

# Project : College Management Application

## Day-1 :

### Topic: Create Your Salesforce Developer Org To Get Started

**Milestone/Activities :** Creating Developer account, Account activation, Login to your Salesforce Account

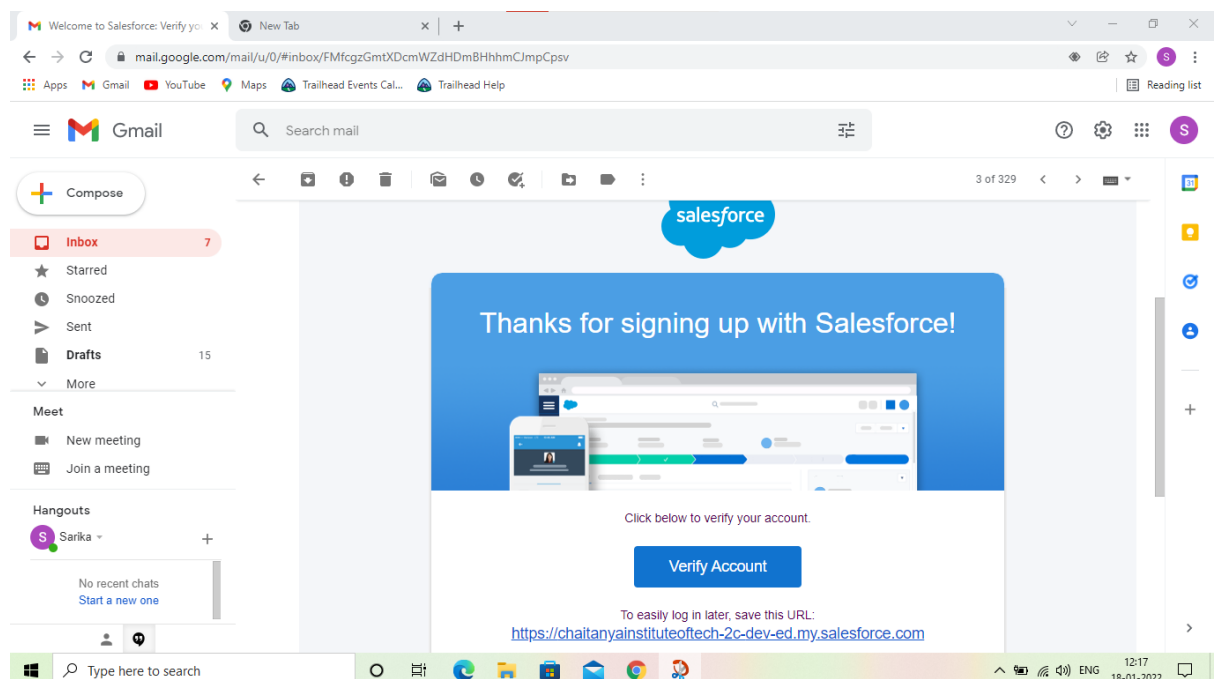
### Description :

#### Creating a developer org in salesforce :

1. Go to [developers.salesforce.com/](https://developers.salesforce.com/)
2. Click on sign up.
3. On the sign up form, enter the following details :
  1. First name : & Last name : sarika kola
  2. Email : sarika.kola2001@gmail.com
  3. Company : Chaitanya Institute of Technology and Science
  4. County : India
  5. Postal Code : 506001
  6. Username : sarika@chaitanyainstituteoftech.com

Click on sign up after filling these.

#### Account Activation :



- Verified mail account and activated the account.

#### Login to your Salesforce Customer Account :

- Logged into salesforce account using <https://login.salesforce.com/> link.
-

## Day-2 :

### Topic: Custom Object Creation

#### Milestone/Activities : Create the custom objects for the Application, Create Fields on college object

##### Description :

Created a College custom object and created Fields for the custom object College.

