon --> Select Developer Console.



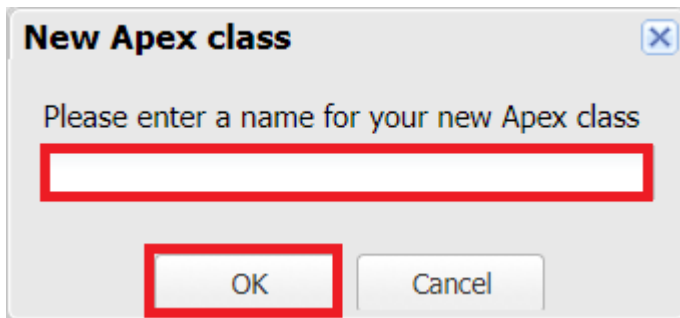
2. Then we can see the Developer console. Click on the developer console and you will navigate to a new console window.

3. To create a new Apex Class follow the below steps:
Click on the file --> New --> Apex Class.



Field service workorder optimization

4. Give the Apex Class name as “WorkOrderClass”.



5. Click ok.

6. Now write the code logic here

```
File Edit Debug Test Workspace Help < >
WorkOrderTrigger.apex WorkOrderClass.apex
Code Coverage: None API Version: 50 Go

1 public class WorkOrderClass {
2     public static void workOrder(List<WorkOrder__c> newListWorkOrder){
3         Map<Integer, List<String>> maptotech = new Map<Integer, List<String>>();
4         integer num = 0;
5         List<WorkOrder__c> properWo = new List<WorkOrder__c>();
6         List<Assignment__c> lstAssignment = new List<Assignment__c>();
7         List<Technician__c> technicianToAssignment = new List<Technician__c>();
8         for(WorkOrder__c iter : newListWorkOrder){
9             List<String> lststring = new List<string>();
10            If(iter.Service_Type__c != null && iter.Location__c != null ){
11                num = num+1;
12                properWo.add(iter);
13                lststring.add(iter.Service_Type__c);
14                lststring.add(iter.Location__c);
15
16                maptotech.put(num,lststring);
17            }
18        }
19        Map<integer,Id> techId = new Map<integer,Id>();
20        Map<Id,Technician__c> allTechnician = new Map<Id,Technician__c>([SELECT Id, Name, Phone__c, Location__c, Skills__c, Availability__c, Name__c, Email__c FROM Technician__c]);
21        integer num2 = 0;
22        For(Technician__c t : allTechnician.values()){
23            num2 = num2+1;
24            if(maptotech.get(num2) != null){
25                List<string> valofmap = maptotech.get(num2);
26                system.debug('error 1 ----> the maptotech is empty ----> ' + maptotech.get(num2));
27                if(valofmap.contains(t.Skills__c) && ValofMap.contains(t.Location__c) && t.Availability__c == 'Available'){
28                    techid.put(num2,t.Id);
29                }
30            }
31        }
32        integer num3 = 0;
33        For(WorkOrder__c M : properWo){
34            // ...

```

Field service workorder optimization

Source Code:

```
public class WorkOrderClass {
    public static void workOrder(List<WorkOrder__C> newListWorkOrder){
        Map<Integer, List<String>> maptotech = new map<Integer,List<String>>();
        integer num = 0;
        List<WorkOrder__c> properWo = new List<WorkOrder__c>();
        List<Assignment__c> IstAssignment = new List<Assignment__c>();
        List<Technician__c> technicianToAssignment = new List<Technician__c>();
        for(WorkOrder__c iter : newListWorkOrder){
            List<String> Iststring = new List<string>();
            If(iter.Service_Type__c != null && iter.Location__c != null ){
                num = num+1;
                properWo.add(iter);
                Iststring.add(iter.Service_Type__c);
                Iststring.add(iter.Location__c);

                maptotech.put(num,Iststring);
            }
        }
        Map<integer,Id> techId = new Map<integer,Id>();
        Map<Id,Technician__c> allTechnician = new Map<Id,Technician__c>([SELECT Id, Name, Phone__c,
        Location__c, Skills__c, Availability__c, Name__c, Email__c FROM Technician__c]);
        integer num2 = 0;
        For(Technician__c T : allTechnician.values()){
            num2 = num2+1;
            if(maptotech.get(num2) != null){
                List<string> valofmap = maptotech.get(num2);
                system.debug('error 1 ----> the maptotech is empty ----> ' + maptotech.get(num2));
                if(valofMap.contains(t.Skills__c) && ValofMap.contains(t.Location__c) && t.Availability__c == 'Available'){
                    techid.put(num2,t.Id);
                }
            }
        }

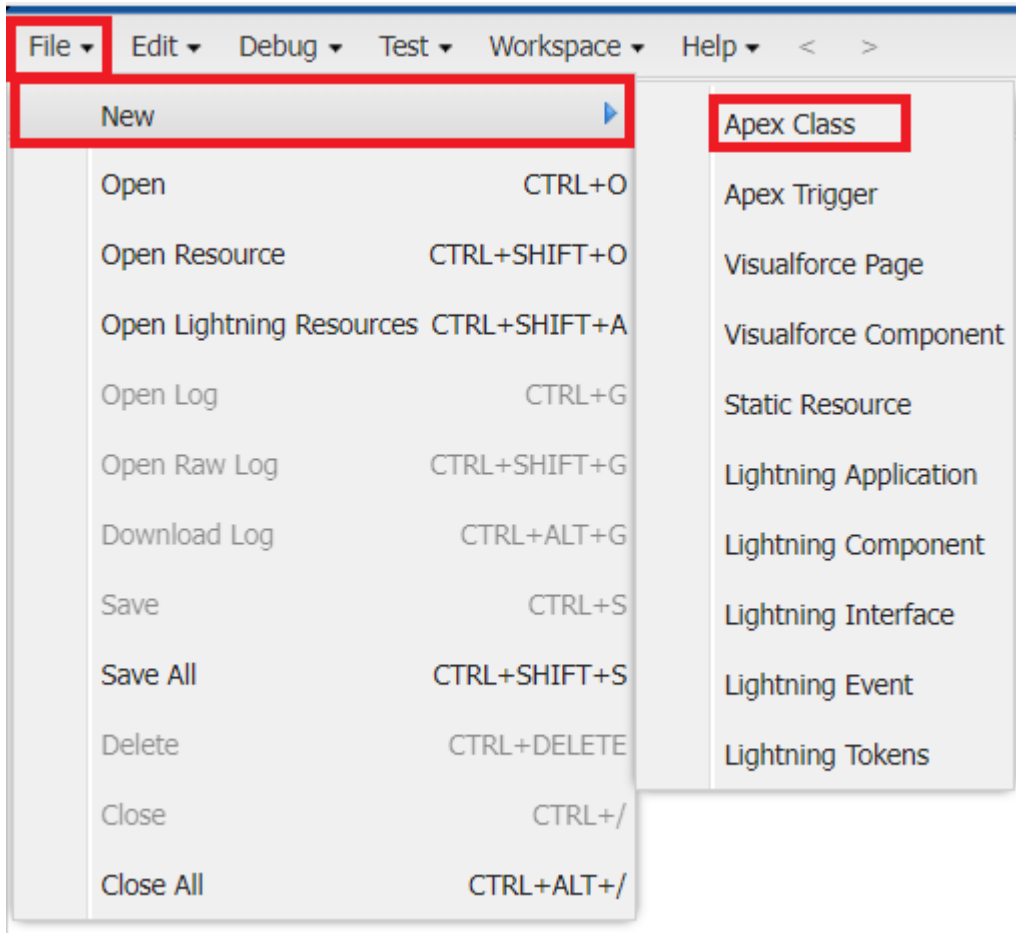
        integer num3 = 0;
        For(WorkOrder__c W : properWo){
            num3 = num3 + 1;
            Assignment__c A = new Assignment__c();
            A.WorkOrder_ID__c = W.Id;
            A.Technician_ID__c = techid.get(num3);
            IstAssignment.add(A);
        }
        If(!IstAssignment.IsEmpty()){
            insert IstAssignment;
        }
    }
}
```

