

Project Title:College Management Application

Day 1 : Create Your Salesforce Developer Org To Get Started

Topic: Creating Developer Account

Milestone / Activities: Created

Detailed Description: Firstly for Creating a Developer org in salesforce I have followed these steps

Go to developers.salesforce.com/

Click on sign up.

On the sign up form, enter the following details :

First name & Last name

Email

Role : Developer

Company : College Name

County : India

Postal Code : pin code

Username : should be a combination of your name and company

This need not be an actual email id, you can give anything in the format :

username@organization.com

Project Title:College Management Application

Topic: Account Activation

Milestone/Activities: Activated

Detailed Description: For this 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, as allocating a developer org will take time

Topic: Login To Your Salesforce Account

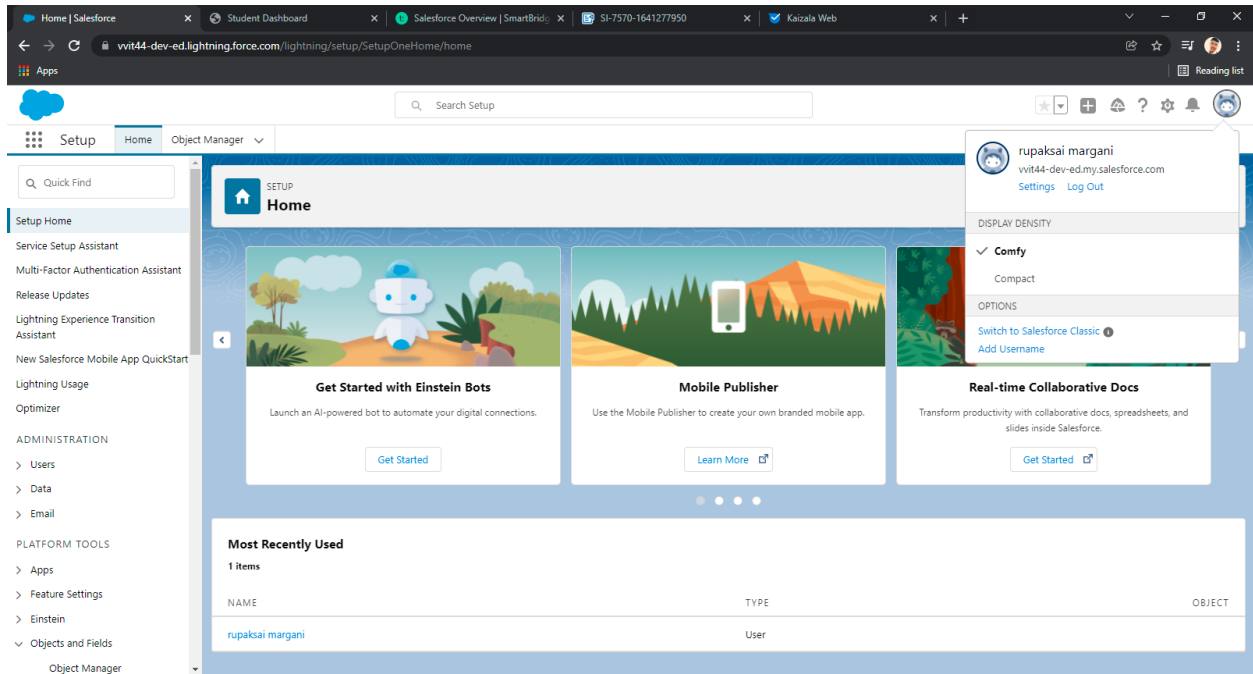
Milestone/Activities: Logged

Detailed Description: To do this

- 1.Go to salesforce.com and click on login.
- 2.Enter the username and password that you just created.
- 3.After login this is the home page which you will see.

Upload the Screenshot the Milestone / Activities :

Project Title: College Management Application



Day-2: Custom Object Creation

Topic: Create The Custom Objects For The Application

Milestone/Activites: Created

Description: The Following Custom Objects are Created

1. College
2. Application Form.
3. Students.
4. Subjects.

Screen shot:

Project Title:College Management Application

Student Dashboard

Object Manager | Salesforce

←

→

↻

vvit44-dev-ed.lightning.force.com/lightning/setup/ObjectManager/home

🔍

Search Setup

🌐

+

🔔

?

⚙️

🔔

👤

Reading list

🌐

Setup

Home

Object Manager

▼

SETUP

Object Manager

1 Items, Sorted by Label

🔍

coll

Schema Builder

Create

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
College	College_c	Custom Object		1/6/2022	✓

Student Dashboard

Object Manager | Salesforce

←

→

↻

vvit44-dev-ed.lightning.force.com/lightning/setup/ObjectManager/home

🔍

Search Setup

🌐

+

🔔

?

