

## Day 1 :

**Topic:**Account creation and logging in

**Milestone / Activities:**creating salesforce developer org to get started,account activation and login to salesforce account

**Detailed Description:** To create a salesforce developer we have to give all the details like name,company ,role and username etc, and click signme up

**Screenshot the Milestone / Activities :**

Student Dashboard x SI-7748-1641227923 x Recently Viewed | Application x Want a Direct Ticket to Salesfo x Developer Edition Signup | Sai x

developer.salesforce.com/signup

**Build enterprise-quality apps fast to bring your ideas to life**

- Build apps fast with drag and drop tools
- Customize your data model with clicks
- Go further with Apex code
- Integrate with anything using powerful APIs
- Stay protected with enterprise-grade security
- Customize UI with clicks or any leading-edge web framework

**Sign up for your Salesforce Developer Edition**  
A full-featured copy of the Platform, for free

First Name\* Sai kishore Last Name\* pallantla

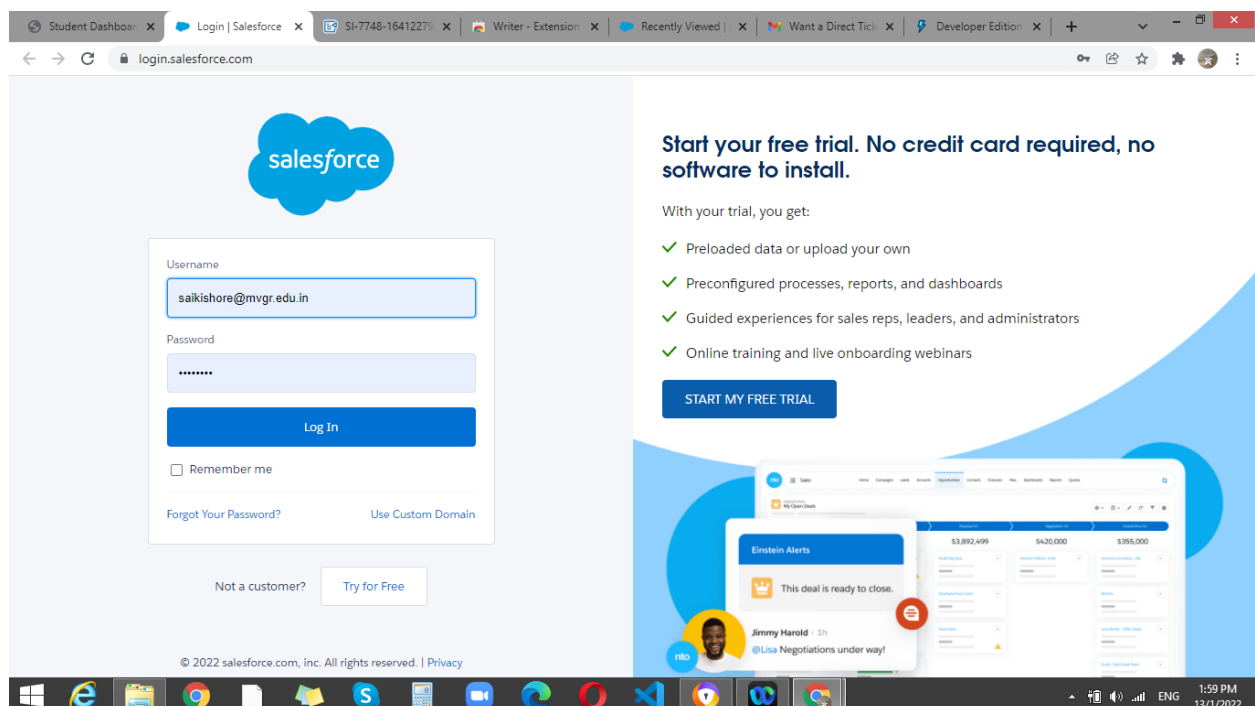
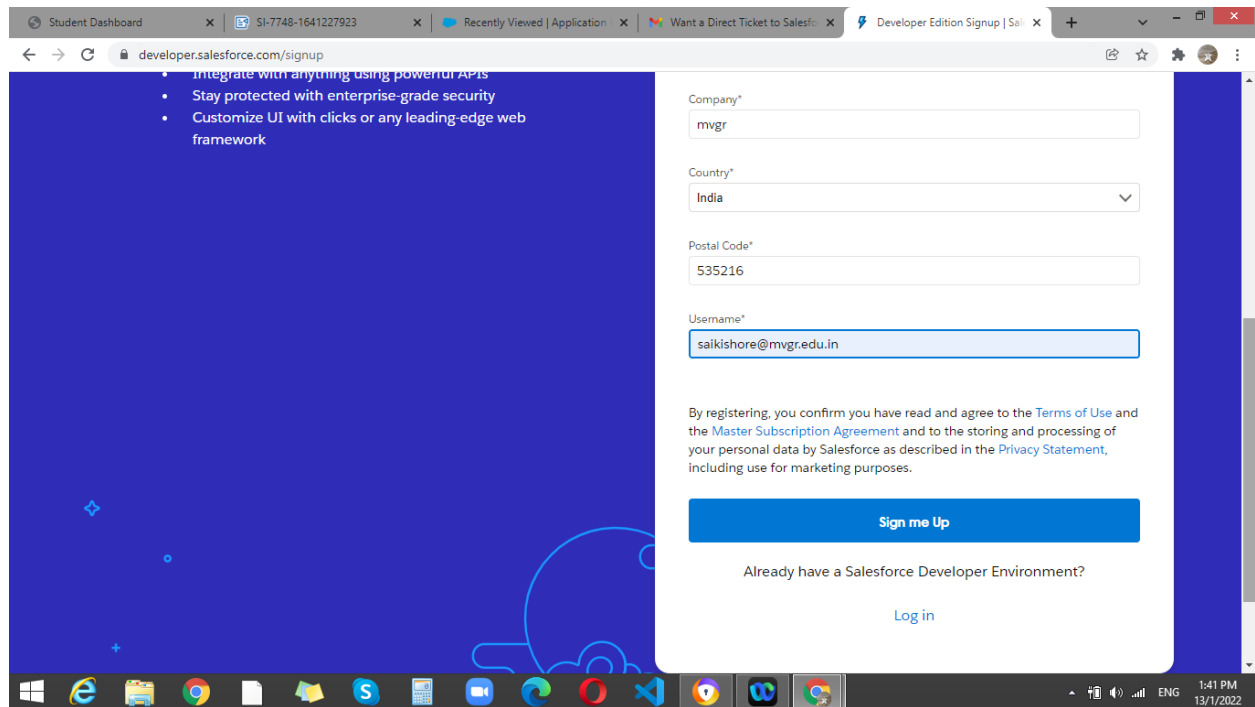
Email\* saikishorepsk@gmail.com