7. Save the code.(click on file --> Save)

Field service workorder optimization

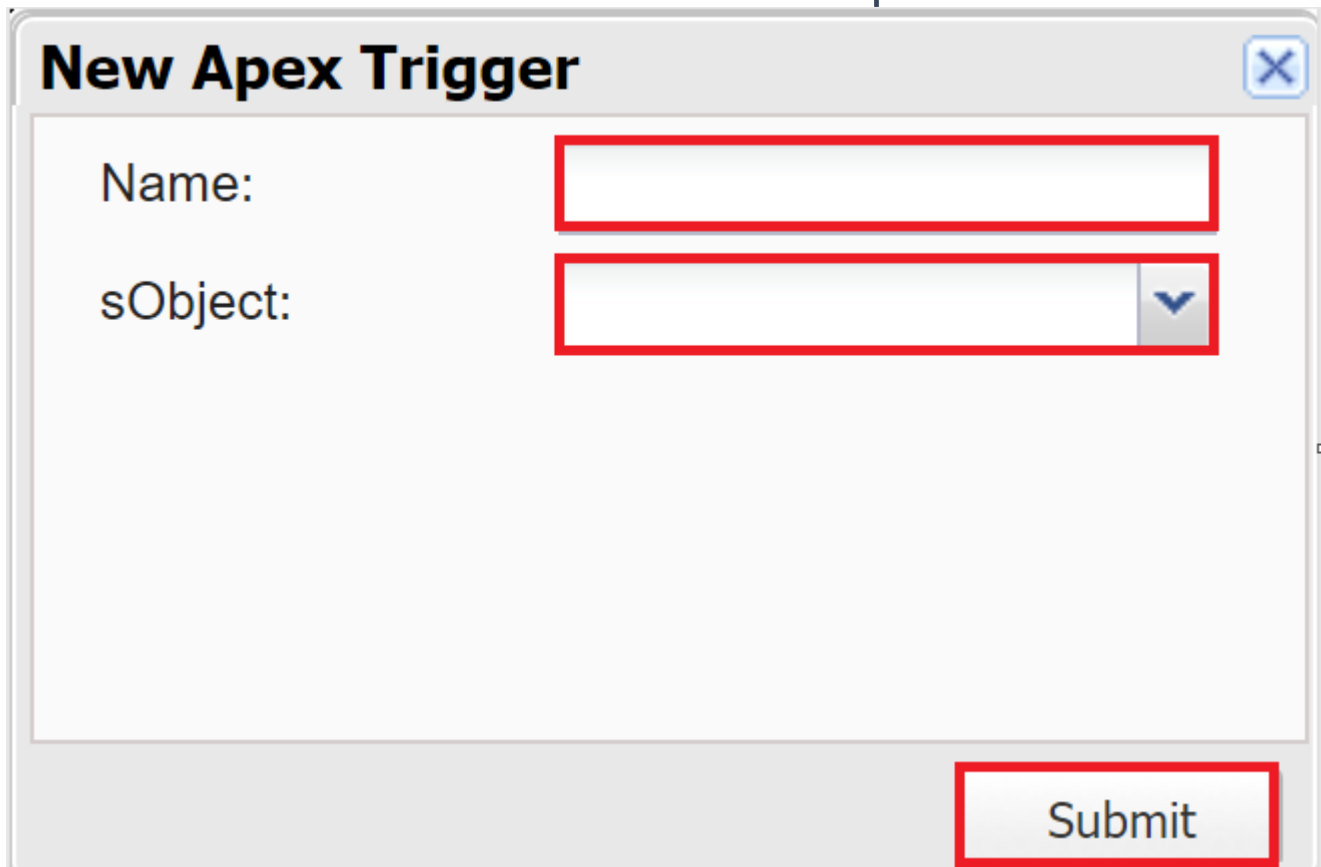
ACTIVITY 2 : Create an Apex Trigger

1. To create a new Apex Class follow the below steps:
Click on the file --> New --> Apex Class.



2. Give the Apex Trigger name as “WorkOrderTrigger”, and select “WorkOrder__c” from the dropdown for sObject.

Field service workorder optimization



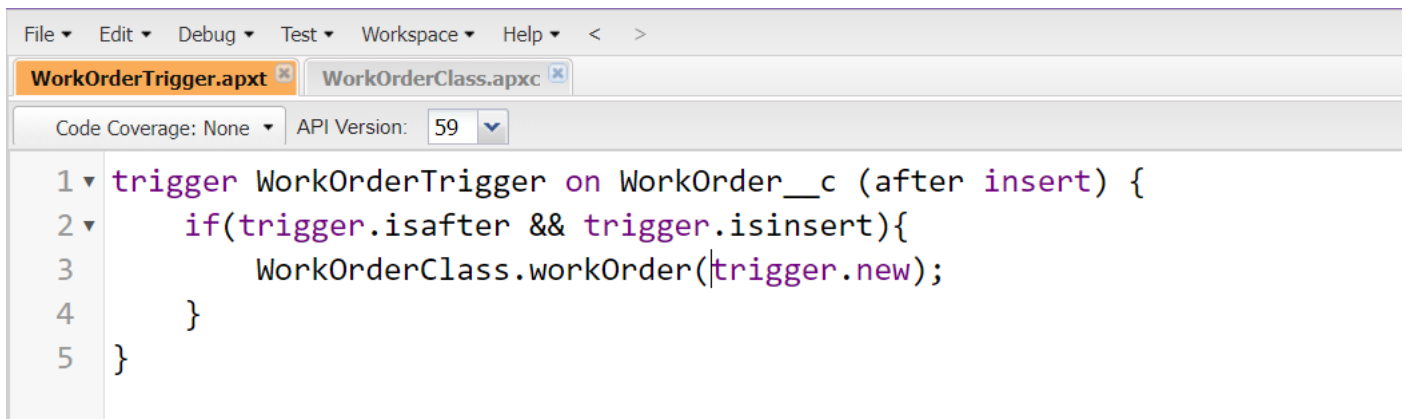
New Apex Trigger

Name:

sObject:

Submit

3. Click Submit.
4. Now write the code logic here



```
File Edit Debug Test Workspace Help < >
WorkOrderTrigger.apxt WorkOrderClass.apxc
Code Coverage: None API Version: 59
1 trigger WorkOrderTrigger on WorkOrder__c (after insert) {
2     if(trigger.isafter && trigger.isinsert){
3         WorkOrderClass.workOrder(trigger.new);
4     }
5 }
```

Source Code:

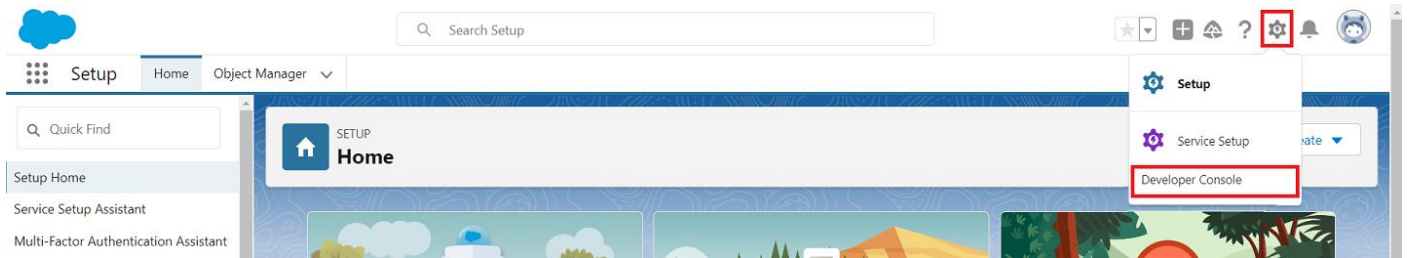
```
trigger WorkOrderTrigger on WorkOrder__c (after insert) {
    if(trigger.isafter && trigger.isinsert){
        WorkOrderClass.workOrder(trigger.new);
    }
}
```

5. Save the code.(click on file --> Save)

Field service workorder optimization

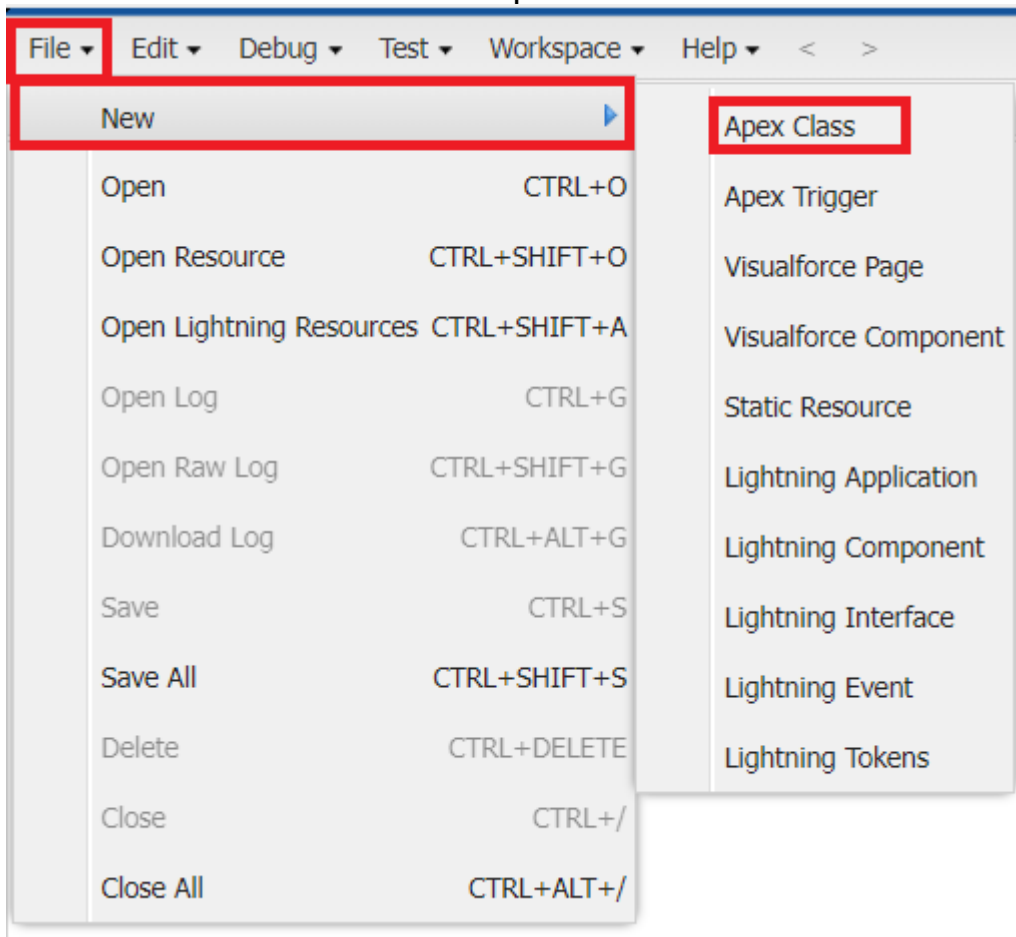
ACTIVITY 3 : Create an Apex Class

1. Go to Setup --> Click on the gear icon --> Select Developer Console.



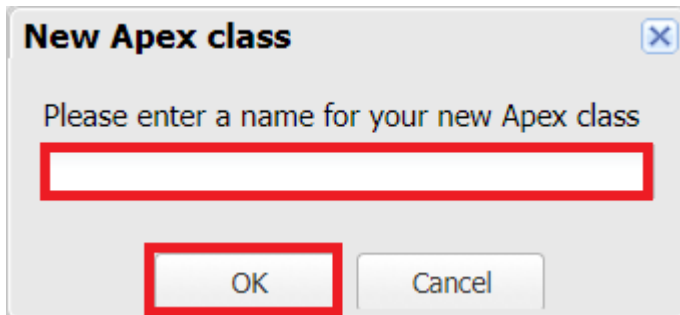
2. Then we can see the Developer console. Click on the developer console and you will navigate to a new console window.