⚙️

🔔

👤

Reading list

🌐

Setup

Home

Object Manager

▼

SETUP

Object Manager

1 Items, Sorted by Label

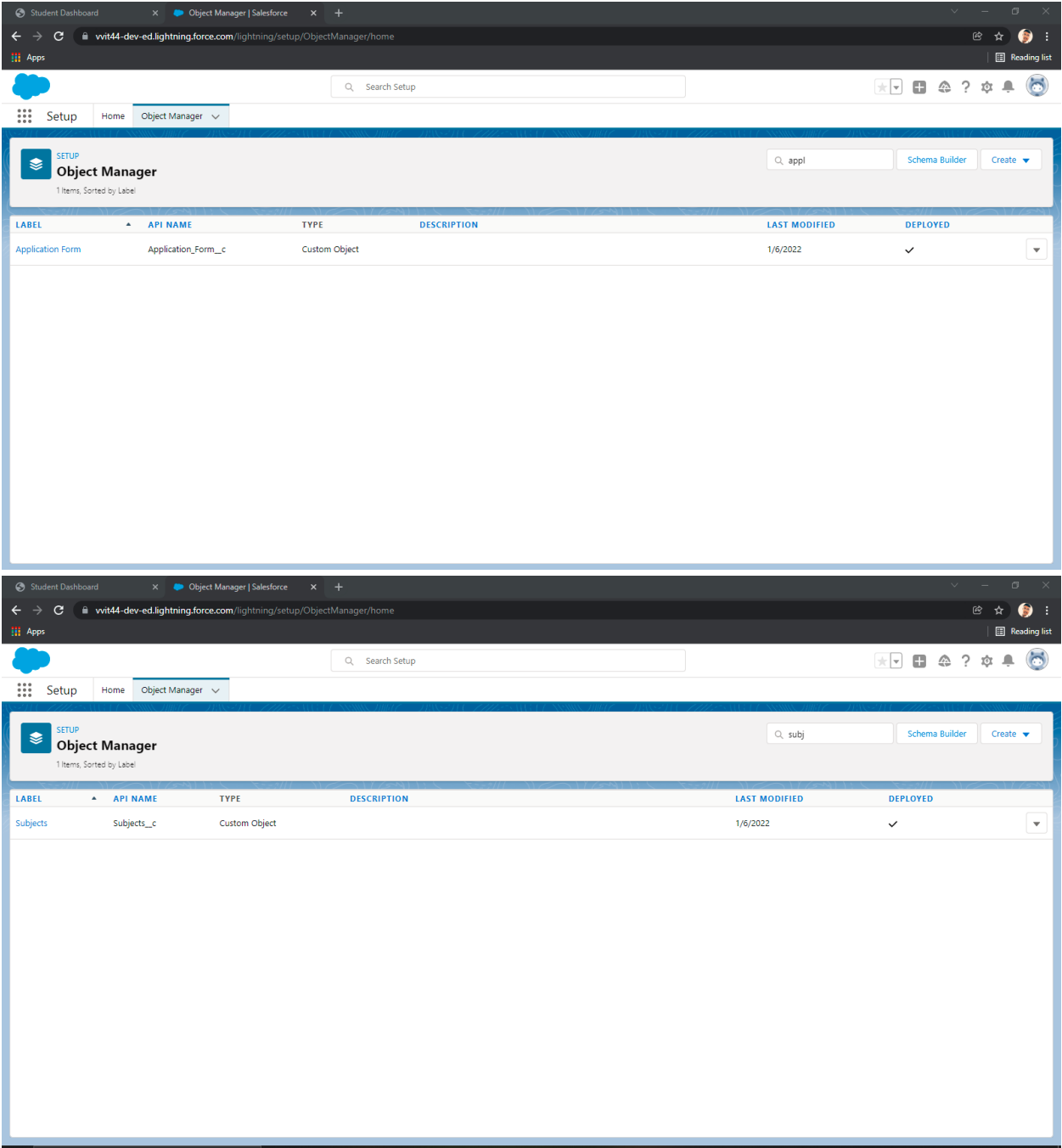
🔍

stude

Schema Builder

Create

Project Title:College Management Application



Topic: Create Fields On College Object

Milestone/Activites: Created

Description: The Following Fields are Created On College Object

Field Name	Data Type	Required	Values
------------	-----------	----------	--------

Project Title:College Management Application

Record Info	Text		
College Fees	Currency(7,2)	Yes	
Hostel Fees	Currency(6,2)	Yes	
College Name	Picklist(Refer Business Logic In Milestones)	Yes	
Email	Picklist		blr@mit.co.in hyd@mit.co.in mum@mit.co.in maa@mit.co.in ccu@mit.co.in del@mit.co.in
Capacity Of Students	Picklist		500-1000, 1000-2500, 2500-6000, 6000-10000

Screen Shot:

The screenshot shows the Salesforce Setup interface for the 'College' object. The 'Fields & Relationships' section is active, displaying a list of 10 fields. The fields are sorted by Field Label. The table below represents the data shown in the screenshot:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Capacity Of Students	Capacity_Of_Students__c	Picklist		
College Fees	College_Fees__c	Currency(7, 2)		
College Name	College_Name__c	Picklist		
College Name	Name	Text(80)		✓
Created By	CreatedById	Lookup(User)		
Email	Email__c	Picklist		
Hostel Fees	Hostel_Fees__c	Currency(6, 2)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Record Info	Record_Info__c	Text(40)		

Topic: Create Fields On Application Form Object

Milestone/Activites: Created

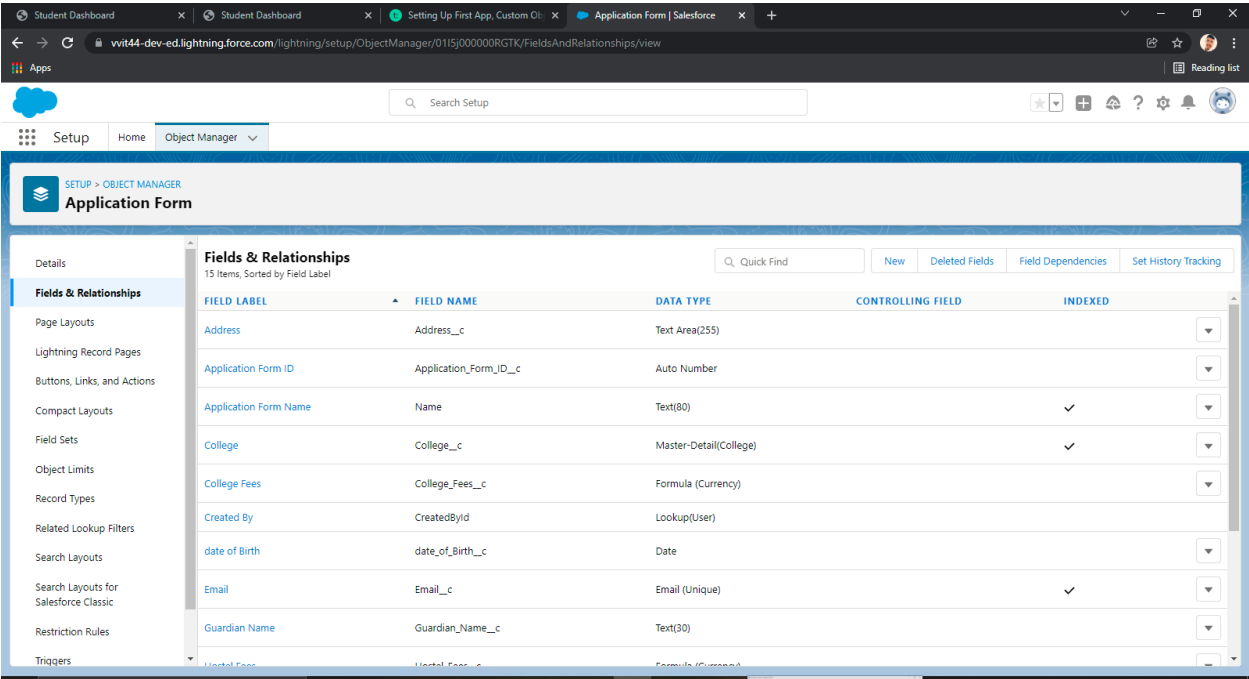
Description: The Following Fileds are Created On Application Form Object

Field Name	Data Type	Required	Values
------------	-----------	----------	--------

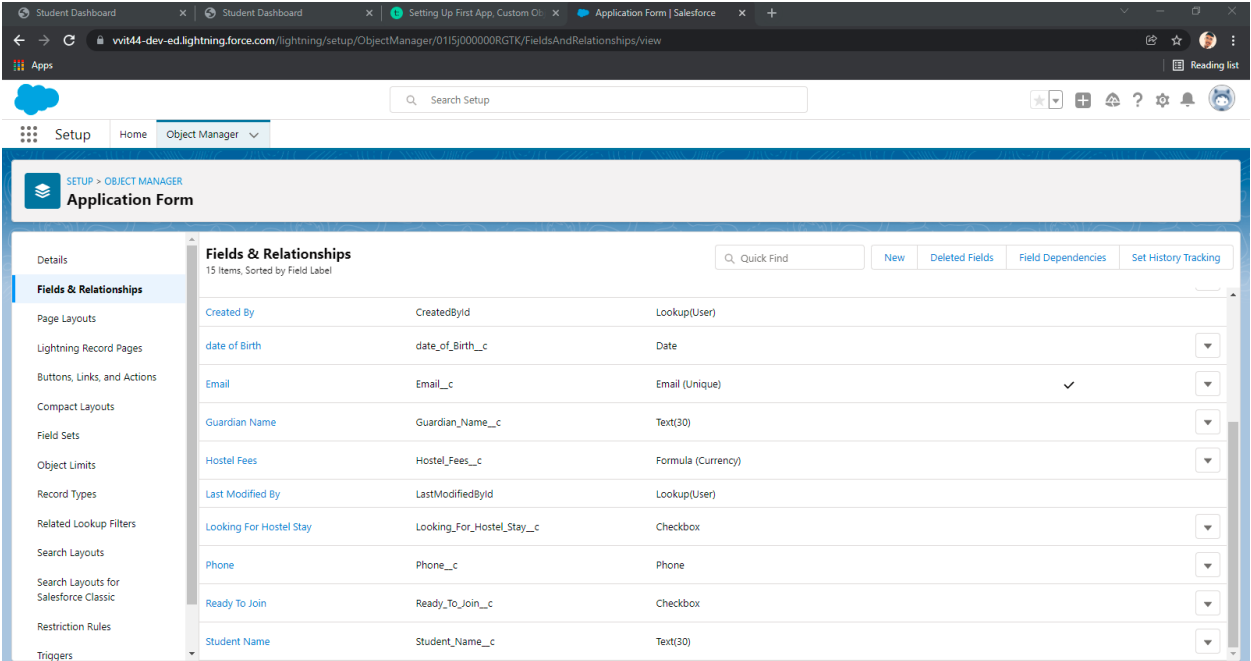
Project Title:College Management Application

