

Field Service Work Optimization



By

Jagilinki Indu

jagilinkiiindu@gmail.com

Project Abstract

The Field Service Work Order Optimization System is designed to enhance the efficiency of operations for companies engaged in installations and repairs. This advanced system leverages a comprehensive database to align work orders with technicians based on their location, availability, and expertise. By employing a sophisticated prioritization algorithm, the system optimally assigns tasks, ensuring that resources are utilized effectively and service quality is maintained. Automated communication features keep technicians informed in real-time, while integrated analytics provide valuable insights for ongoing operational improvements. This solution not only streamlines field service management but also significantly reduces operational costs and boosts customer satisfaction, addressing the complex demands of modern field service environments.

INDEX Page

Topics

Salesforce -----

Creating Developer Account

Account Activation

Creating Objects -----

Create Technician Object

Create Workorder Object

Create Assignment Object

Creating Tabs -----

Creating A Custom Tab

The Lightning App -----

Create A Lightning App

Fields & Relationships -----

Creating Lookup Field In Assignment Object

Manage Your Picklist Values

Creating Formula Field In Workorder Object

Creating Remaining Fields For The Respective Objects

Profiles-----

Technician Profile

Users -----

Create User

Apex Triggers & Classes-----

Create An Apex Class

Create An Apex Trigger

Create An Asynchronous Apex Class

Create An Apex Schedule

Create A Schedule Apex

Reports & Dashboards -----

Create Reports

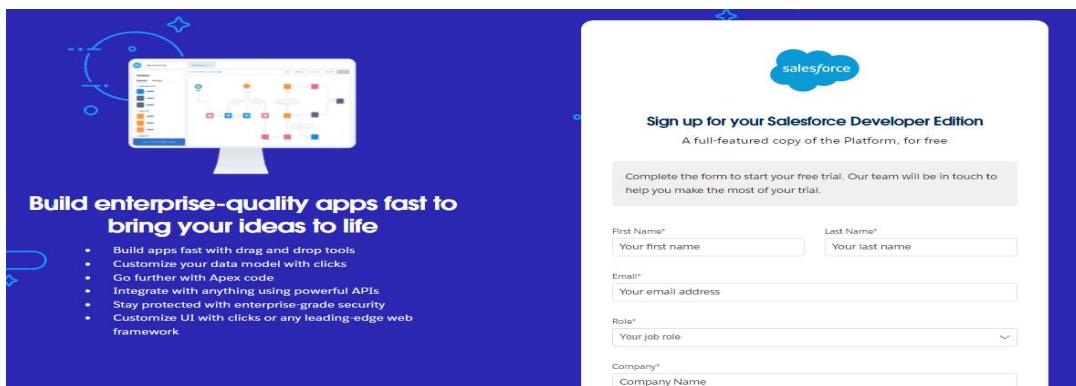
Create Dashboards

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 technicians 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.

Task 1: Creating Salesforce Account

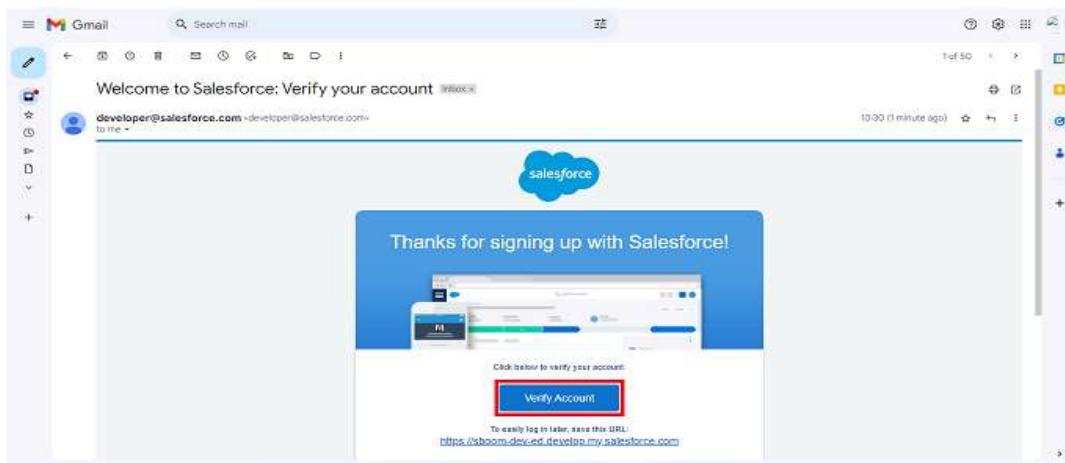
STEP 1 : Creating Developer Account



Go to <https://developer.salesforce.com/signup>

If you had an existing account in salesforce developer then directly sign in without signup

STEP 2 : Activation Account



Task 2: Creating Objects

Step 3 : Create Technician Object

1. Download and open [this spreadsheet](#), save it as Technician.csv.

	A	B	C	D	E	F	G
1	Technician ID	Name	Phone	Email	Location	Availability	Skills
2	T-0001	Raghu	7892341560	example@gmail	Hyderabad	Available	Machine Installat
3	T-0002	Raghav	7892341560	example@gmail	Pune	Not Available	Hardware Repair
4	T-0003	Shyam	7892341560	example@gmail	Nasik	Not Available	Troubleshoot/Det
5	T-0004	Mastan	7892341560	example@gmail	Warangal	Available	Maintenance
6	T-0005	Ramesh	7892341560	example@gmail	Nanded	Available	Lane-Manageme
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							
21							
22							
23							
24							
25							
26							
27							

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.

CSV File Details

Encoding Format: Unicode (UTF8) Values Separated By: Comma Field Label Source: Detect from row * Field Labels Row: 1 Import 5 rows of Data? No, skip import Yes, import data

Record Name Field: Technician ID

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

Back Next

12. Click Next and enter the following settings.
13. Click Finish. The Technician object is successfully created and data imported, all within minutes.