3. To create a new Apex Class follow the below steps:
Click on the file --> New --> Apex Class.



4. Give the Apex Class name as "AssigningEmail".

Field service workorder optimization



4. Click ok.

5. Now write the code logic here

```
File Edit Debug Test Workspace Help < >
AssigningEmail.apxc
Code Coverage: None API Version: 59
1 public class AssigningEmail {
2     public static void sendEmailmsg(List<Assignment__c> assRec){
3         List<messaging.SingleEmailMessage> myVar = new List<messaging.SingleEmailMessage>();
4         Map<Id,Technician__c> technicians = new Map<Id,Technician__c>([SELECT Id, Phone__c, Location__c, Skills__c, Name__c, Email__c, Availability__c, Name FROM Technician__c]);
5         try{
6             for(Assignment__c con : assRec){
7                 if(con.Technician_ID__c != null){
8                     messaging.SingleEmailMessage mail = new messaging.SingleEmailMessage();
9                     List<String> sendTo = new List<String>();
10                    sendTo.add(technicians.Get(con.Technician_ID__c).Email__c);
11                    mail.setToAddresses(sendTo);
12                    string subject = 'email sub';
13                    mail.setSubject(subject);
14                    string body = 'email body ';
15                    mail.setHTMLbody(body);
16                    myVar.add(mail);
17                }
18            }
19            Messaging.sendEmail(myVar);
20        }
21        catch(exception e){
22            system.debug('Error -----> ' + e.getMessage());
23        }
24    }
25 }
26 }
```

Field service workorder optimization

Source Code:

```
public class AssigningEmail {
    public static void sendEmailmsg(List<Assignment__c> assRec){
        List<messaging.SingleEmailMessage> myVar = new List<messaging.SingleEmailMessage>();
        Map<id,Technician__c> technicians = new Map<id,Technician__c>([SELECT Id, Phone__c, Location__c,
Skills__c, Name__c, Email__c, Availability__c, Name FROM Technician__c]);
        try{
            for(Assignment__c con : assRec){
                if(con.Technician_ID__c != null){
                    messaging.SingleEmailMessage mail = new messaging.SingleEmailMessage();
                    List<String> sendTo = new List<String>();
                    sendTo.add(technicians.Get(con.Technician_ID__c).Email__c);
                    mail.setToAddresses(sendTo);
                    string subject = 'WorkOrder Assignment ';
                    mail.setSubject(subject);
                    string body = 'The following WorkOrder has been assigned to you ';
                    mail.setHTMLbody(body);
                    myVar.add(mail);
                }
            }
            Messaging.sendEmail(myvar);
        }
        catch(exception e){
            system.debug('Error -----> ' + e.getMessage());
        }
    }
}
```

Source Code:

```
public class AssigningEmail {
    public static void sendEmailmsg(List<Assignment__c> assRec){
        List<messaging.SingleEmailMessage> myVar = new List<messaging.SingleEmailMessage>();
        Map<id,Technician__c> technicians = new Map<id,Technician__c>([SELECT Id, Phone__c, Location__c,
Skills__c, Name__c, Email__c, Availability__c, Name FROM Technician__c]);
        try{
            for(Assignment__c con : assRec){
                if(con.Technician_ID__c != null){
                    messaging.SingleEmailMessage mail = new messaging.SingleEmailMessage();
                    List<String> sendTo = new List<String>();
                    sendTo.add(technicians.Get(con.Technician_ID__c).Email__c);
                    mail.setToAddresses(sendTo);
                    string subject = 'WorkOrder Assignment ';
                    mail.setSubject(subject);
                    string body = 'The following WorkOrder has been assigned to you ';
                    mail.setHTMLbody(body);
                    myVar.add(mail);
                }
            }
            Messaging.sendEmail(myvar);
        }
        catch(exception e){
            system.debug('Error -----> ' + e.getMessage());
        }
    }
}
```

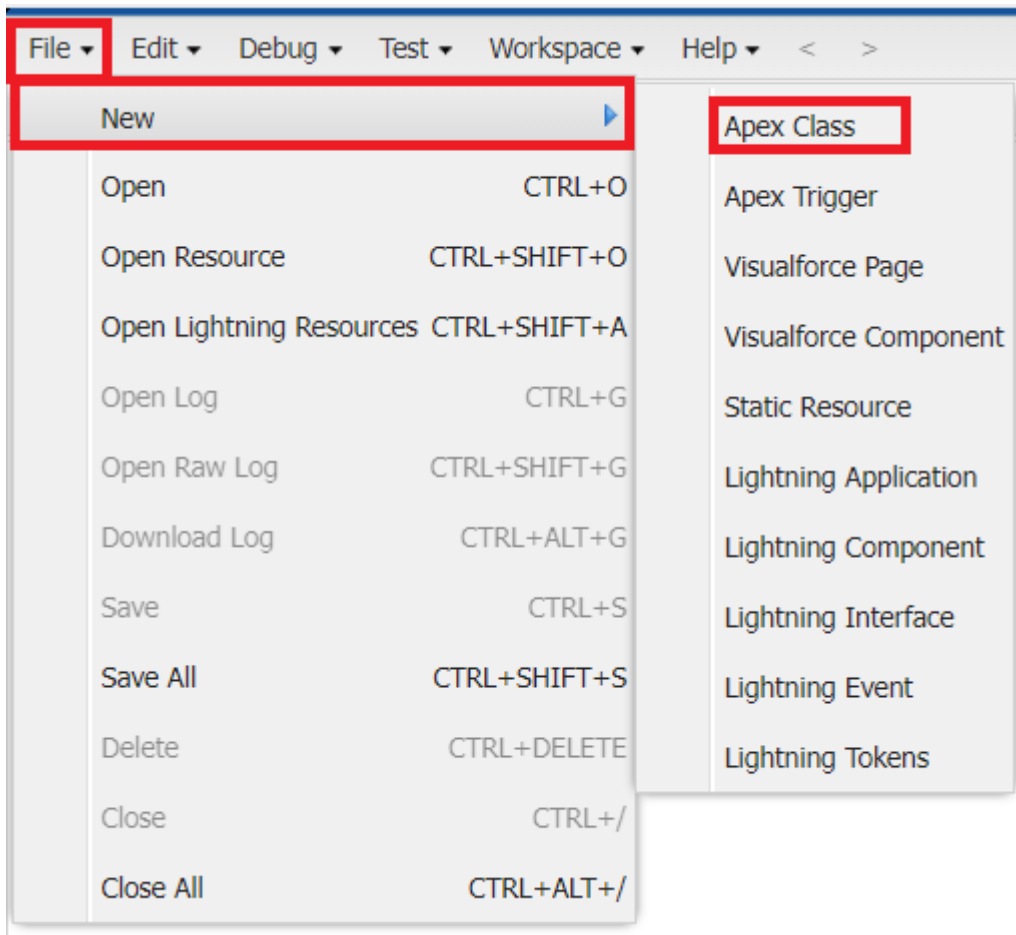
6. Save the code.(click on file --> Save)