Role\* Developer

Company\* mvgr

Country\* India

1:41 PM 13/1/2022



## Day 2 :

**Topic:** custom object creation

**Milestone / Activities:** creating custom objects for college management application

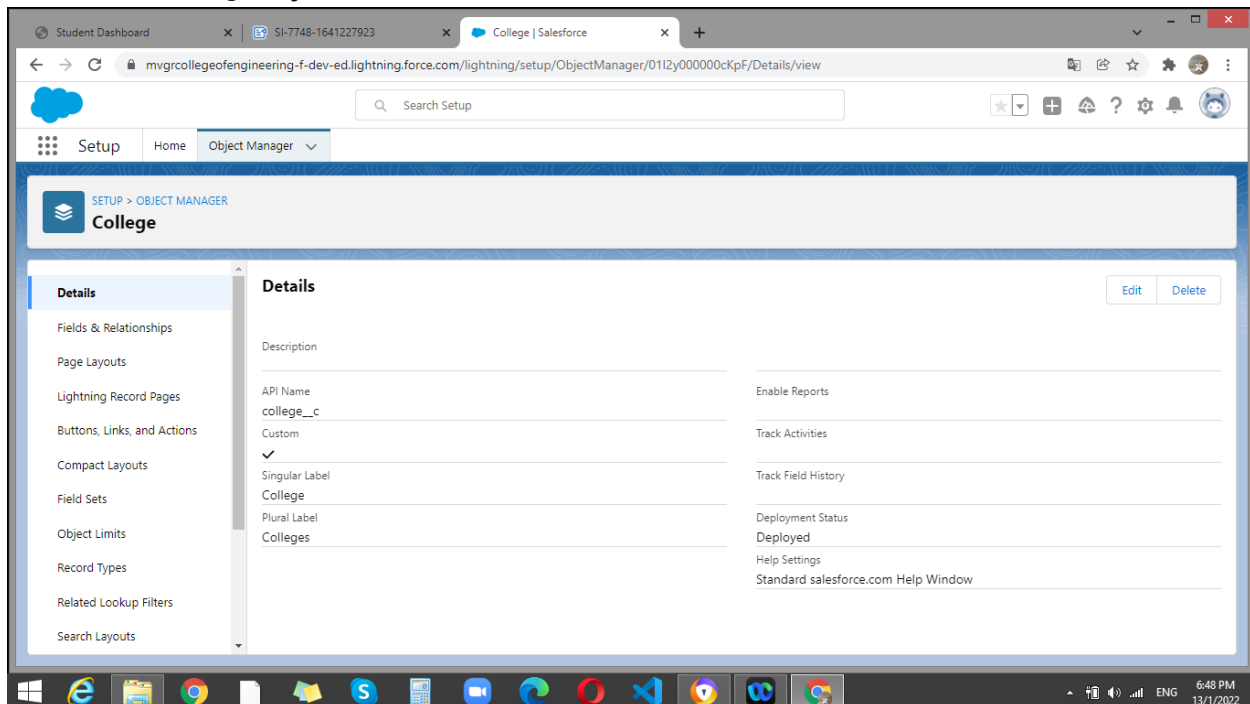
**Detailed Description:** the custom objects needed for the college management

application are college,application form,student,subject.

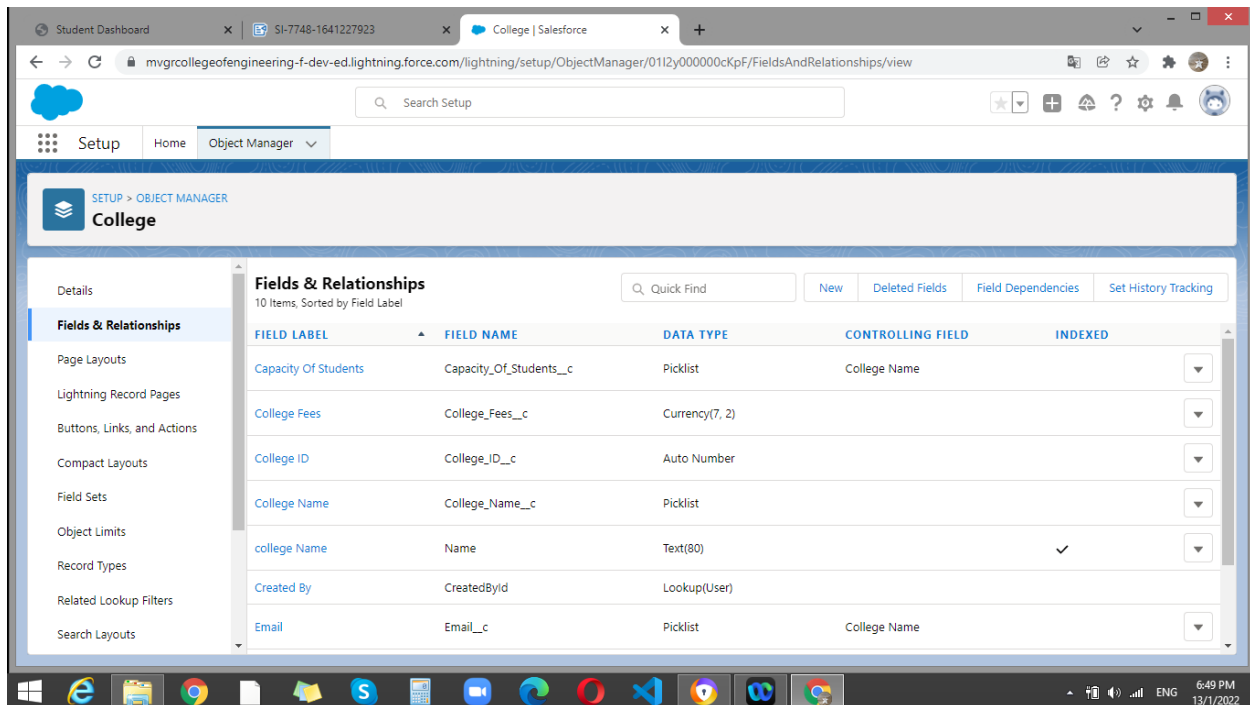
The college object contains the fields like record info of datatype text ,college fees of datatype currency ,hostelfees of datatype currency ,college name of datatype text ,email of datatype picklist ,capacityof students of datatype picklist.

## Screenshot the Milestone / Activities :

### Creation of collegeObject



### Adding Fields and relations



### Day 3 :

**Topic:** custom object creation

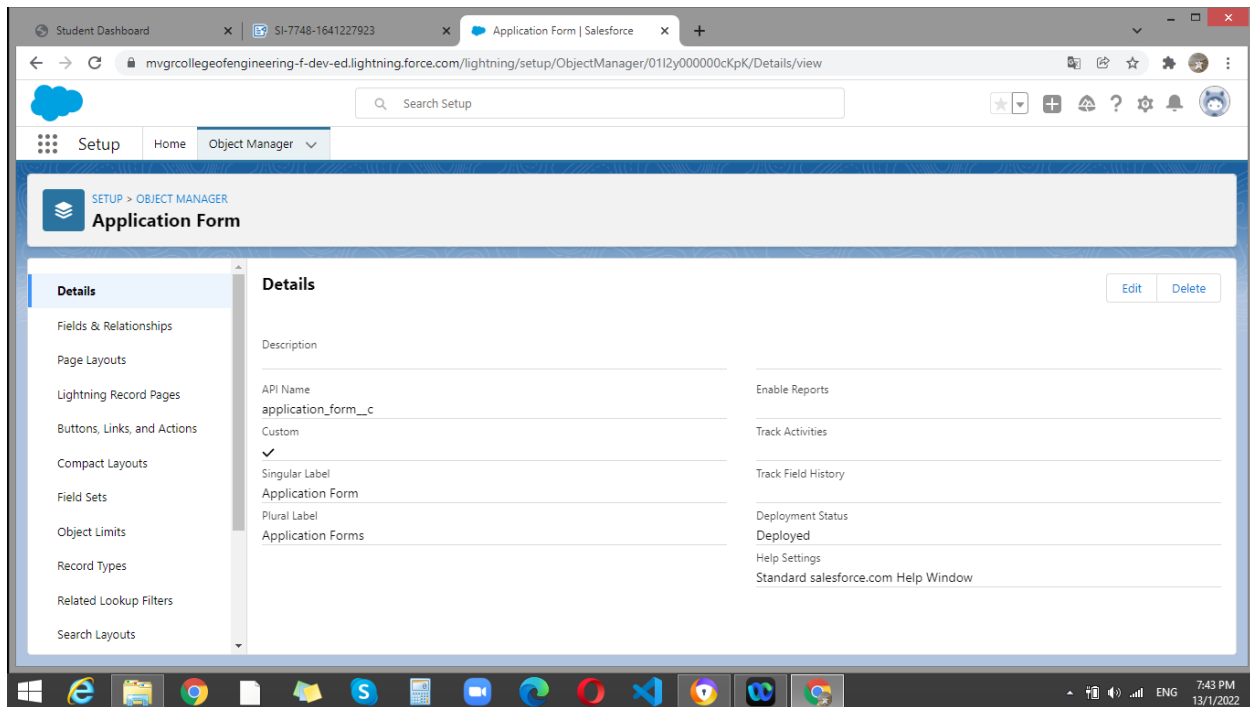
**Milestone / Activities:** creating Application form custom objects for college management application

**Detailed Description:** the custom objects needed for the college management application are college, application form, student, subject.