Application Form ID	Autonumber		F-{000000} Starting Number=1
Address	Text(255)	Yes	
College	Master-Detail(College)	Yes	
College Fees	Formula(Currency)		
Hostel Fees	Formula(Currency)		
date of Birth	Date	Yes	
Email	Email(Unique)	Yes	
Guardian Name	Text(30)	Yes	
Looking For Hostel Stay	Checkbox(default=Uncheck)		
Ready To Join	Checkbox(default=Uncheck)		
Student Name	Text(30)	Yes	
Phone	Phone	Yes	

Screen Shot:



Project Title:College Management Application



Topic: Create Fields On Student Object

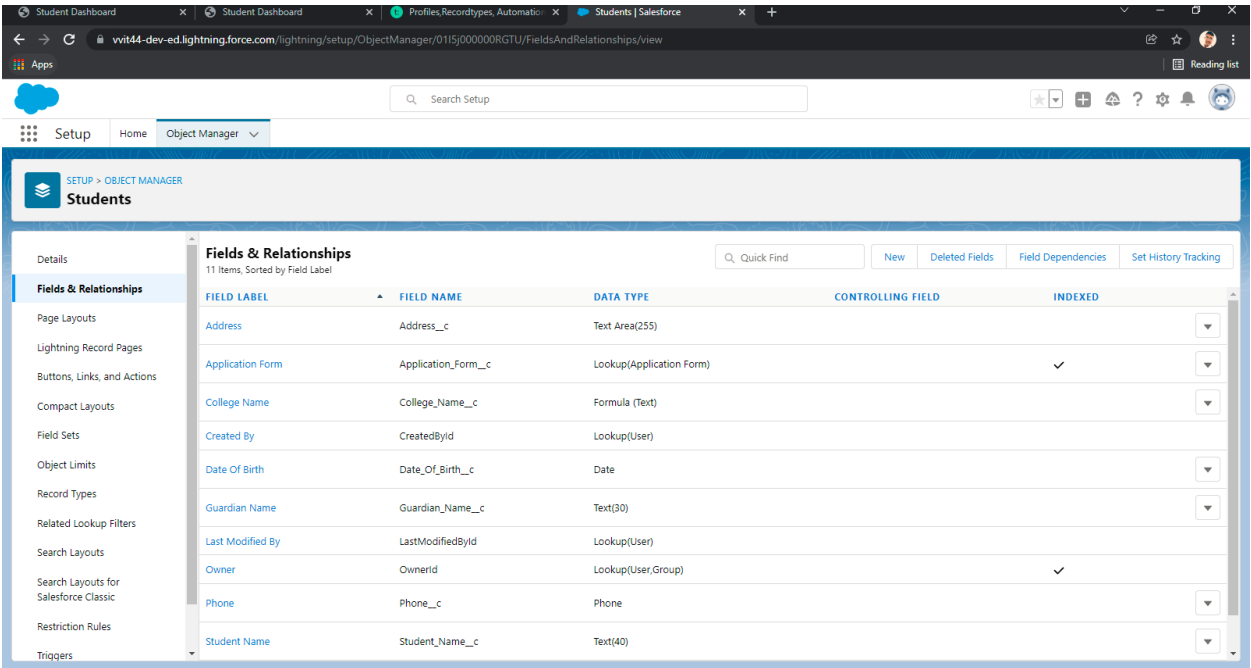
Milestone/Activites: Created

Description: The Following Fields are Created On Student Object

Field Name	Data Type	Required	Values
Student Name	Text		
Address	Text(255)	Yes	
Application Form	Lookup(Application Form)		
College Name	Formula(Text)		
Date Of Birth	Date	Yes	
Guardian Name	Text(30)	Yes	
Phone	Phone	Yes	

Screen Shot:

Project Title:College Management Application



Topic: Create Fields On Subject Object

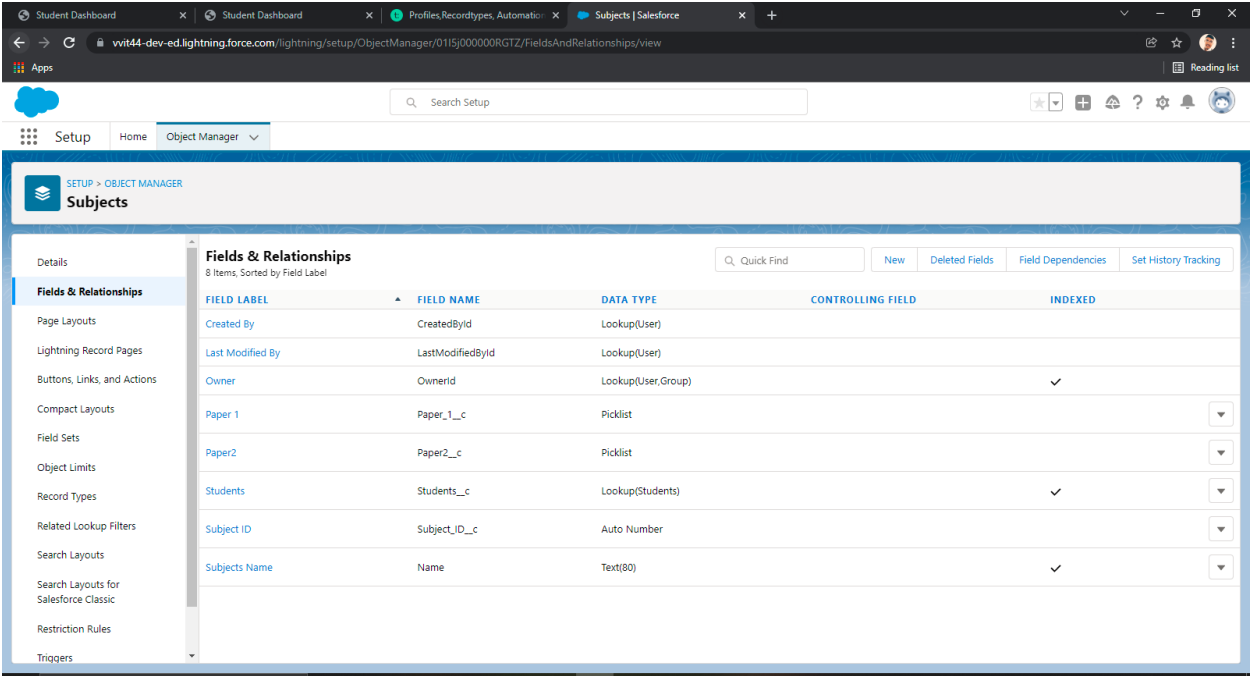
Milestone/Activities: Created

Description:

Field Name	Data Type	Required	Values
Subject ID	AutoNumber		S-{000000} Starting Number=1
Paper 1	Picklist(Refer Business Logic Milestone)		
Paper2	Picklist(Refer Business logic Milestone)		
Student	Lookup(Student)		

Screen Shot:

Project Title:College Management Application



Day-3: Adding Business Logic To Application

Topic: Creating Global Picklist Value Sets

Milestone/Activites: Created

Description: Created the following global picklist value sets for the application.

a)College

Picklist Value Name	Values
Colllege	MIT-HYD MIT-BLR MIT-MUM MIT-MAA MIT-DEL MIT-CCU

b)Paper1

Picklist Value Name	Values
Paper 1	APEX JAVA C

Project Title:College Management Application

	C++
--	-----

c)Paper2

Picklist Value Name	Values
Paper2	MATHEMATICS ENGLISH STATISTICS

Screen Shot:

Student Dashboard

Profiles, Record Types, Automation

Picklist Value Sets | Salesforce

vvit44-dev-ed.lightning.force.com/lightning/setup/Picklists/page?address=%2F0Nt5j0000000omkc

Apps

Reading list

Setup

Home

Object Manager

Quick Find

Search Setup

SETUP

Picklist Value Sets

Global Value Set

Back to List

Values (5)

Inactive Values (0)

Fields Where Used (0)

Global Value Set Detail

Edit

Delete

Information

Label

College

Name

College

Description

Picklist Values Used

Active and inactive picklist values

6 (1,000 max)

Edit

Delete

Values

New

Reorder

Replace

Printable View

Chart Colors

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	MIT-HYD	MIT-HYD	<input type="checkbox"/>	Assigned dynamically	rueakhsal.margani 1/8/2022, 12:37 AM
Edit Del Deactivate	MIT-BLR	MIT-BLR	<input type="checkbox"/>	Assigned dynamically	rueakhsal.margani 1/8/2022, 12:37 AM
Edit Del Deactivate	MIT-MUM	MIT-MUM	<input type="checkbox"/>	Assigned dynamically	rueakhsal.margani 1/8/2022, 12:37 AM
Edit Del Deactivate	MIT-MAA	MIT-MAA	<input type="checkbox"/>	Assigned dynamically	rueakhsal.margani 1/8/2022, 12:37 AM
Edit Del Deactivate	MIT-DEL	MIT-DEL	<input type="checkbox"/>	Assigned dynamically	rueakhsal.margani 1/8/2022, 12:37 AM
Edit Del Deactivate	MIT-CCU	MIT-CCU	<input type="checkbox"/>	Assigned dynamically	rueakhsal.margani 1/8/2022, 12:37 AM

Project Title: College Management Application

The image displays two screenshots of the Salesforce Setup interface, specifically the 'Picklist Value Sets' configuration page. The top screenshot shows a 'Global Value Set' named 'Paper_1' with four active picklist values: APEX, JAVA, C, and C++. The bottom screenshot shows the same 'Global Value Set' but with a different name, 'Paper2', and three active picklist values: MATHEMATICS, ENGLISH, and STATISTICS. Both screenshots show the 'Values' table with columns for Action, Values, API Name, Default, Chart Colors, and Modified By. The 'Inactive Values' section is empty in both.

Top Screenshot: Global Value Set Detail

Global Value Set Detail

Information

Label: Paper_1

Name: Paper_1

Description:

Picklist Values Used

Active and inactive picklist values: 4 (1,000 max)

Values

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	APEX	APEX	<input type="checkbox"/>	Assigned dynamically	ruakhsai margani 1/8/2022, 12:39 AM
Edit Del Deactivate	JAVA	JAVA	<input type="checkbox"/>	Assigned dynamically	ruakhsai margani 1/8/2022, 12:39 AM
Edit Del Deactivate	C	C	<input type="checkbox"/>	Assigned dynamically	ruakhsai margani 1/8/2022, 12:39 AM
Edit Del Deactivate	C++	C++	<input type="checkbox"/>	Assigned dynamically	ruakhsai margani 1/8/2022, 12:39 AM

Inactive Values

No Inactive Values values defined.

Bottom Screenshot: Global Value Set Detail

Global Value Set Detail

Information

Label: Paper2

Name: Paper2

Description:

Picklist Values Used

Active and inactive picklist values: 3 (1,000 max)

Values

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	MATHEMATICS	MATHEMATICS	<input type="checkbox"/>	Assigned dynamically	ruakhsai margani 1/8/2022, 12:40 AM
Edit Del Deactivate	ENGLISH	ENGLISH	<input type="checkbox"/>	Assigned dynamically	ruakhsai margani 1/8/2022, 12:40 AM
Edit Del Deactivate	STATISTICS	STATISTICS	<input type="checkbox"/>	Assigned dynamically	ruakhsai margani 1/8/2022, 12:40 AM

Inactive Values

No Inactive Values values defined.

Topic: Creating Field Dependencies

Milestone/Activites: Created

Description: 1. Created field dependency between college Name and Email, where the controlling field is college Name and dependent field is Email. Selected the email ids according to the college names.

Project Title: College Management Application

2. Created field dependency between college Name and capacity of students, where the controlling field is college Name and dependent field is Capacity of Students. Selected the values according to my wish.

Screen Shot:

The first screenshot shows the 'Edit Field Dependency' configuration for the 'Email' field. The controlling field is 'College Name' and the dependent field is 'Email'. The interface includes instructions, a legend for 'Excluded Value' (grey) and 'Included Value' (yellow), and a table of values.

College Name	MIT-BLR	MIT-HYD	MIT-MUM	MIT-MAA	MIT-CCU	MIT-DEL
Email	blr@mit.co.in	hyd@mit.co.in	mum@mit.co.in	maa@mit.co.in	ccu@mit.co.in	del@mit.co.in

The second screenshot shows the 'Edit Field Dependency' configuration for the 'Capacity Of Students' field. The controlling field is 'College Name' and the dependent field is 'Capacity Of Students'. The interface includes instructions, a legend for 'Excluded Value' (grey) and 'Included Value' (yellow), and a table of values.

College Name	MIT-BLR	MIT-HYD	MIT-MUM	MIT-MAA	MIT-CCU	MIT-DEL
Capacity Of Students	500-1000	500-1000	500-1000	500-1000	500-1000	500-1000
	1000-2500	1000-2500	1000-2500	1000-2500	1000-2500	1000-2500
	2500-6000	2500-6000	2500-6000	2500-6000	2500-6000	2500-6000
	6000-10000	6000-10000	6000-10000	6000-10000	6000-10000	6000-10000

Topic: Creating Validation Rules

Project Title:College Management Application

Milestone/Activites: Created

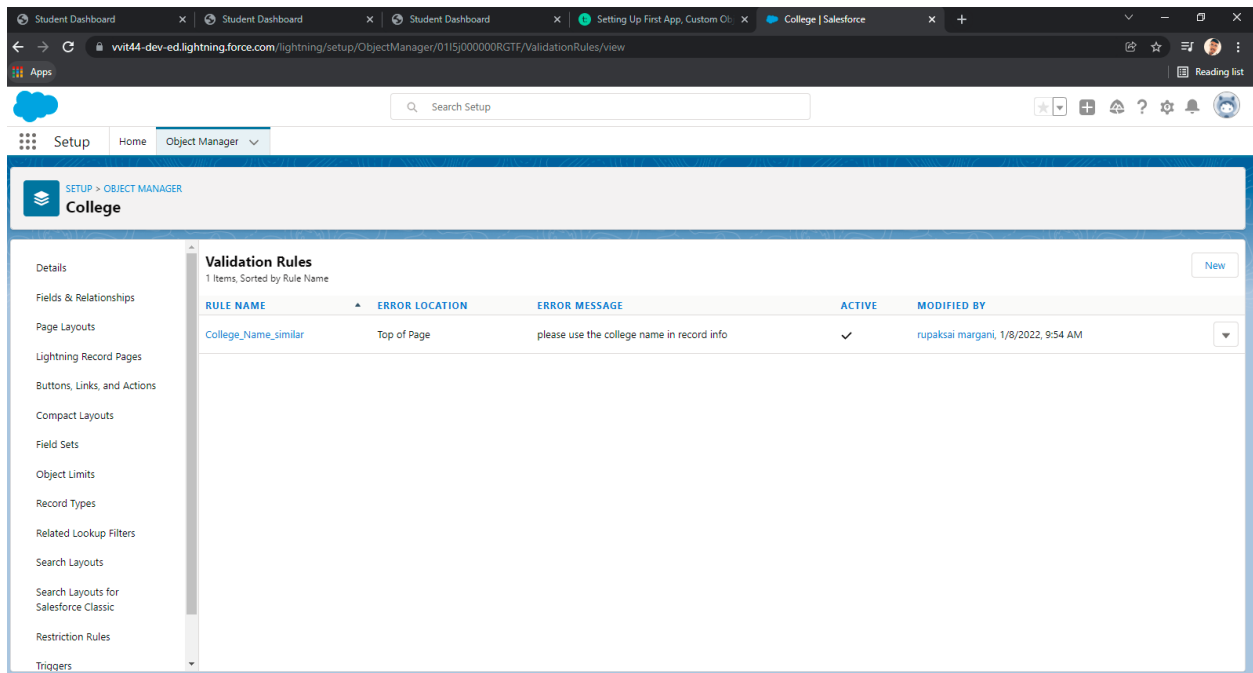
Description: 1. Created a validation rule on the college object such that the college name and record info should have the same name.