Field service workorder optimization

ACTIVITY 4: Create an Apex Trigger

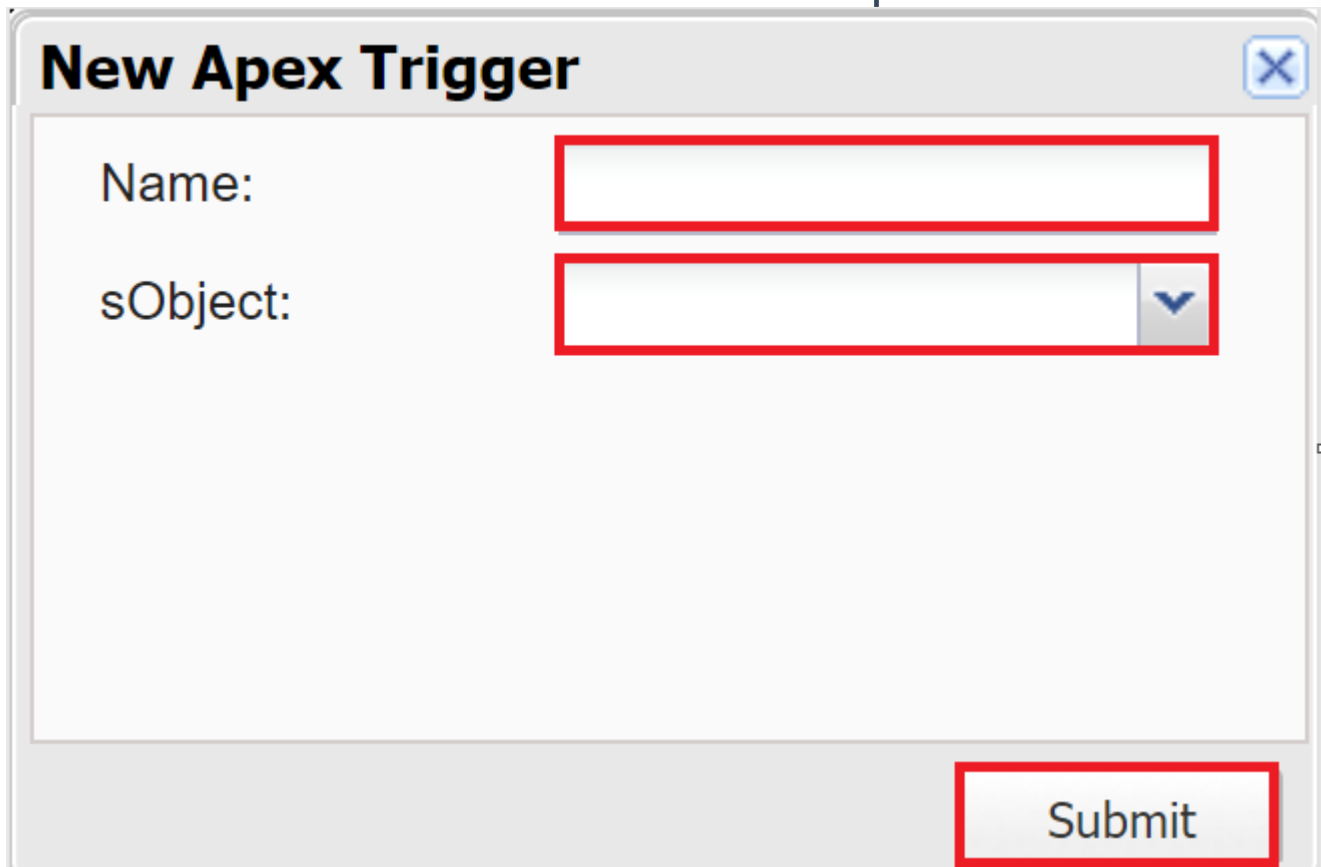
To create a new Apex Class follow the below steps:

1. Click on the file --> New --> Apex Class.



2. Give the Apex Trigger name as "AssignmentTrigger", and select "Assignment__c" from the dropdown for sObject.

Field service workorder optimization

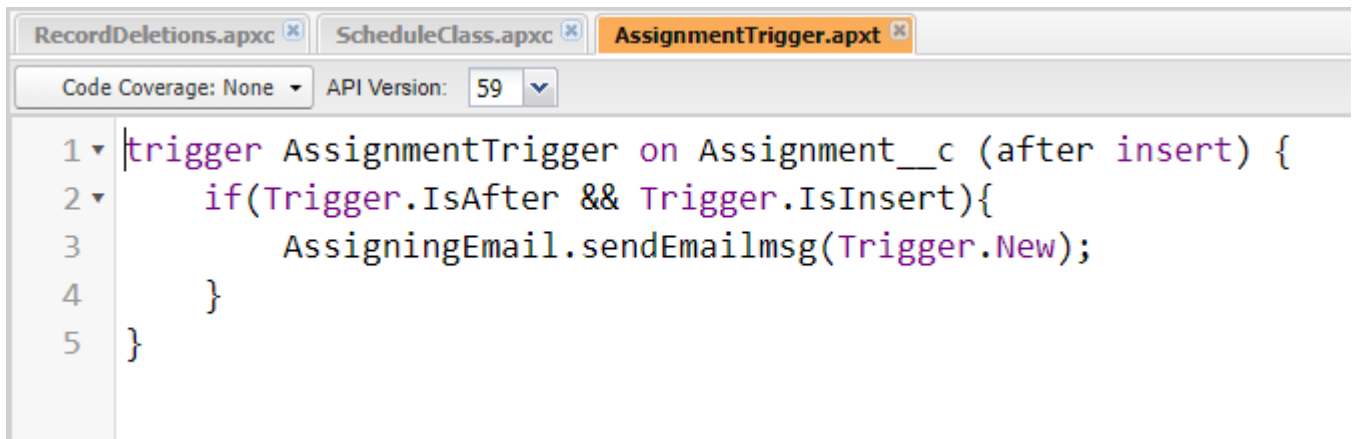


New Apex Trigger

Name:

sObject:

3. Click Submit.
4. Now write the code logic here



```
1 trigger AssignmentTrigger on Assignment__c (after insert) {
2     if(Trigger.IsAfter && Trigger.IsInsert){
3         AssigningEmail.sendEmailmsg(Trigger.New);
4     }
5 }
```

Source Code:

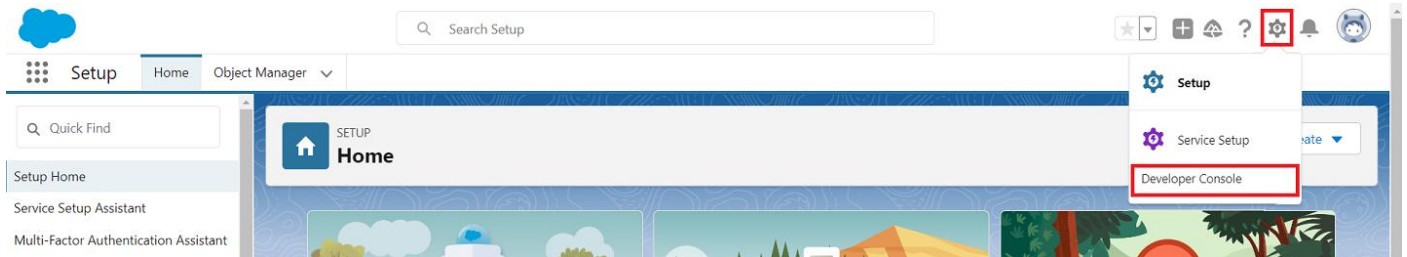
```
trigger AssignmentTrigger on Assignment__c (after insert) {
    if(Trigger.IsAfter && Trigger.IsInsert){
        AssigningEmail.sendEmailmsg(Trigger.New);
    }
}
```

5. Save the code.(click on file --> Save)

Field service workorder optimization

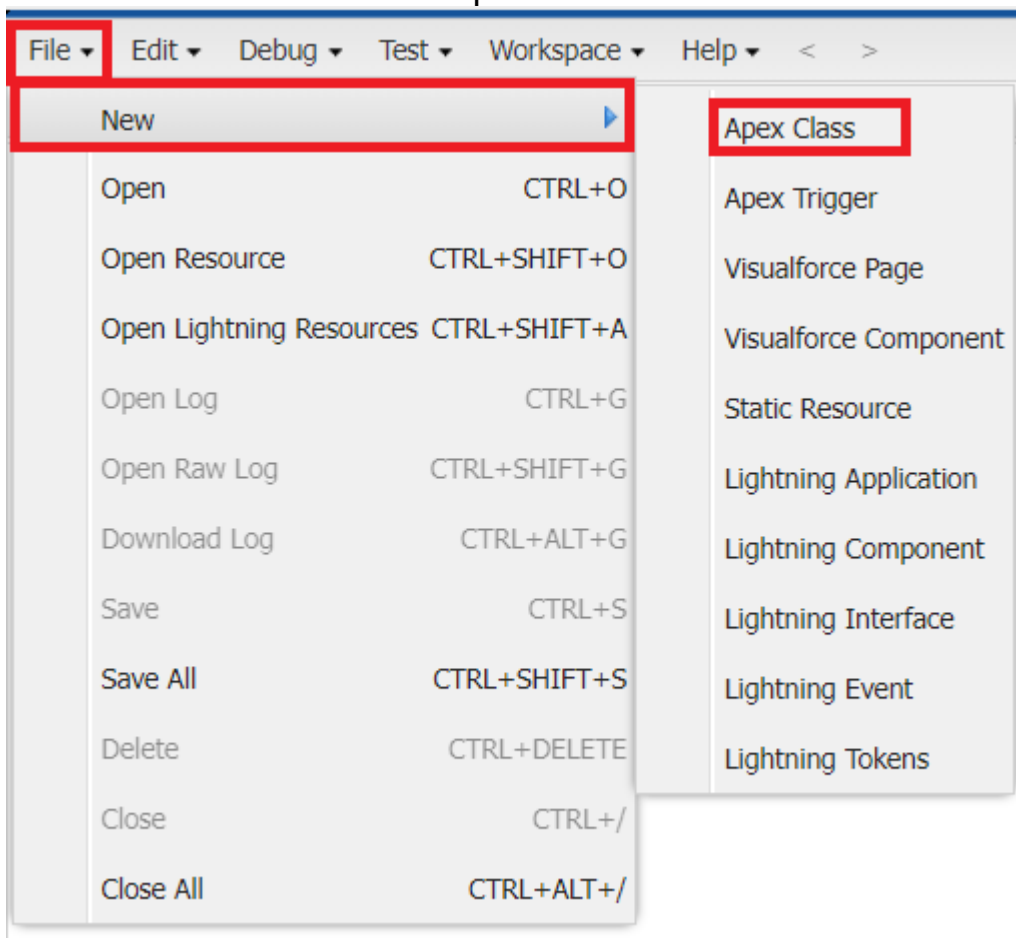
ACTIVITY 5 : Create an Apex Class

1. Go to Setup --> Click on the gear icon --> Select Developer Console.



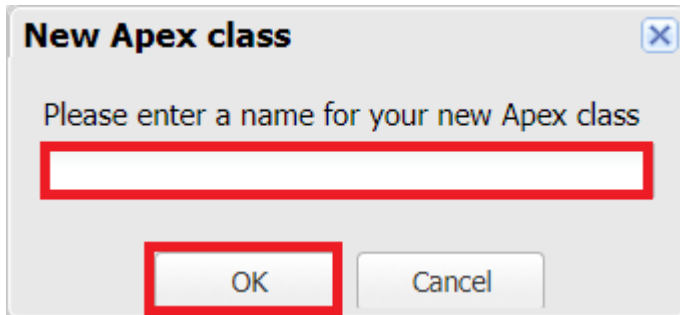
2. Then we can see the Developer console. Click on the developer console and you will navigate to a new console window.