Step 3 : Create WorkOrder Object

Repeat the same steps as we done for the Technician ID. By making sure to do field mapping with proper field type as shown below

SETUP > OBJECT MANAGER
WorkOrder

Details

Custom Object Definition Edit

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: WorkOrder Example: Account
Plural Label: WorkOrders Example: Accounts
Starts with vowel sound:

The Object Name is used when referencing the object via the API.
Object Name: WorkOrder_ID Example: Account

Description:

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

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: WorkOrder ID Example: Account Name
Data Type: Text 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

By entering the values Data mentioned in the picture and follow up by the fields types

CSV File Details

Fields 7 of 7 to import Hide mapped fields

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

Import 0 rows of Data? No, skip import Yes, import data

Record Name Field

Back Next

Step 3 : Create Assignment Object

SETUP > OBJECT MANAGER

Assignment

Custom Object Definition Edit

Details

Custom Object Information

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

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:

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.
Display Format: Example: A-(000) What is This?

Optional Features

Allow Reports Allow Activities

Create the Assignment object with the following details mentioned in the picture and save it

Task 3: Creating Tabs

Step 1 : Creating A Custom Tab

Action	Label	Tab Style
Edit Del	Assignments	Guitar

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

Task 4: Create a Lightning App

Step 1 :

Go to setup page → search “app manager” in quick find → select “app manager”
→ click on New lightning App

Then fill the details below mentioned in picture and save and finish

Task 5: Fields & Relationship

Step 1 : Creating Lookup Field in Assignment Object

Go to setup → click on Object Manager → type Assignment object name in quick find bar → click on the object.

The screenshot shows the Salesforce Object Manager interface. At the top, there are navigation links: Setup, Home, and Object Manager. A red arrow points to the 'Object Manager' link. Below it, a search bar contains the text 'assignment'. Another red arrow points to this search bar. The main area displays a table with two rows. The first row is for the 'Assignment' object, which is highlighted with a red box. The second row is for 'Location Group Assignment'. The columns in the table are: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED.

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

Now click on “Fields & Relationships” → New

The screenshot shows the 'Assignment' object's Fields & Relationships page. On the left, there is a sidebar with options like Details, Fields & Relationships (which is selected and highlighted with a red box), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, and Field Sets. The main area is titled 'Fields & Relationships' and shows a table with four items. The columns are: FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The items listed are Assignment ID, Created By, Last Modified By, and Owner.

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

Select Data type as “Lookup”.

The screenshot shows the 'Step 1. Choose the field type' dialog. At the top right, there are 'Next' and 'Cancel' buttons. The main area has a sub-header 'Data Type'. It lists several options: None Selected, Auto Number, Formula, Roll-Up Summary, Lookup Relationship (which is selected and highlighted with a red box), and Master-Detail Relationship. Each option has a brief description below it.

Click on Next

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

Assignment
New Relationship

Help for this Page

Step 2. Choose the related object

Select the other object to which this object is related.

Related To **WorkOrder**

Step 2

Previous Next Cancel

Previous Next Cancel

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

Assignment
New Relationship

Help for this Page

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

Step 3 of 6

Field Label **WorkOrder ID**

Field Name **WorkOrder_ID**

Description

Help Text

Previous Next Cancel

WorkOrder ID WorkOrder_ID_c Lookup(WorkOrder)

Next → Next → Save & New

Step 2 : Manage your picklist values

- From rom the setup page go to object manager
- Search and Select Workorder object.
- Go to fields & relationship, select Location field, scroll down to values and click “New”.

Values

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

Values

						New	Reorder	Replace	Printable View	Chart Colors	Values Help
						Delete Selected	Deactivate Selected	Replace Selected			
Action	Values	API Name	Default	Chart Colors	Modified By						
<input type="checkbox"/>	Edit Del Deactivate	Pune	<input type="checkbox"/>	Assigned dynamically	Jagilinki Indu, 07/08/2024, 3:58 pm						
<input type="checkbox"/>	Edit Del Deactivate	Hyderabad	<input type="checkbox"/>	Assigned dynamically	Jagilinki Indu, 07/08/2024, 3:58 pm						
<input type="checkbox"/>	Edit Del Deactivate	Nasik	<input type="checkbox"/>	Assigned dynamically	Jagilinki Indu, 07/08/2024, 11:23 pm						
<input type="checkbox"/>	Edit Del Deactivate	Warangal	<input type="checkbox"/>	Assigned dynamically	Jagilinki Indu, 07/08/2024, 11:23 pm						
<input type="checkbox"/>	Edit Del Deactivate	Nanded	<input type="checkbox"/>	Assigned dynamically	Jagilinki Indu, 07/08/2024, 11:23 pm						

Step 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

Fields & Relationships				
16 Items, Sorted by Data Type				
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
WorkOrder ID	Name	Text(80)		✓
Status	Status_c	Picklist		▼
Priority	Priority_c	Picklist		▼
Location	Location_c	Picklist		▼
Service Type	Service_Type_c	Picklist		▼
WorkOrder ID	WorkOrder_ID_c	Lookup(WorkOrder)		✓

Step 3 : Creating Formula Field in WorkOrder Objects

1. Go to setup --> click on Object Manager --> type object name(WorkOrder) in quick find bar--> click on the object.
2. Now click on “Fields & Relationships” --> New
3. Select Data type as “Formula” and click Next.
4. Give Field Label and Field Name as “Date_c” and select formula return type as “Date_c” and click next.