Fields for College are:

- Record Info
- College Name
- Email
- College Fees
- Hostel Fees
- Capacity of Students
- Validation Rules

The screenshot shows the Salesforce Setup interface for a custom object named 'college'. The 'Fields & Relationships' section is active, displaying a list of 7 fields sorted by Field Label. The fields are:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
College Email	College_Email__c	Picklist		
College Name	College_Name__c	Picklist		
college Name	Name	Text(80)		✓
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Record Info	Record_Info__c	Text(100)		

## Day-3 :

### Topic: Custom Object Creation

# Milestone/Activities : Create the custom objects for the Application Form, Student, Subject, Create Fields on Application Form object, Student object, Subject object, Adding Business Logic to Application

## Description :

Created a Application Form custom object and created Fields for the custom object Application Form.

The screenshot shows the Salesforce Object Manager interface for the 'Application Form' custom object. The left sidebar contains a navigation menu with options: Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts. The main area displays the 'Fields & Relationships' section with 12 items, sorted by Field Label. The table below lists the fields:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text(255)		
Application Form ID	Application_Form_ID__c	Picklist		
Application Form Name	Name	Text(80)		✓
ApplicationForm Name	ApplicationForm_Name__c	Text(80)		
college	college__c	Master-Detail(college)		✓
College Fee	College_Fee__c	Formula (Currency)		
Created By	CreatedById	Lookup(User)		

Created a student custom object and created Fields for the custom object students.

## Student Object :

The screenshot shows the Salesforce Object Manager interface for the 'Student' custom object. The left sidebar contains a navigation menu with options: Details (selected), Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts. The main area displays the 'Details' section with the following configuration:

Field Name	Value
Description	
API Name	Student__c
Custom	✓
Singular Label	Student
Plural Label	Students
Enable Reports	
Track Activities	
Track Field History	
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

## Student Fields :

The screenshot shows the Salesforce Object Manager interface for the 'Student' object. The left sidebar contains a navigation menu with options: Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts. The main content area is titled 'Fields & Relationships' and shows a list of 8 fields, sorted by Field Label. The fields are:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Date of Birth	Date_of_Birth__c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
LastModifiedById	LastModifiedById__c	Lookup(User)		✓
Owner	Owner__c	Lookup(User)		✓
Owner	OwnerId	Lookup(User,Group)		✓
Phone	Phone__c	Phone		

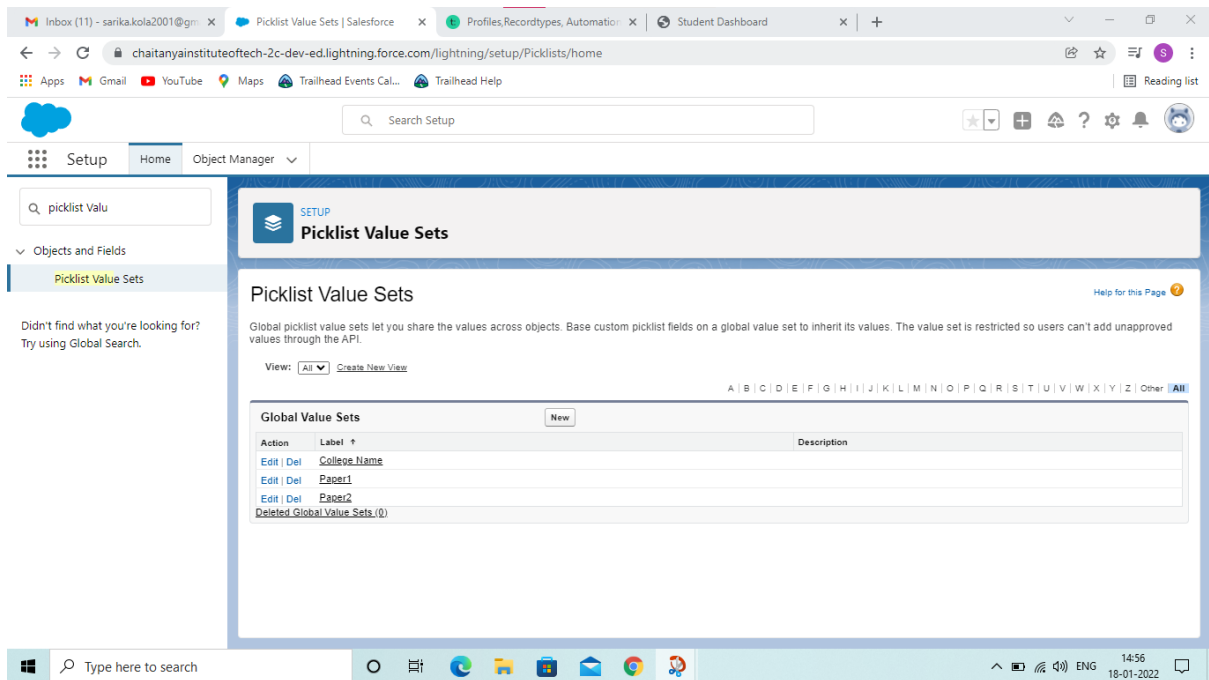
In *Picklist* values sets Created 3 Picklist values sets

