

Day 1 : 03/01/2022,Monday.

Topic: Create your salesforce developer org to get started.

Milestone / Activities : Creating Developer Account.

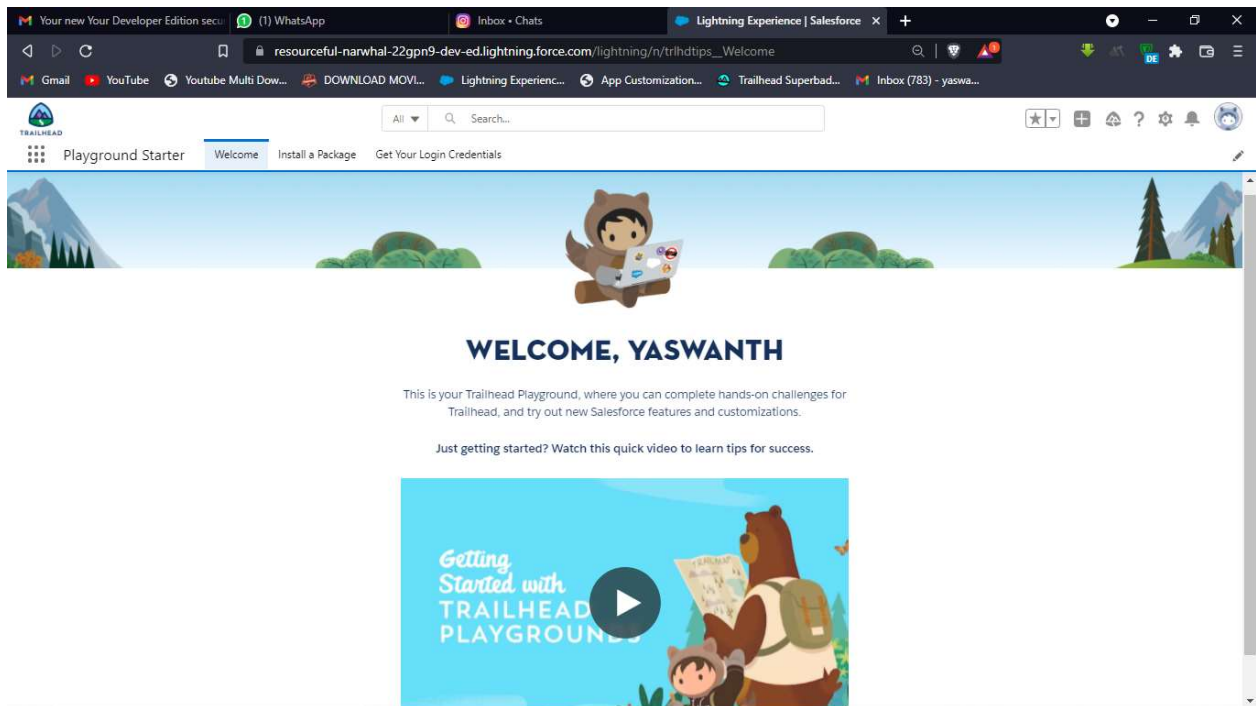
Account activation.

Login to your salesforce Account.

Detailed Description : I have created the salesforce developer account,

- > Then I activated my developer account by verifying the mail,
- > At last I logged into the developer account.

Upload the Screenshot of the Milestone / Activities :



Day 2 : 04/01/2022,Tuesday.

Topic: Custom Object Creation.

Milestone / Activities : Create the custom objects for the application.

Create fields on college object.

Create fields on application form object.

Create fields on student object.

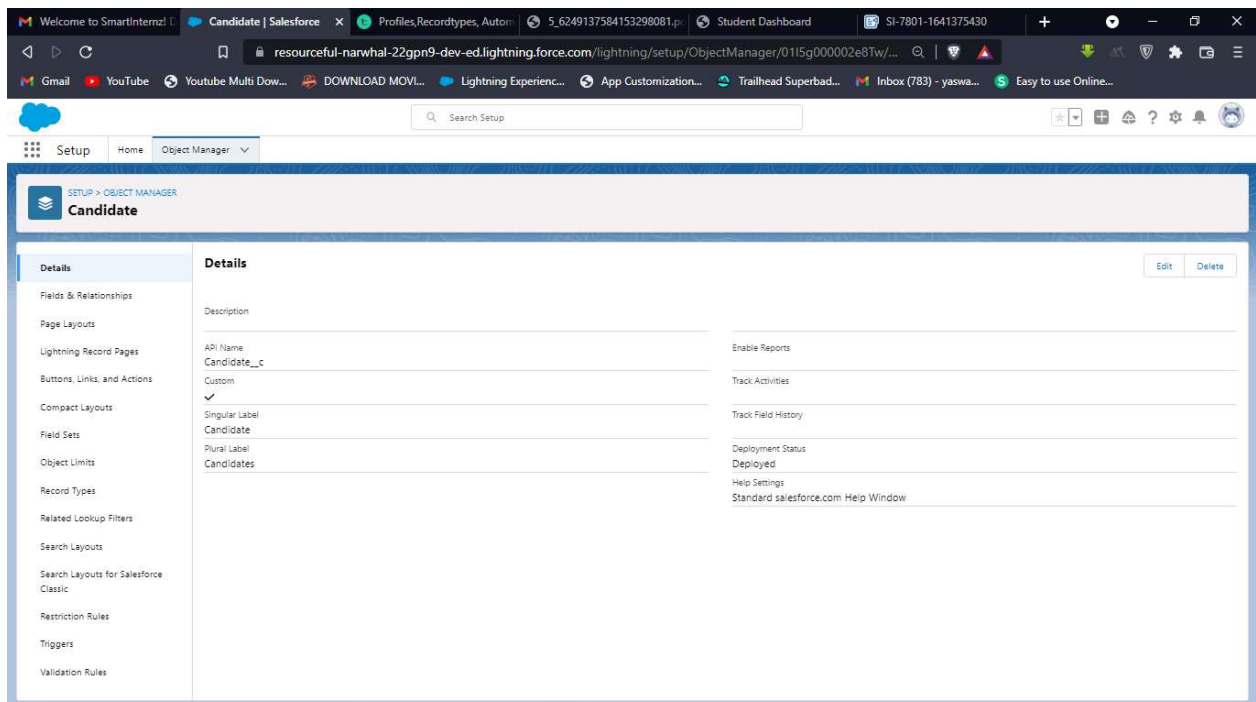
Create fields on subject object.

Detailed Description : First I have created the custom objects as mentioned.