Step 2. Choose output type

Step 2 of 5
Previous **Next** Cancel

Field Label	<input type="text"/>
Field Name	<input type="text"/>

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

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|`
- 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|`

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

Formula: CreatedDate

6. Next--> Next--> Save.

SETUP > OBJECT MANAGER
WorkOrder

Details Fields & Relationships

Page Layouts Lightning Record Pages Buttons, Links, and Actions Compact Layouts Field Sets Object Limits Record Types Related Lookup Filters Search Layouts List View Button Layout Restriction Rules Scoping Rules Triggers Flow Triggers Validation Rules

WorkOrder Custom Field
Date__c
Back to WorkOrder

Custom Field Definition Detail

Field Information

Field Label	Date__c	Object Name	WorkOrder
Field Name	Date__c		
API Name	Date_c__c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	Jagilink Indu 07/08/2024, 11:40 pm	Modified By	Jagilink Indu 07/08/2024, 11:40 pm

Formula Options

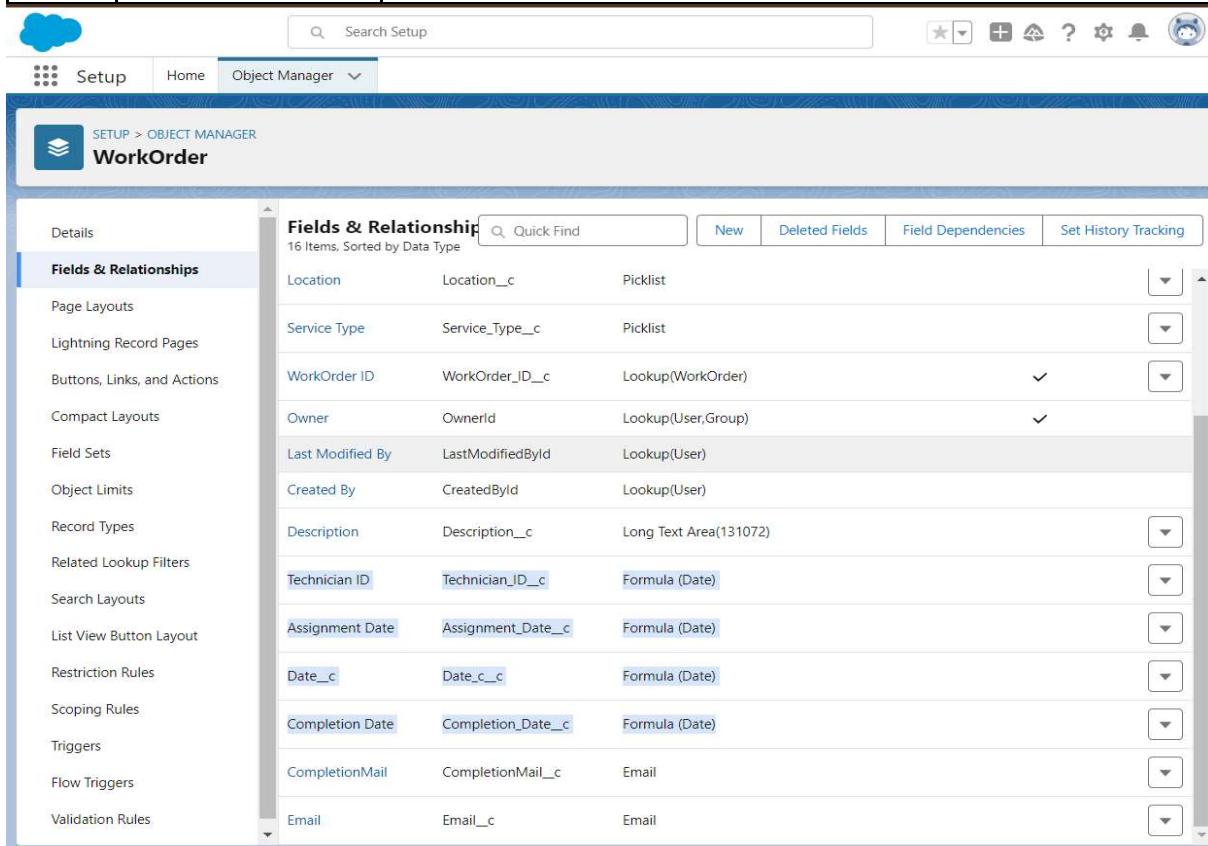
CreatedDate	Data Type	Formula
-------------	-----------	---------

Step 3 : 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

		Field Name	Datatype
1	Assignment	<ul style="list-style-type: none"> ● Technician ID ● Assignment Date ● Completion Date 	Lookup(Technician) Formula: return type : Date (WorkOrder_ID__r.Date__c) Formula: return type : Date IF(ISPICKVAL(WorkOrder_ID__r.Status__c , 'Resolved'), WorkOrder_ID__r.LastModifiedD ate , NULL)



The screenshot shows the Salesforce Object Manager interface for the 'WorkOrder' object. The top navigation bar includes a cloud icon, a search bar with 'Search Setup', and various global buttons. The main area displays the 'Fields & Relationships' section for the WorkOrder object. On the left, a sidebar lists various configuration options like Details, Fields & Relationships (which is selected), Page Layouts, Lightning Record Pages, etc. The main table lists 16 items, sorted by Data Type, with columns for Field Name, Data Type, and Description.

Field Name	Data Type	Description
Location	Picklist	Location_c
Service Type	Picklist	Service_Type_c
WorkOrder ID	Lookup(WorkOrder)	WorkOrder_ID__c
Owner	Lookup(User,Group)	OwnerId
Last Modified By	Lookup(User)	LastModifiedById
Created By	Lookup(User)	CreatedById
Description	Long Text Area(131072)	Description_c
Technician ID	Formula (Date)	Technician_ID__c
Assignment Date	Formula (Date)	Assignment_Date__c
Date_c	Formula (Date)	Date_c_c
Completion Date	Formula (Date)	Completion_Date__c
CompletionMail	Email	CompletionMail_c
Email	Email	Email_c

Task 6: Profiles

Step 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.
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"/>				
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"/>

Custom Object Permissions						
	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"/>
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"/>				
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"/>				

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.
7. Click on Edit, enable the check box for the status field.
8. Click on Save.

The screenshot shows the Salesforce Setup interface with the 'Profiles' page open. The 'Technician3' profile is selected. In the 'Page Layouts' section, under 'Standard Object Layouts', the 'Object Milestone' layout is chosen for the 'WorkOrder' object. Other objects listed include Global, Lead, Object Milestone, Operating Hours, Order, Order Product, Payment, and Payment Authorization.