College Name

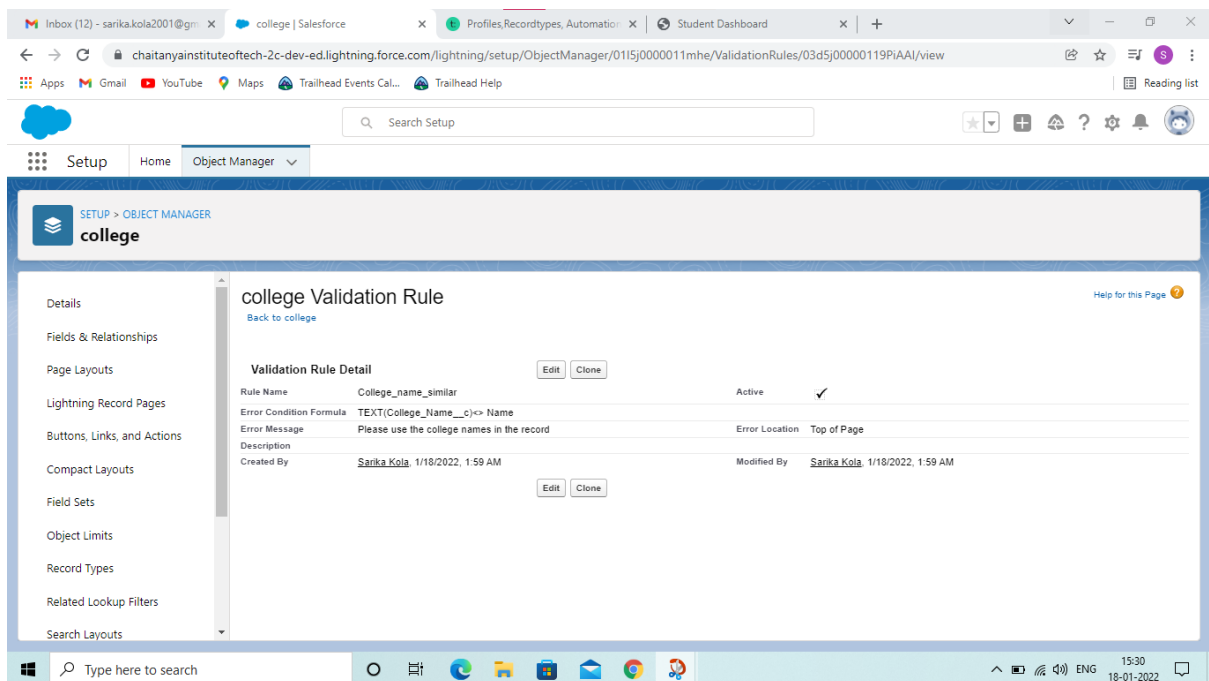
Paper 1

Paper 2

## Creating Global Picklist Values :



## Creating Validation Rule :



## Creating Field Dependencies :

Created field dependencies for college object

controlling field as college name and depending field is college email.

controlling field as college name and depending field is capacity of students

The screenshot shows the Salesforce Setup interface for the 'college' object. The left sidebar contains navigation links: Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts. 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
<a href="#">Edit</a>   <a href="#">Del</a>	College Name	Capacity of Student	<a href="#">Sarika Kola</a> 1/18/2022, 1:23 AM
<a href="#">Edit</a>   <a href="#">Del</a>	College Name	Email	<a href="#">Sarika Kola</a> 1/18/2022, 1:22 AM

The bottom of the screen shows the Windows taskbar with various application icons and the system clock displaying 14:53 on 18-01-2022.

## Day-4 :

### Topic : Validation Rule

### Description :

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

The screenshot shows the Salesforce Setup interface for the 'Application Form' object. The left sidebar contains navigation links: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts. The main content area is titled 'Application Form Validation Rule' and includes a 'Back to Application Form' link. Below the title, a 'Validation Rule Detail' section shows the rule configuration:

Rule Name	Error Condition Formula	Error Message	Description	Created By	Modified By
No_Modification	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 ) ) )	No modification are allowed		<a href="#">Sarika Kola</a> 1/18/2022, 2:20 AM	<a href="#">Sarika Kola</a> 1/18/2022, 2:20 AM

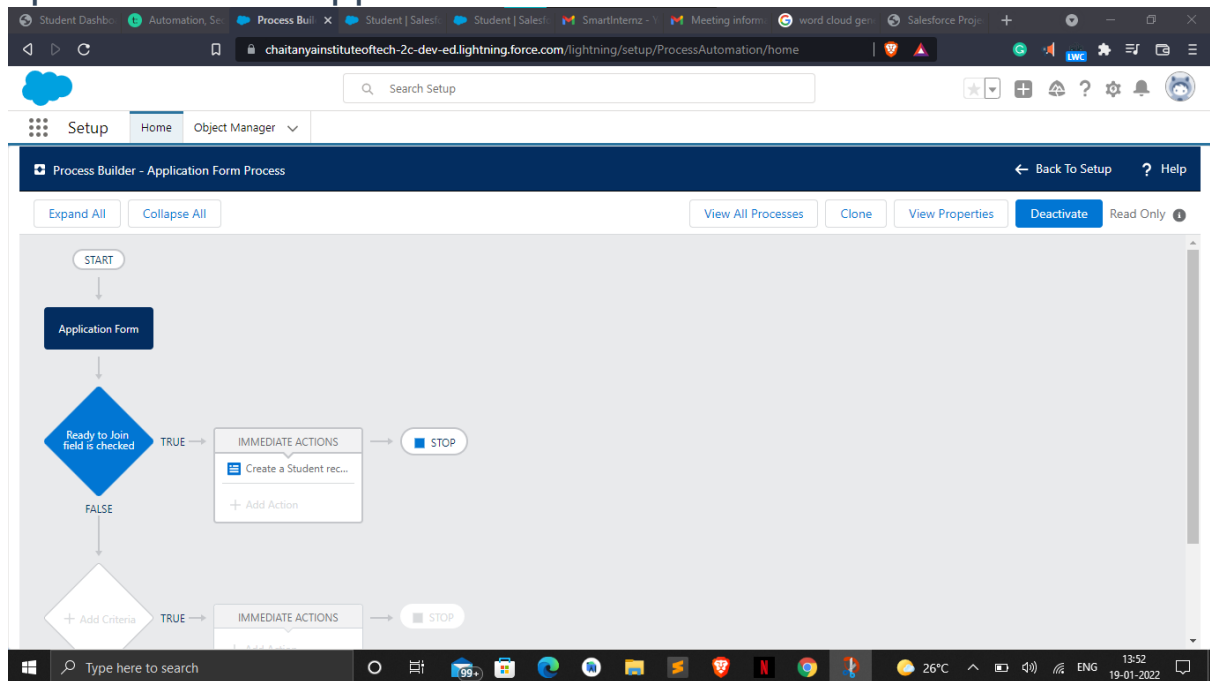
The 'Active' checkbox is checked. The 'Error Location' is set to 'Top of Page'. The bottom of the screen shows the Windows taskbar with various application icons and the system clock displaying 15:51 on 18-01-2022.

**Day-5 :**

**Topic : Process Automation**

**Description :**

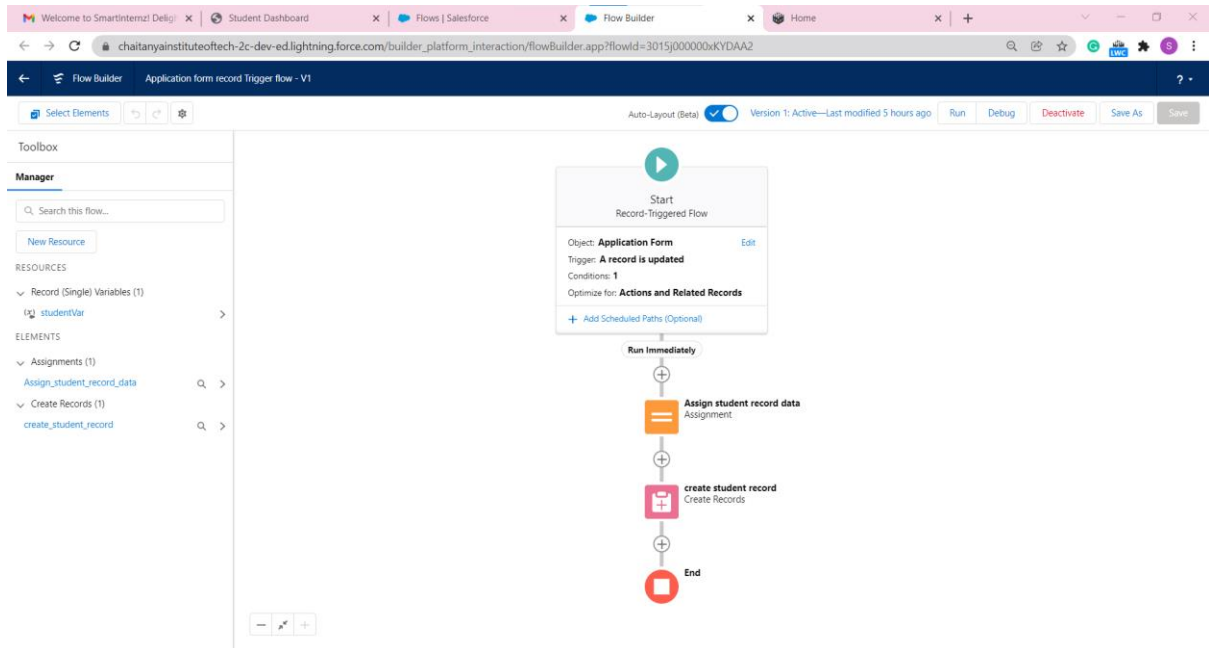
Created an automation process such that when the "ready to join" field is checked on the application form object it will to create the student record automatically with the information specified in the application form record.



**Day-6 :**

**Topic: Create the student record using flow**

## Milestone/Activities :Adding Business Logic to Application Description :



**Day-7 :**

**Topic: BATCH APEX**

**Milestone/Activities : Create A Batch-apex For Application Form,  
Create A Scheduler Class**

**Description : created  
apex batches**

- Applicationbatchtest
- applicationschedule



The screenshot shows the Salesforce Developer Console with the 'ApplicationBatchTest.apex' file open. The code defines a class that implements Database.Batchable<subject> and Database.Stateful. It includes methods for start, execute, and finish. The start method sets up a query to select application forms. The execute method iterates through a list of application forms and updates their status based on a 'Ready\_To\_Join\_\_c' field.

```
1 public class ApplicationBatchTest implements Database.Batchable<subject>, Database.Stateful{
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 }
```

The screenshot shows the Salesforce Developer Console with the 'applicationSchedule.apex' file open. The code defines a class that implements Schedulable. It includes an execute method that creates an instance of ApplicationBatchTest and executes a batch of application forms.

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

## Day-8:

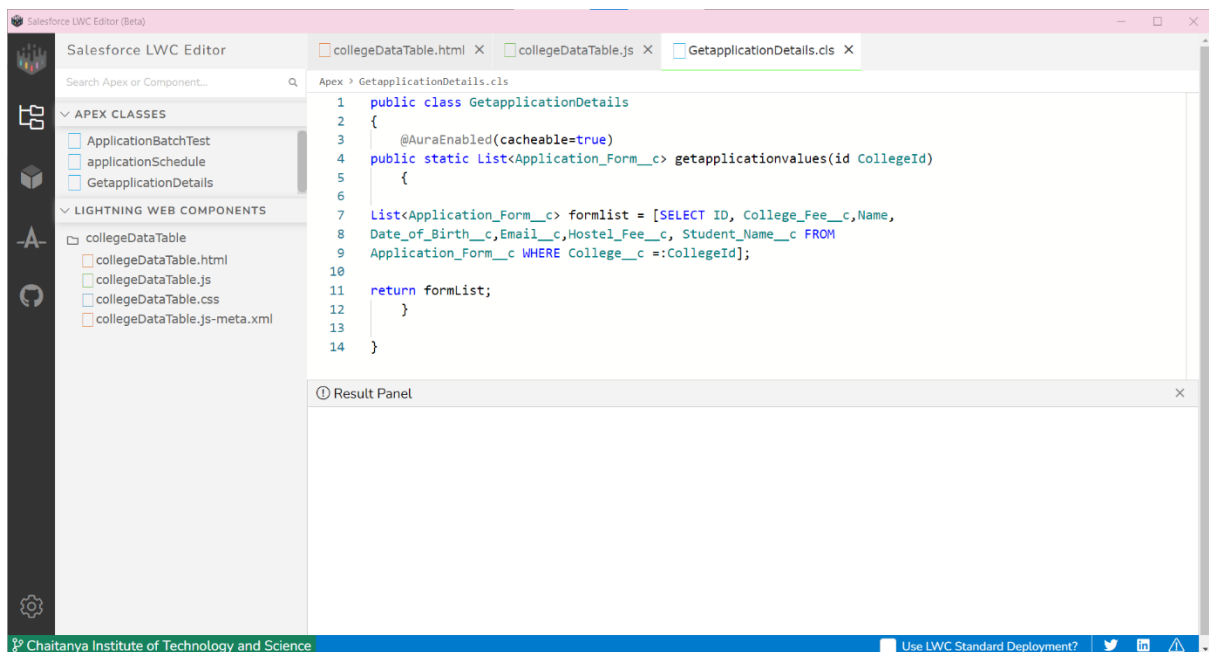
### Topic : Lightning Web Components

#### Milestone/Activity :

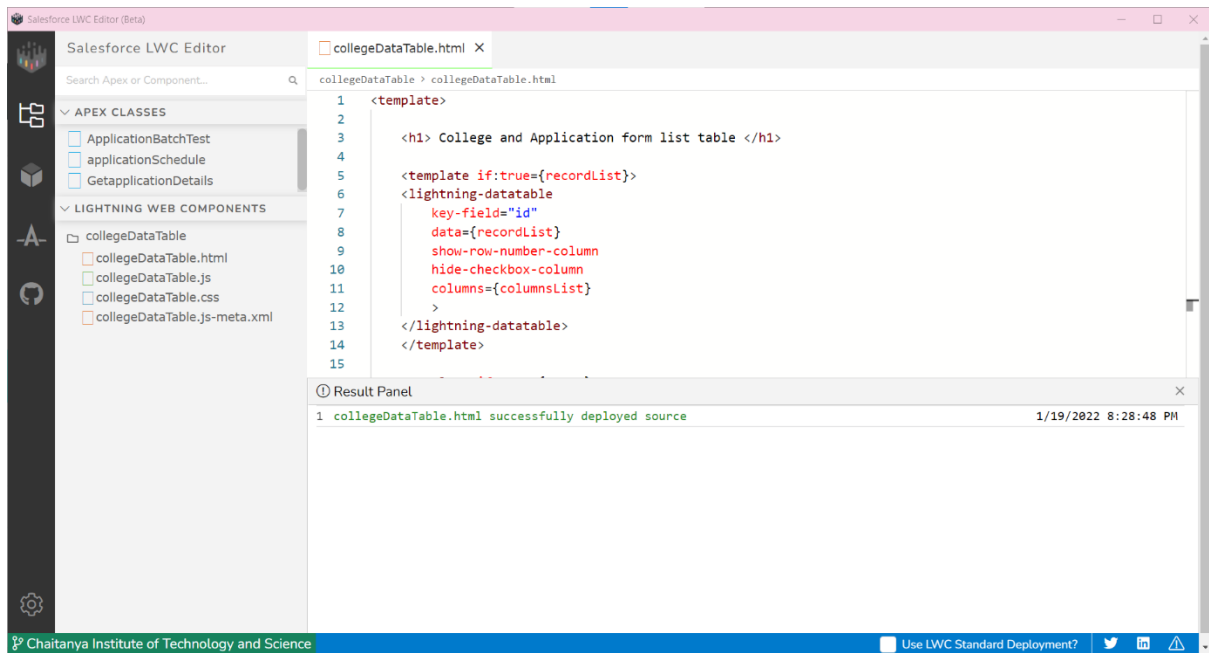
- Creating college data table component( Apex class )
- creating college data table component( HTML File )
- creating college data table component( Java Script )
- creating college data table component( Meta File )

## Description :

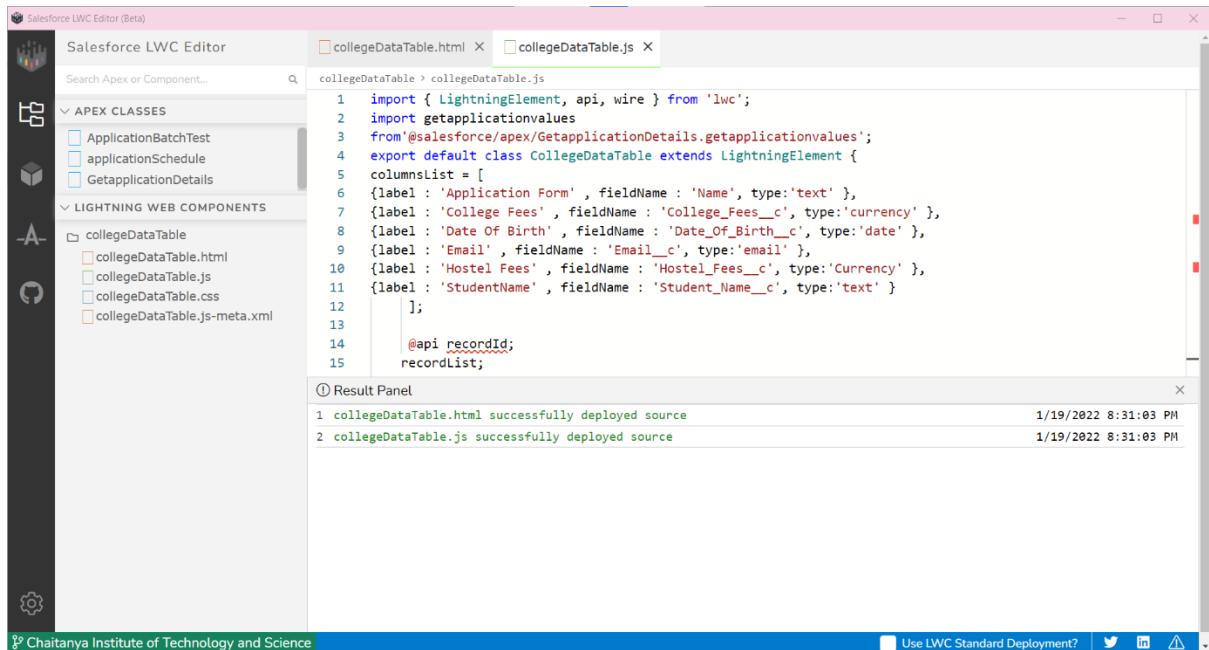
## Creating college data table component( Apex class ):



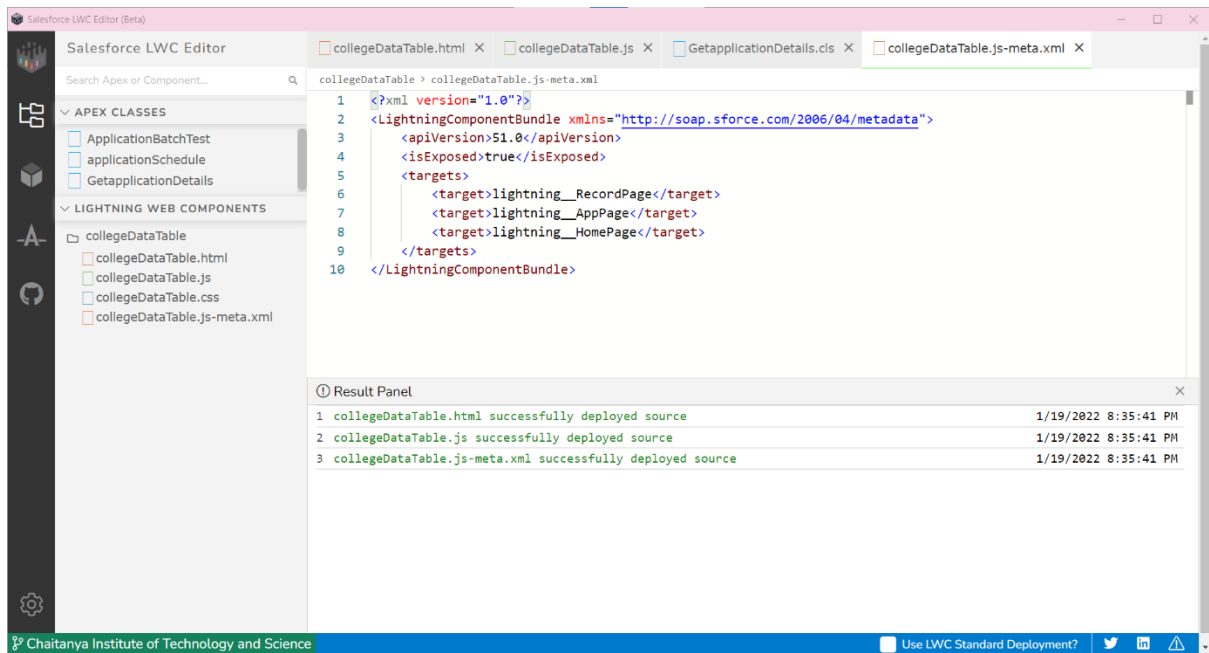
## Creating college data table component( HTML File ) :



## Creating college data table component( Java Script File ) :



## Creating college data table component( Meta File ) :



## Data table in college object :