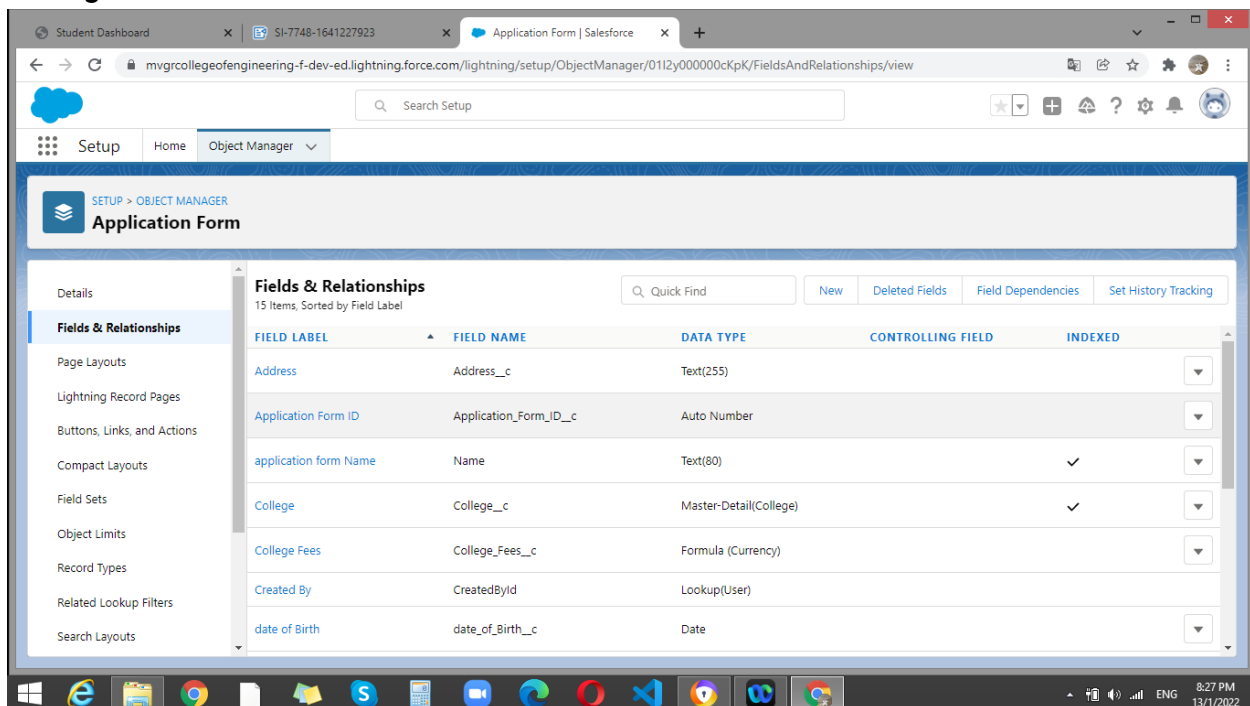
The application form object form contains fields like college form Id, college fees, hostelfees, gaurdian name, date of birth ,email, ready to join ,student name etc

**Screenshot the Milestone / Activities :**

**Creation of Application form object**



## Adding Fields and Relations



## Day 4 :

**Topic:** custom object creation

**Milestone / Activities:** creating student custom objects for college management application

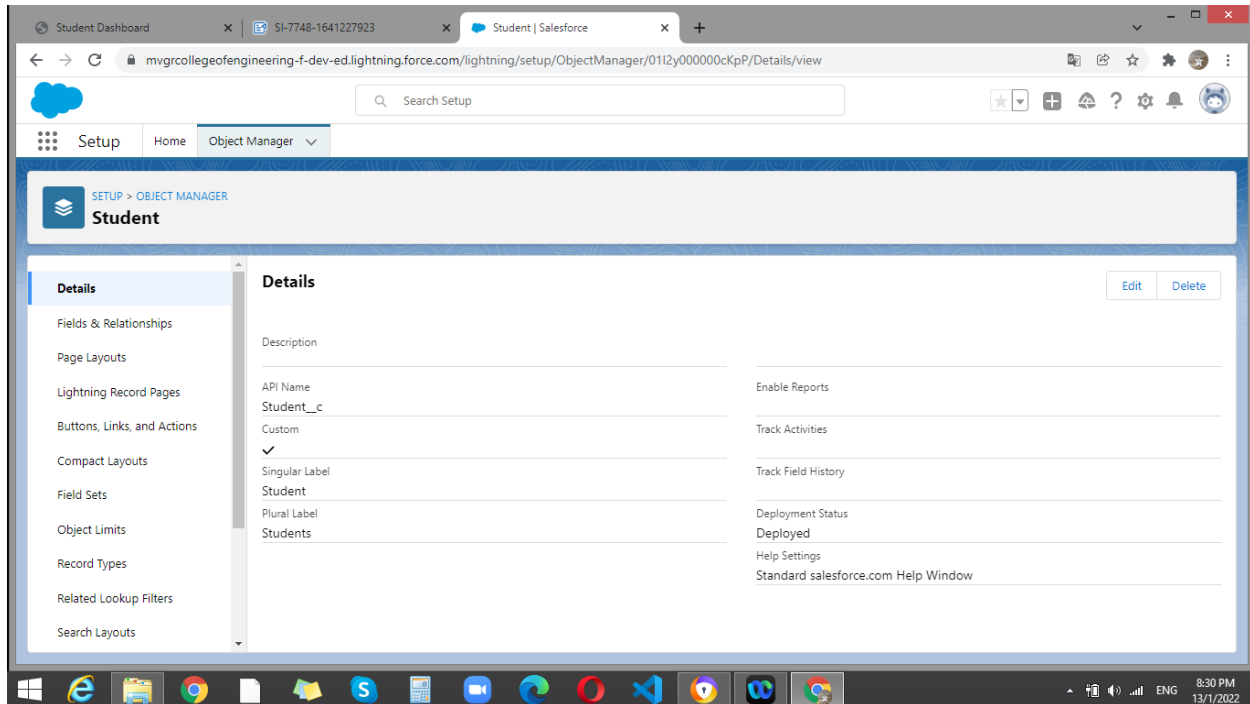
**Detailed Description:** the custom objects needed for the college management

application are college,application form,student,subject.