Task 7: Users

Step 1:

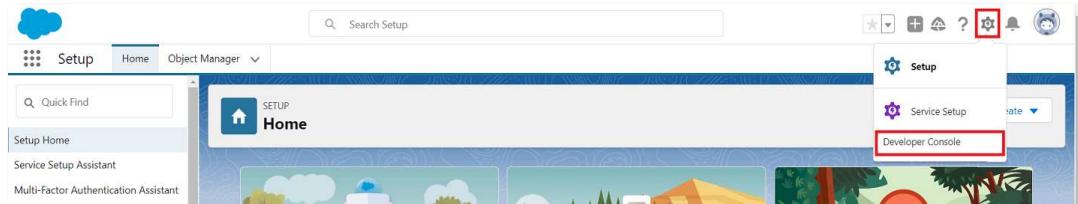
1. Go to setup --> type users in quick find box --> select users --> click New user.
2. Fill in the fields
1. First Name : Elina
2. Last Name : Gilbert
3. Alias : Give a Alias Name
4. Email id : Give your Personal Email id
5. Username : Username should be in this form: text@text.text
6. Nick Name : Give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

The screenshot shows the Salesforce Setup interface with the 'Users' tab selected. On the left, a sidebar lists categories like 'Permission Set Groups', 'Profiles', 'Public Groups', 'Queues', 'Roles', 'User Management Settings', and 'Users'. The 'Users' section is expanded, showing sub-options: 'Feature Settings', 'Data.com', and 'Prospector Users'. A search bar at the top right contains the text 'Search Setup'. The main content area is titled 'Elina Gilbert' and shows the 'User Detail' page. The 'User Detail' section includes fields for Name (Elina Gilbert), Alias (elina), Email (jagilinkindu@gmail.com), Username (jagilinkindu@gmail.com), Nickname (elina), Title, Company, Department, Division, Address, Time Zone (GMT+05:30) India Standard Time (Asia/Kolkata), Locale (English (India)), Language (English), Delegated Approver, Manager, Receive Approval Request Emails (Only if I am an approver), Federation ID, App Registration: One-Time Password Authenticator, App Registration: Salesforce Authenticator, Security Key (U2F or WebAuthn), Lightning Login, Temporary Verification Code (Expires in 1 to 24 Hours), and Generated (Temporary Verification Code). The 'Role' section shows 'Salesforce Platform' selected. The 'User License' section shows 'Technician3' selected. The 'Profiles' section shows 'Marketing User', 'Offline User', 'Knowledge User', 'Flow User', 'Service Cloud User', 'Site.com Contributor User', 'Site.com Publisher User', 'WDC User', and 'Mobile Push Registration' (View). The 'Data.com User Type' section shows 'Data.com User Type' (View). The 'Accessibility Mode (Classic Only)' section shows 'Accessibility Mode (Classic Only)' (View). The 'Debug Mode' section shows 'Debug Mode' (View). The 'High-Contrast Palette on Charts' section shows 'High-Contrast Palette on Charts' (View). The 'Load Lightning Pages While Scrolling' section shows 'Load Lightning Pages While Scrolling' checked. The 'Salesforce CRM Content Use' section shows 'Salesforce CRM Content Use' checked. The 'Receive Salesforce CRM Content Email Alerts' section shows 'Receive Salesforce CRM Content Email Alerts' checked. The 'Receive Salesforce CRM Content Alerts as Daily Digest' section shows 'Receive Salesforce CRM Content Alerts as Daily Digest' checked. The 'Make Setup My Default Landing Page' section shows 'Make Setup My Default Landing Page' (View). The 'Allow Forecasting' section shows 'Allow Forecasting' (View). The 'No MRU Updates' section shows 'No MRU Updates' (View).