3. To create a new Apex Class follow the below steps:
Click on the file --> New --> Apex Class.



4. Give the Apex Class name as "CompletionMail".

Field service workorder optimization



5. Click ok.

6. Now write the code logic here

```
RecordDeletions.apxc | ScheduleClass.apxc | CompletionMail.apxc
Code Coverage: None | API Version: 59
1 public class CompletionMail {
2     public static void sendEmailMsg(List<WorkOrder__c> workOrderList){
3         List<messaging.SingleEmailMessage> myVar = new List<messaging.SingleEmailMessage>();
4         for(WorkOrder__c con : workOrderList){
5             if(con.Status__c == 'Resolved'){
6                 messaging.SingleEmailMessage mail = new messaging.SingleEmailMessage();
7                 List<String> sendTo = new List<String>();
8                 sendTo.add(con.Email__c);
9                 mail.setToAddresses(sendTo);
10                string subject = 'Status Updated';
11                mail.setSubject(subject);
12                string body = 'email body ';
13                mail.setHTMLbody(body);
14                myVar.add(mail);
15            }
16        }
17        Messaging.sendEmail(myvar);
18    }
19 }
```


Field service workorder optimization

Source Code:

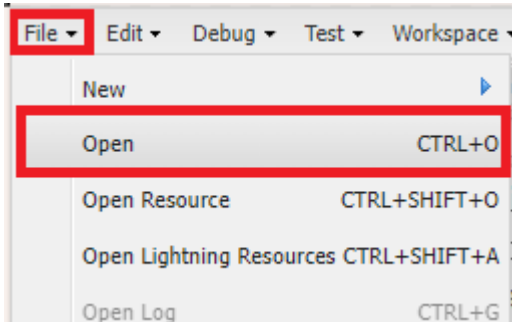
```
public class CompletionMail {
    public static void sendEmailMsg(List<WorkOrder__c> workOrderList){
        List<messaging.SingleEmailMessage> myVar = new
List<messaging.SingleEmailMessage>();
        for(WorkOrder__c con : workOrderList){
            if(con.Status__c == 'Resolved'){
                messaging.SingleEmailMessage mail = new messaging.SingleEmailMessage();
                List<String> sendTo = new List<String>();
                sendTo.add(con.Email__c);
                mail.setToAddresses(sendTo);
                string subject = 'Status Updated';
                mail.setSubject(subject);
                string body = 'email body ';
                mail.setHTMLbody(body);
                myVar.add(mail);
            }
        }
        Messaging.sendEmail(myvar);
    }
}
```

7. Save the code.(click on file --> Save)

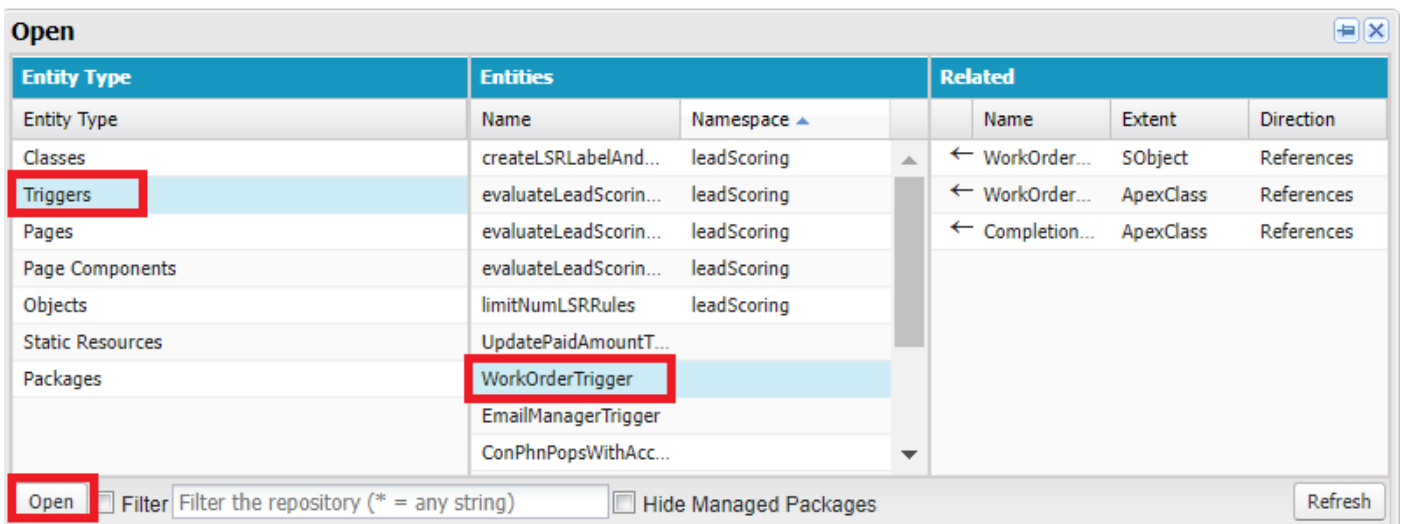
Field service workorder optimization

ACTIVITY 6 : Create an Apex Trigger

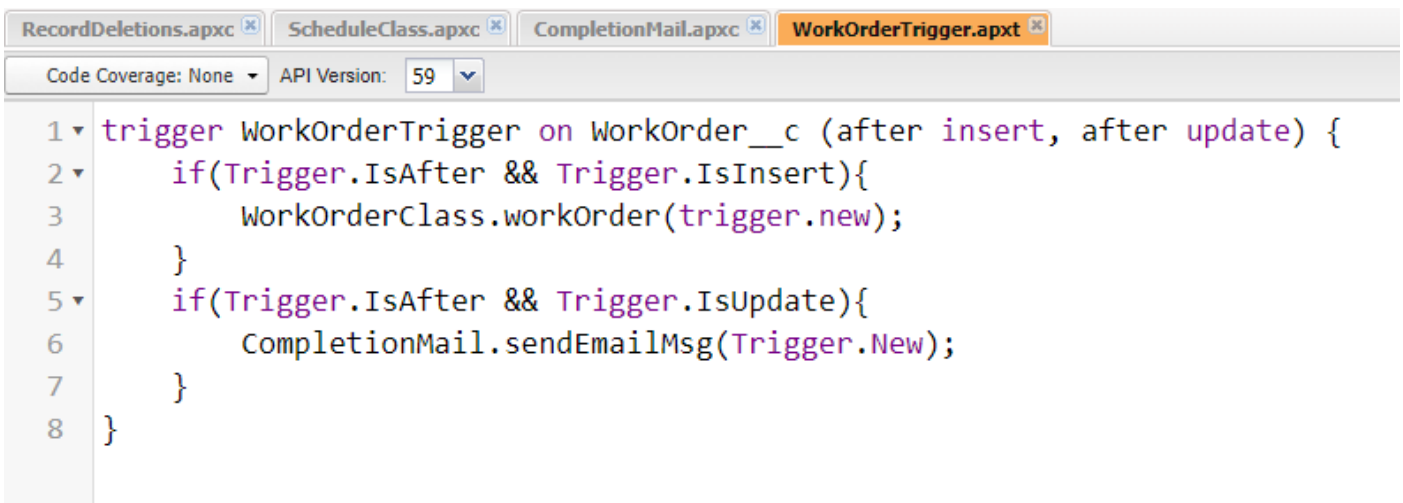
1. Click on the file --> Open.