TEXT(College_Name__c) <> Name.

2. Created a validation rule on the application form object to stop any modification on the application form once a student record is created.

```
AND( Ready_To_Join__c == true,  
OR(  
ISCHANGED( Address__c ) ,  
ISCHANGED( College__c ) ,  
ISCHANGED( Date_Of_Birth__c ) ,  
ISCHANGED( Email__c ) ,  
ISCHANGED( Guardian_Name__c ) ,  
ISCHANGED( Phone__c )  
)  
)
```

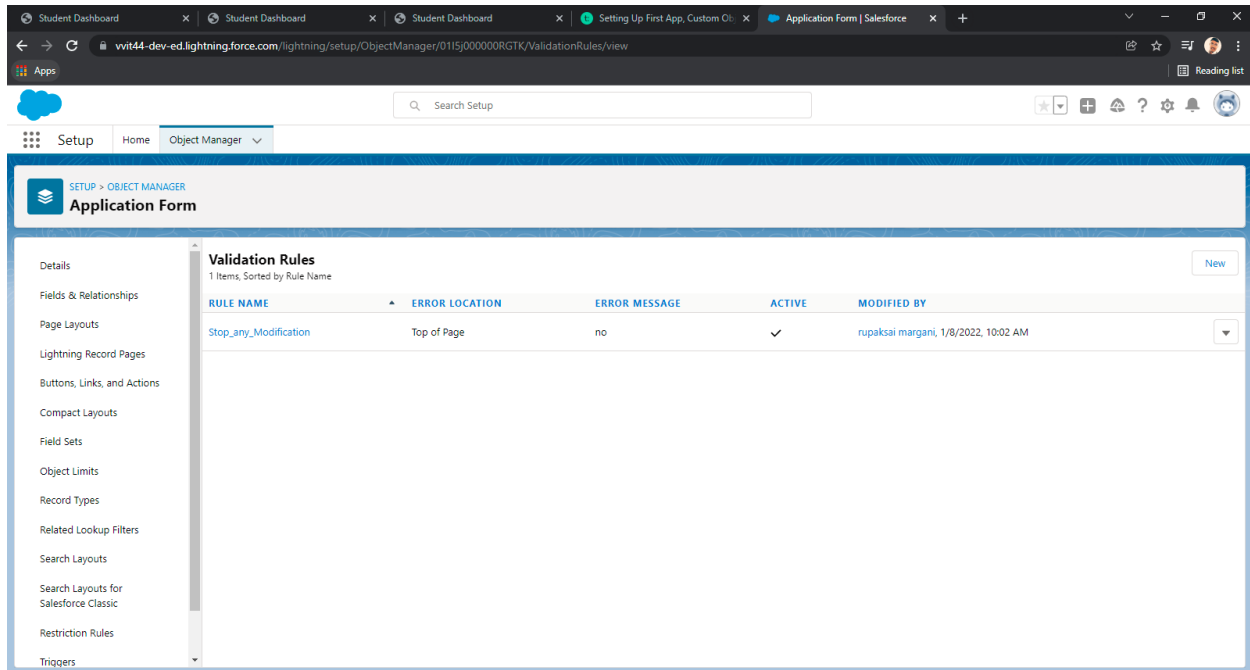
Screen Shot:



The screenshot shows the Salesforce Setup interface. The left sidebar contains a navigation menu with options like 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, Search Layouts for Salesforce Classic, Restriction Rules, and Triggers. The main content area is titled 'Validation Rules' and shows a table with one rule named 'College_Name_similar'. The table has columns for Rule Name, Error Location, Error Message, Active status, and Modified By. The rule is active and was modified by 'rupaksai margani' on 1/8/2022 at 9:54 AM.

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
College_Name_similar	Top of Page	please use the college name in record info	✓	rupaksai margani, 1/8/2022, 9:54 AM

Project Title: College Management Application



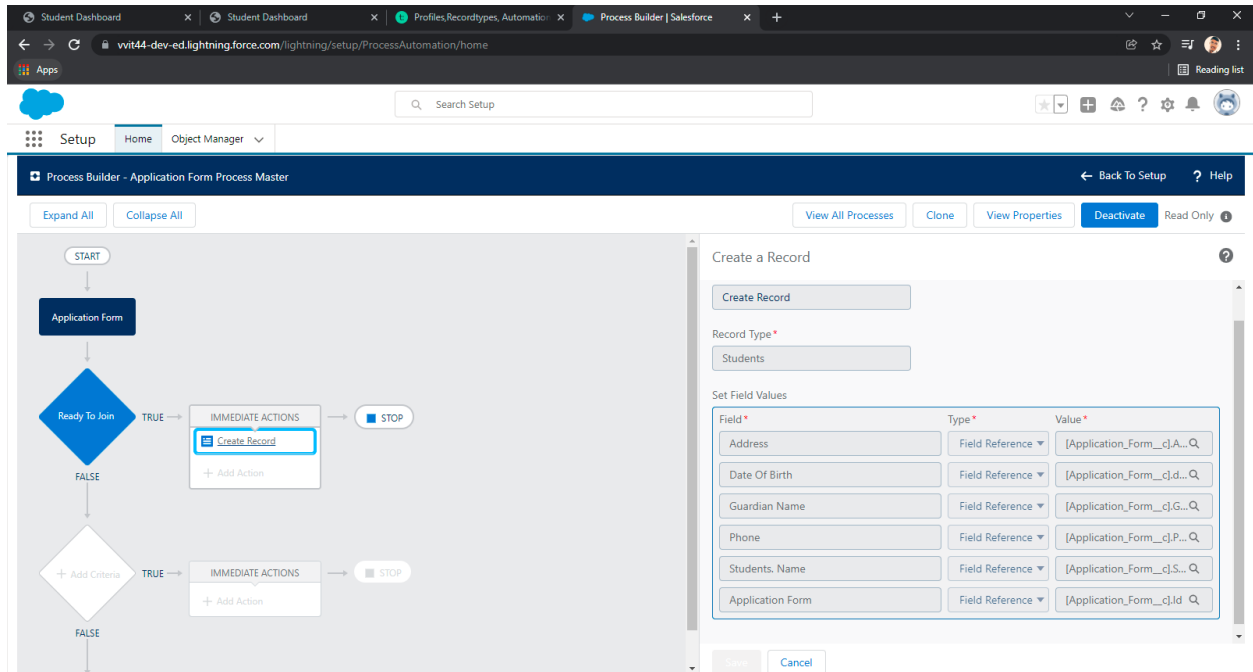
Topic: Process Automation

Milestone/Activites: Created

- Description:**
1. Create an automation process such that when the "ready to join" field is checked on the application form object we need to create the student record automatically with the information specified in the application form record.
 2. Go to Setup -> select "Process Builder" from quick find. Create a Process Builder on the "Application Form" object with a condition as "When a record change". And select "When a record is created or edited".
 - a. In the diamond shape box(called nodes), select the criteria which trigger the Process builder to fire. In our example, it is "When Ready to Join field is checked."
 - b. Once the node is setup, click on the adjacent box called "Immediate action". And select create a record on the student object.