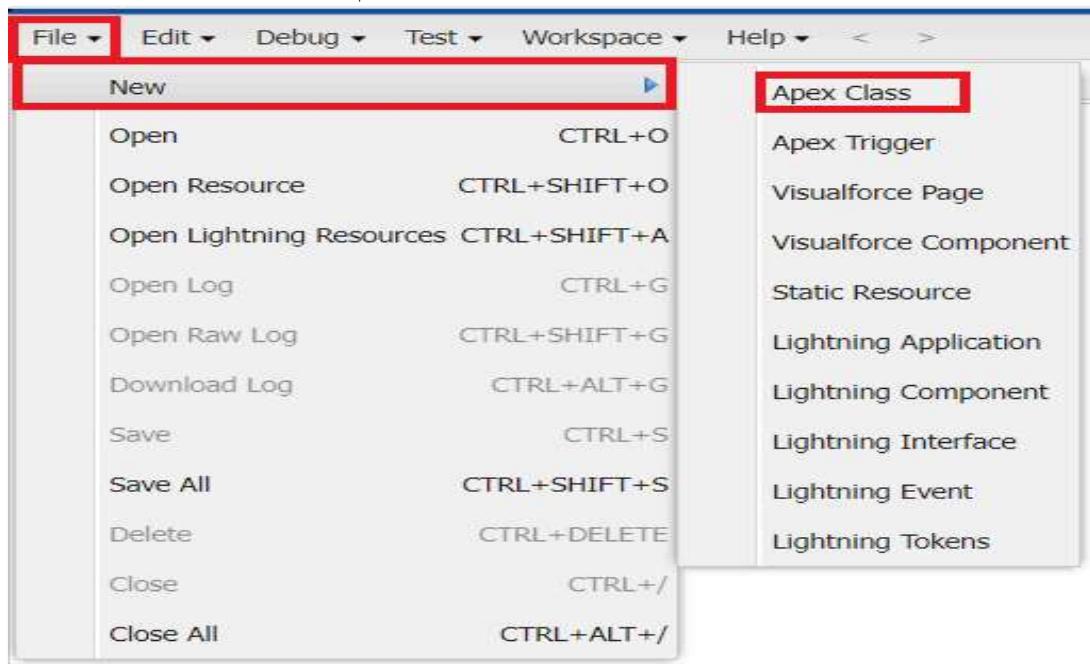
Task 8: Apex Trigger

Step 1: Create An Apex Class

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



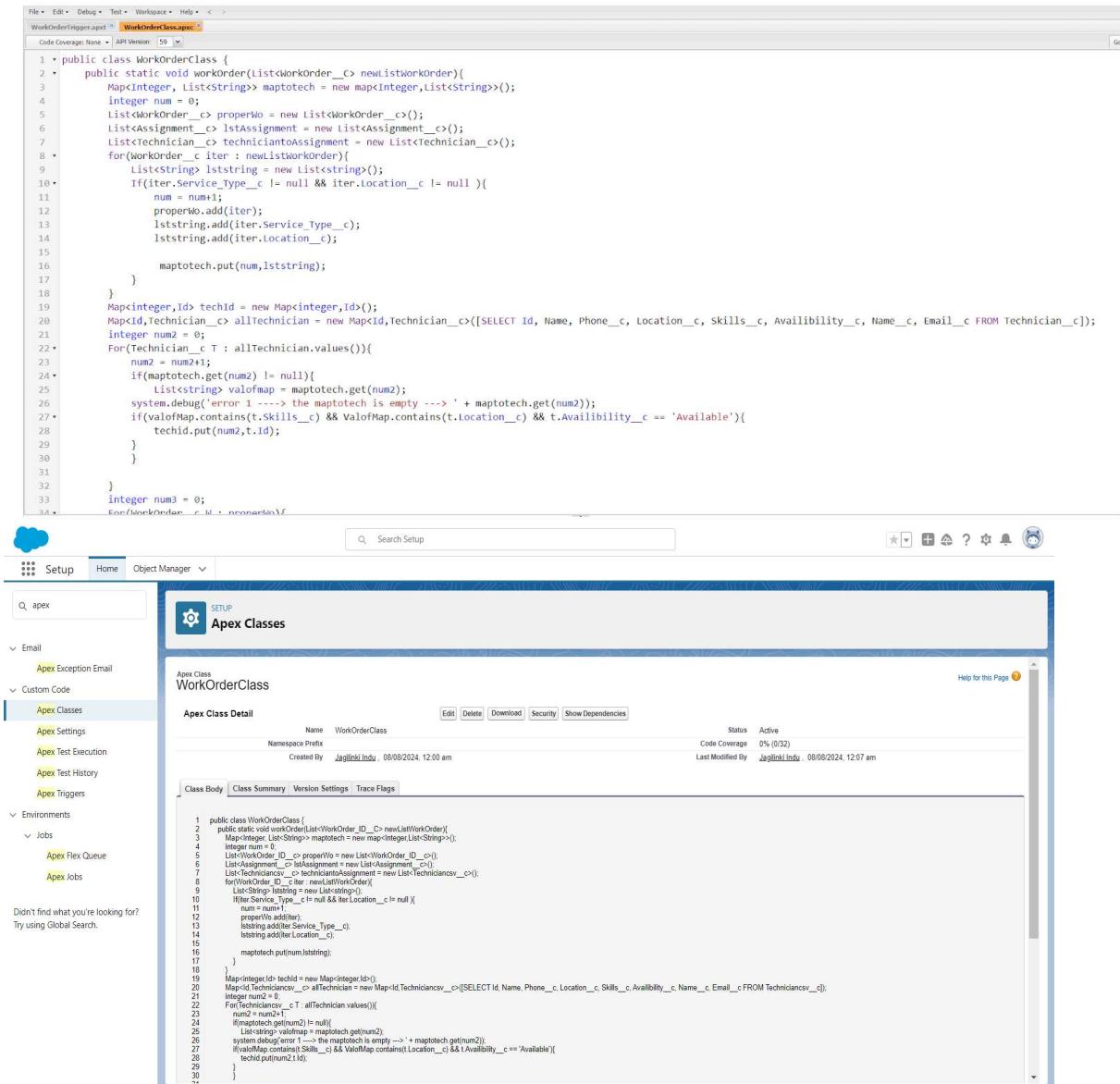
2. Then we can see the Developer console. Click on the developer console and you will navigate to a new console window.
3. To create a new Apex Class follow the below steps:
Click on the file --> New --> Apex Class.



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



5. Click ok.
6. Now write the code logic here



```

1 public class WorkOrderClass {
2     public static void worker(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> listAssignment = new List<Assignment__c>();
7         List<Technician__c> technicianAssignment = 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                mapToTech.put(num,lstString);
16            }
17        }
18        Map<integer,id> techId = new Map<integer,id>();
19        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]);
20        Integer num2 = 0;
21        For(Technician__c t : allTechnician.values()){
22            num2 = num2+1;
23            If(mapToTech.get(num2) != null){
24                List<string> valMap = mapToTech.get(num2);
25                system.debug('error 1 ---> the maptoTech is empty ---> ' + mapToTech.get(num2));
26                If(valMap.contains(t.skills__c) && ValMap.contains(t.location__c) && t.Availability__c == 'Available'){
27                    techId.put(num2,t.Id);
28                }
29            }
30        }
31    }
32    Integer num3 = 0;
33    For(WorkOrder__c iter : properWo){
34        num3 = num3+1;
35        If(iter.Service_Type__c != null && iter.Location__c != null ){
36            num = num+1;
37            properWo.add(iter);
38            lstString.add(iter.Service_Type__c);
39            lstString.add(iter.Location__c);
40            mapToTech.put(num,lstString);
41        }
42    }
43    Map<integer,id> techId = new Map<integer,id>();
44    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]);
45    Integer num2 = 0;
46    For(Technician__c t : allTechnician.values()){
47        num2 = num2+1;
48        If(mapToTech.get(num2) != null){
49            List<string> valMap = mapToTech.get(num2);
50            system.debug('error 1 ---> the maptoTech is empty ---> ' + mapToTech.get(num2));
51            If(valMap.contains(t.skills__c) && ValMap.contains(t.location__c) && t.Availability__c == 'Available'){
52                techId.put(num2,t.Id);
53            }
54        }
55    }
56 }

```

Step 2: Create An Apex Trigger

- To create a new Apex Class follow the below steps:
Click on the file --> New --> Apex Class.
- Give the Apex Trigger name as “WorkOrderTrigger”, and select “WorkOrder__c” from the dropdown for sObject.
- Click Submit.
- Now write the code logic here

The screenshot shows the Salesforce Setup interface with the search bar set to "apex tri". The results list "Apex Triggers" under "Custom Code". The "WorkOrderTrigger" is selected, showing its detail page. The trigger name is "WorkOrderTrigger" and it is defined on the "WorkOrder_ID__c" object. The trigger code is as follows:

```

1 trigger WorkOrderTrigger on WorkOrder_ID__c (before insert)
2 {
3     if(trigger.isAfter && trigger.isInsert)
4     {
5         WorkOrderClass.workOrder(trigger.new);
6     }
7     if(Trigger.IsAfter && Trigger.isUpdate)
8     {
9         CompletionMail.sendEmailMsg(Trigger.New);
10    }
11}

```

Step 3: Create An Apex Class

Now create apex class by giving it name as "AssigningEmail".