2. A pop up window opens click on Triggers, then select "WorkOrderTrigger" and click on "Open"



3. Now write the code logic here.



Field service workorder optimization

```
WorkOrderClass.workOrder(trigger.new);
}
if(trigger.IsAfter && Trigger.IsUpdate){
    CompletionMail.sendEmailMsg(trigger.New);
}
}
```

4. Save the code.(click on file --> Save)

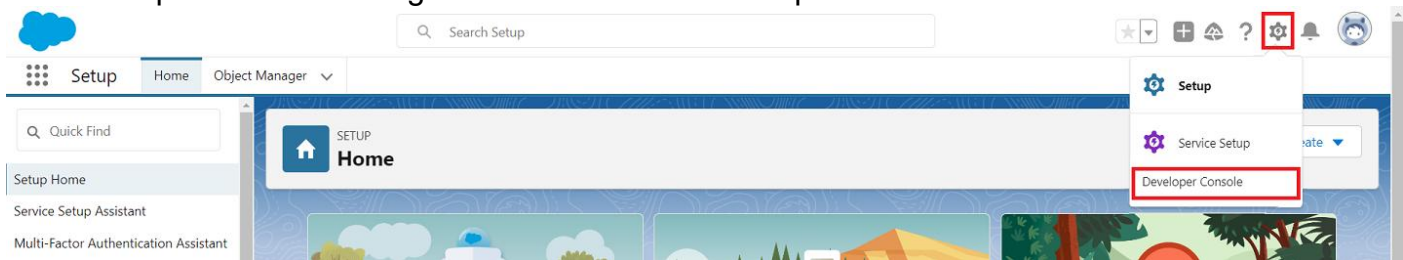
ACTIVITY 7 : Create an Asynchronous Apex Class

Create an Apex Class to Delete all the WorkOrder records which meets the following criteriaL

1. Completed date should be more than 30 days.
2. Status should be 'Resolved'.

Create an Apex Class

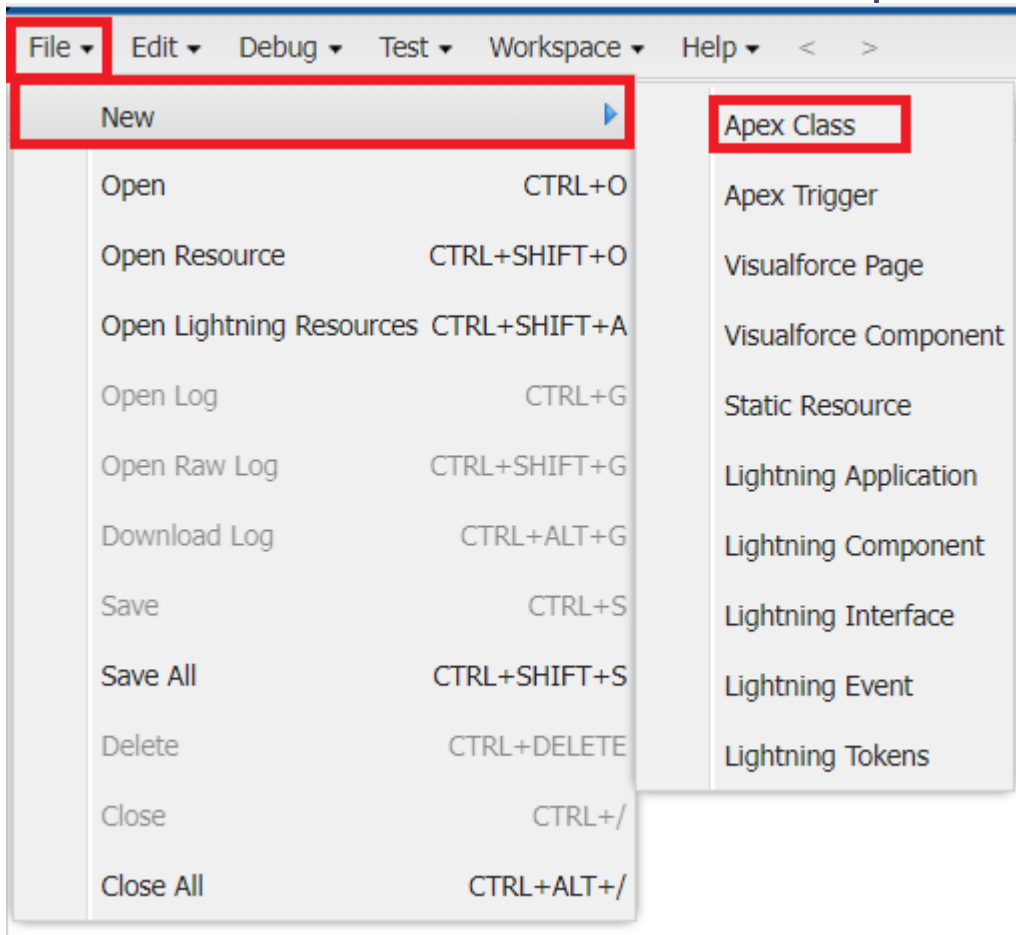
1. Go to Setup --> Click on the gear icon --> Select Developer Console.



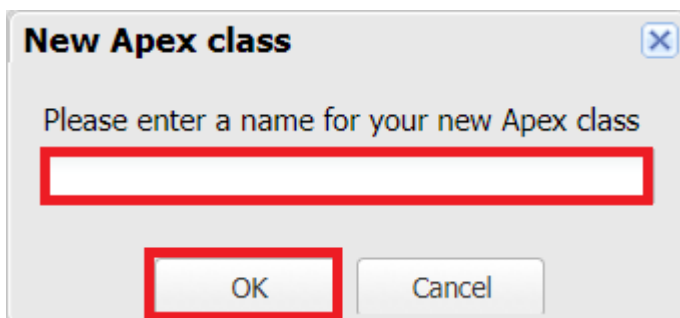
2. Then we can see the Developer console. Click on the developer console and you will navigate to a new console window.

3. To create a new Apex Class follow the below steps:
Click on the file --> New --> Apex Class.

Field service workorder optimization

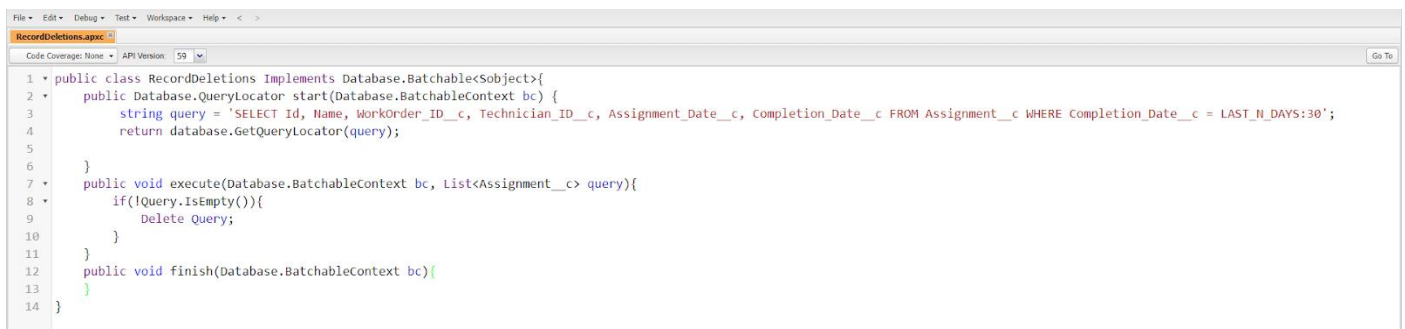


4. Give the Apex Class name as "RecordDeletion".



5. Click ok.

6. Now write the code logic here



Field service workorder optimization

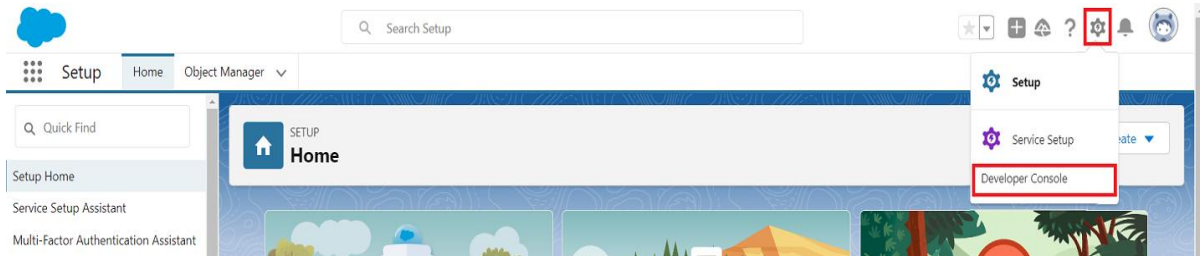
```
public class RecordDeletions Implements Database.Batchable<Sobject>{
    public Database.QueryLocator start(Database.BatchableContext bc) {
        string query = 'SELECT Id, Name, WorkOrder_ID__c, Technician_ID__c,
        Assignment_Date__c, Completion_Date__c FROM Assignment__c WHERE
        Completion_Date__c = LAST_N_DAYS:30';
        return database.GetQueryLocator(query);
    }
    public void execute(Database.BatchableContext bc, List<Assignment__c> query){
        if(!Query.IsEmpty()){
            Delete Query;
        }
    }
    public void finish(Database.BatchableContext bc){
    }
}
```