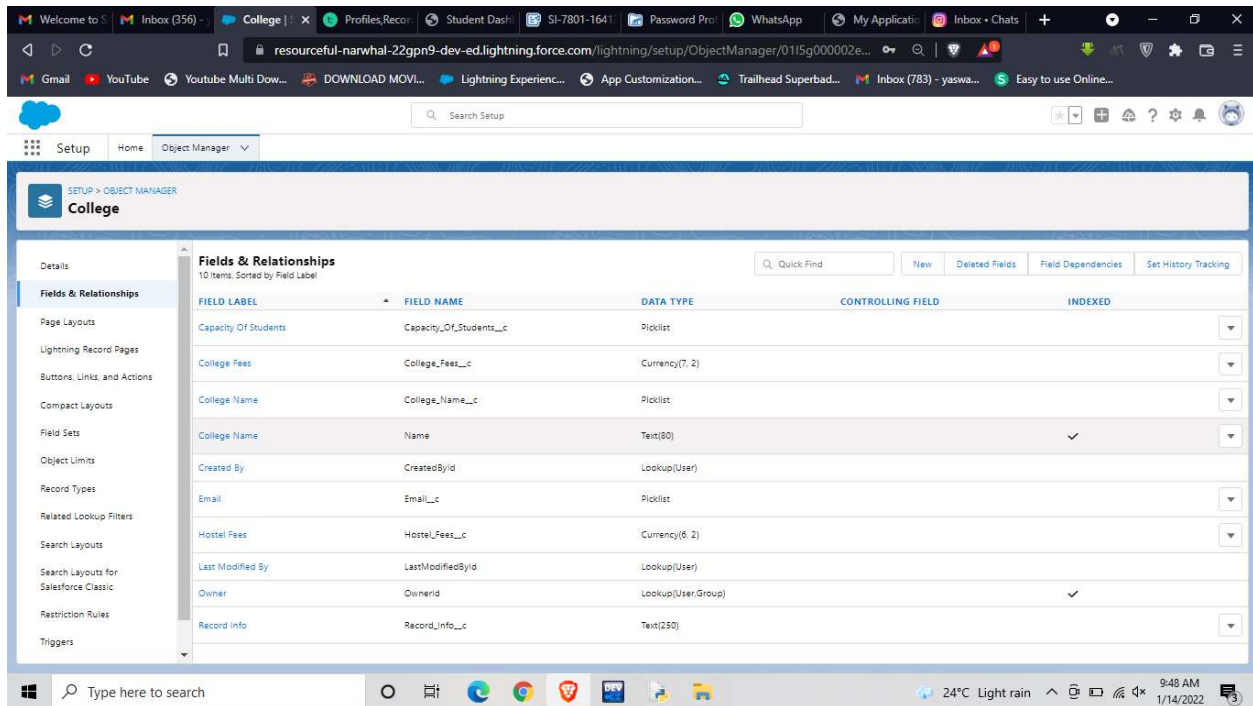
-> After that I have created the required fields for the custom objects below:

-> College,
Application Form,
Student,
Subject.

Upload the Screenshot of the Milestone / Activities :



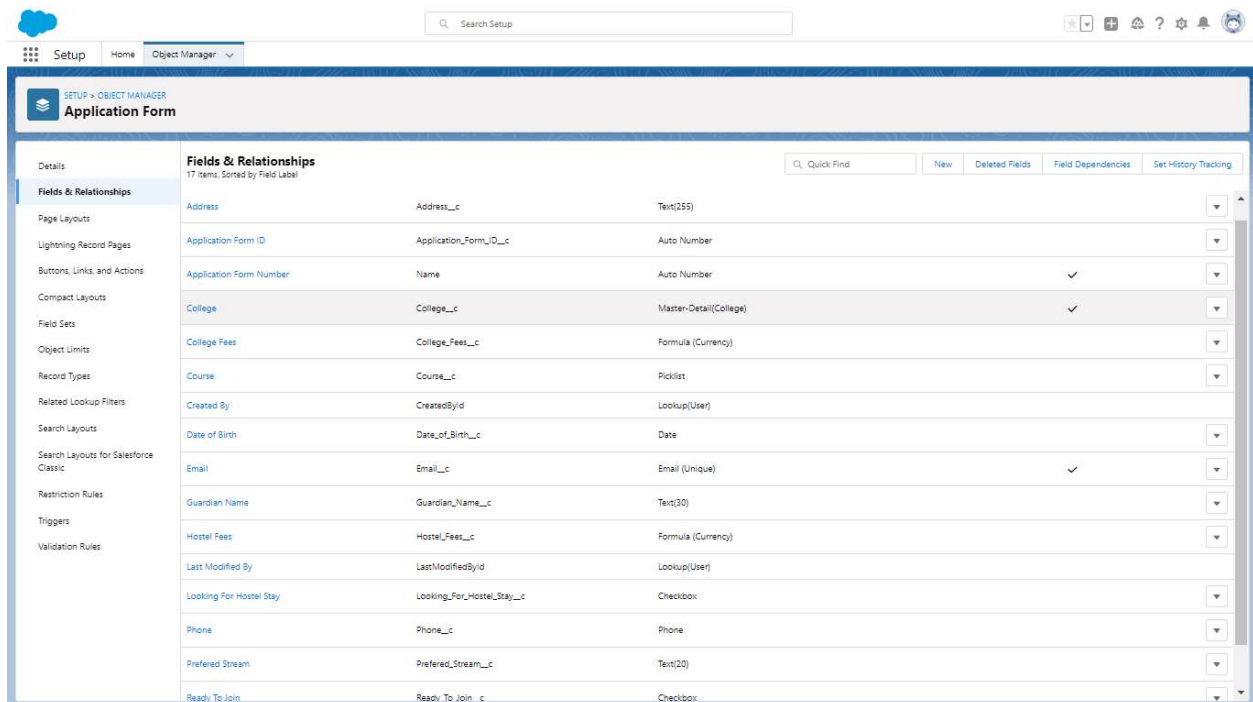
Custom Field on College Object:



The screenshot shows the Salesforce Setup interface for the 'College' object. The 'Fields & Relationships' section is active, displaying a list of 10 fields. The table below summarizes the fields shown:

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(250)		

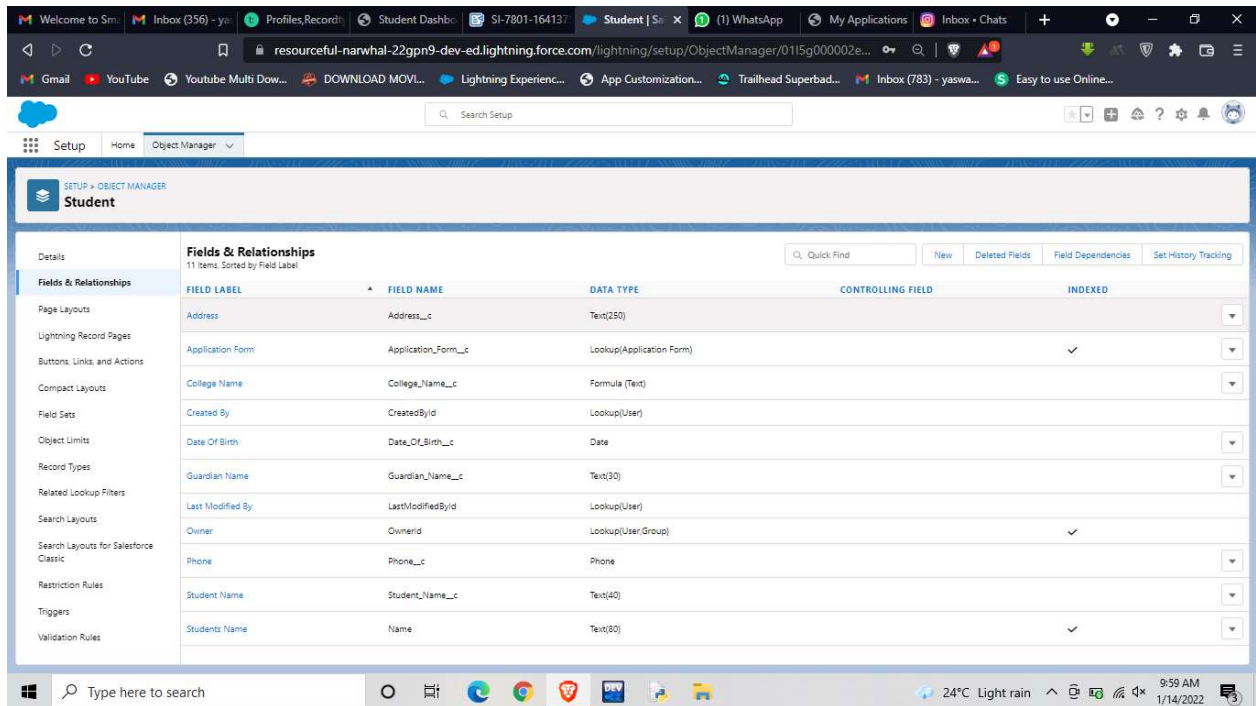
Custom object and fields Creation (Application Form):