Screen Shot:

Project Title: College Management Application



Topic: Create The Student Record Using Flow

Milestone/Activites: Created

- Description:
1. First deactivate the process builder which we created earlier.
 2. Now search for flows and select new flow ->record triggered flow
 3. For object select application form, in configure trigger select when the record is updated for entry criteria select as ready to join equals to true.
 4. Now create a variable named student in the resource section.
 5. Now add the assignment as follows

Screen Shot:

Project Title: College Management Application

The screenshot displays the Salesforce Flow Builder interface. The top section shows the 'Edit Assignment' dialog for the 'Assign student record data' assignment. The dialog lists several variables and their values, all set to 'Equals'.

Variable	Operator	Value
Student > Address	Equals	Student > Address
Student > Application Form	Equals	Student > Application Form
Student > College Name	Equals	Student > College Name
Student > Date Of Birth	Equals	Student > Date Of Birth
Student > Guardian Name	Equals	Student > Guardian Name
Student > Phone	Equals	Student > Phone
Student > Student Name	Equals	Student > Student Name

The bottom section shows the flow canvas with the following steps:

- Record-Triggered Flow Start
- Run Immediately
- Assign student record data (Assignment)
- create student record (Create Records)
- End

Day-4: Batch Apex

Topic: Create A Batch Apex For Application Form

Milestone/Activites: Created

Description: 1. From the developer console created a new apex class and entered the following

Project Title:College Management Application

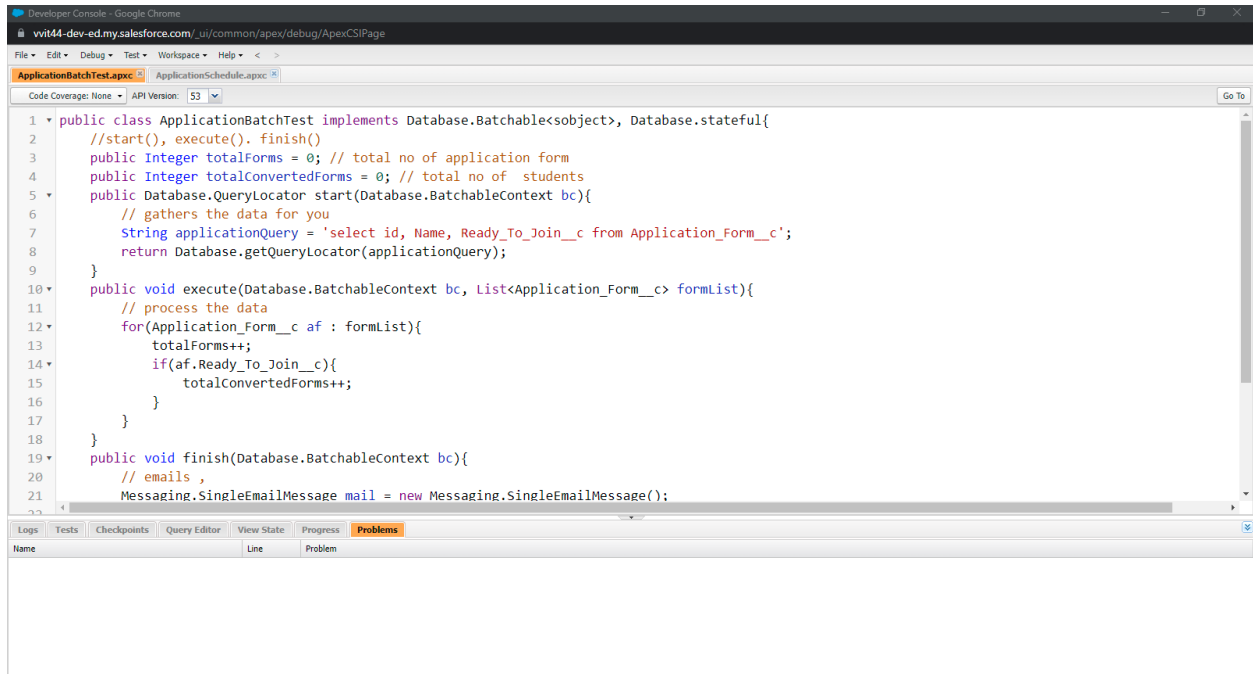
code.

```
Public class ApplicationBatchTest implements Database.Batchable<sObject>{

    //start(), execute(). finish()
    public Integer totalForms = 0; // total no of application form
    public Integer totalConvertedForms = 0; // total no of students
    public Database.QueryLocator start(Database.BatchableContext bc){
        // gathers the data for you
        String applicationQuery = 'select id, Name, Ready_To_Join__c from ApplicationForm__c';
        return Database.getQueryLocator(applicationQuery);
    }
    public void execute(Database.BatchableContext bc, List<ApplicationForm__c> formList){
        // process the data
        for(ApplicationForm__c af : formList){
            totalForms++;
            if(af.Ready_To_Join__c){
                totalConvertedForms++;
            }
        }
    }
    public void finish(Database.BatchableContext bc){
        // emails ,
        Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
        // address, subject, content( data to sent to admins)
        mail.setSubject(' Application form and student record data as of today ');
        mail.setPlainTextBody(' Total no of application form records are : '+totalForms+ ' out of
which no of students as per today : '+totalConvertedForms);
        String[] emailAddress = new String[]{'your email address'};
        mail.setToAddresses(emailAddress);
        Messaging.sendEmail(new Messaging.SingleEmailMessage[]{ mail } );
    }
}
```

ScreenShot:

Project Title: College Management Application



Topic: Create A Scheduler Class

Milestone/Activities: Created

Description: 1. From the developer console created a new apex class and entered the following code.

```
public class applicationschedule implements Schedulable{
```

```
    public void execute(SchedulableContext sc){
```

```
        ApplicationBatchTest abt = new ApplicationBatchTest();
```

```
        Database.executeBatch(abt, 400); // 200 to 2000
```