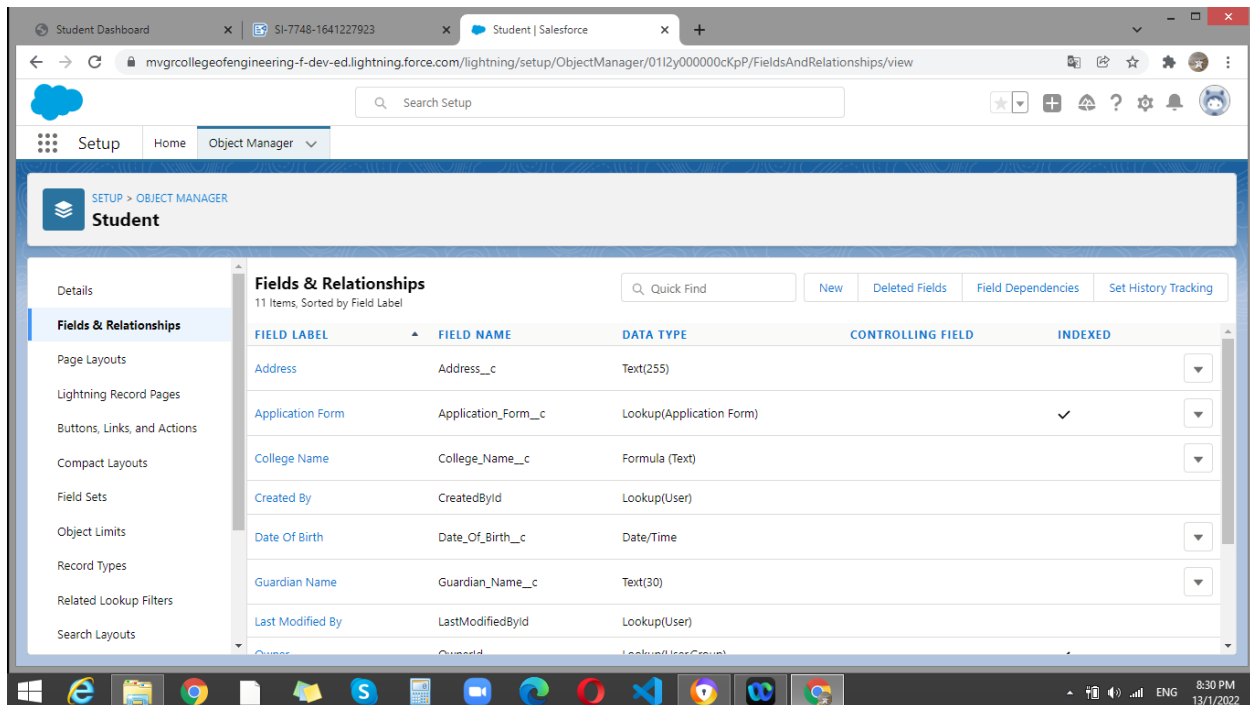
The student object contains fields like student name ,address ,application form ,college name ,guardian name ,phone.

## Screenshot the Milestone / Activities :

### Creation of student object



### Adding of fields and relations



## Day 5 :

**Topic:** subject custom object creation

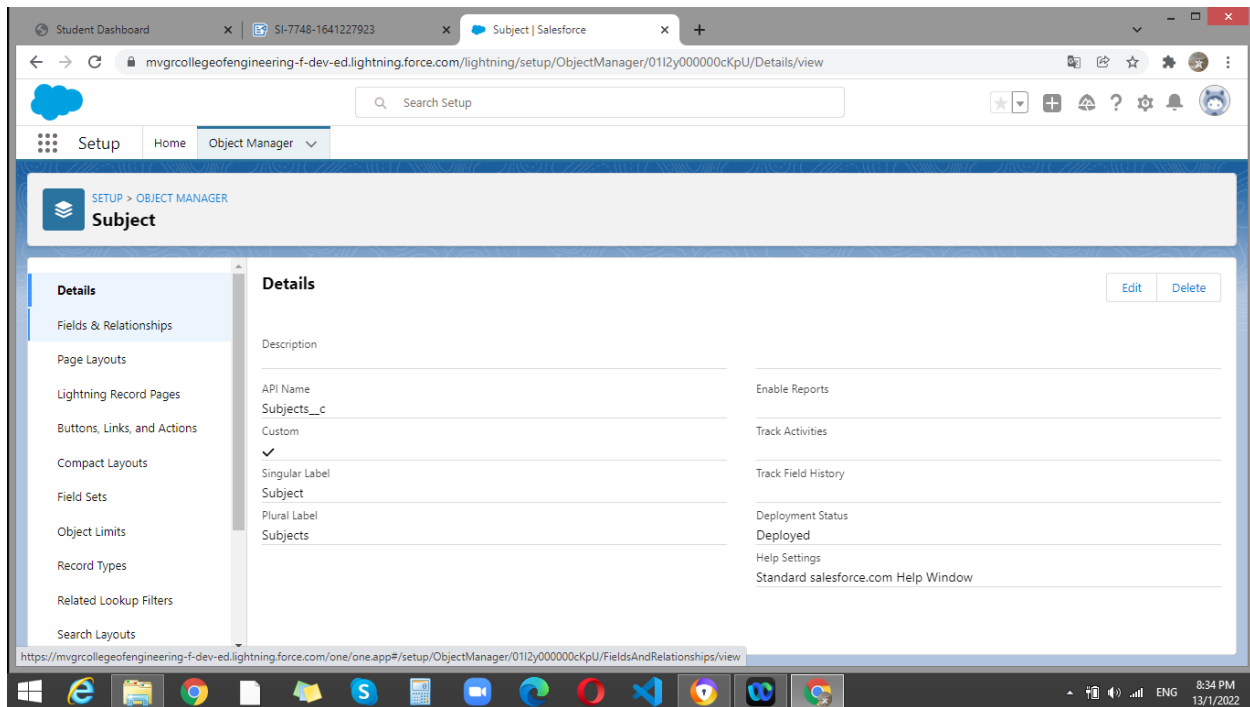
**Milestone / Activities:** creating subject custom objects for college management application

**Detailed Description:** the custom objects needed for the college management application are college, application form, student, subject.

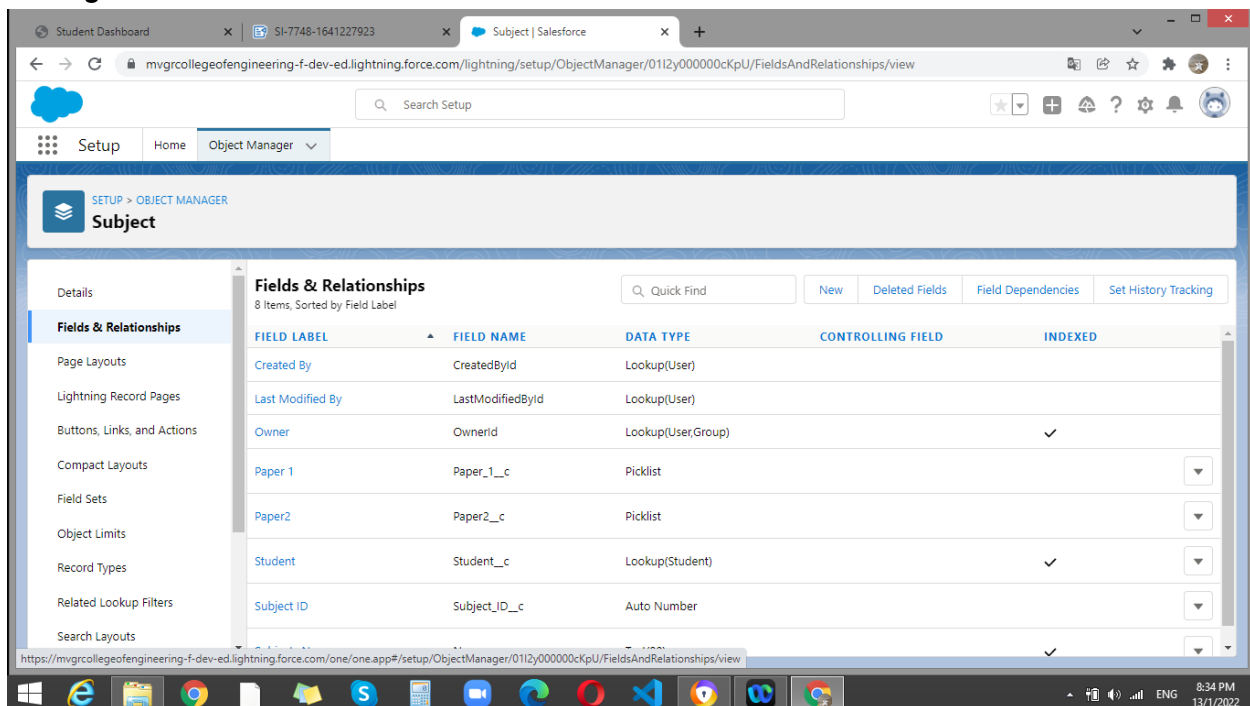
The subject object has the fields subject Id of datatype autonumber, paper1 of datatype picklist, paper2 of datatype picklist, student of datatype lookup(student object)