7. Save the code.(click on file --> Save)

Field service workorder optimization

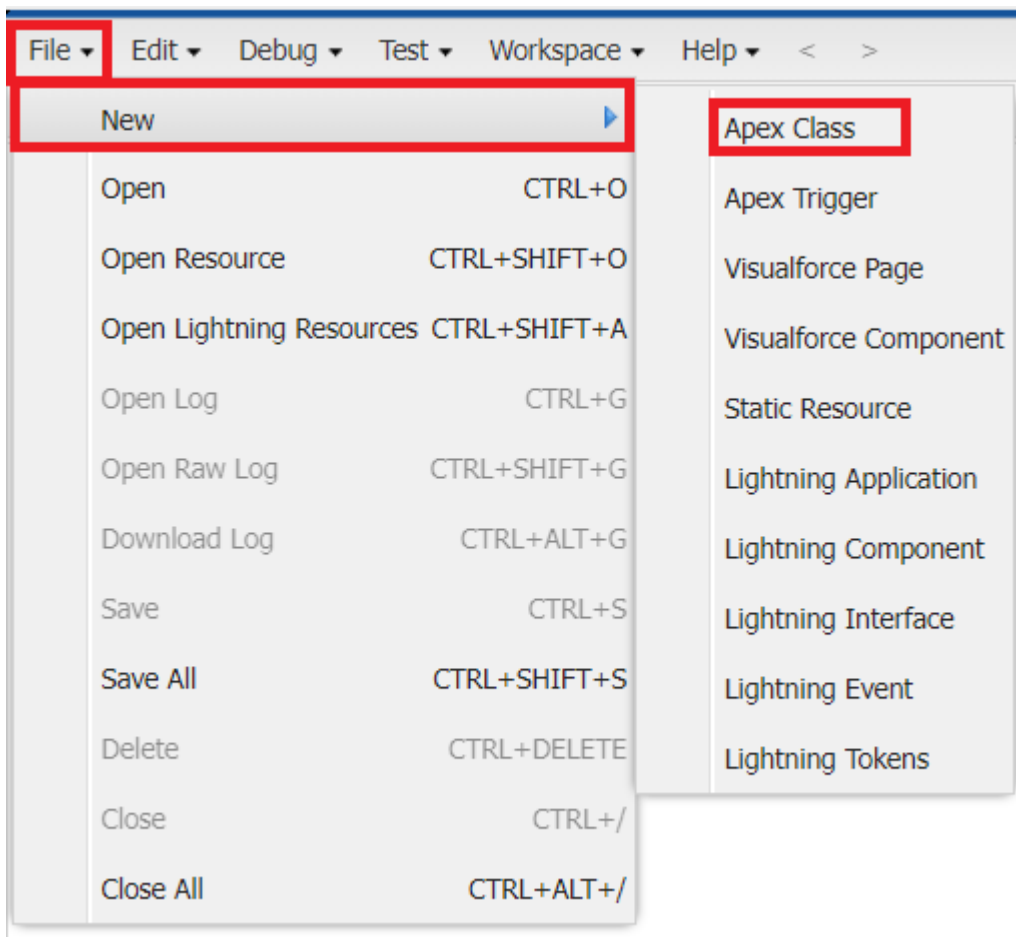
ACTIVITY 8 : Create an Apex Schedule Class

1. Go to Setup --> Click on the gear icon --> Select Developer Console.



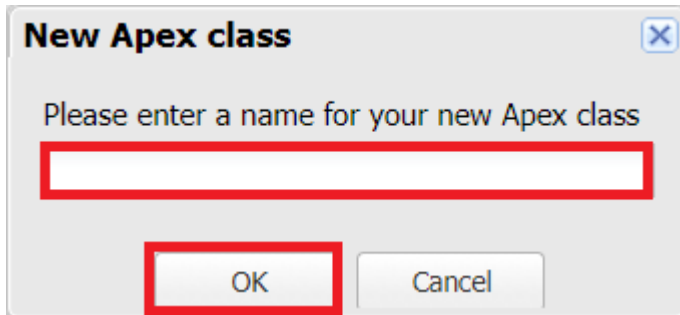
2. Then we can see the Developer console. Click on the developer console and you will navigate to a new console window.

3. To create a new Apex Class follow the below steps:
Click on the file --> New --> Apex Class.



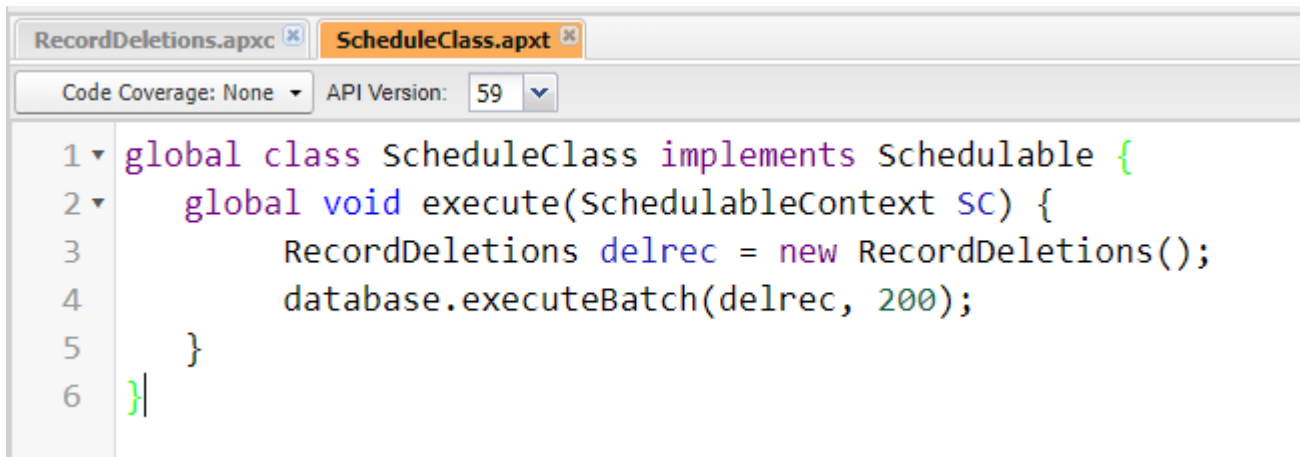
4. Give the Apex Class name as "ScheduleClass".

Field service workorder optimization



5. Click ok.

6. Now write the code logic here



Source Code:

```
global class ScheduleClass implements Schedulable {  
    global void execute(SchedulableContext SC) {  
        RecordDeletions delrec = new RecordDeletions();  
        database.executeBatch(delrec, 200);  
    }  
}
```

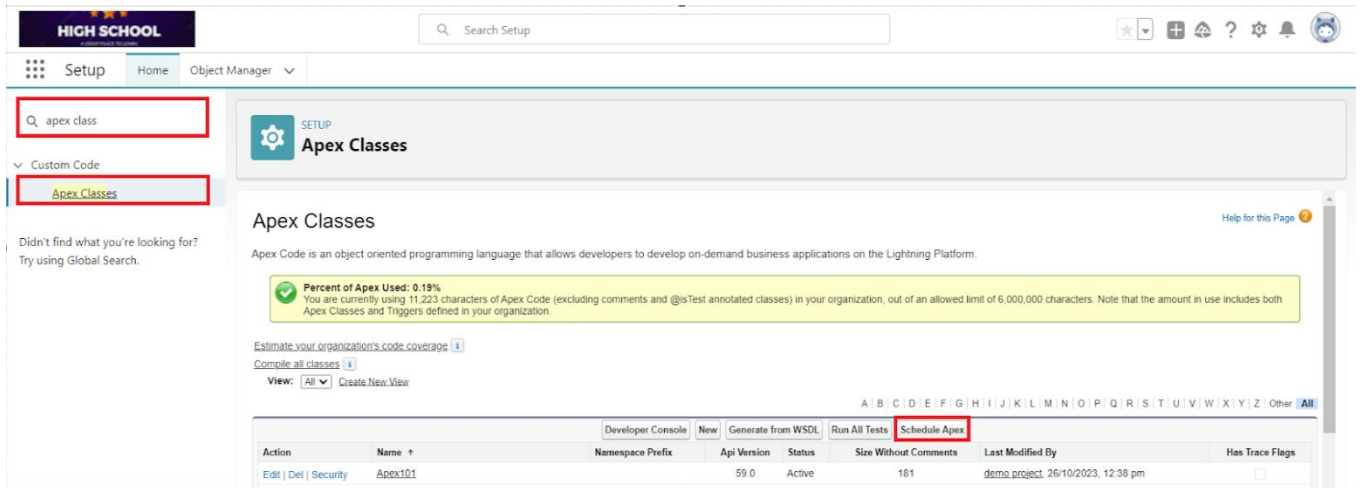
7. Save the code. (click on file ? Save)

Field service workorder optimization

ACTIVITY 9 : Create a Schedule Apex

Schedule the Apex class:

1. From the Setup page search for “Apex Classes” in quick search.
2. Click on “Schedule Apex” as shown below.



3. Click on Schedule Apex and enter the Job name.

- a. Job Name : DeleteAssignmentSchedule
- b. Apex Class : ScheduleClass (from clicking on lookup icon)
- c. Frequency : Monthly
- d. Preferred Start Time : Select any time

The screenshot shows the 'Schedule Apex' configuration page. The 'Job Name' field is set to 'DeleteAssignmentSchedule' and the 'Apex Class' field is set to 'ScheduleClass'. The 'Frequency' is set to 'Monthly'. The 'Start' date is '06/12/2023' and the 'End' date is '06/01/2024'. The 'Preferred Start Time' is set to '4:00 pm'. The 'Save' button is highlighted.