The screenshot shows the Salesforce Setup interface with the search bar set to "apex cl". The results list "Apex Classes" under "Custom Code". The "AssigningEmail" class is selected, showing its detail page. The class name is "AssigningEmail" and it has a namespace prefix. The class code is as follows:

```

1 public class AssigningEmail {
2     public static void sendEmails(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 Techniciansv__c};
5         try{
6             for(Assignment__c c : assRec){
7                 Assignment__c assRec = new Assignment__c();
8                 messaging.SingleEmailMessage mail = new messaging.SingleEmailMessage();
9                 List<String> sendTo = new List<String>();
10                String[] Techniciansv__c_email = Techniciansv__c.Email__c;
11                mail.setToAddresses(sendTo);
12                string subject = 'WorkOrder Assignment';
13                mail.setSubject(subject);
14                string body = 'The following WorkOrder has been assigned to you';
15                mail.setHTMLBody(body);
16                myVar.add(mail);
17            }
18        } Messaging.sendEmail(myVar);
19    }
20    catch(exception e){
21        system.debug('Error ----> ' + e.getMessage());
22    }
23}
24
25}

```

Step 4: Create An Apex Trigger

Now create apex trigger by giving it name as "AssignmentTrigger".

The screenshot shows the Salesforce Setup Apex Triggers page. The search bar at the top contains 'apex tr'. The left sidebar has 'Custom Code' expanded, with 'Apex Triggers' selected. The main area displays the 'AssignmentTrigger' details:

- Apex Trigger Detail:**
 - Name: AssignmentTrigger
 - Code Coverage: 0% (0/2)
 - Created By: Jagilalki Indu | 03/08/2024, 7:08 pm
 - Namespace Prefix:
- Apex Trigger Body:**

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

Step 5: Create An Apex Class

Now create apex class by giving it name as "CompletionMail".

The screenshot shows the Salesforce Setup Apex Classes page. The search bar at the top contains 'apex d'. The left sidebar has 'Custom Code' expanded, with 'Apex Classes' selected. The main area displays the 'CompletionMail' details:

- Apex Class Detail:**
 - Name: CompletionMail
 - Namespace Prefix:
 - Created By: Jagilalki Indu | 08/08/2024, 12:25 am
 - Status: Active
 - Code Coverage: 0% (0/14)
 - Last Modified By: Jagilalki Indu | 08/08/2024, 12:26 am
- Class Body:**

```
1 public class CompletionMail {
2     public static void sendEmails(List<WorkOrder_ID__c> workOrderList){
3         List<messaging.SingleEmailMessage> myVar = new List<messaging.SingleEmailMessage>();
4         for(WorkOrder_ID__c con : workOrderList){
5             messaging.SingleEmailMessage mail = new messaging.SingleEmailMessage();
6             List<String> sendTo = new List<String>();
7             sendTo.add(con.getOwner().getUsername());
8             mail.setToAddresses(sendTo);
9             mail.setSubject('Status Updated');
10            mail.setHTMLBody('
11                string body = 'email body';
12                mail.setHTMLBody(body);
13                myVar.add(mail);
14            }
15        };
16        Messaging.sendEmail(myVar);
17    }
18 }
19 }
```

Step 6: Create An Apex Trigger

Now create apex trigger by giving it name as "WorkOrder Trigger".

The screenshot shows the Salesforce Setup Apex Triggers page. A search bar at the top left contains 'apex tri'. The main area displays a table for the 'Apex Trigger Detail' of 'WorkOrderTrigger'. The trigger's code is listed below:

```

1 trigger WorkOrderTrigger on WorkOrder__c (before insert)
2 {
3     if(trigger.isAfter && trigger.isInsert)
4     {
5         WorkOrderClass.workOrder(trigger.new);
6     }
7     if(Trigger.isAfter && Trigger.isUpdate)
8     {
9         CompletionMail.sendEmailMsg(Trigger.New);
10    }
11}

```

Step 7: Create An Asynchronous Apex Class

Now create apex class by giving it name as "RecordDeletion".

The screenshot shows the Salesforce Setup Apex Classes page. A search bar at the top left contains 'apex cl'. The main area displays a table for the 'Apex Class Detail' of 'RecordDeletions'. The class's code is listed below:

```

1 public class RecordDeletions implements Database.Batchable<SObject>
2 {
3     public Database.QueryLocator start(Database.BatchableContext bc) {
4         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';
5         return database.GetQueryLocator(query);
6     }
7     public void execute(Database.BatchableContext bc, List<Assignment__c> query) {
8         if(Query.isEmpty()){
9             Delete query;
10        }
11    }
12    public void finish(Database.BatchableContext bc) {
13    }
14 }

```

Step 8: Create An Apex Schedule Class

Now create apex class by giving it name as "ScheduleClass".

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

The screenshot shows the Salesforce Setup interface. In the top navigation bar, 'Setup' is selected. A search bar at the top right contains the query 'apex class'. Below the search bar, there's a dropdown menu with 'Custom Code' and 'Apex Classes' selected. The main content area displays a list of Apex classes. A green box highlights the 'Percent of Apex Used: 0.19%' message, which states that 11,223 characters are used out of a limit of 6,000,000. At the bottom of the list, the 'Schedule Apex' tab is highlighted with a red box.