**Screenshot the Milestone / Activities :**

**Creation of subject Object**



## Adding Fields and relations



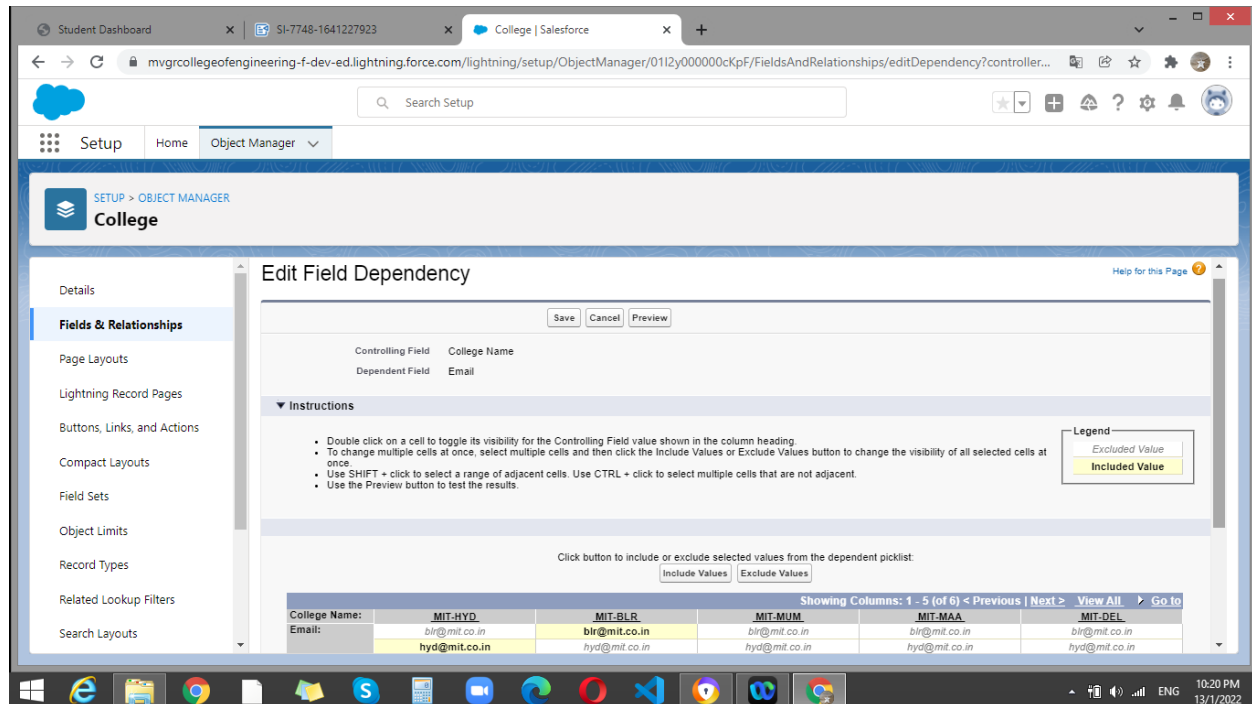
**Day 6 :**

**Topic: field dependencies**



## Milestone / Activities:creation of field dependencies

### Screenshot the Milestone / Activities :



## Day 7:

**Topic:**Adding business logic to application.

**Milestone / Activities:**creating global picklists.

**Detailed Description:** The global picklist are the picklists that are available for every field of datatype picklist.when a same picklist have to use many number of times then global picklist comes handy.

**Screenshot the Milestone / Activities :**

**Picklist Value Sets**

Global picklist value sets let you share the values across objects. Base custom picklist fields on a global value set to inherit its values. The value set is restricted so users can't add unapproved values through the API.

View: **All** [Create New View](#)

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | **All**

Action	Label	Description
<a href="#">Edit</a>   <a href="#">Del</a>	College	
<a href="#">Edit</a>   <a href="#">Del</a>	Paper 1	
<a href="#">Edit</a>   <a href="#">Del</a>	Paper 2	
Deleted Global Value Sets (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
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	MIT-HYD	MIT-HYD	<input type="checkbox"/>	Assigned dynamically	<a href="#">Sai Kishore Pallantia</a> 1/4/2022, 1:54 AM
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	MIT-BLR	MIT-BLR	<input type="checkbox"/>	Assigned dynamically	<a href="#">Sai Kishore Pallantia</a> 1/4/2022, 1:54 AM
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	MIT-MUM	MIT-MUM	<input type="checkbox"/>	Assigned dynamically	<a href="#">Sai Kishore Pallantia</a> 1/4/2022, 1:54 AM
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	MIT-MAA	MIT-MAA	<input type="checkbox"/>	Assigned dynamically	<a href="#">Sai Kishore Pallantia</a> 1/4/2022, 1:54 AM
<a href="#">Edit</a>   <a href="#">Del</a>   <a href="#">Deactivate</a>	MIT-DEL	MIT-DEL	<input type="checkbox"/>	Assigned dynamically	<a href="#">Sai Kishore Pallantia</a> 1/4/2022, 1:54 AM

**Day 8:**

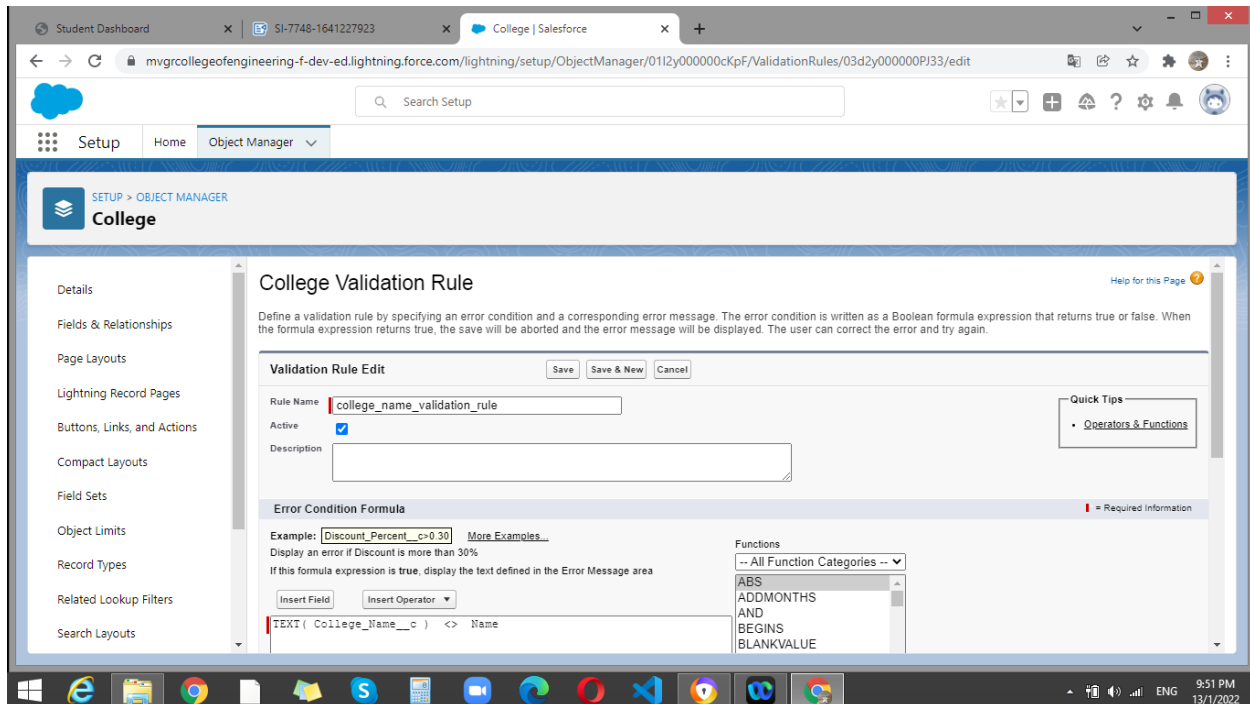
**Topic:** Adding business logic to application.

