Day 1 :- College Management Application

Topic :- create your Salesforce Developer Org To Get Start

Milestone / Activity :-

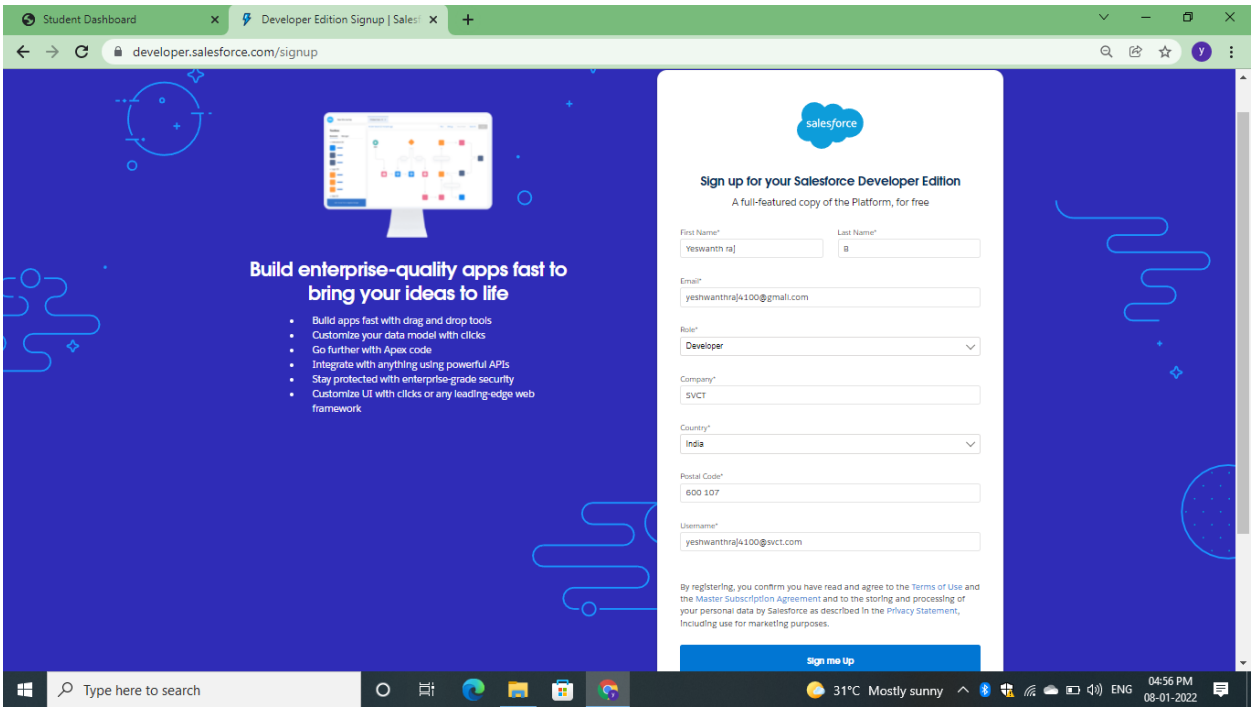
- Creating a Developer Account
- Account Activation
- Login to your Salsforce account
    - ☆ Salesforce Login

Detailed Description :-

To create the Developer Account first we have to login to the Account using the URL :- Developer.Salesforce.com and get into the sign in button. If we already have an account in Salesforce Login using the **Username and the correct password.**

Developer Account will started load to display your account screen. While clicking Setup button it will display a screen as shown bellow:

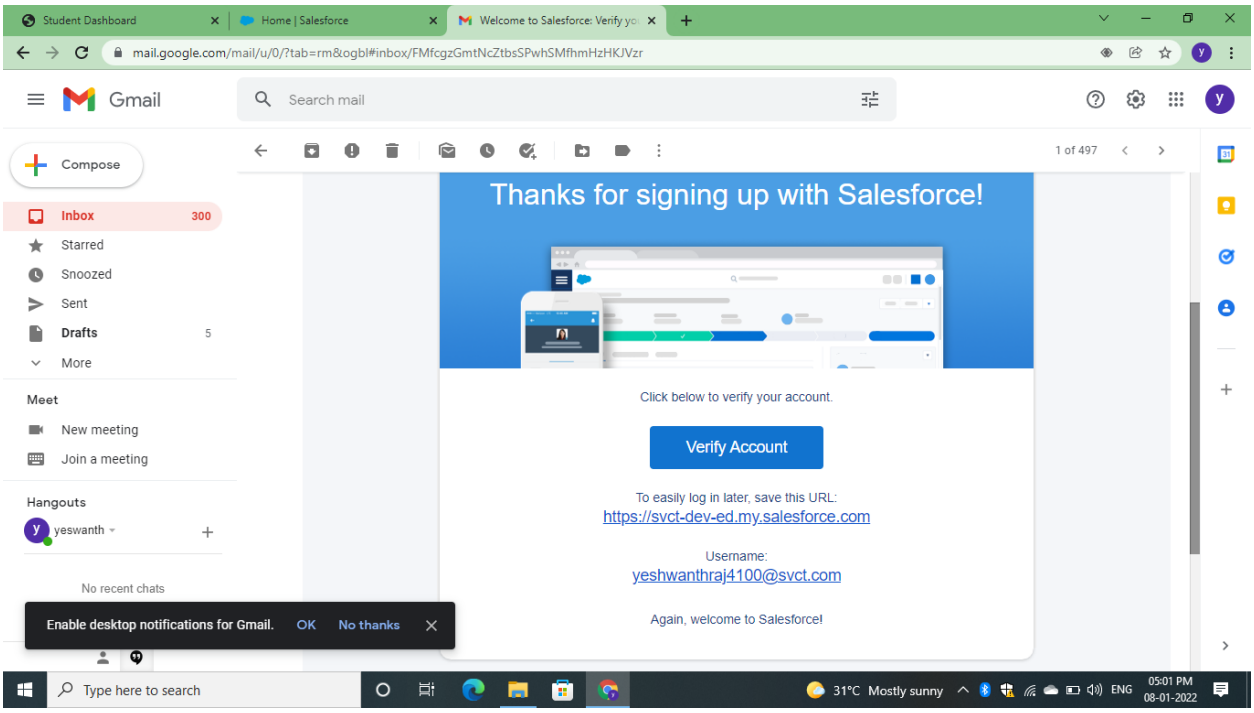
Creating a Developer Account :-



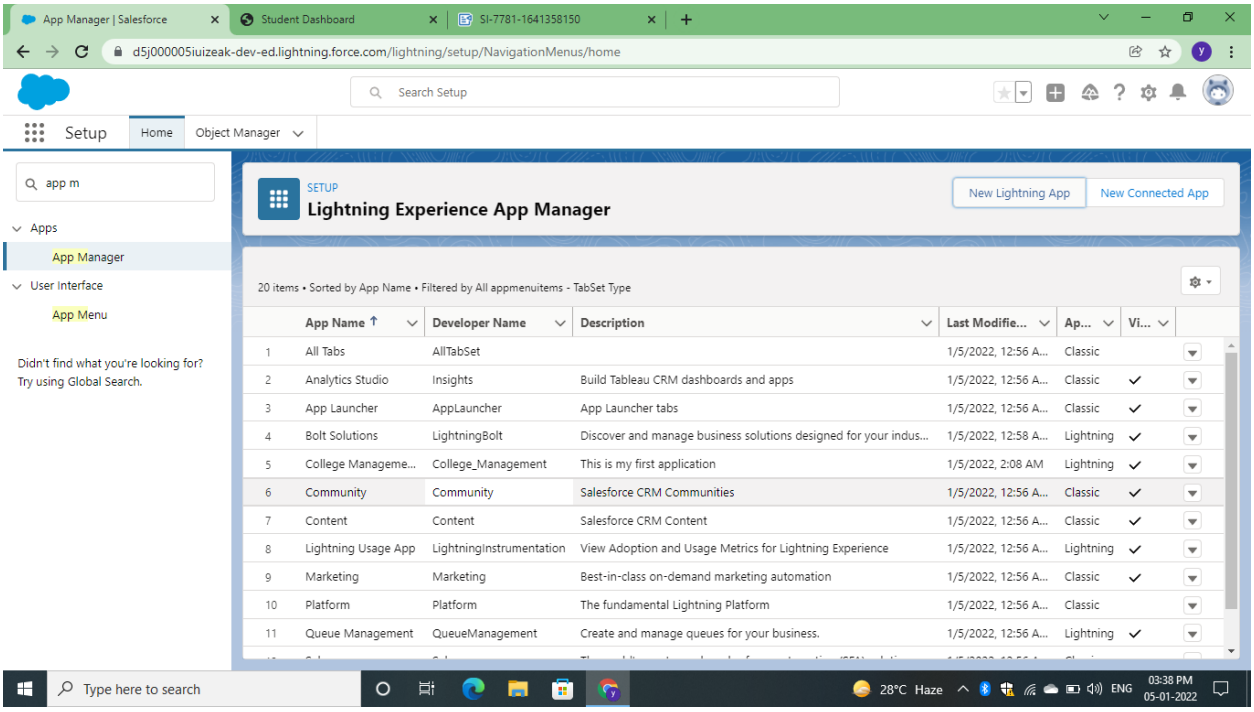
Object Manager :- It has two type of object are

a. Customer Object :- It allows the user to create their own object.
b. Standard Object :- It is standardly built by the salesforce.

Account Activation :-



Creating the Application called College management using App Managerr in New Lightning App**.** The screen shot is shown below :-



Day 2 :-

Topic :- Customer Object Creation

Milestone / Activity :-

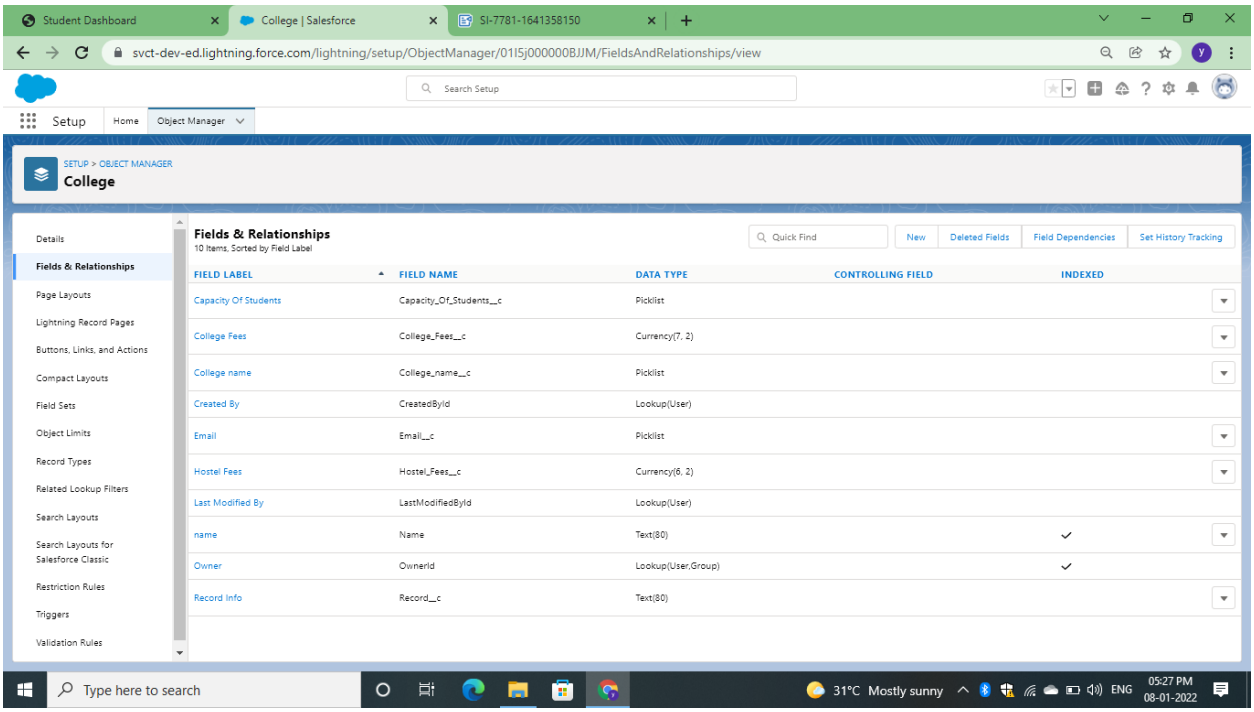
Creating the following object for the College Management Application

- ☆ College
- ☆ Application Form
- ☆ Student
- ☆ Subject

Detailed Description :-

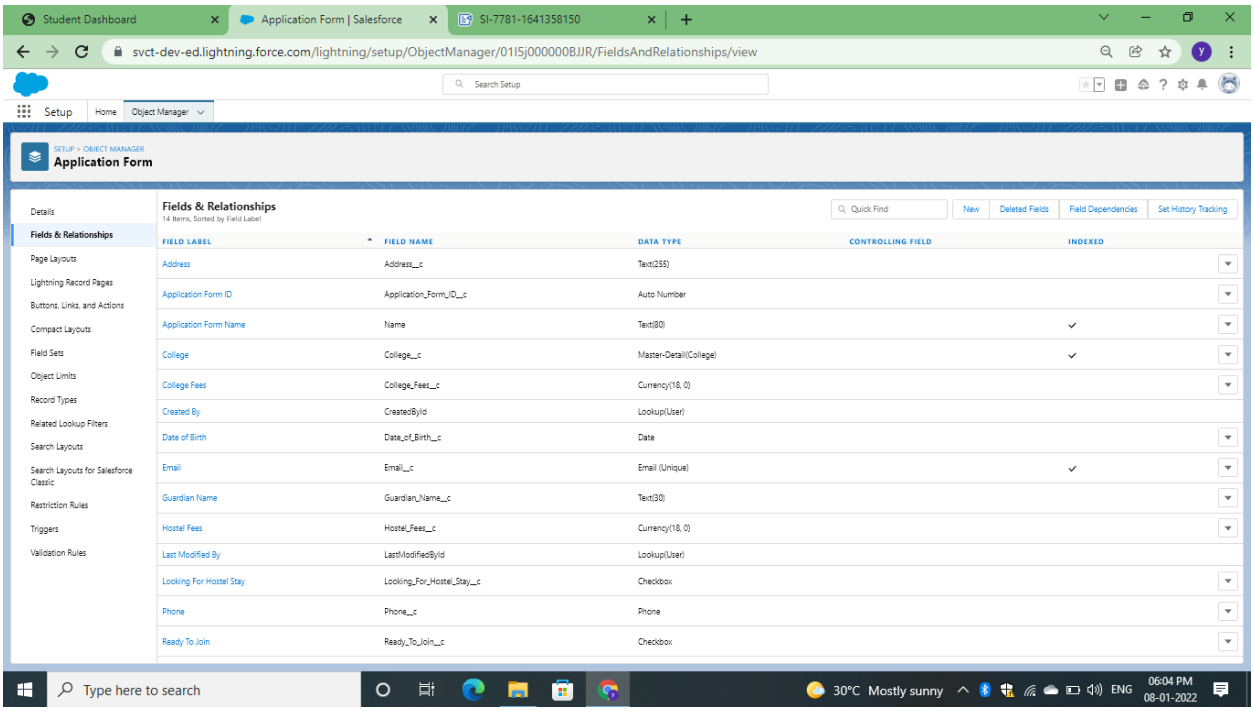
In College Object we creating the field according to the relationship in the managements.

- ☆ Record Info
- ☆ College Name
- ☆ Hostel Fees
- ☆ College Fees
- ☆ Email
- ☆ Capacity OfStudents



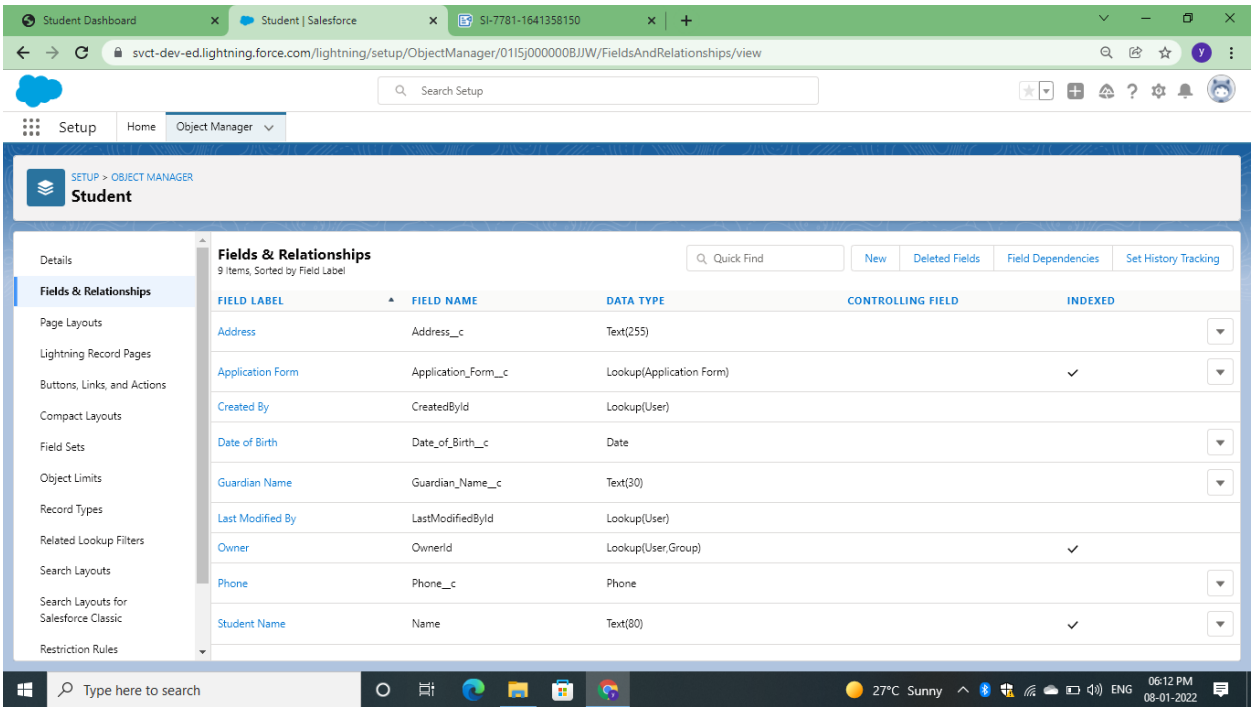
In Application Object :- Creating the field, as shown below

- ☆ Application Form ID and Looking For Hostel Stay
- ☆ Address and Phone
- ☆ College and College Fees and Hostel Fees
- ☆ Date Of Birth and Email
- ☆ Guardian Name and Student Name



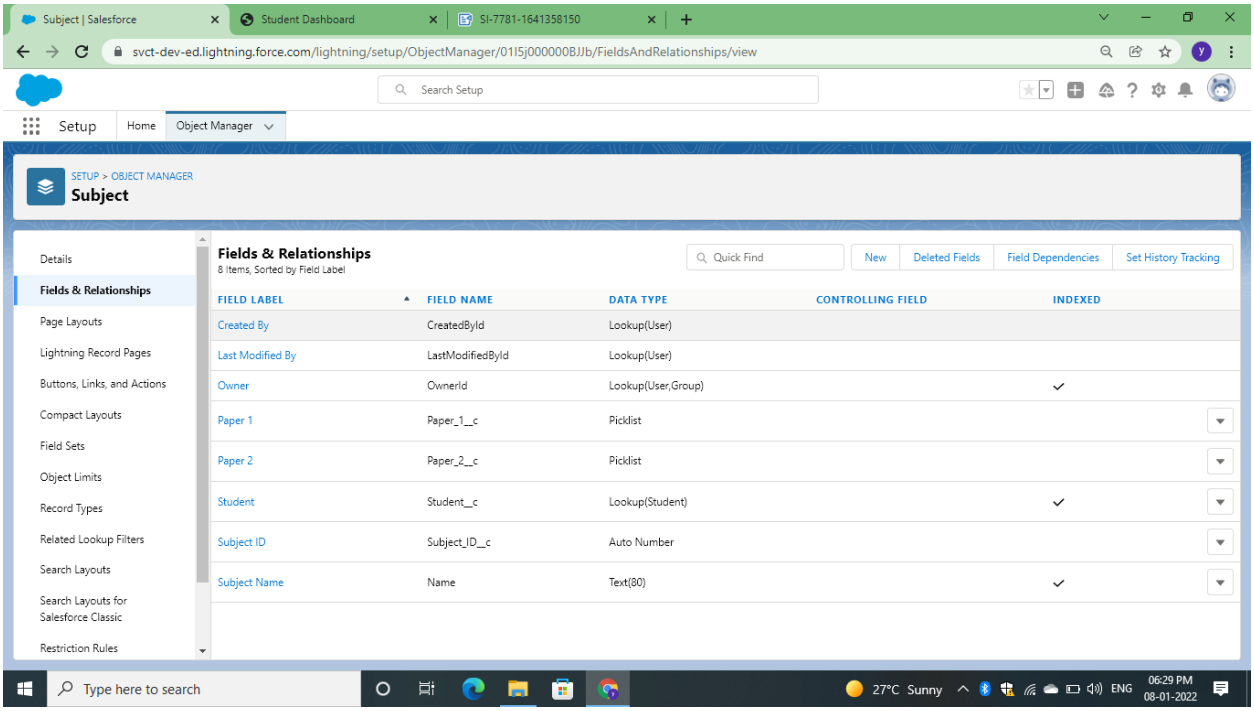
Creating the field on Student Object :-

- ☆ Student Name and Phone
- ☆ Application Form abd Guardian Name
- ☆ Address and Date Of Birth



Creating the field on Subject Object : -

- ☆ Student and Student ID
- ☆ Paper 1 and Paper 2



Adding the Business Logic to Application

1.Creating the Global Picklist Value Sets :-

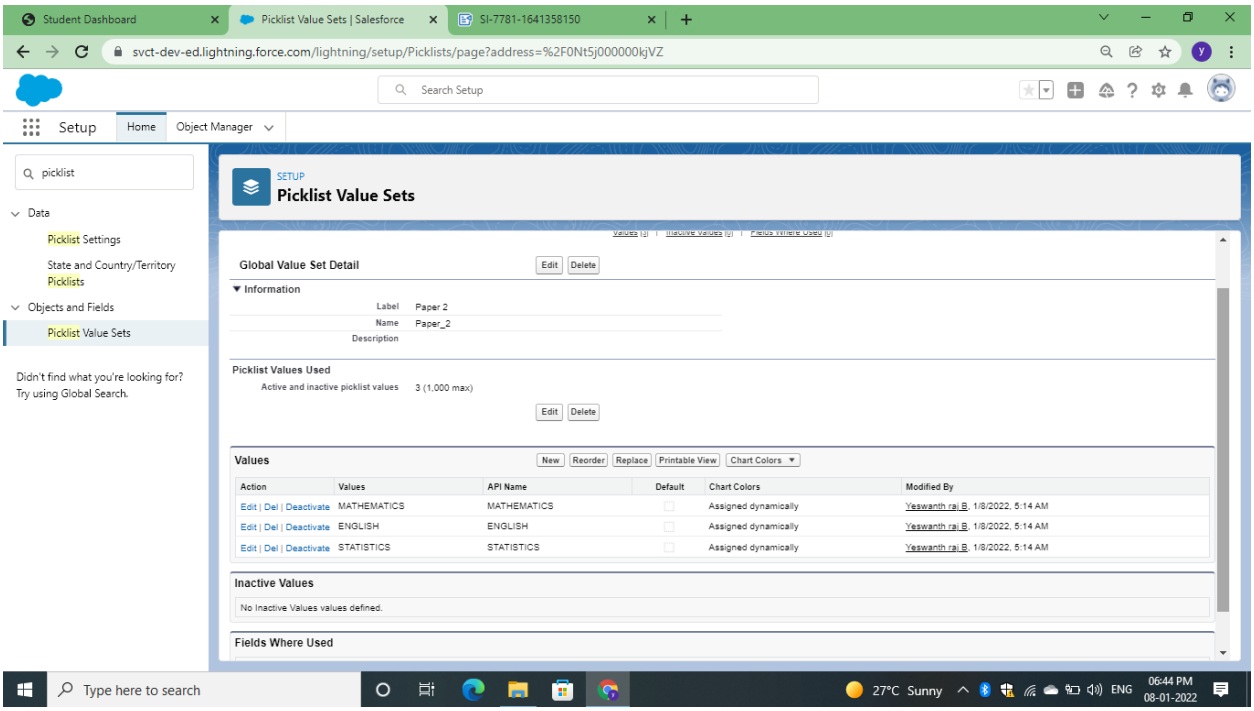
We creating the global picklist value set in the project called College Management. Its used to share values across object and the customer fields and to restrict the picklist to only the values that you specify.**(1.1)**

2.Creating the Field dependencies : -

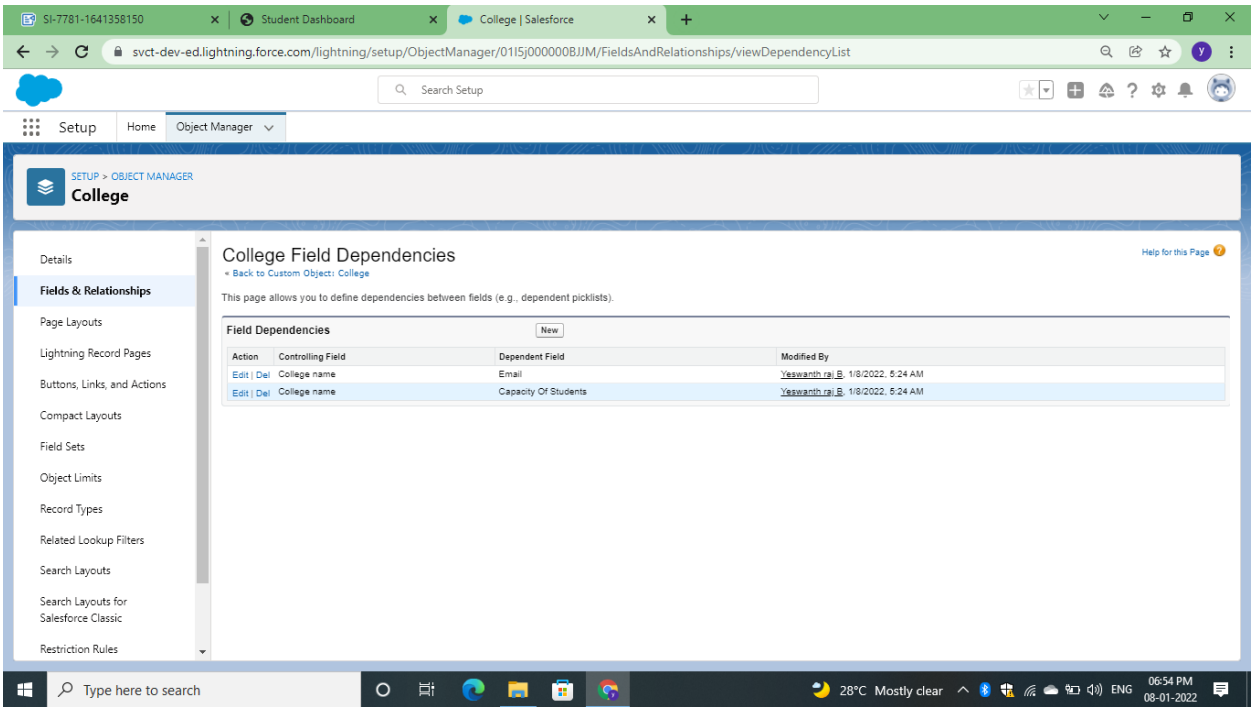
That allows us to change the content of a picklist based on the value of another field. The bellow screenshot Shows th field dependencies **(2.1)**

3. Validation Rule :-

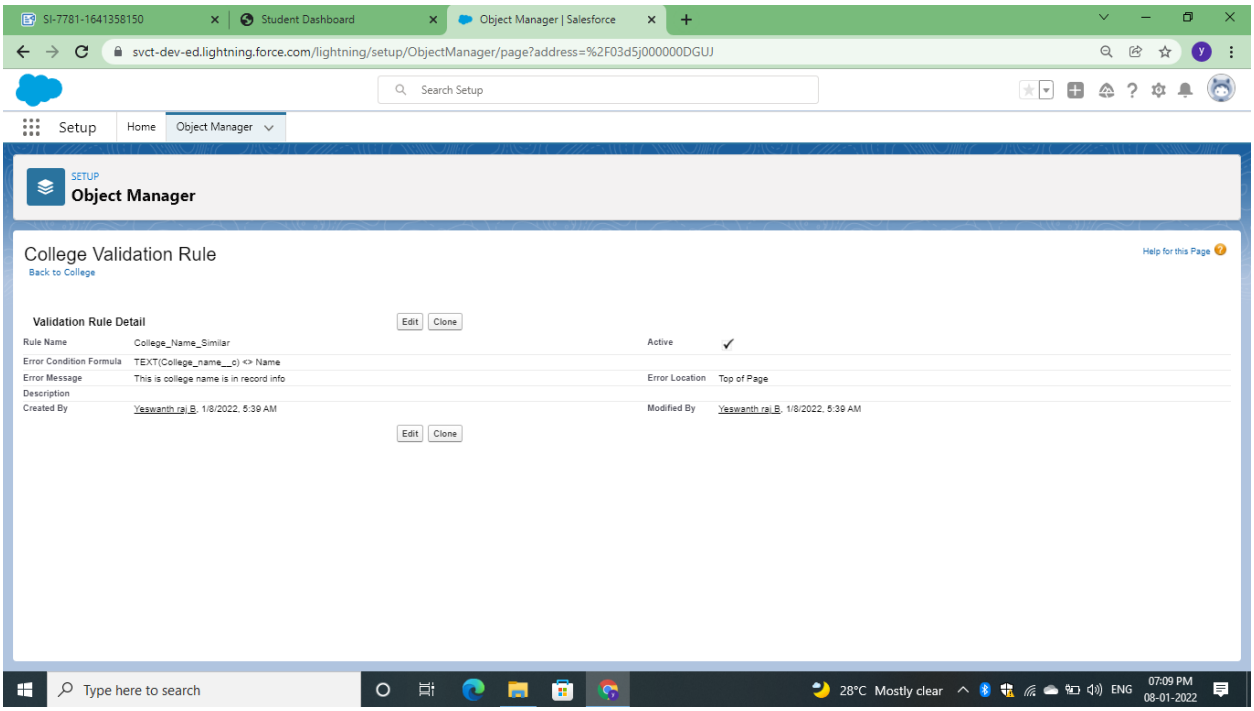
- ☆ **College Object (3.1)**
- ☆ **Application Form Object (3.2)**



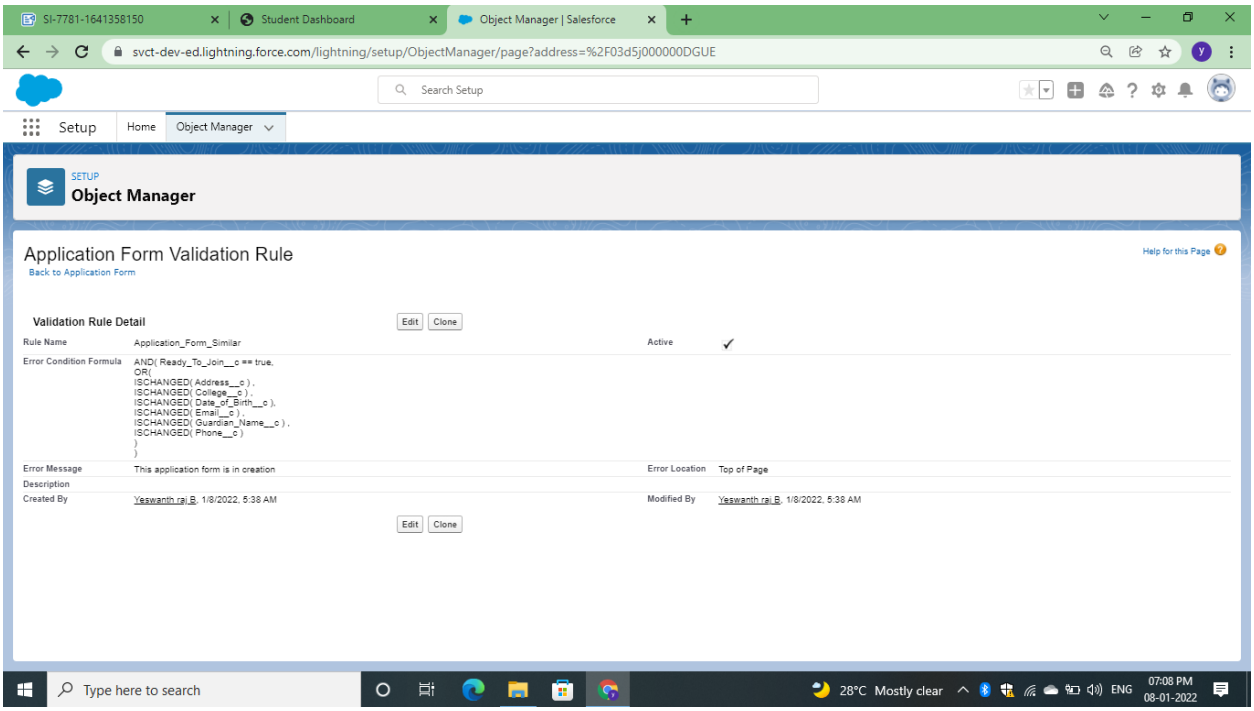
Global Picklist value Sets (1.1)



Field Dependencies (1.2)



Validation Rule - College Object (3.1)



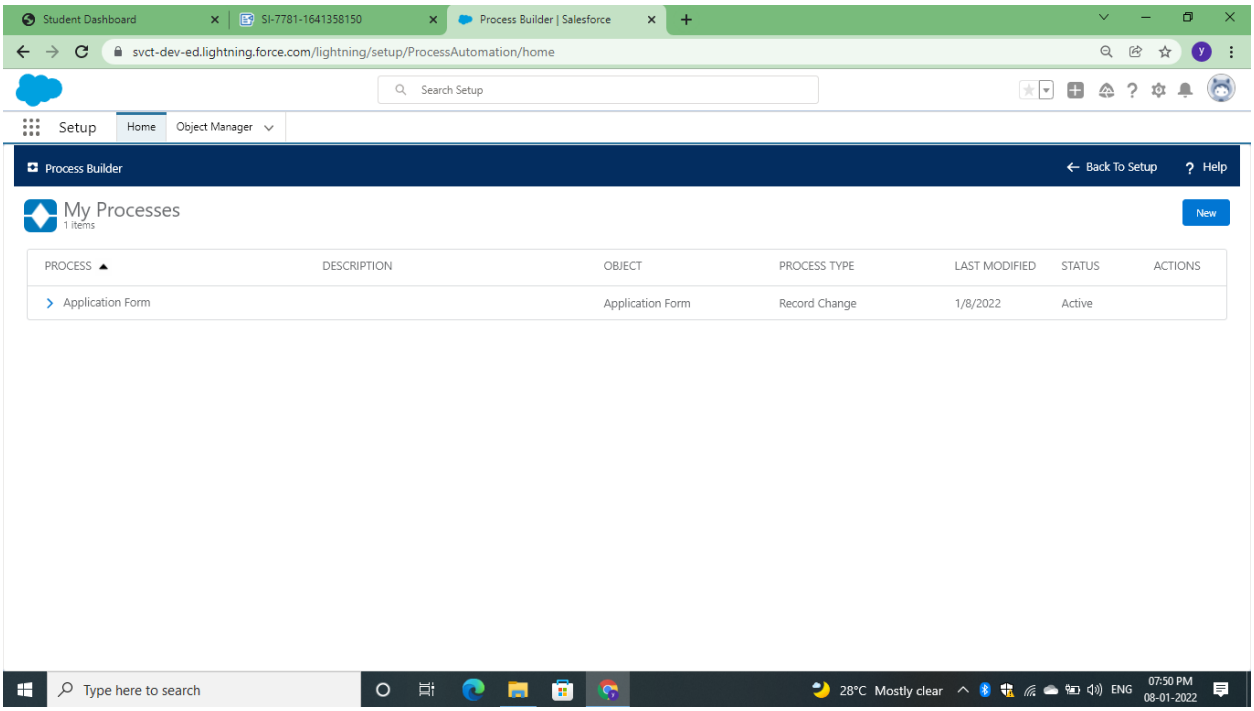
Validation Rule - Application Form Object (3.2)

Day 3

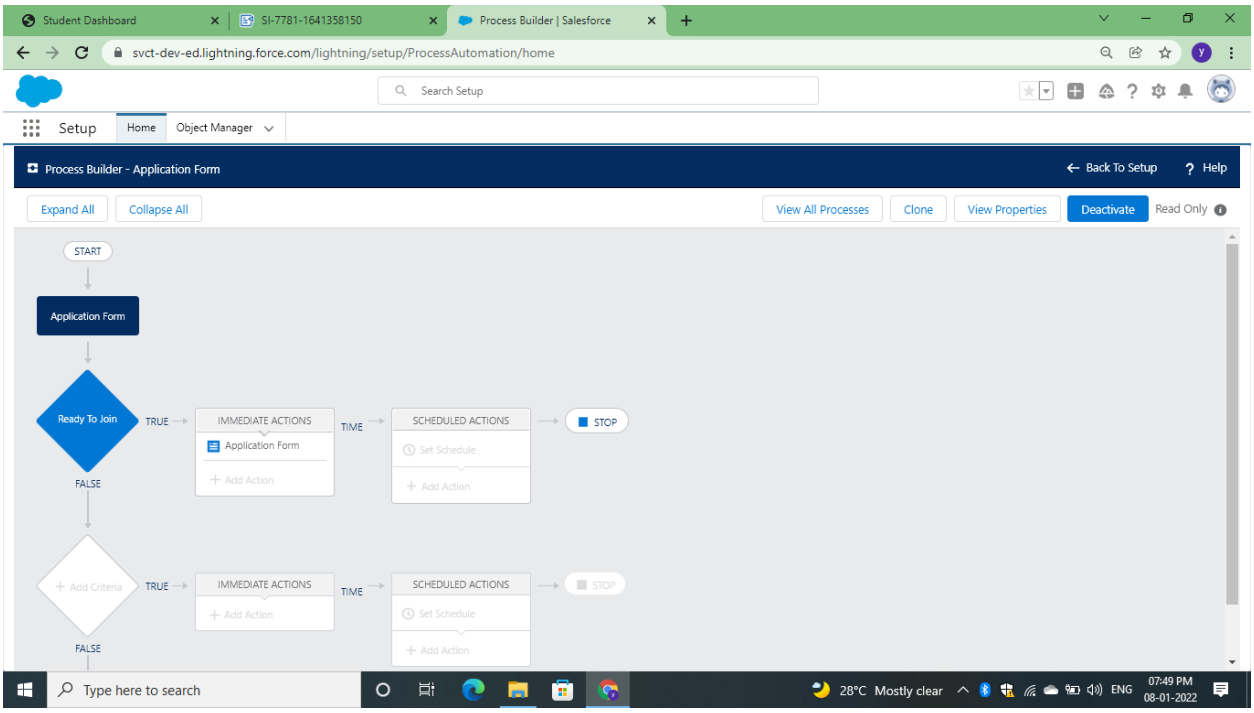
Topic :- Process Automation :-

Milestone / Activity :-

Process Builder is the tool for admins and also capable of performing everything workflow with the exception of sending outbound message - message to another person.Better example is that we doing the College Management Application.



Go to setup ---> in quick finder search **Process Builder** ---> click process builder then ---> click new ---> Create a process builder on "**Application Form**" process start when a record changeh

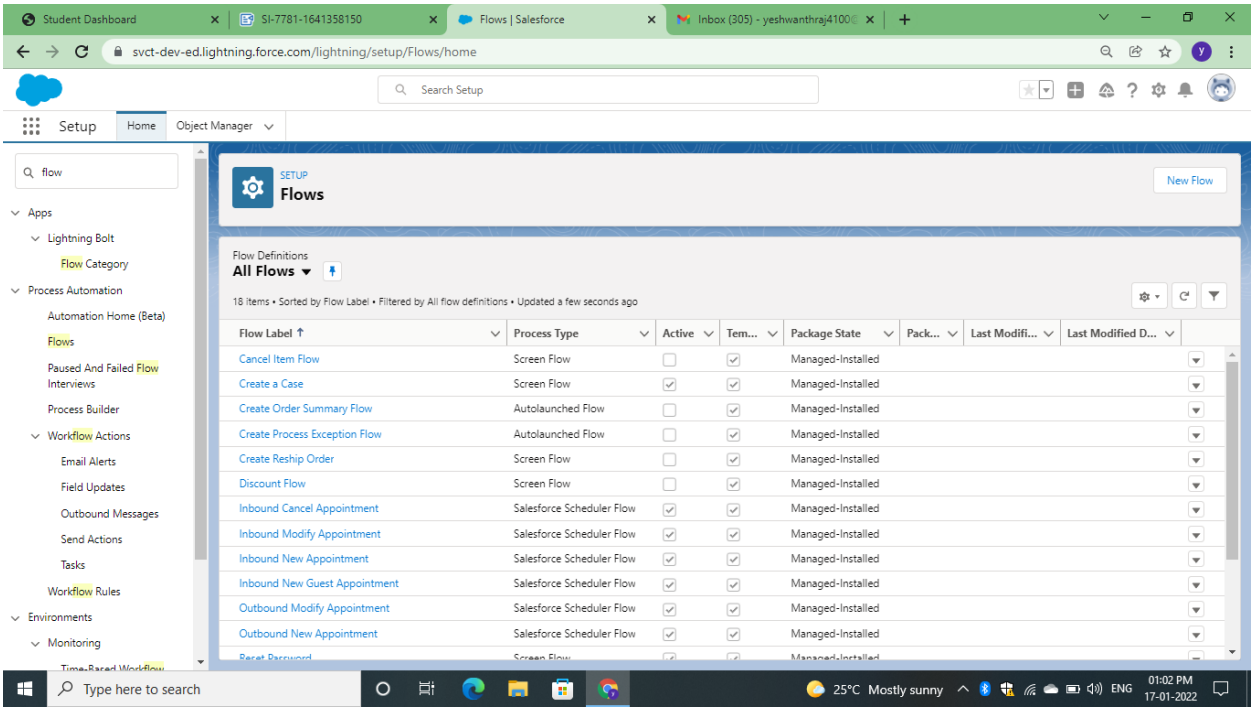


Day 4

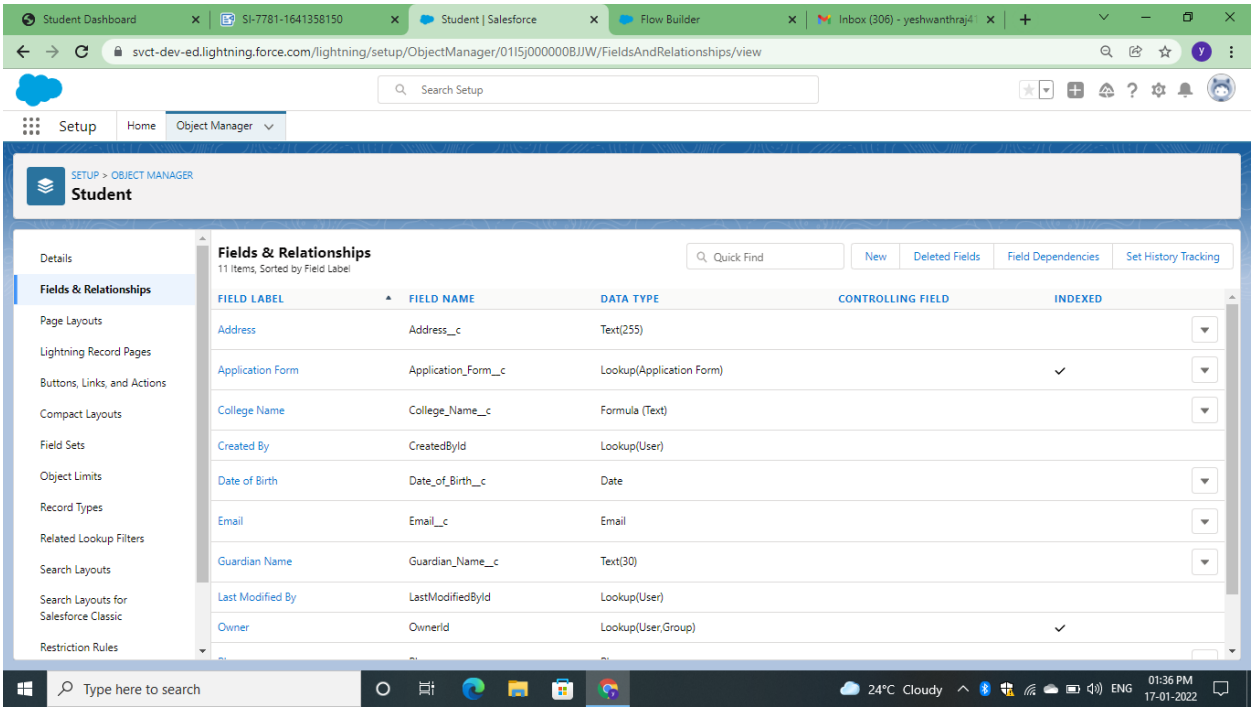
Topic :- Flow builder

Milestone / Activity :-

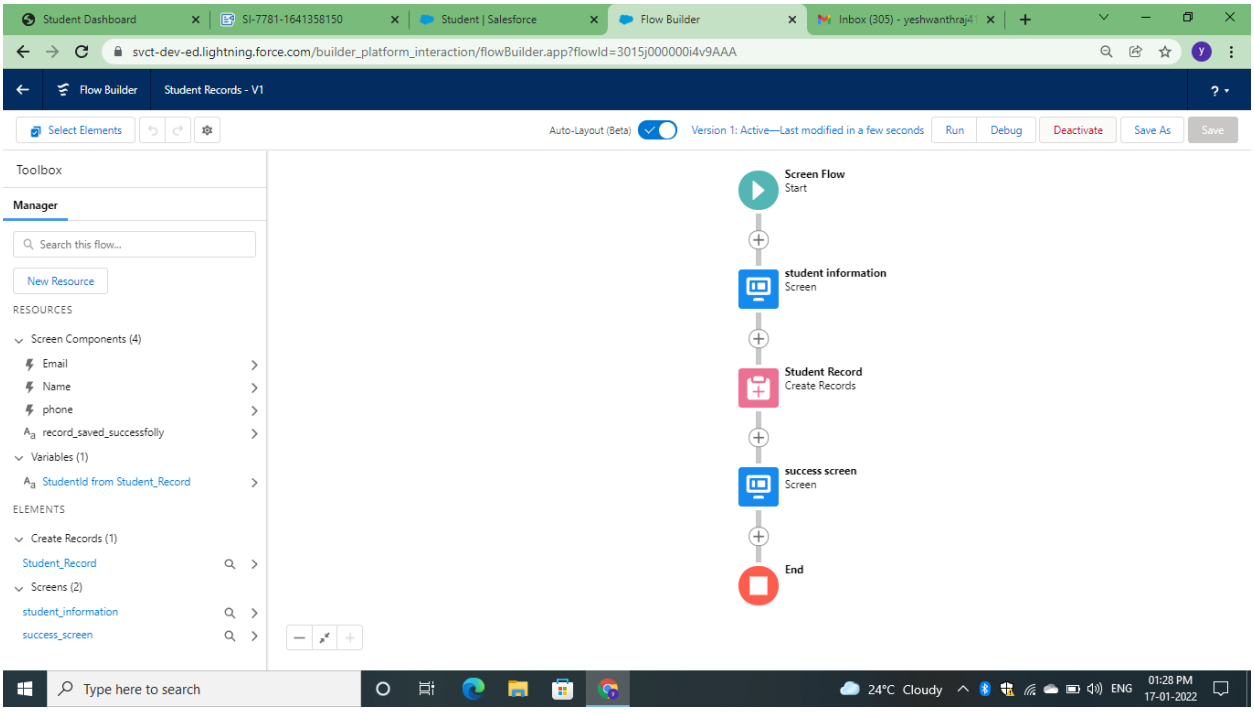
1. Navigate to setup --> Search flows in **Quick Finder** --> **click on New Flow.**



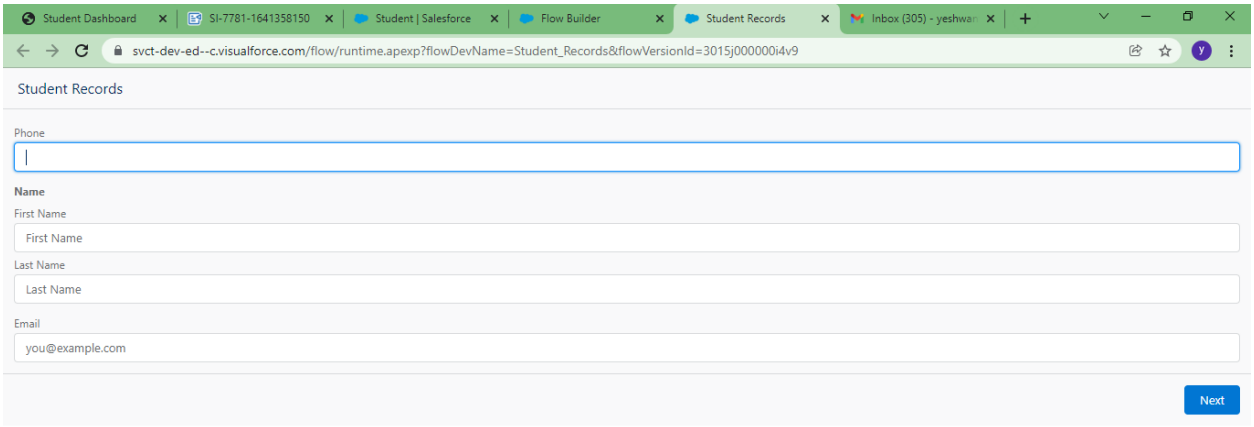
2. In object manage, we already created the customer object **Student** with phone, email, name and etc.



Student Record is created using element called **Create Record.** Next creating the screen name the element has Student Information and Success Screen.



Output :-



Day 5 :- Apex

Topic :- Developer console

Milestone / Activity :-

1. **querying from child to parent**

select Student_Name __c, Gaudian_Name__c, Date_of_Birth__c from Application_Form__c

2. **querying from parent to child**

select id, College_Name__c, (select id, Student_Name __c, Gaudian_Name__c, Date_of_Birth__c from Application_Form__c)

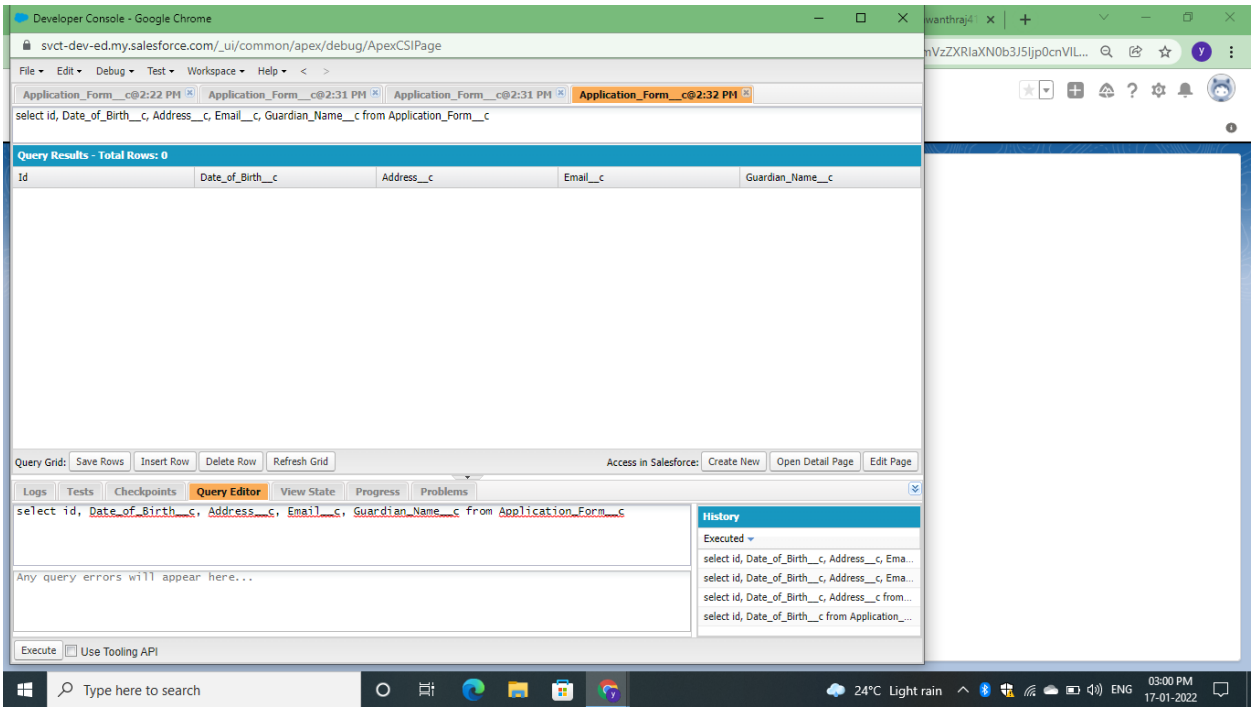
Negiotation to setup --> click Developer Console

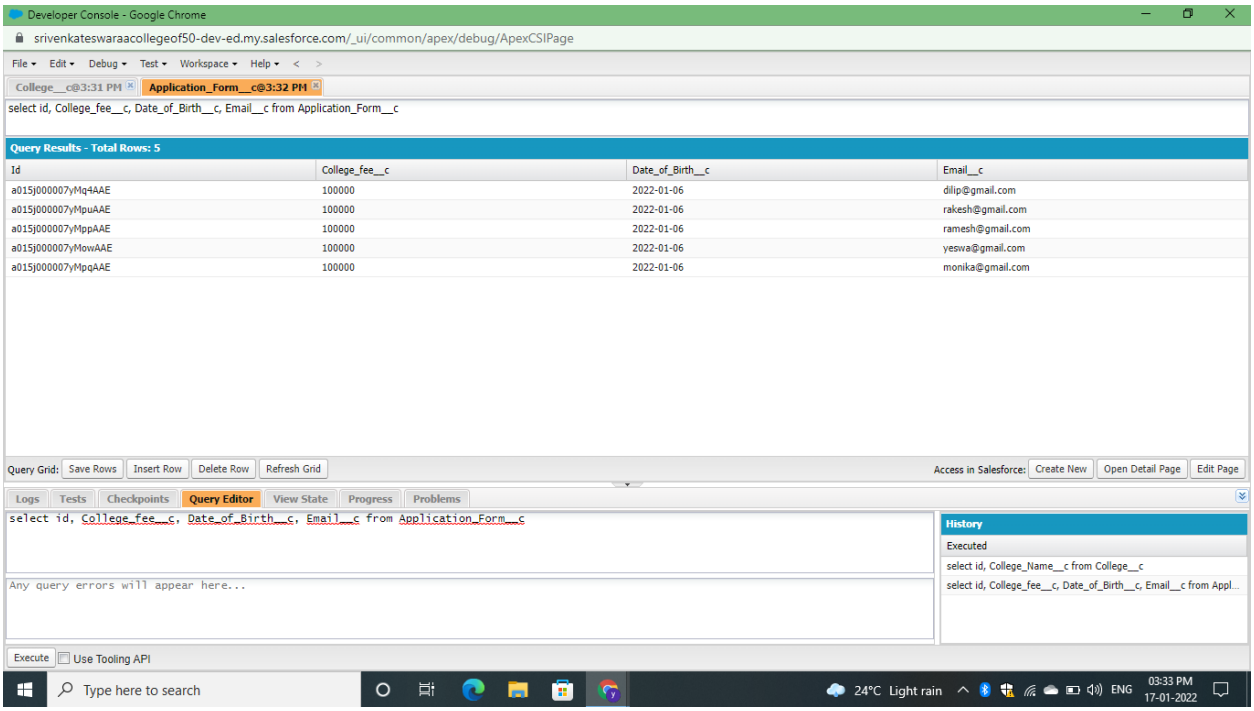
Apex Trigger that will execute beforce or after the following operation

- ☆ insert
- ☆ update
- ☆ delete

Before an object record are inserted into the database, after record have been deleted or even after a reord is restored from the recycle bin. There are two type of triggers - before and after trigger. In the developer Console , click File | new | apex trigger.

Then enter College Management - trigger management


Developer Console - Google Chrome
svct-dev-ed.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help
Application_Form__c@2:22 PM
Application_Form__c@2:31 PM
Application_Form__c@2:31 PM
Application_Form__c@2:32 PM
select id, Date_of_Birth__c, Address__c, Email__c, Guardian_Name__c from Application_Form__c
Query Results - Total Rows: 0
Id
Date_of_Birth__c
Address__c
Email__c
Guardian_Name__c
Query Grid: Save Rows Insert Row Delete Row Refresh Grid
Access in Salesforce: Create New Open Detail Page Edit Page
Logs Tests Checkpoints Query Editor View State Progress Problems
select id, Date_of_Birth__c, Address__c, Email__c, Guardian_Name__c from Application_Form__c
Any query errors will appear here...
History
Executed
select id, Date_of_Birth__c, Address__c, Ema...
select id, Date_of_Birth__c, Address__c, Ema...
select id, Date_of_Birth__c, Address__c from...
select id, Date_of_Birth__c from Application_...
Execute
Use Tooling API
Type here to search
24°C Light rain
ENG
03:00 PM
17-01-2022


Developer Console - Google Chrome
srivenkateswaraacollegeof50-dev-ed.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File Edit Debug Test Workspace Help
College__c@3:31 PM
Application_Form__c@3:32 PM
select id, College_fee__c, Date_of_Birth__c, Email__c from Application_Form__c
Query Results - Total Rows: 5
Id
College_fee__c
Date_of_Birth__c
Email__c
a015j000007yMq4AAE 100000 2022-01-06 dilip@gmail.com
a015j000007yMpuAAE 100000 2022-01-06 rakesh@gmail.com
a015j000007yMppAAE 100000 2022-01-06 ramesh@gmail.com
a015j000007yMowAAE 100000 2022-01-06 yeswa@gmail.com
a015j000007yMpqAAE 100000 2022-01-06 monika@gmail.com
Query Grid: Save Rows Insert Row Delete Row Refresh Grid
Access in Salesforce: Create New Open Detail Page Edit Page
Logs Tests Checkpoints Query Editor View State Progress Problems
select id, College_fee__c, Date_of_Birth__c, Email__c from Application_Form__c
Any query errors will appear here...
History
Executed
select id, College_Name__c from College__c
select id, College_fee__c, Date_of_Birth__c, Email__c from Appl...
Execute
Use Tooling API
Type here to search
24°C Light rain
ENG
03:33 PM
17-01-2022

Day 6 :- Batch Apex

Topic :- start(), execute() and finish()

Milestone / activity :-

Basic syntax :-

```
public class ApplicationBatchTest implements Database.Batchable<sobject>
{
//start(), execute(), finish().
public Database.QueryLocator start (Database.BatchableCContext bc) {
        }
public void exeute (Database.BatchableContext bc, List<sobjects>listobj) {
        }
public void finish(Datbase.Batch ableContext bc) {
        }
}
```

Using batch apex, you can process records asynchronously in batches to stay within platform limits.

Start()

Used to collect the records or object to be passed to the interface method execute forprocessing. It is the begining of a batch apex job.

Execute()

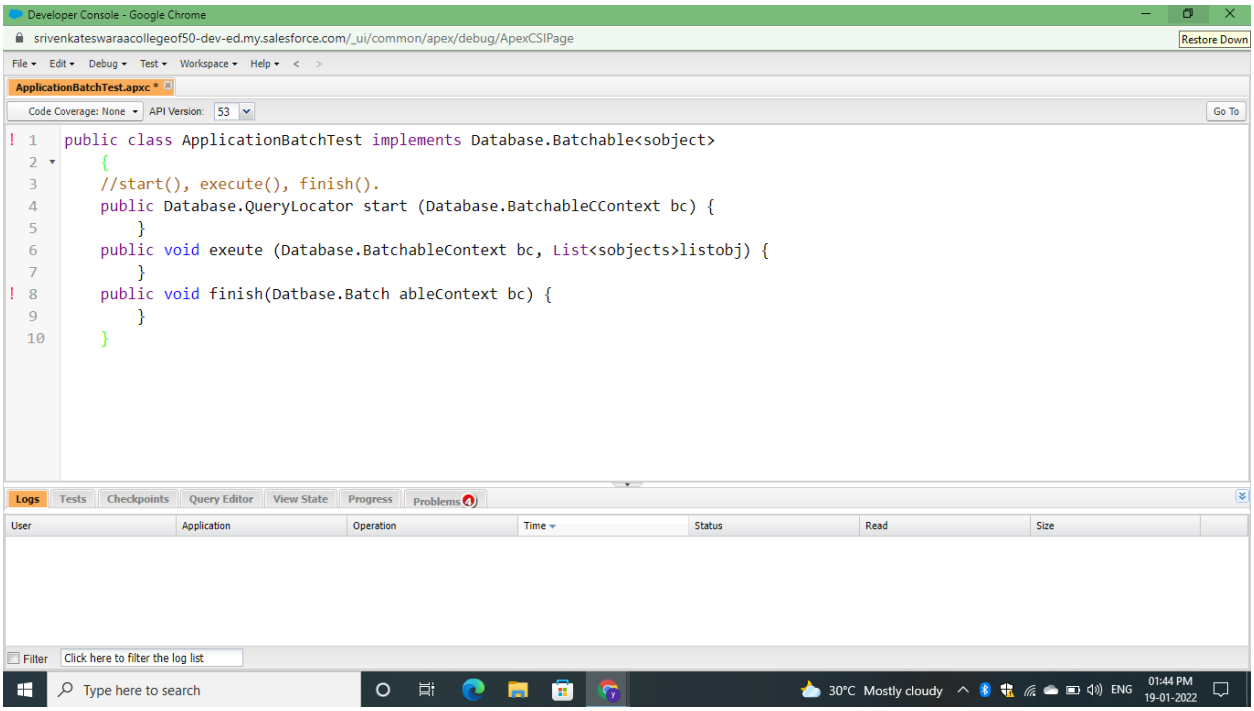
It perform the actual processing for each batch of data passed to the method.

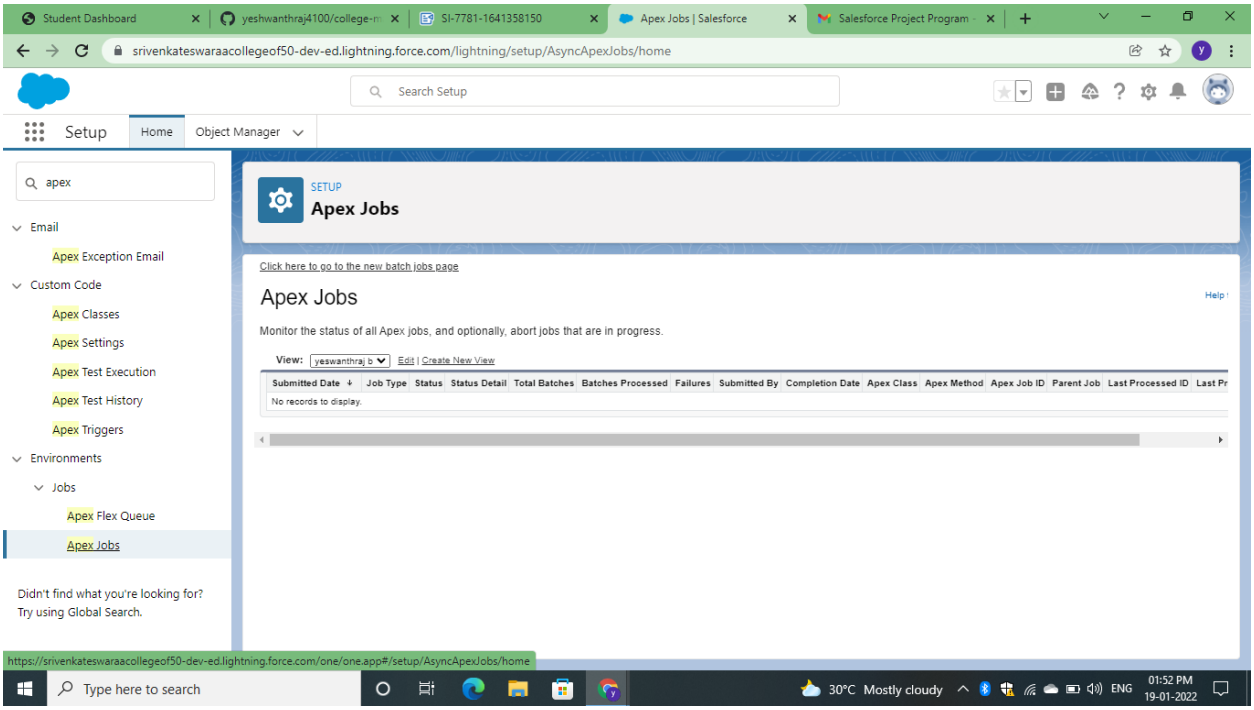
finish()

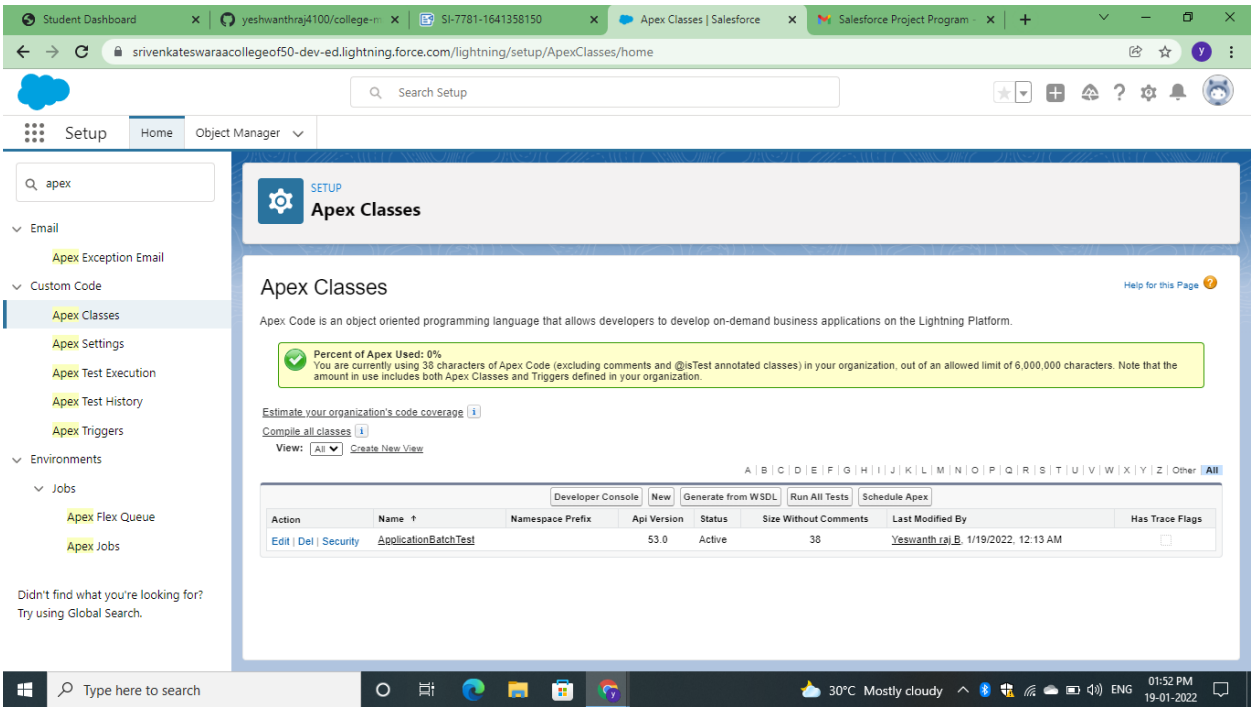
it allow to execute the post processing operation and onces the batch has been process, it will called.

The salesforce provided the Database.Batchable and then invoke. From Setup, enter Apex jobs in the Quick Finder box, then select Apex Jobs.

Apex Classes :- its used to implement the actions associated with an object.


Developer Console - Google Chrome
srivenkateswaraacollegeof50-dev-ed.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
Restore Down
File Edit Debug Test Workspace Help
ApplicationBatchTest.apxc *
Code Coverage: None
API Version: 53
Go To
1 public class ApplicationBatchTest implements Database.Batchable<sobject>
2 {
3 //start(), execute(), finish().
4 public Database.QueryLocator start (Database.BatchableCContext bc) {
5 }
6 public void exeute (Database.BatchableContext bc, List<objects>listobj) {
7 }
8 public void finish(Datbase.Batch ableContext bc) {
9 }
10 }
Logs
Tests
Checkpoints
Query Editor
View State
Progress
Problems
User
Application
Operation
Time
Status
Read
Size
Filter
Click here to filter the log list
Type here to search
30°C Mostly cloudy
ENG
01:44 PM
19-01-2022


Student Dashboard
yeshwanthraj4100/college-
SI-7781-1641358150
Apex Jobs | Salesforce
Salesforce Project Program
srivenkateswaraacollegeof50-dev-ed.lightning.force.com/lightning/setup/AsyncApexJobs/home
Search Setup
Setup
Home
Object Manager
apex
Email
Apex Exception Email
Custom Code
Apex Classes
Apex Settings
Apex Test Execution
Apex Test History
Apex Triggers
Environments
Jobs
Apex Flex Queue
Apex Jobs
Didn't find what you're looking for?
Try using Global Search.
SETUP
Apex Jobs
Click here to go to the new batch jobs page
Apex Jobs
Help
Monitor the status of all Apex jobs, and optionally, abort jobs that are in progress.
View:
yeswanthraj b
Edit | Create New View
Submitted Date
Job Type
Status
Status Detail
Total Batches
Batches Processed
Failures
Submitted By
Completion Date
Apex Class
Apex Method
Apex Job ID
Parent Job
Last Processed ID
No records to display.
https://srivenkateswaraacollegeof50-dev-ed.lightning.force.com/one/one.app#/setup/AsyncApexJobs/home
Type here to search
30°C Mostly cloudy
ENG
01:52 PM
19-01-2022



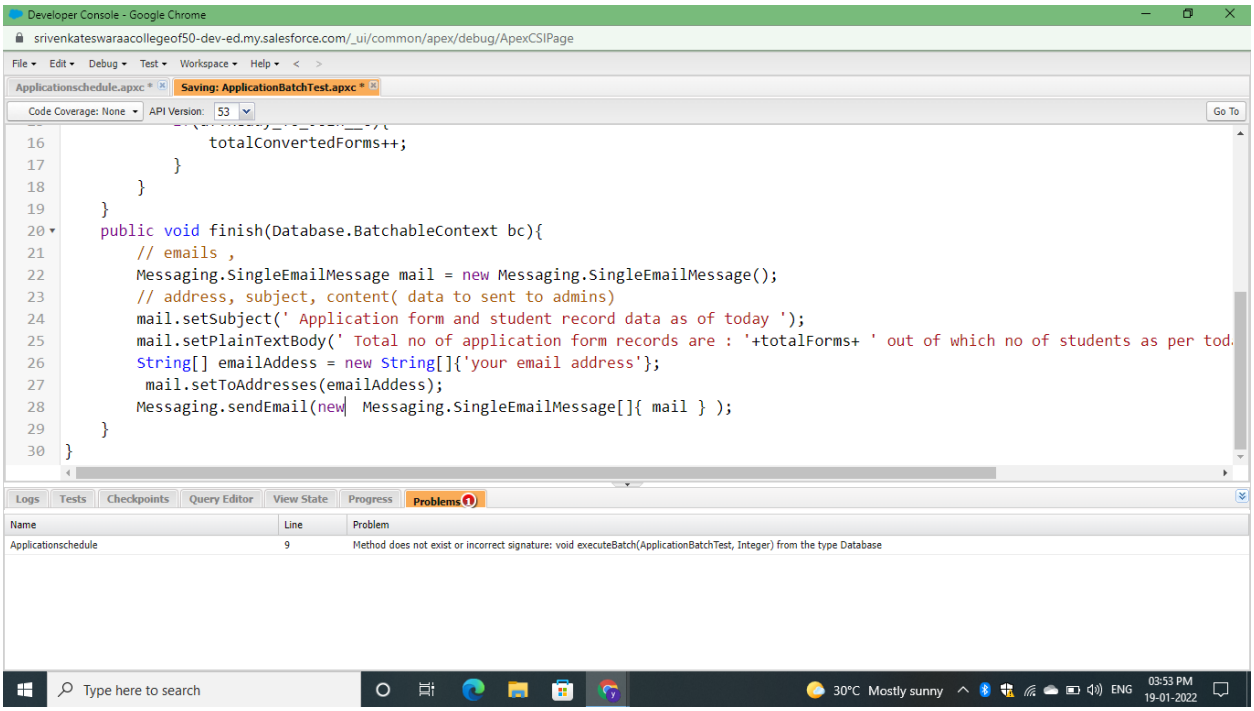
Day 7 :-

Topic :- Schedular Classes

Milestone / Activity :-

The scheduler runs as system—all classes are executed, whether or not the user has permission to execute the class.

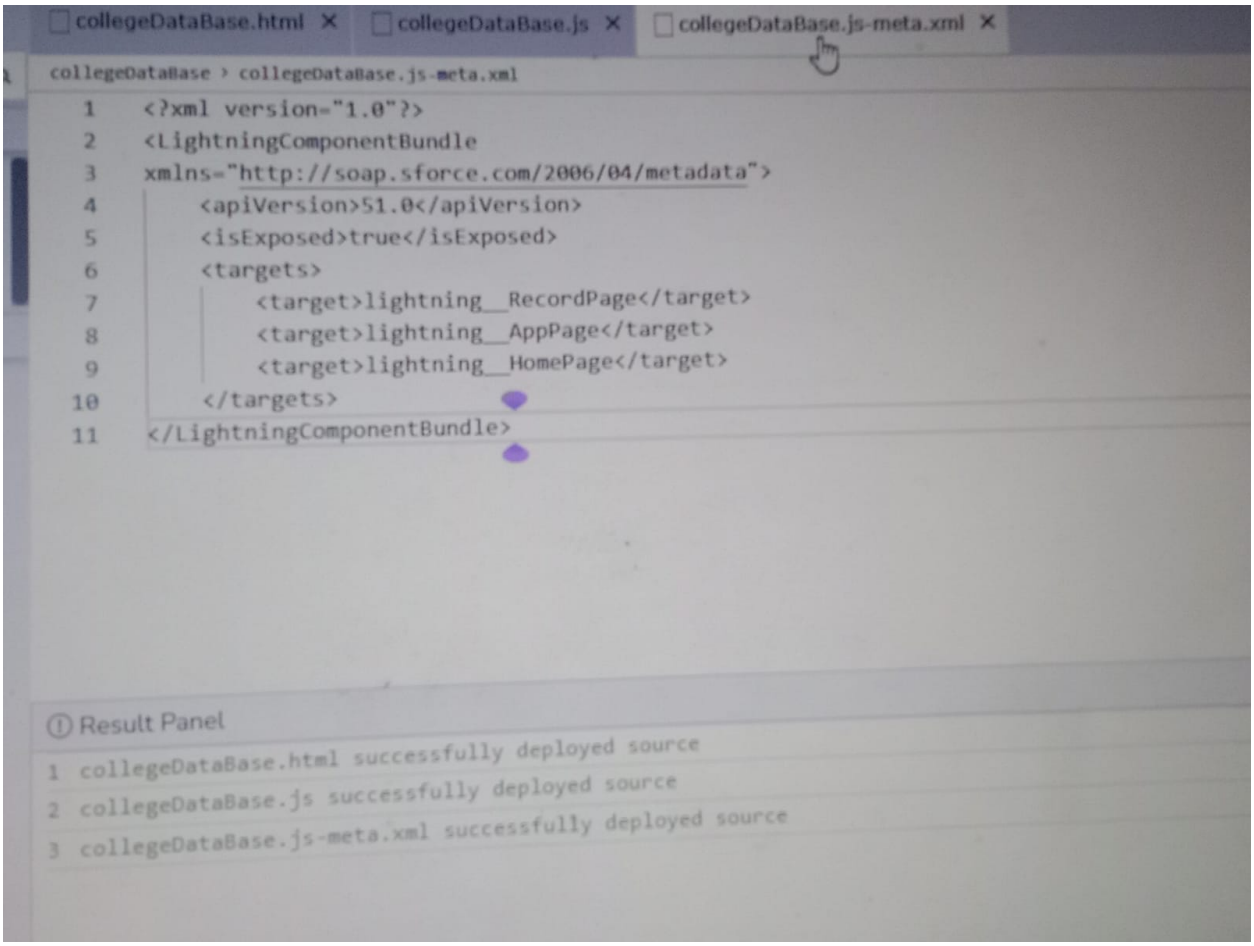
```
global void execute(SchedulableContext sc)
{
}
```