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

1. Job Name : DeleteAssignmentSchedule
2. Apex Class : ScheduleClass (from clicking on lookup icon)
3. Frequency : Monthly
4. Preferred Start Time : Select any time

The screenshot shows the 'Schedule Apex' configuration page. It has fields for 'Job Name' (set to 'DeleteAssignmentSchedule') and 'Apex Class' (set to 'ScheduleClass'). Under 'Schedule Apex Execution', 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 '4:00 pm'. At the bottom, there are 'Save' and 'Cancel' buttons, with 'Save' highlighted with a red box.

4. Click Save.

Step 9: Create A Schedule Class

Now create apex class by giving it name as "ScheduleClass".

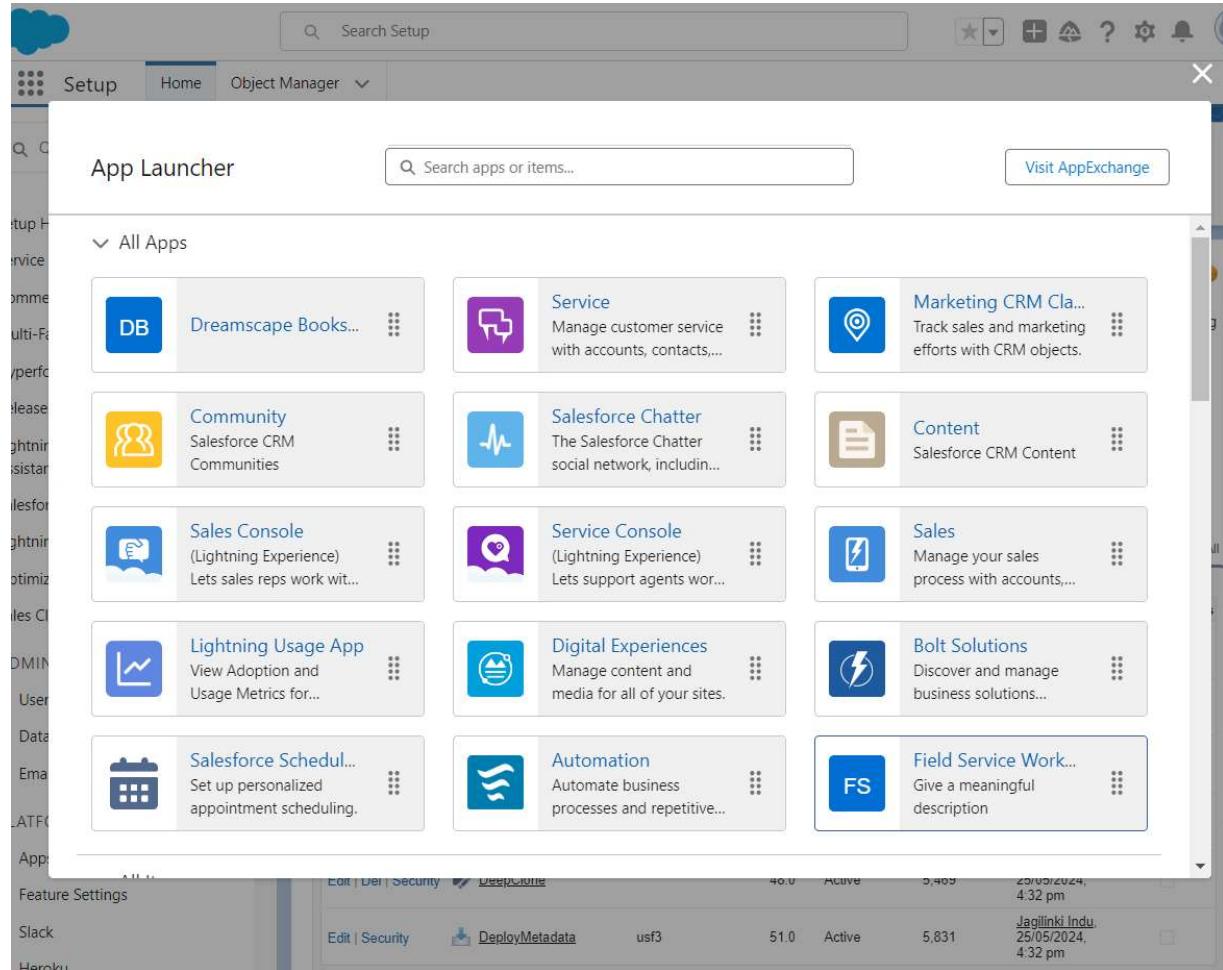
The screenshot shows the Salesforce Setup Apex Classes page. The URL is [https://workdrive.zohoexternal.com/writer/open/lbayc1372339e3a7e48e4b3ae07fc73f8f47c?authId=%7B"linkId"%3A"5k2wApayC00-LYmlU"%7D](#). The page title is "Apex Classes". The main content displays the "Apex Class Detail" for "ScheduleClass". The class is active and was created by Jagilink Indu on 03/08/2024 at 7:15 pm. The code coverage is 0% (0/3). The class body contains the following Apex code:

```
1 global class ScheduleClass implements Schedulable {
2     global void execute(SchedulableContext SC) {
3         RecordDeletions delrec = new RecordDeletions();
4         database.executeBatch(delrec, 200);
5     }
6 }
```

Task 9: Reports & Dashboards

Step 1: Report

1. Go to the app --> Click and open Field Service WorkOrder



2. click on the reports tab
3. Click New Report.
3. Select report type from category or from report type panel or from search panel --> click on start report.
4. Customize your report
 - Add fields from left pane as shown below
 - Grouped by workorder ID

The screenshot shows the 'Employee Management' software interface. A report titled 'New Employees Report' is displayed for the 'Employees' section. The report preview shows two rows of data with columns: Employee, Employee Name, Reports to, Login Time, Logout Time, Mode of Work, and LinkedIn Profile. On the left, there are sections for 'Groups' (with 'GROUP ROWS' selected) and 'Columns' (with 'Add column...' highlighted). The top right features standard report controls: Save & Run, Save, Close, and Run.