The screenshot shows the Salesforce Setup interface for the 'Application Form' object. The 'Fields & Relationships' section is active, displaying a list of 17 fields. The table below summarizes the fields shown:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text(255)		
Application Form ID	Application_Form_ID__c	Auto Number		
Application Form Number	Name	Auto Number		✓
College	College__c	Master-Detail(College)		✓
College Fees	College_Fees__c	Formula (Currency)		
Course	Course__c	Picklist		
Created By	CreatedById	Lookup(User)		
Date of Birth	Date_of_Birth__c	Date		
Email	Email__c	Email (Unique)		✓
Guardian Name	Guardian_Name__c	Text(30)		
Hostel Fees	Hostel_Fees__c	Formula (Currency)		
Last Modified By	LastModifiedById	Lookup(User)		
Looking For Hostel Stay	Looking_For_Hostel_Stay__c	Checkbox		
Phone	Phone__c	Phone		
Preferred Stream	Preferred_Stream__c	Text(20)		
Ready To Join	Ready_To_Join__c	Checkbox		

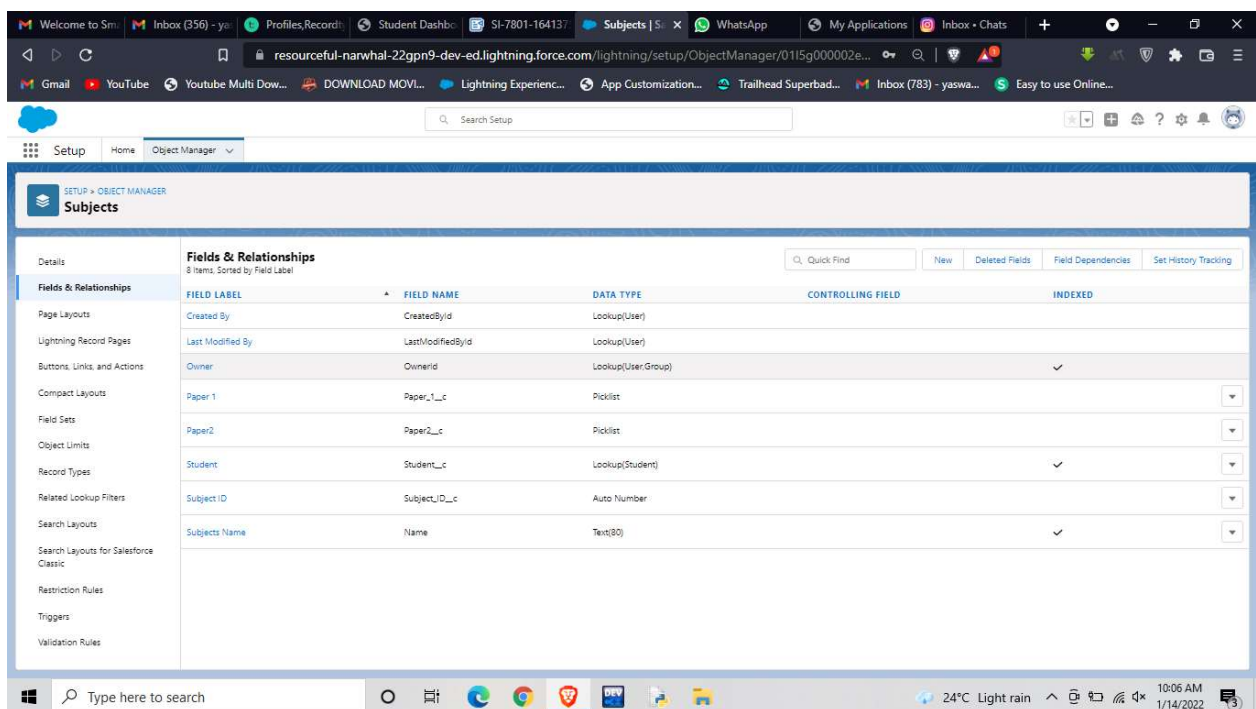
Custom Object Creation of Student Object and its Fields:



The screenshot shows the Salesforce Lightning Setup interface for the 'Student' object. The 'Fields & Relationships' tab is active, displaying a table of fields. The table has columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The fields listed are: Address, Application Form, College Name, Created By, Date Of Birth, Guardian Name, Last Modified By, Owner, Phone, Student Name, and Students Name. The 'Owner' field is marked as indexed and has a checkmark in the CONTROLLING FIELD column.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text(250)		
Application Form	Application_Form__c	Lookup(Application Form)		✓
College Name	College_Name__c	Formula (Text)		
Created By	CreatedById	Lookup(User)		
Date Of Birth	Date_Of_Birth__c	Date		
Guardian Name	Guardian_Name__c	Text(30)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User:Group)		✓
Phone	Phone__c	Phone		
Student Name	Student_Name__c	Text(40)		
Students Name	Name	Text(80)		✓

Custom Object Creation of Subject and its Fields :



The screenshot shows the Salesforce Lightning Setup interface for the 'Subject' object. The 'Fields & Relationships' tab is active, displaying a table of fields. The table has columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The fields listed are: Created By, Last Modified By, Owner, Paper 1, Paper 2, Student, Subject ID, and Subjects Name. The 'Owner' field is marked as indexed and has a checkmark in the CONTROLLING FIELD column.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User:Group)		✓
Paper 1	Paper_1__c	Picklist		
Paper 2	Paper_2__c	Picklist		
Student	Student__c	Lookup(Student)		✓
Subject ID	Subject_ID__c	Auto Number		
Subjects Name	Name	Text(80)		✓

Day 3 : 05/01/2022, Wednesday.

Topic: Adding Business Logic to Application.

Milestone / Activities : Creating Global Picklist Value Sets.

Creating Field Dependencies.