4. Click Save.

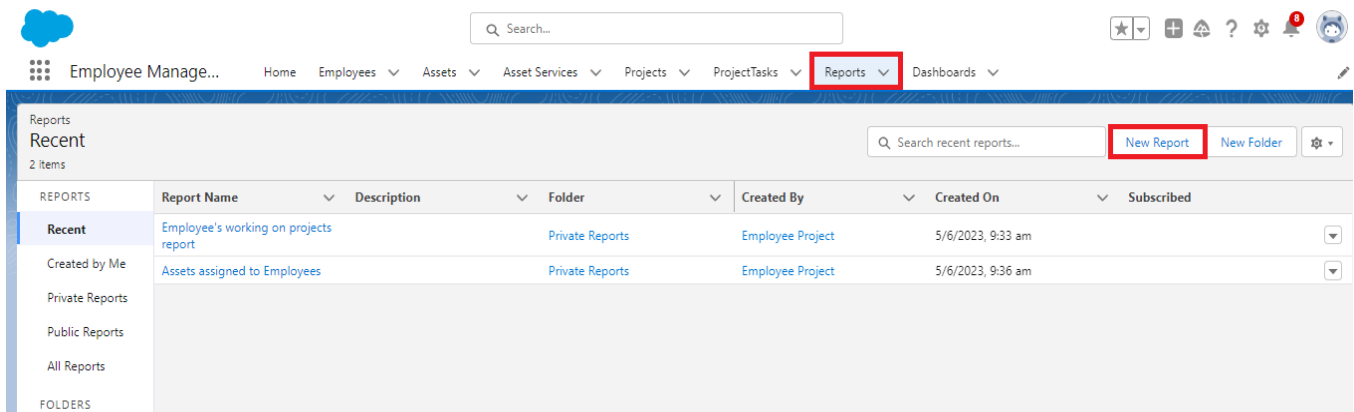
Field service workorder optimization

Reports & Dashboards

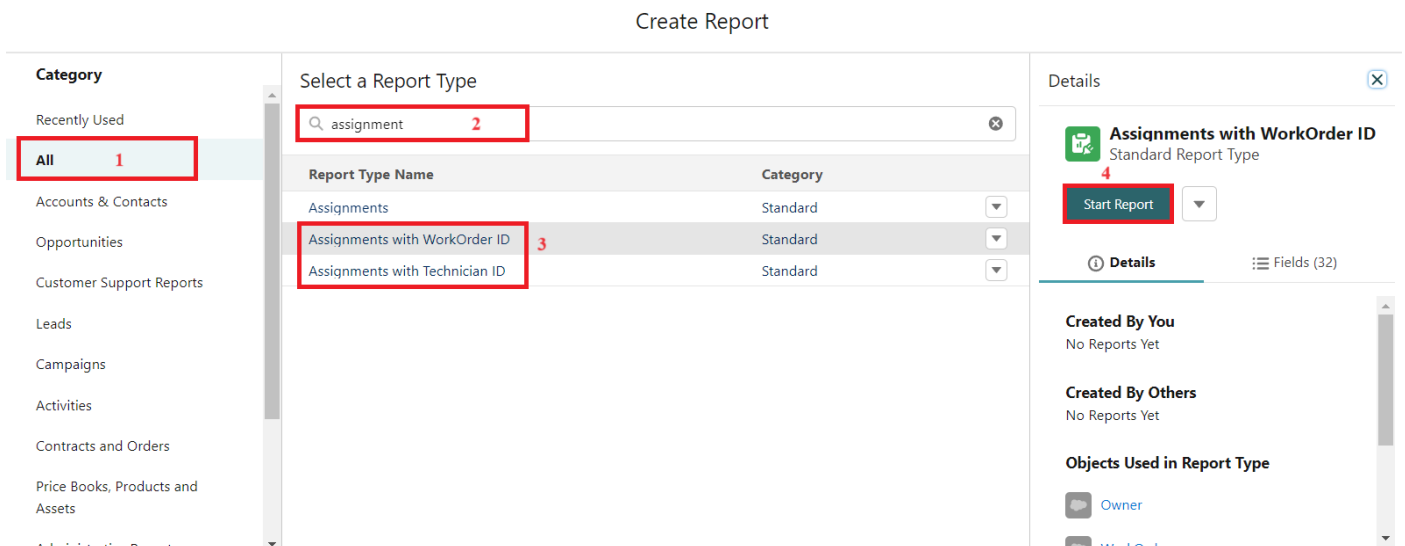
Salesforce Reports and Dashboards are powerful tools that empower users to visualize and analyze data within the Salesforce platform. They play a crucial role in providing insights, monitoring performance, and making informed business decisions.

ACTIVITY 1 : Report

1. Go to the app --> click on the reports tab
2. Click New Report.

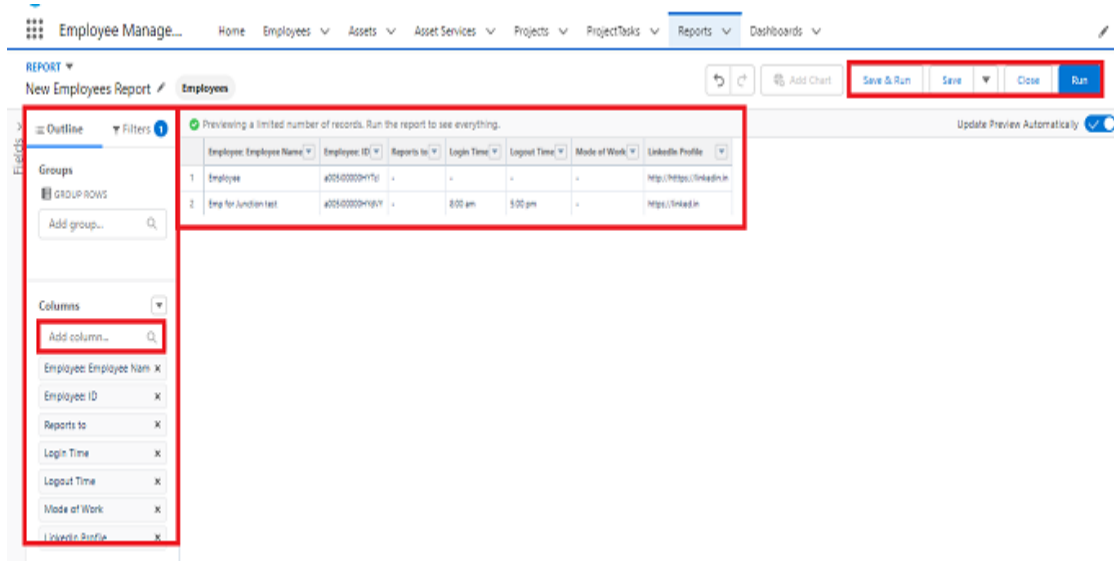


3. Select report type from category or from report type panel or from search panel --> click on start report.



Field service workorder optimization

4. Customize your report
 - Add fields from left pane as shown below
 - Grouped by workorder ID



5. Save or run it.

Note: Reports may get varied from the above pictures as the data might be different.

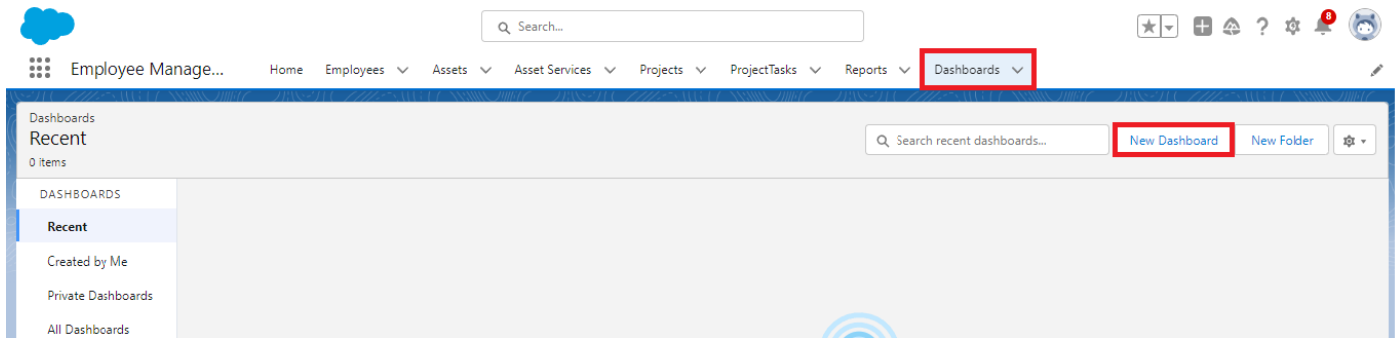
ACTIVITY 2 : Create Reports

1. Create a report with report type: "WorkOrders Status Reports".
2. Create a report with report type: "Technician and Assignment Details Reports".

Field service workorder optimization

ACTIVITY 3 : Dashboard

1. Go to the app --> click on the Dashboards tabs.



2. Give a Name and click on Create.

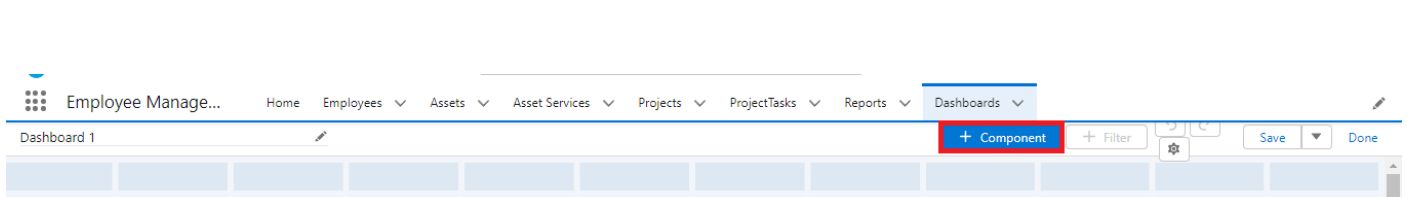
New Dashboard

*** Name**

Description

Folder

3. Select add component.



4 Select a Report which we have created in the previous activities and click on select.

Field service workorder optimization

Select Report

Reports

Recent

Created by Me

Private Reports

Public Reports

All Reports

Folders

Created by Me

Shared with Me

Select Report

Q Search Reports and Folders...

Reports and Folders ▼

Employee's working on projects report
Employee Project · 05-Jun-2023, 9:37 am · [Private Reports](#)

Assets assigned to Employees
Employee Project · 05-Jun-2023, 9:36 am · [Private Reports](#)

Cancel Select

5. Click Add then click on Save and then click on Done.

ACTIVITY 4 : Create Dashboards

Create another Dashboard as we discussed in activity 3 which shows the details of completed workorder status in a vertical bar graph.

Field service workorder optimization