5. Save or run it.

The screenshot shows the 'Field Service WorkOrder' software interface. The 'Reports' section is active, with the 'Recent' tab selected. It lists several reports: 'WorkOrders Status Reports' (Private Reports, Jagilink Indu, 8/8/2024, 12:54 am), 'Technician and Assignment Details Report' (Private Reports, Jagilink Indu, 4/8/2024, 6:35 pm), 'New Technician.csv Report' (Public Reports, Jagilink Indu, 7/8/2024, 3:33 pm), 'New Assignments with WorkOrder ID Report' (Private Reports, Jagilink Indu, 4/8/2024, 1:57 am), and 'Sample Flow Report: Screen Flows' (Public Reports, Automated Process, 25/5/2024, 4:32 pm). The left sidebar provides navigation for Reports, Recent, Created by Me, Private Reports, Public Reports, All Reports, Folders, All Folders, Created by Me, Shared with Me, Favorites, and All Favorites.

Step 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".

Report Name	Description	Folder	Created By	Created On	Subscribed
WorkOrders Status Reports		Private Reports	Jagilink Indu	8/8/2024, 12:54 am	
Technician and Assignment Details Report		Private Reports	Jagilink Indu	4/8/2024, 6:35 pm	
New Technician.csv Report		Public Reports	Jagilink Indu	7/8/2024, 3:33 pm	
New Assignments with WorkOrder ID Report		Private Reports	Jagilink Indu	4/8/2024, 1:57 am	
Sample Flow Report: Screen Flows	Which flows run, what's the status of each interview, and how long do users take to complete the screens?	Public Reports	Automated Process	25/5/2024, 4:32 pm	

Step 3: Dashboards

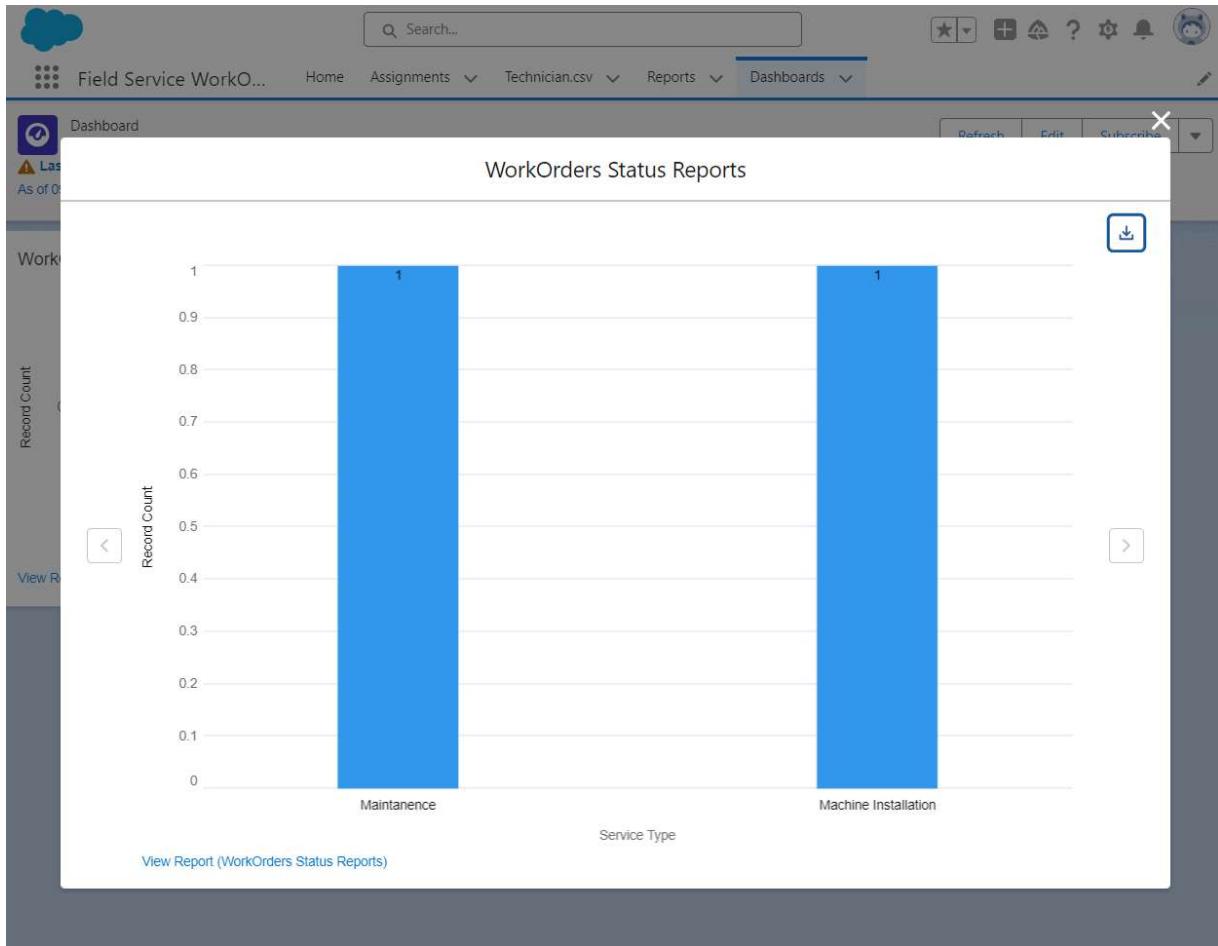
1. Go to the app --> click on the Dashboards tabs.
2. Give a Name (Dashboard1) and click on Create.
3. Select add component.
4. Select a Report which we have created in the previous activities and click on select.
5. Click Add then click on Save and then click on Done.

And Repeat the steps and create Dashboard2

Dashboard Name	Description	Folder	Created By	Created On	Subscribed
Dashboard 2		Private Dashboards	Jagilink Indu	7/8/2024, 2:52 pm	
Dashboard 1		Private Dashboards	Jagilink Indu	4/8/2024, 6:42 pm	

Step 3: Create Dashboards

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



THANKYOU