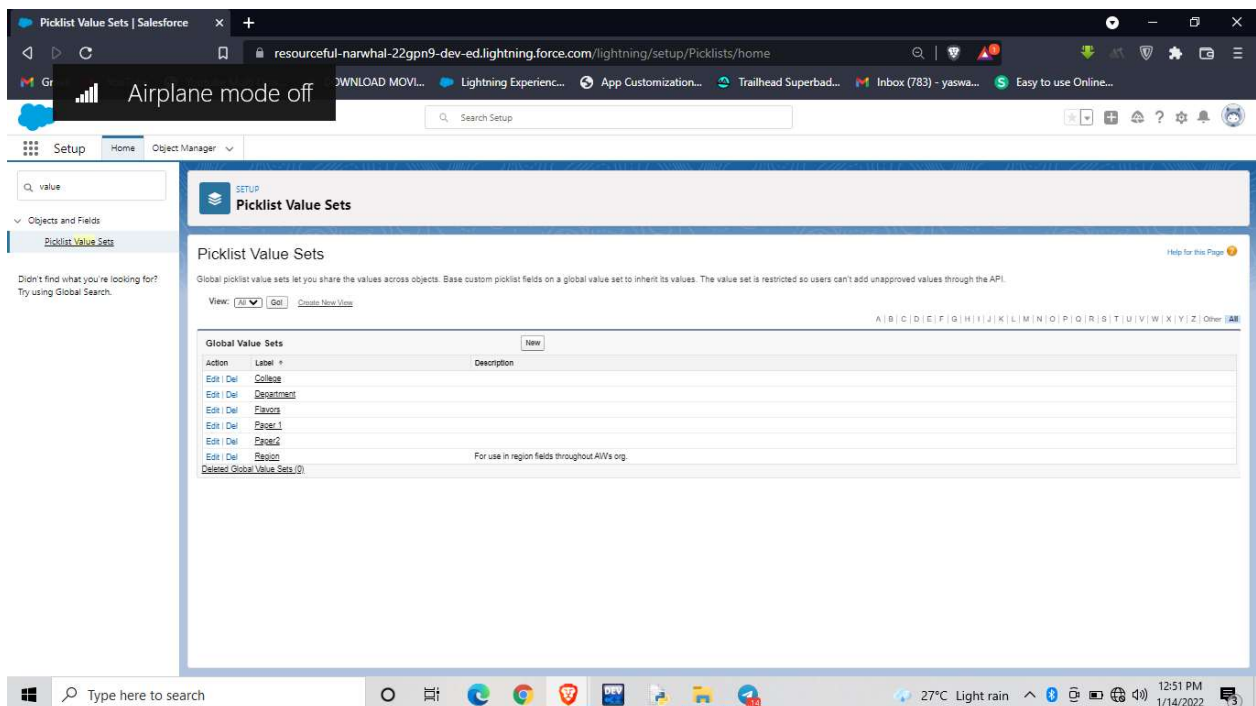
Creating Validation Rules.

Detailed Description :

- > First I have created global picklist value sets by following the below process.
Setup -> Quick Find box -> picklist value sets -> new, then I have created the required Global Picklist Values.
- > Next I have created the Field Dependencies by following below process,
setup -> object manager -> college -> fields and relationships -> field dependencies -> new.
- > Then I have created validation rules on College and ApplicationForm Objects as follows,
setup -> object manager -> college -> validation rules -> new.

Upload the Screenshot of the Milestone / Activities :

Global Picklist Value sets :



Field Dependencies :

The screenshot shows the Salesforce Lightning Setup interface for the 'College' object. 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, Triggers, and Validation Rules. The main content area is titled 'College Field Dependencies' and includes a 'Back to Custom Object: College' link. Below the title, a message states: 'This page allows you to define dependencies between fields (e.g., dependent picklists)'. A table titled 'Field Dependencies' lists two dependencies:

Action	Controlling Field	Dependent Field	Modified By
Edit Del	College Name	Email	Vignesh Thammara 1/13/2022, 9:07 PM
Edit Del	College Name	Capacity Of Students	Vignesh Thammara 1/13/2022, 9:08 PM

Validation Rules : For College Object:

The screenshot shows the Salesforce Lightning Setup interface for the 'College' object, specifically the 'College Validation Rule' page. The left sidebar is the same as in the previous screenshot. The main content area is titled 'College Validation Rule' and includes a 'Back to College' link. Below the title, a 'Validation Rule Detail' section shows the following information:

Field	Value
Rule Name	College_name_and_Record_Info_must_be_sam
Error Condition Formula	TEXT(College_Name__c) <> Record_Info__c
Error Message	college name and record info has to be same
Description	
Created By	Vignesh Thammara 1/13/2022, 9:24 PM
Modified By	Vignesh Thammara 1/13/2022, 9:24 PM

For Application Form :

The screenshot shows the Salesforce Lightning Setup interface for configuring a Validation Rule for the 'Application Form' object. The browser address bar shows the URL: `resourceful-nanwhal-22gpn9-dev-ed.lightning.force.com/lightning/setup/ObjectManager/0115g000002eYLR/...`. The left sidebar contains a navigation menu with the following items: 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, Triggers, and Validation Rules (which is currently selected). The main content area is titled 'Application Form Validation Rule' and includes a 'Back to Application Form' link. Below the title is the 'Validation Rule Detail' section, which contains the following information:

Validation Rule Detail	
Rule Name	No_Modifications_to_application_form
Active	✓
Error Condition Formula	AND(Ready_To_Join__c, OR(ISCHANGED(Address__c), ISCHANGED(College__c), ISCHANGED(Date_of_Birth__c), ISCHANGED(Email__c), ISCHANGED(Guardian_Name__c), ISCHANGED(Phone__c)))
Error Message	U cannot change modify the application form right now
Error Location	Top of Page
Description	
Created By	Vasanth Thammara 1/13/2022, 9:37 PM
Modified By	Vasanth Thammara 1/13/2022, 9:37 PM

At the bottom of the page, there is a Windows taskbar showing the system clock as 1:02 PM on 1/14/2022, and the weather as 27°C Light rain.

Day 4 : 06/01/2022,Thursday.

Topic: Adding Business Logic to Application.

Milestone / Activities : Process Automation.

Detailed Description : I have created a Process named Create Student record as follows,

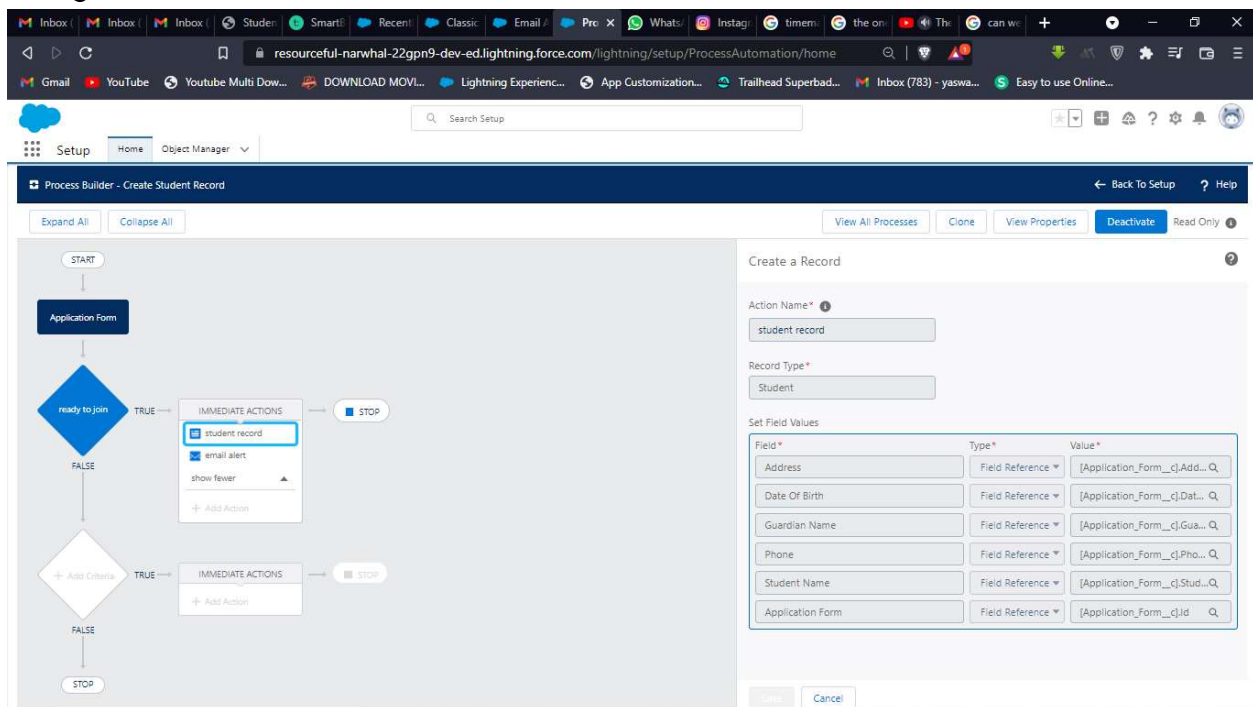
-> setup -> quick find -> process builder -> new -> created as per the requirements for the project.

-> Also the process can send an email to the administrator and the student with the details ,when the ready to join will be checked.

Upload the Screenshot of the Milestone / Activities :

Process Automation :

Using Process Builder :



Day 5 : 07/01/2022, Friday.

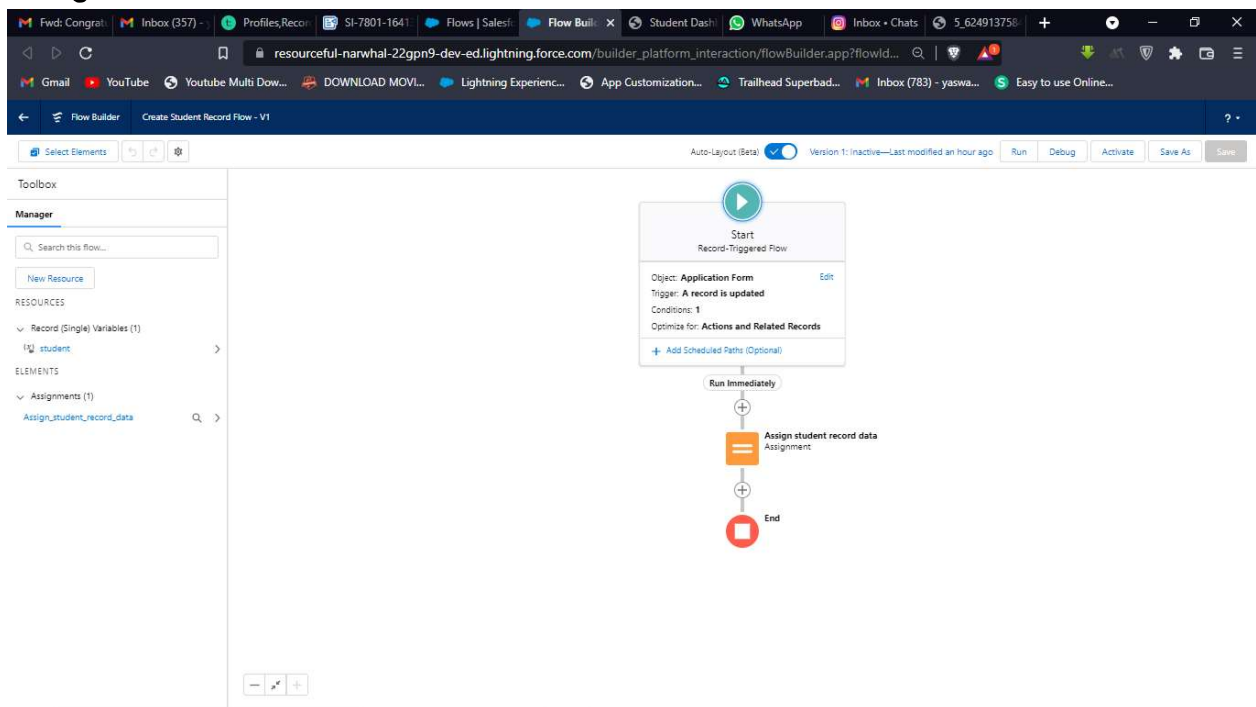
Topic: Adding Business Logic to Application.

Milestone / Activities : Create The Student Record Using Flow.

Detailed Description :

-> I have created a Record trigger Flow as mentioned below:
Setup -> Quick find -> Flow -> NewFlow -> Record trigger Flow.

Using Flow Builder :



Day 6 : 10/01/2022,Monday.

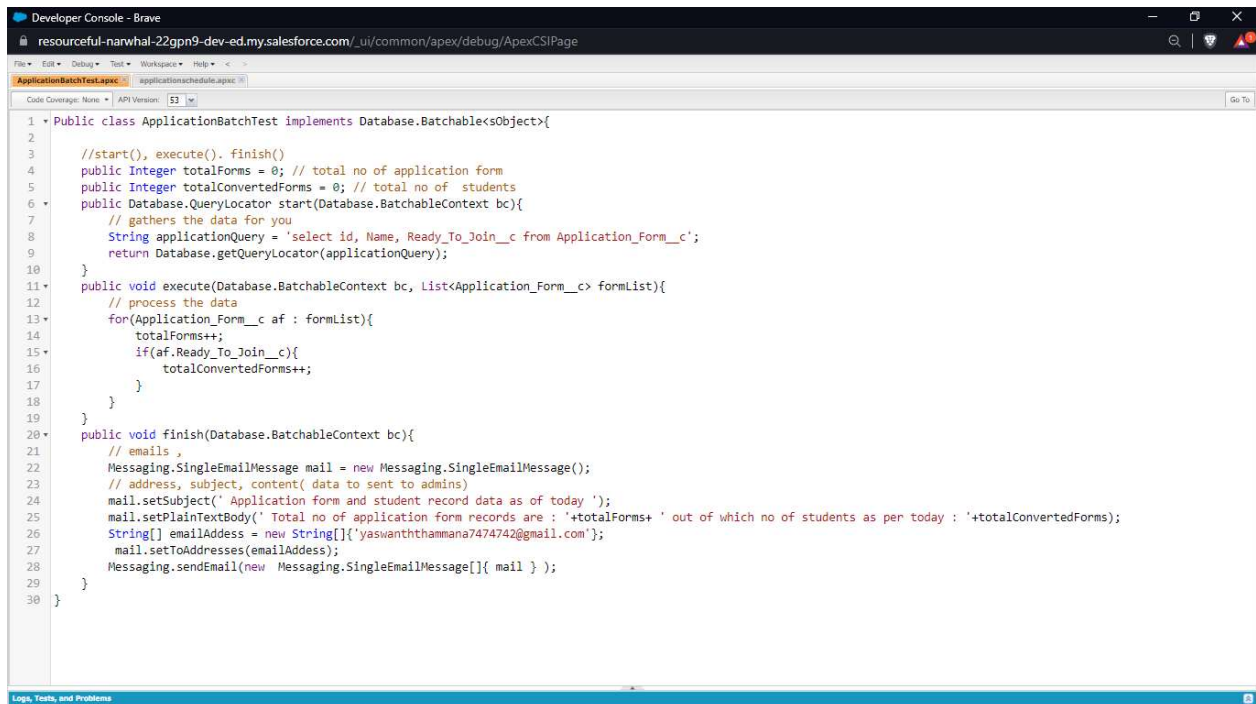
Topic: Batch Apex.

Milestone / Activities : Create a Batch Apex for Application Form.

Detailed Description :

- > First I have created the Batch Apex for Custom Object Application Form as Followed :
- Setup -> Developer Console -> New -> Apex Class -> enter code -> save.

Upload the Screenshot of the Milestone / Activities :
For Batch apex :



```
1  public class ApplicationBatchTest implements Database.Batchable<Object>{
2
3      //start(), execute(), finish()
4      public Integer totalForms = 0; // total no of application form
5      public Integer totalConvertedForms = 0; // total no of students
6      public Database.QueryLocator start(Database.BatchableContext bc){
7          // gathers the data for you
8          String applicationQuery = 'select id, Name, Ready_To_Join__c from Application_Form__c';
9          return Database.getQueryLocator(applicationQuery);
10     }
11     public void execute(Database.BatchableContext bc, List<Application_Form__c> formList){
12         // process the data
13         for(Application_Form__c af : formList){
14             totalForms++;
15             if(af.Ready_To_Join__c){
16                 totalConvertedForms++;
17             }
18         }
19     }
20     public void finish(Database.BatchableContext bc){
21         // emails
22         Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
23         // address, subject, content( data to sent to admins)
24         mail.setSubject(' Application form and student record data as of today ');
25         mail.setPlainTextBody(' Total no of application form records are : '+totalForms+ ' out of which no of students as per today : '+totalConvertedForms);
26         String[] emailAddresses = new String[]{'yaswanththammana74742@gmail.com'};
27         mail.setToAddresses(emailAddresses);
28         Messaging.sendEmail(new Messaging.SingleEmailMessage[] { mail } );
29     }
30 }
```

Day 7 : 11/01/2022,Tuesday.

Topic: Batch Apex.

Milestone / Activities : Create a Scheduler Class.

Detailed Description :

- > I have created the Scheduler Class as follows :
Setup -> Developer Console -> Apex Class -> enter code -> Save.
- > Then created the Schedule job as follows :
Setup -> Quick find -> Apex classes -> Schedule Apex -> Filled as per requirement -> Save.

Upload the Screenshot of the Milestone / Activities :

For Scheduler Apex :

The screenshot displays the Salesforce Developer Console interface. The main editor shows the following Apex code for the `applicationschedule` class, which implements the `Schedulable` interface:

```
1 public class applicationschedule implements Schedulable{
2
3
4
5     public void execute(SchedulableContext sc){
6
7         ApplicationBatchTest abt = new ApplicationBatchTest();
8
9         Database.executeBatch(abt, 400); // 200 to 2000
10
11     }
12
13 }
14
15 }
```

At the bottom of the console, the 'Overall Code Coverage' table is visible, showing the following data:

Class	Percent	Lines
Overall	78%	
AccountAddressTrigger	66%	2/3
AccountManager	100%	6/6
AccountProcessor	100%	7/7
AdPrimaryContact	100%	13/13
AnimalLocator	100%	11/11

For Schedule Job :

The screenshot shows the Salesforce Setup interface for scheduling an Apex class. The browser tabs include 'Fwd: Congratu...', 'Inbox (357) - y...', 'Profiles, Record...', 'Apex Class...', 'SI-7801-164137', 'Student Dashb...', 'WhatsApp', 'Inbox - Chats', '5_62491375841', and '+'. The address bar shows the URL: 'resourceful-nanwhal-22gpn9-dev-ed.lightning.force.com/lightning/setup/ApexClasses/page?address=%2Fui...'. The left sidebar contains a search bar with 'sched...' and a list of navigation items: 'Feature Settings', 'Sales', 'Products', 'Product Schedules Settings', 'Environments', 'Jobs', and 'Scheduled Jobs'. The main content area is titled 'Apex Classes' and 'Schedule Apex'. It includes a description: 'Schedule an Apex class that implements the 'Schedulable' interface to be automatically executed on a weekly or monthly interval.' Below this is a form with fields for 'Job Name' (applicationschedule), 'Apex Class' (applicationschedule), and 'Schedule Apex Execution'. The 'Frequency' section has radio buttons for 'Weekly' (selected) and 'Monthly'. A dropdown menu for 'Recur every week on' shows days of the week: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, and Saturday. The 'Start' date is 1/13/2022 and the 'End' date is 2/13/2022. The 'Preferred Start Time' is 8:00 PM. A note at the bottom states: 'Exact start time will depend on job queue activity'.

Apex Classes

Schedule Apex

Schedule an Apex class that implements the 'Schedulable' interface to be automatically executed on a weekly or monthly interval.

[Cancel](#) [Details](#)

Job Name: applicationschedule
Apex Class: applicationschedule
Schedule Apex Execution

Frequency: ☒ Weekly ☐ Monthly

Recur every week on:

- ☒ Sunday
- ☒ Monday
- ☒ Tuesday
- ☒ Wednesday
- ☒ Thursday
- ☒ Friday
- ☒ Saturday

Start: 1/13/2022 [1/14/2022]
End: 2/13/2022 [1/14/2022]
Preferred Start Time: 8:00 PM

Exact start time will depend on job queue activity

[Cancel](#) [Details](#)

Day 8 : 12/01/2022,Wednesday.

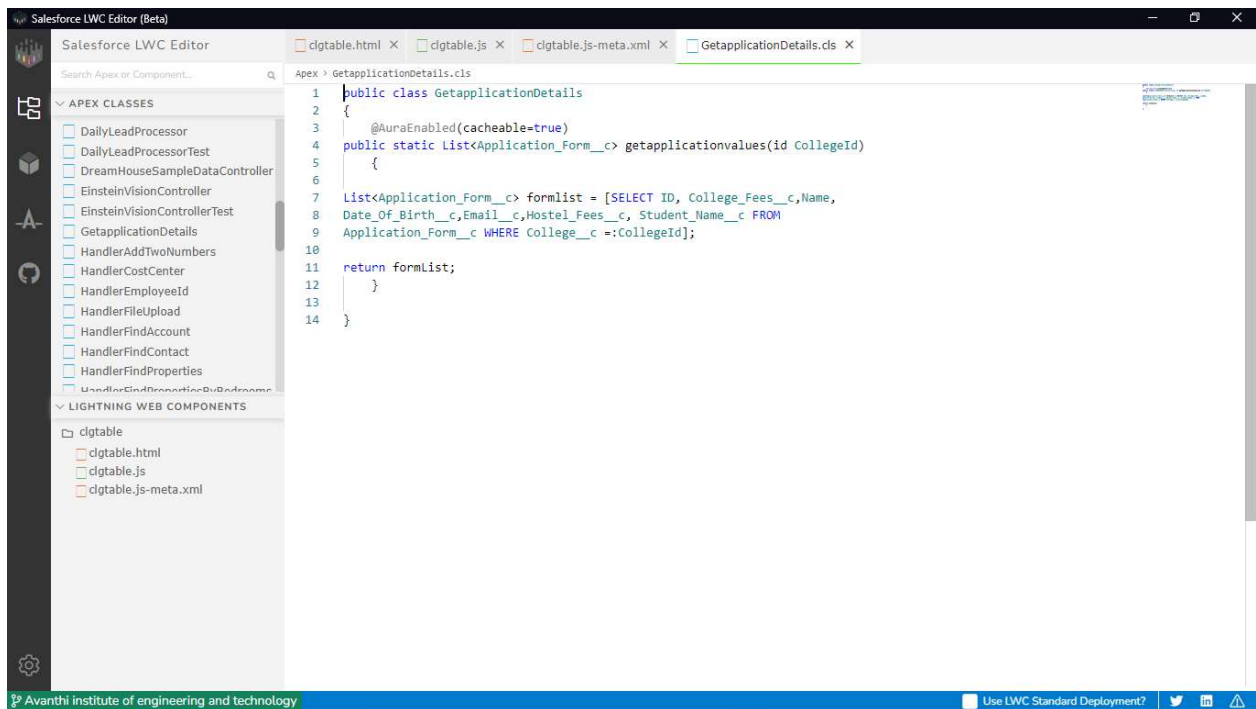
Topic: Lightning Web Components.

Milestone / Activities : Create College Data Table Component

Detailed Description : I have created a apex class for creating the College Data Table.

-> Developer Console -> New -> Apex Class -> Enter Code -> Save.

Upload the Screenshot of the Milestone / Activities :



Day 9 : 13/01/2022,Thursday.

Topic: Lightning Web Components.

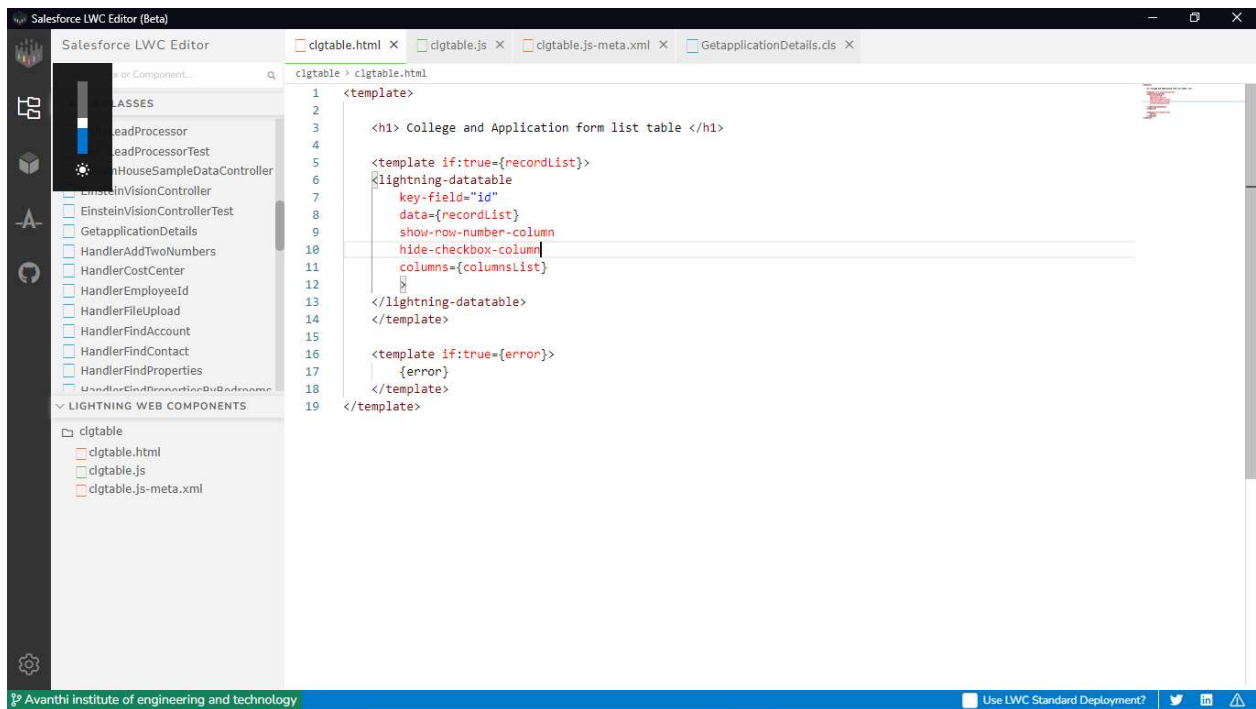
Milestone / Activities : Create College Data Table Component (HTML File).

Detailed Description :

-> I have created the HTML File for the College Data Table as follows :

-> App Launcher -> College Management -> Colleges -> Adding Lwc Extension -> Open Lwc Editor -> Create HTML File -> Save.

Upload the Screenshot of the Milestone / Activities :



Day 10 : 14/01/2022, Friday.

Topic: Lightning Web Components.

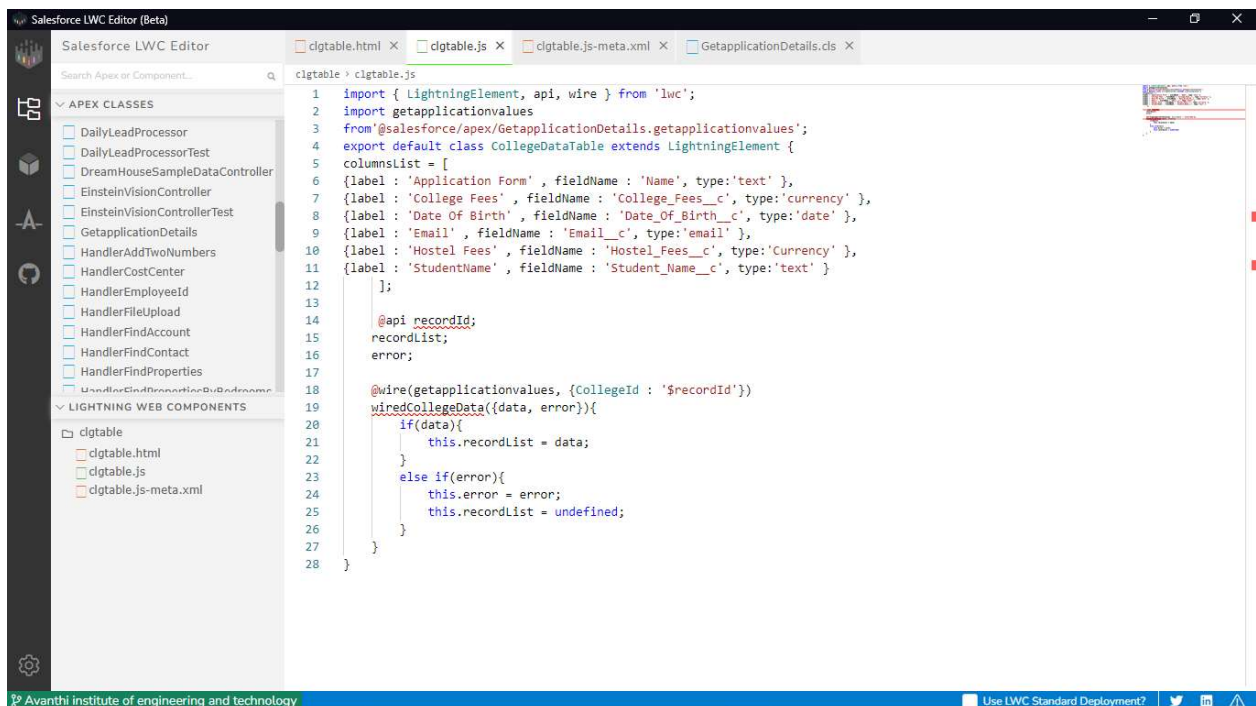
Milestone / Activities : Create College Data Table Component (JAVA Script File).

Detailed Description :

-> I have created the JAVA Script File for the College Data Table as follows :

-> App Launcher -> College Management -> Colleges -> Open Lwc Editor -> Create JAVA Script File -> Save.

Upload the Screenshot of the Milestone / Activities :



The screenshot displays the Salesforce LWC Editor (Beta) interface. On the left, a sidebar shows the 'APEX CLASSES' and 'LIGHTNING WEB COMPONENTS' sections. The 'LIGHTNING WEB COMPONENTS' section is expanded, showing a folder named 'cdgtable' containing three files: 'cdgtable.html', 'cdgtable.js', and 'cdgtable.js-meta.xml'. The main editor area shows the code for 'cdgtable.js'. The code defines a class 'CollegeDataTable' that extends 'LightningElement'. It includes a 'columnsList' array with six columns: 'Application Form', 'College Fees', 'Date Of Birth', 'Email', 'Hostel Fees', and 'StudentName'. The class also includes a '@api recordId' property, a 'recordList' property, and an '@wire' decorator for 'getapplicationvalues' with a wired function 'wiredCollegeData' that handles data and errors.

```
1 import { LightningElement, api, wire } from 'lwc';
2 import getapplicationvalues
3 from '@salesforce/apex/GetapplicationDetails.getapplicationvalues';
4 export default class CollegeDataTable extends LightningElement {
5   columnsList = [
6     {label: 'Application Form', fieldName: 'Name', type: 'text' },
7     {label: 'College Fees', fieldName: 'College_Fees__c', type: 'currency' },
8     {label: 'Date Of Birth', fieldName: 'Date_Of_Birth__c', type: 'date' },
9     {label: 'Email', fieldName: 'Email__c', type: 'email' },
10    {label: 'Hostel Fees', fieldName: 'Hostel_Fees__c', type: 'Currency' },
11    {label: 'StudentName', fieldName: 'Student_Name__c', type: 'text' }
12  ];
13
14  @api recordId;
15  recordList;
16  error;
17
18  @wire(getapplicationvalues, {CollegeId: '$recordId'})
19  wiredCollegeData({data, error}){
20    if(data){
21      this.recordList = data;
22    }
23    else if(error){
24      this.error = error;
25      this.recordList = undefined;
26    }
27  }
28 }
```


Day 11 : 17/01/2022,Monday.

Topic: Lightning Web Components.

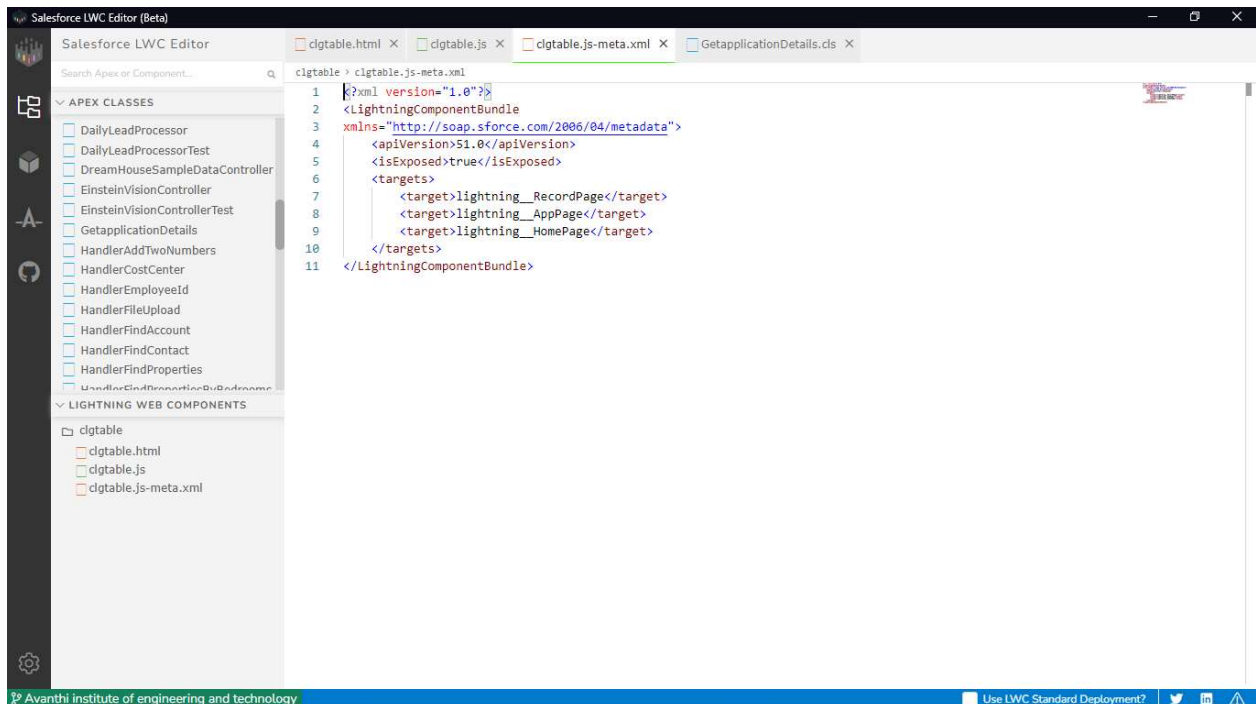
Milestone / Activities : Create College Data Table Component (Meta File).

Detailed Description :

-> I have created the Meta File for the College Data Table as follows :

-> App Launcher -> College Management -> Colleges -> Open Lwc Editor -> Create Meta File -> Save.

Upload the Screenshot of the Milestone / Activities :



Day 12 : 18/01/2022,Tuesday.

Topic: Lightning Web Components.

Milestone / Activities : Create Data Table Tab in the Lightning app page.

Detailed Description :

-> I have created the Data Table in the Lightning app page by adding new lightning component to the new Tab

-> App Launcher -> College Management -> Colleges -> Edit Page -> Add New Tab -> Light Components -> ClgTable -> Save.

Upload the Screenshot of the Milestone / Activities :