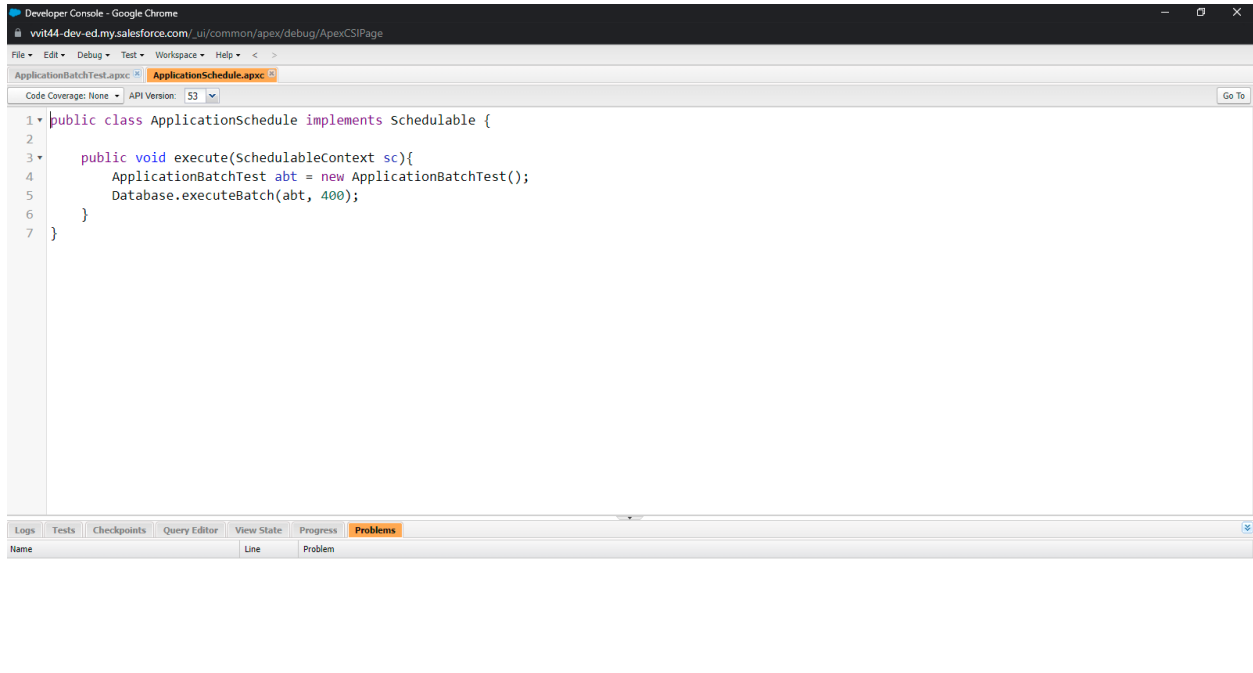
```
    }
```

```
}
```

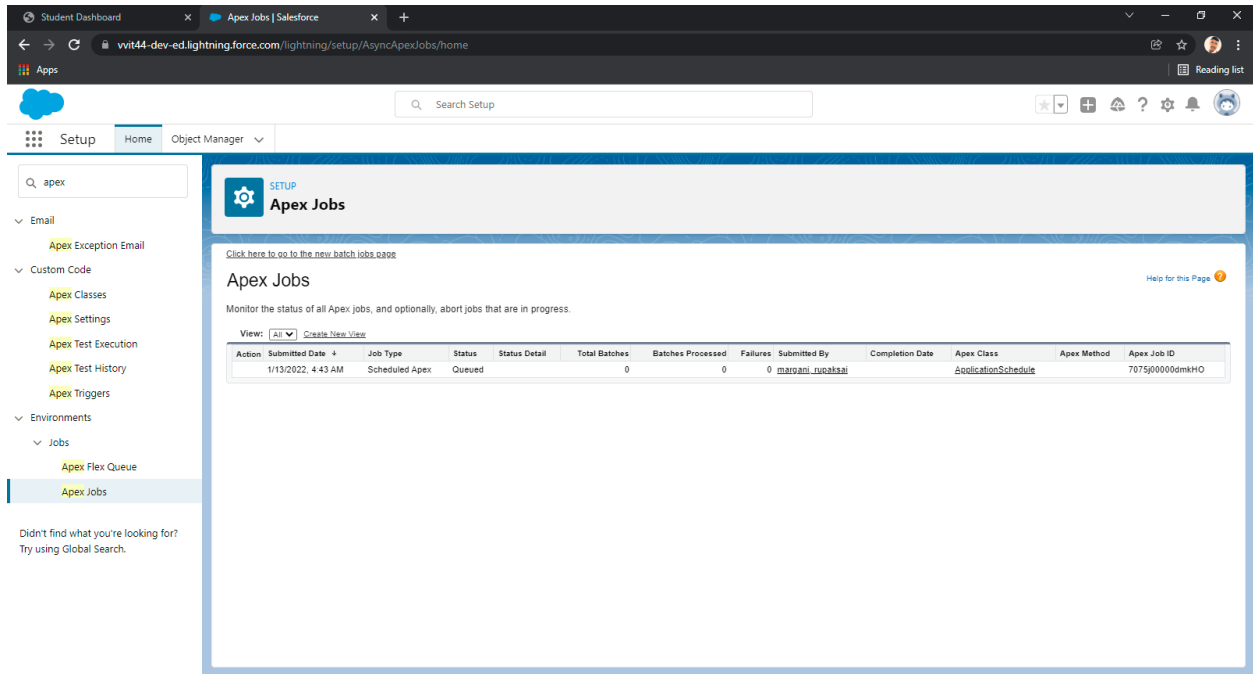
Project Title:College Management Application

From setup searched for apex class and clicked on schedule jobs and filled the details as per requirements.

ScreenShot:



Project Title: College Management Application



Day-5: LIGHTNING WEB COMPONENTS

Topic: Create College DataTable Component(APEX CLASS)

Milestone/Activites: Created

Description: Here we are going to create the college data table, for this we need to create an apex class, from which we are going to retrieve the data and Html, javascript files for UI .

Created an Apex Class With The Required Functionality

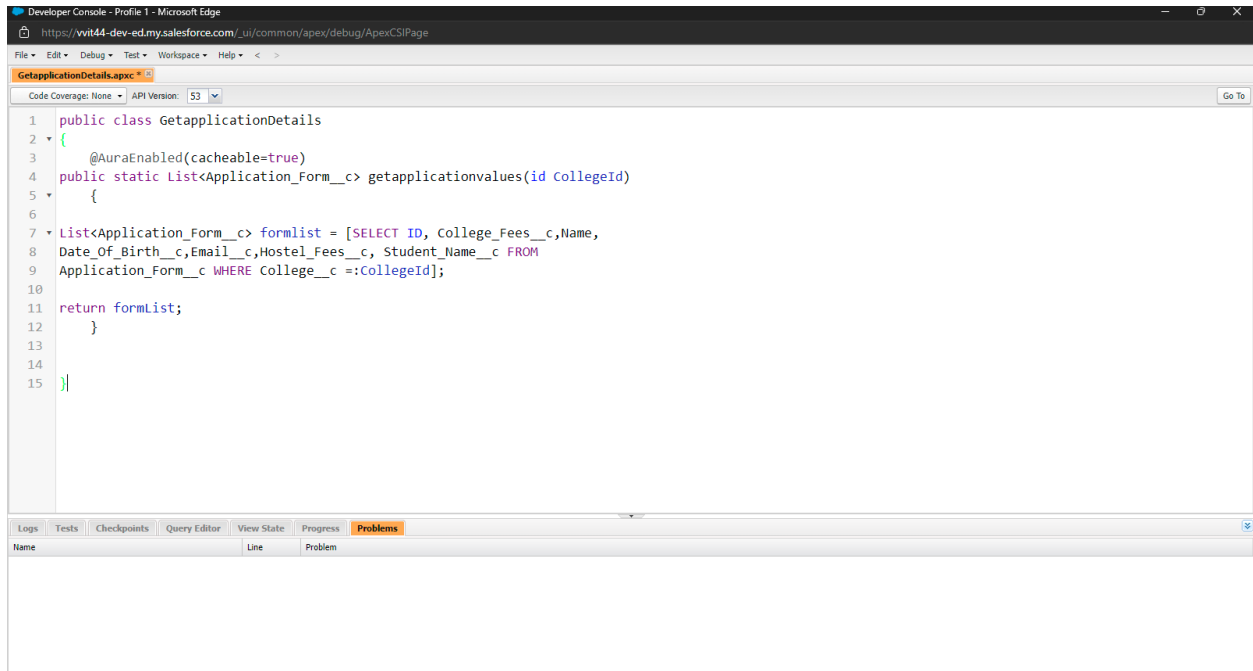
```
public class GetapplicationDetails
{
    @AuraEnabled(cacheable=true)
    public static List<ApplicationForm__c> getapplicationvalues(id Collegeld)
    {
        List<ApplicationForm__c> formlist = [SELECT ID, College_Fees__c,Name,
        Date_Of_Birth__c,Email__c,Hostel_Fees__c, Student_Name__c FROM
        ApplicationForm__c WHERE College__c =:Collegeld];

        return formList;
    }
}
```

Project Title:College Management Application

```
}  
  
}
```

Screenshot:



Topic: Create College DataTable Component (HTML FILE)

Milestone/Activities: Created

Description: Created the component with the following files.