**Milestone / Activities:** creating validation rules.

**Detailed Description:** The validation rules are the rules that are checked when data is entered in the data fields. In the application the validation rule used is the college name

of picklist value must be equal to college name of text type.

### Screenshot the Milestone / Activities :



### Day9:

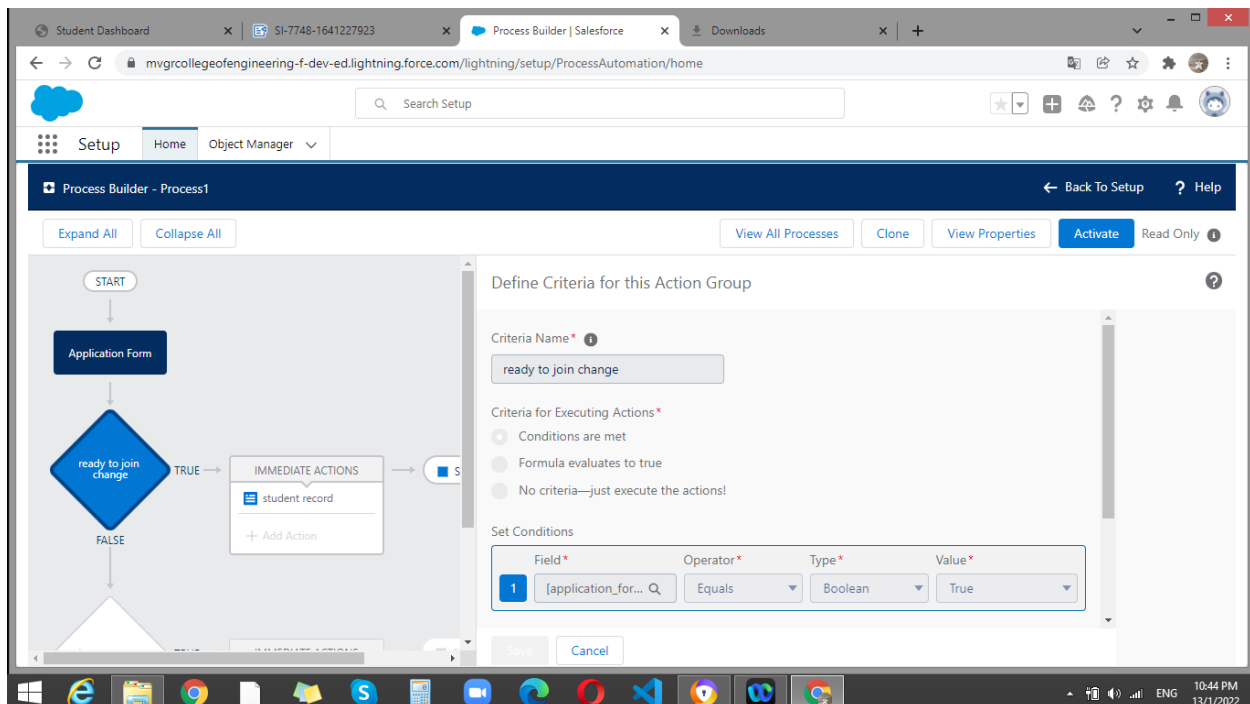
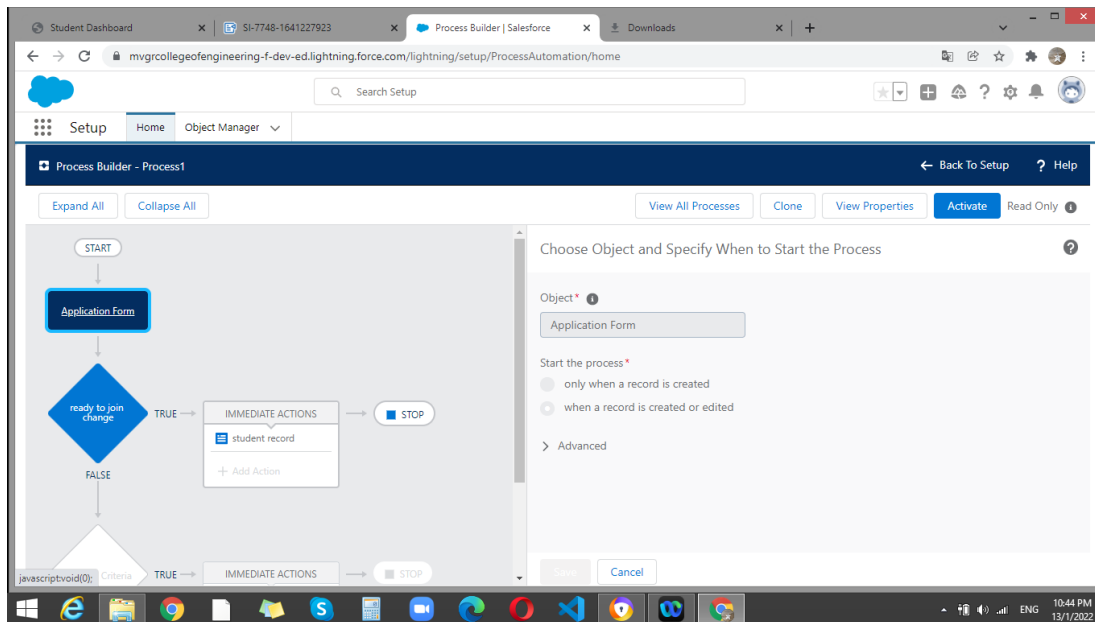
**Topic:**adding business logic to application

**Milestone / Activities:**process automation

**Detailed Description:** Process automation is the execution of certain task when the given condition becomes true.

In the application process automation is used to create a student record when the ready to join is checked.

### Screenshot the Milestone / Activities :



## **Day 10 :**

**Topic:** Adding business logic to application

**Milestone / Activities:** create the student record using flow

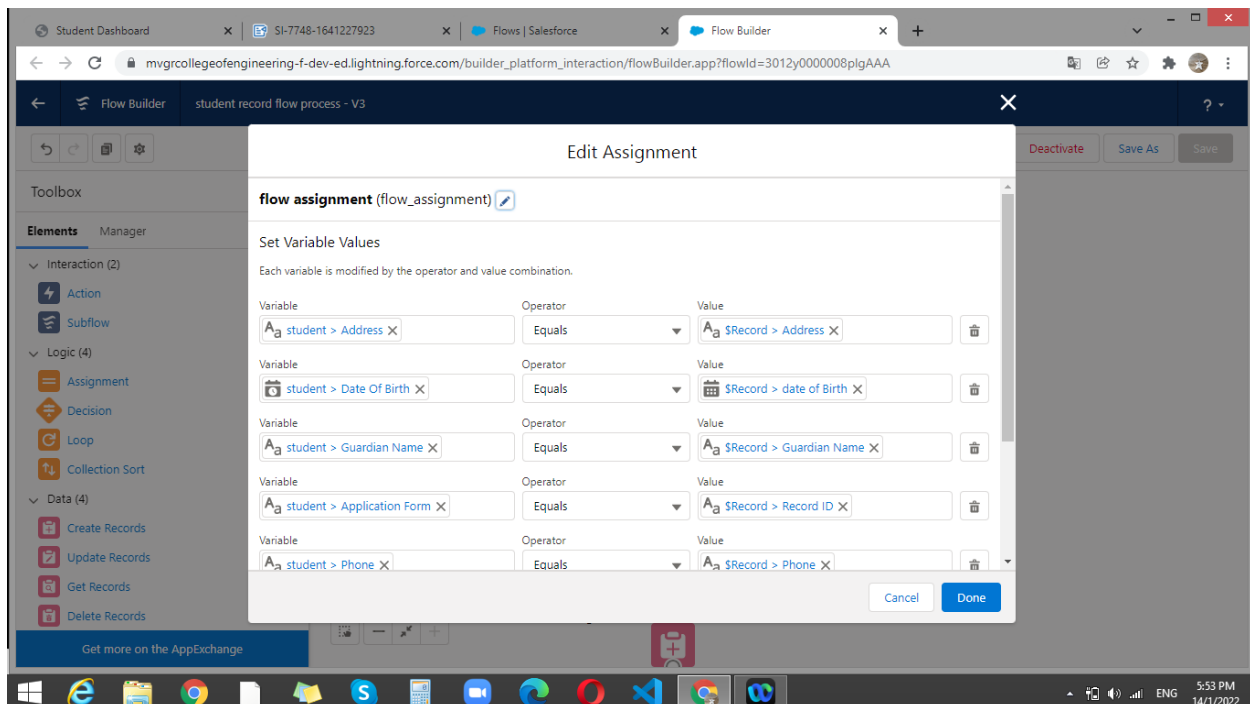
**Detailed Description:** flows are the special feature in salesforce that can do the works

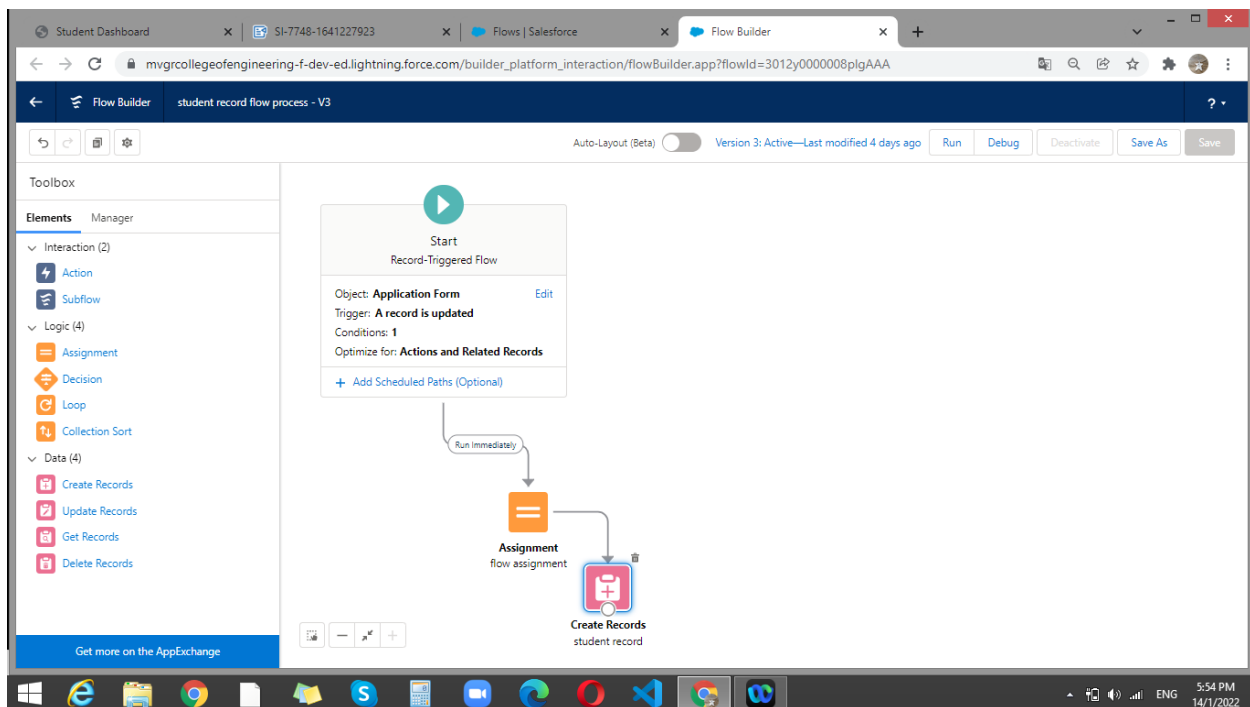
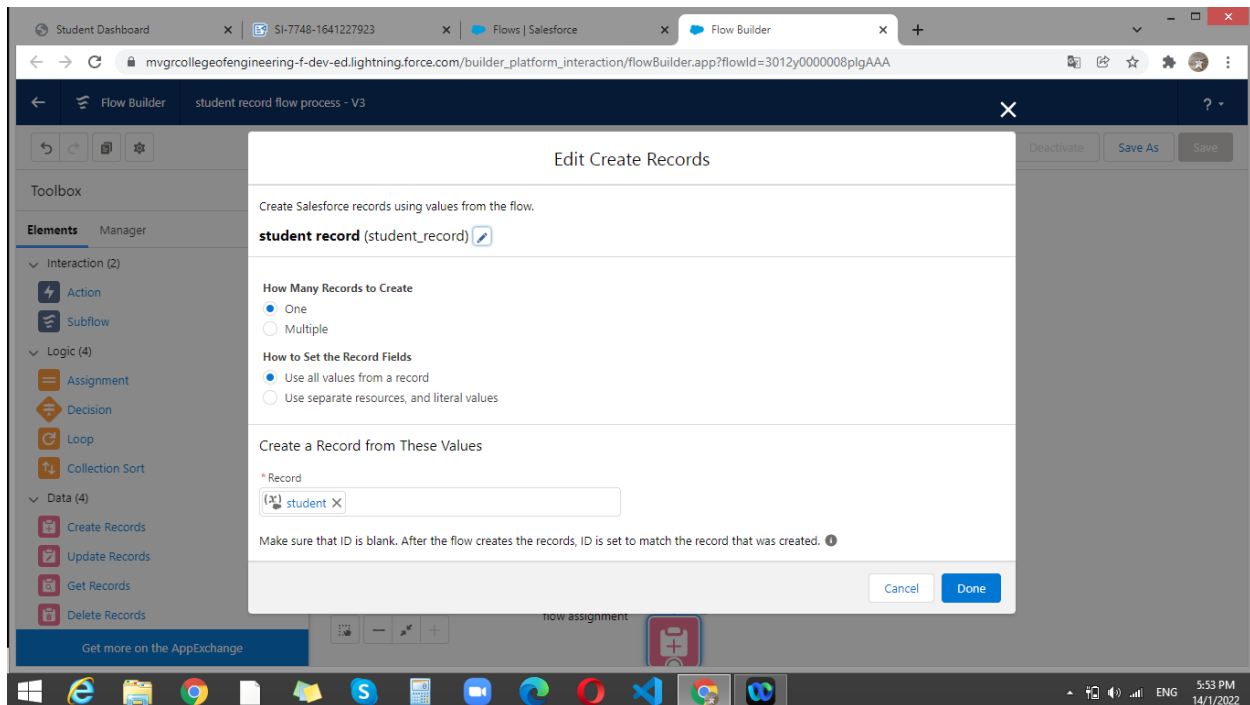
of workflow and the process builder.

In our application we use the flow to create student record when the ready to join in the application form is edited.

while creating flow we use record triggerd flow,and also create a new resource of name student and assign the values of application form to student record and create records of student.

## Screenshot the Milestone / Activities



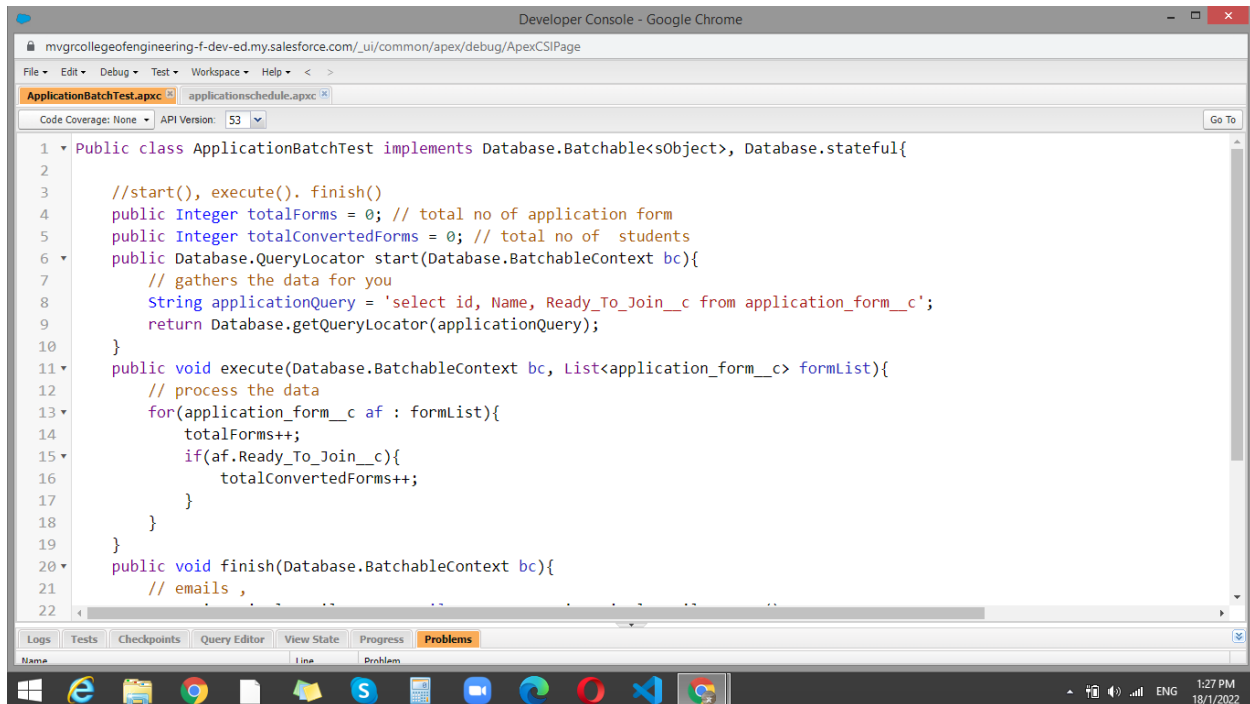


**Day 11:**  
**Topic:**Batch apex

## Milestone / Activities: creating batch apex for application form

**Detailed Description:** Batch Apex is used to run large jobs (think thousands or millions of records!) that would exceed normal processing limits. Using Batch Apex, you can process records asynchronously in batches to stay within platform limits.

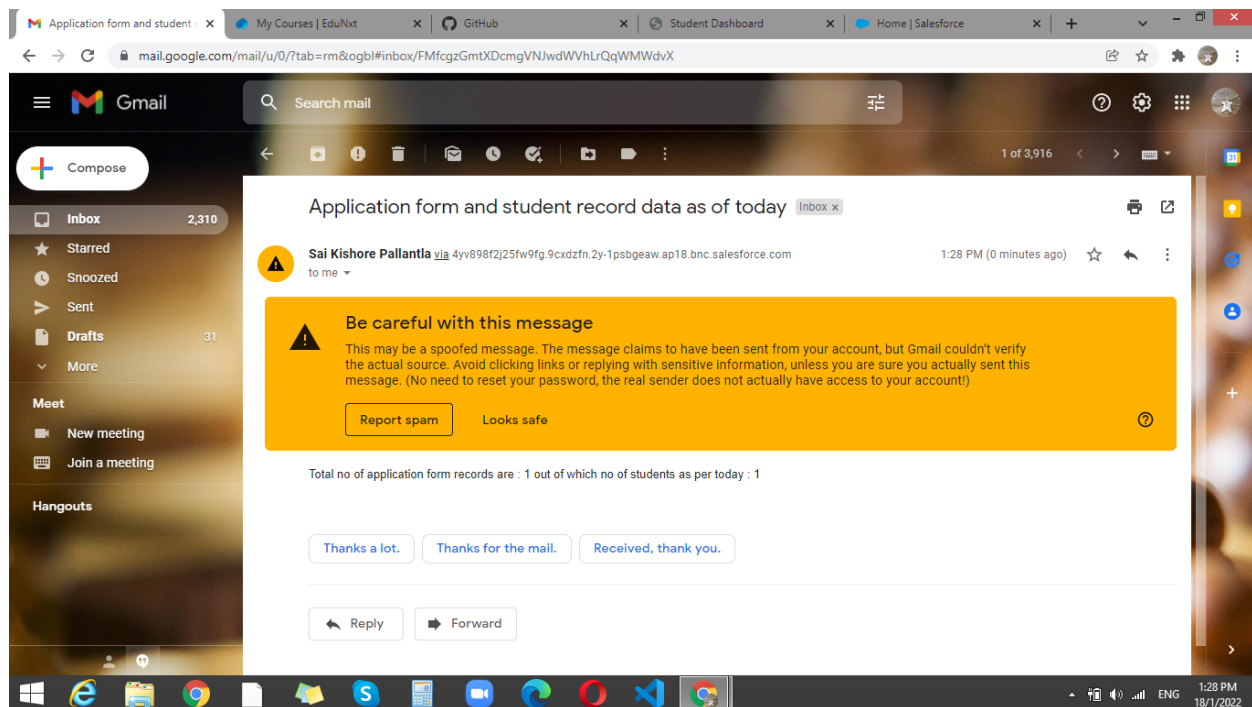
## Screenshot the Milestone / Activities



The screenshot displays the Salesforce Developer Console in Google Chrome. The browser address bar shows the URL: `mvgrcollegeofengineering-f-dev-ed.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage`. The console's file explorer shows the file `ApplicationBatchTest.apxc` is open. The code editor displays the following Apex code:

```
1 Public class ApplicationBatchTest implements Database.Batchable<Object>, 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    public void finish(Database.BatchableContext bc){
21        // emails ,
22    }
```

The console interface includes tabs for Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Windows taskbar at the bottom shows various application icons and the system clock indicating 1:27 PM on 18/1/2022.



## **Day 12:**

**Topic:**Batch apex

**Milestone / Activities:** creating scheduler class

**Detailed Description:** when a class needs to execute at a certain time every day then scheduler class comes handy.

It executes the class at certain time that is fixed.

**Screenshot the Milestone / Activities**



Developer Console - Google Chrome

mvgrcollegeofengineering-f-dev-ed.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage

ApplicationBatchTest.apxc applicationschedule.apxc

Code Coverage: None API Version: 53 Go To

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

Logs Tests Checkpoints Query Editor View State Progress Problems

Name	Line	Problem
------	------	---------

Windows taskbar: 4:39 PM 18/1/2022