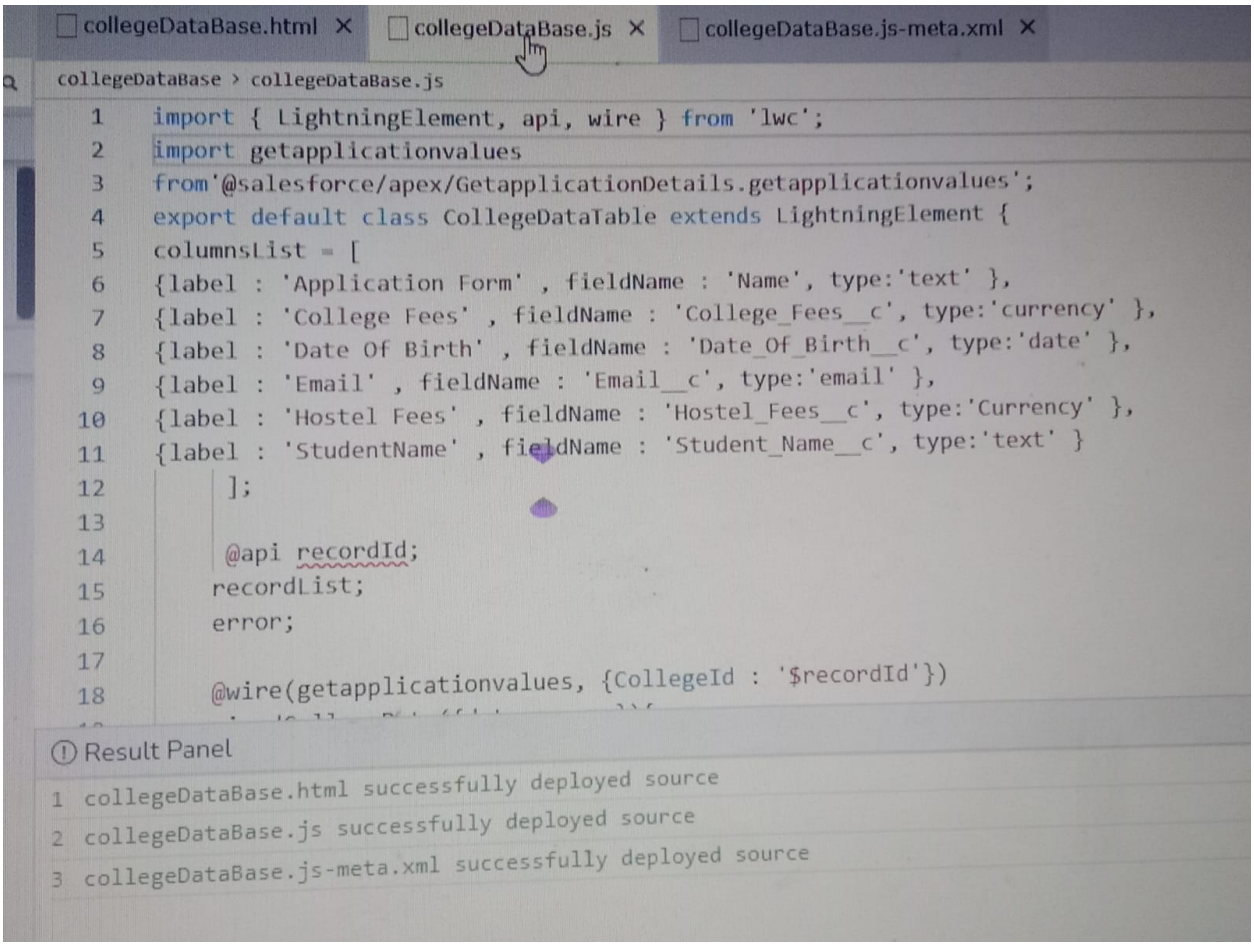


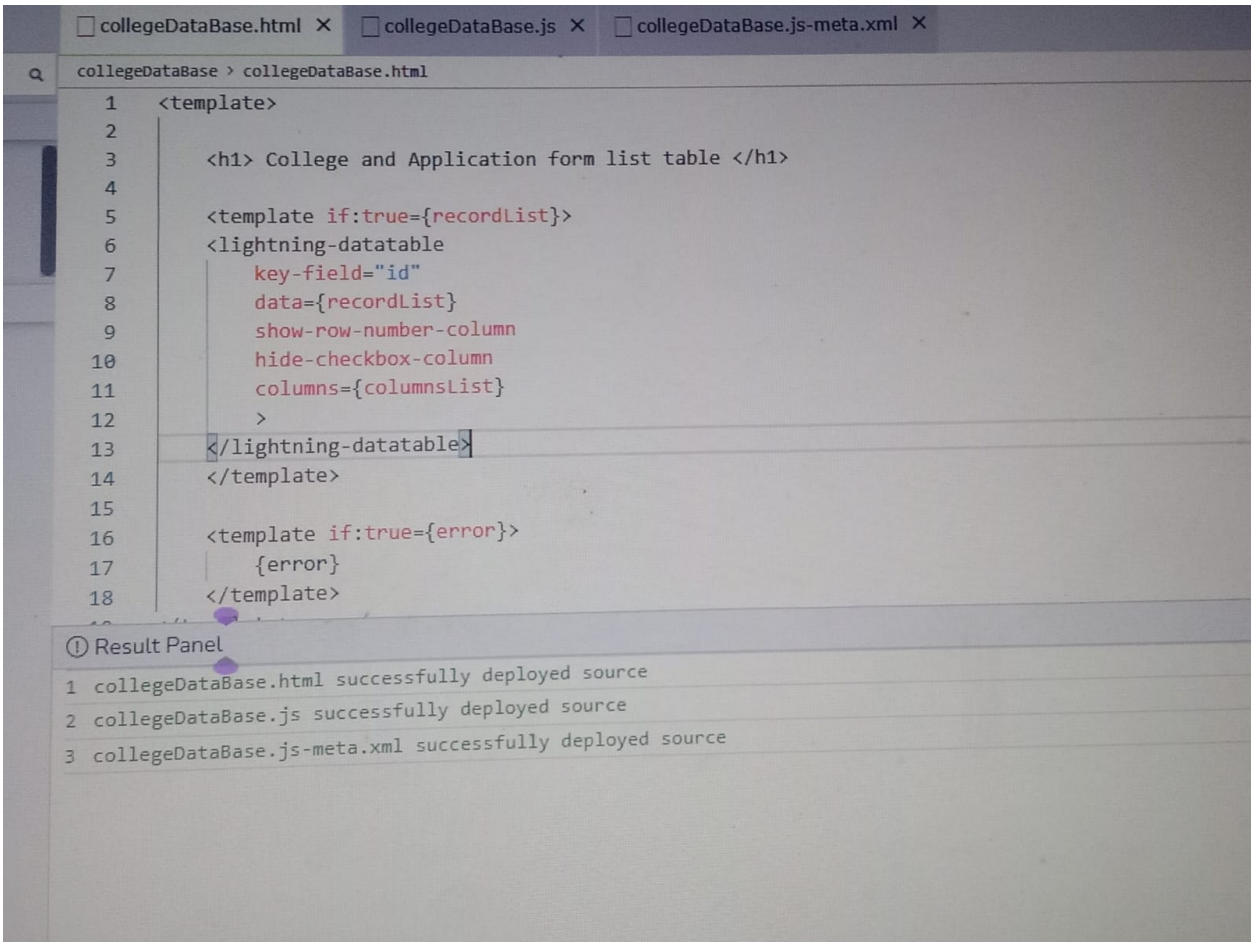
Day 8, 9, 10, 11:-

Topic :- Lightning web comjponents

- **Creating College Data Table Component**
  ==>Apex Class
  ==>HTML File
  ==>Javascript (Js) File
  ==>Meta File


collegeDataBase.html
collegeDataBase.js
collegeDataBase.js-meta.xml
collegeDataBase › collegeDataBase.js-meta.xml
1 <?xml version="1.0"?>
2 <LightningComponentBundle
3 xmlns="http://soap.sforce.com/2006/04/metadata">
4 <apiVersion>51.0</apiVersion>
5 <isExposed>true</isExposed>
6 <targets>
7 <target>lightning__RecordPage</target>
8 <target>lightning__AppPage</target>
9 <target>lightning__HomePage</target>
10 </targets>
11 </LightningComponentBundle>
Result Panel
1 collegeDataBase.html successfully deployed source
2 collegeDataBase.js successfully deployed source
3 collegeDataBase.js-meta.xml successfully deployed source


collegeDataBase.html
collegeDataBase.js
collegeDataBase.js-meta.xml
collegeDataBase > collegeDataBase.js
import { LightningElement, api, wire } from 'lwc';
import getapplicationvalues
from'@salesforce/apex/GetapplicationDetails.getapplicationvalues';
export default class CollegeDataTable extends LightningElement {
columnsList = [
{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' },
{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'})
Result Panel
1 collegeDataBase.html successfully deployed source
2 collegeDataBase.js successfully deployed source
3 collegeDataBase.js-meta.xml successfully deployed source


collegeDataBase.html
collegeDataBase.js
collegeDataBase.js-meta.xml
collegeDataBase > collegeDataBase.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}
>
</lightning-datatable>
</template>
<template if:true={error}>
{error}
</template>
Result Panel
1 collegeDataBase.html successfully deployed source
2 collegeDataBase.js successfully deployed source
3 collegeDataBase.js-meta.xml successfully deployed source