HTML:

<template>

<h1> College and Application form list table </h1>

<template if:true={recordList}>

<lightning-datatable

key-field="id"

data={recordList}

show-row-number-column

hide-checkbox-column

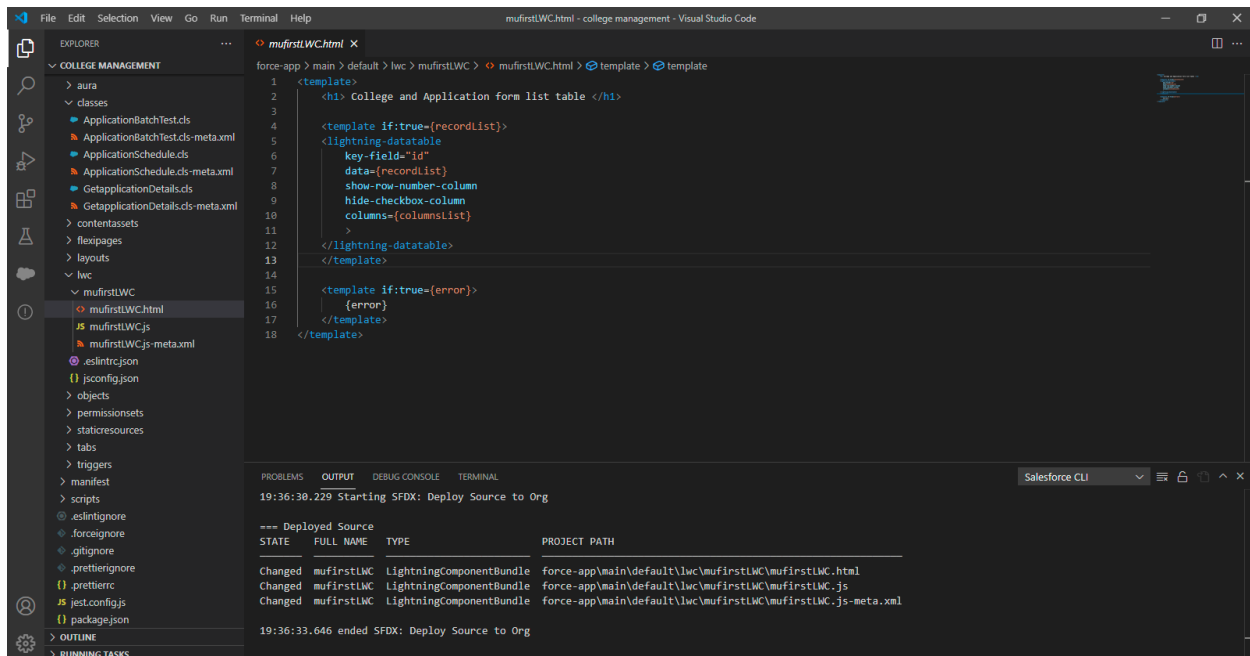
columns={columnsList}>

Project Title:College Management Application

```
>
</lightning-datatable>
</template>

<template if:true={error}>
  {error}
</template>
</template>
```

Screenshot:



Topic: Create College DataTable Component(JAVA SCRIPT FILE)

Milestone/Activites: Created

Description: After the creation of the Html file created the following js file

```
import { LightningElement, api, wire } from 'lwc';
import getapplicationvalues
from '@salesforce/apex/GetapplicationDetails.getapplicationvalues';
export default class CollegeDataTable extends LightningElement {
  columnList = [
    {label : 'Application Form' , fieldName : 'Name', type:'text' },
    {label : 'College Fees' , fieldName : 'College_Fees__c', type:'currency' },
    {label : 'Date Of Birth' , fieldName : 'Date_Of_Birth__c', type:'date' },
```

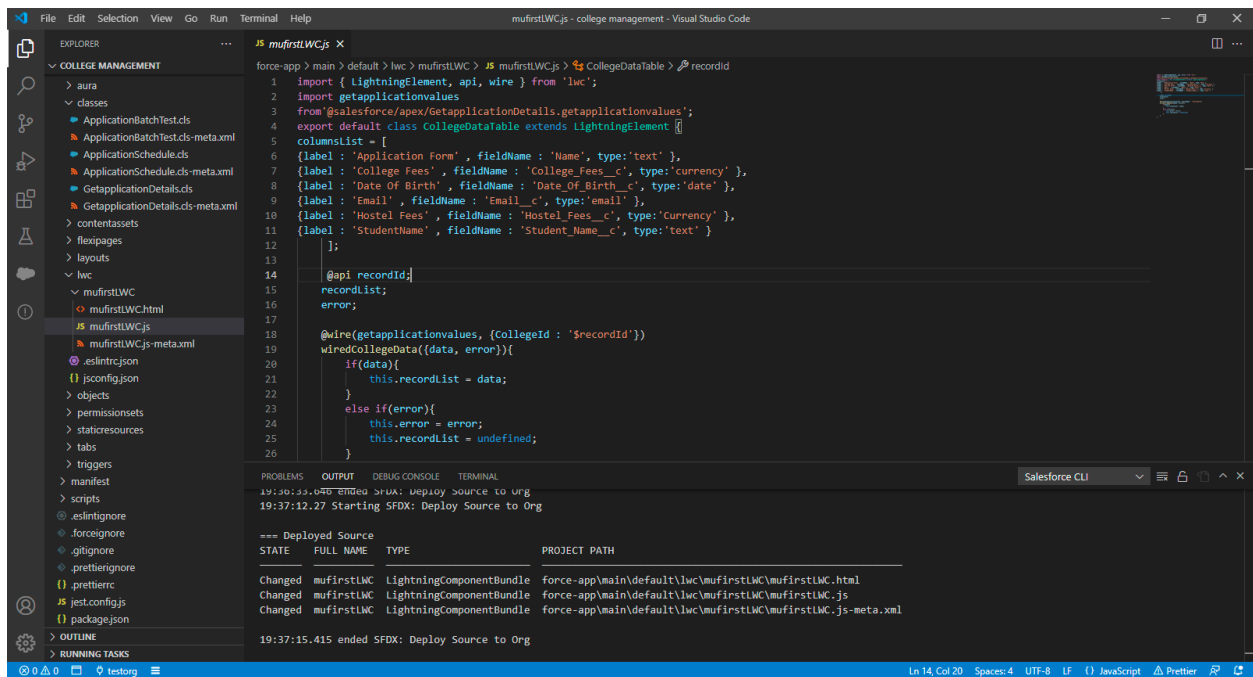
Project Title:College Management Application

```
{label : 'Email' , fieldName : 'Email__c', type:'email' },
{label : 'Hostel Fees' , fieldName : 'Hostel_Fees__c', type:'Currency' },
{label : 'StudentName' , fieldName : 'Student_Name__c', type:'text' }
];
```

```
@api recordId;
recordList;
error;
```

```
@wire(getapplicationvalues, {CollegeId : '$recordId'})
wiredCollegeData({data, error}){
    if(data){
        this.recordList = data;
    }
    else if(error){
        this.error = error;
        this.recordList = undefined;
    }
}
```

Screenshot:



Topic: Create College DataTable Component(META FILE)

Project Title: College Management Application

Milestone/Activities: Created

Description: Changed the Meta File as follows:

```
<?xml version="1.0"?>
<LightningComponentBundle
xmlns="http://soap.sforce.com/2006/04/metadata">
  <apiVersion>51.0</apiVersion>
  <isExposed>true</isExposed>
  <targets>
    <target>lightning__RecordPage</target>
    <target>lightning__AppPage</target>
    <target>lightning__HomePage</target>
  </targets>
</LightningComponentBundle>
```

Screenshot:

