Day 1:

**Topic:** creating salesforce developer org to get started

Activities:
1. Creating Developer Account
2. Account Activation
3. Login to your salesforce account

Detailed Description:
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.
I have account in my laptop so I directly entered into salesforce account

Screenshots:
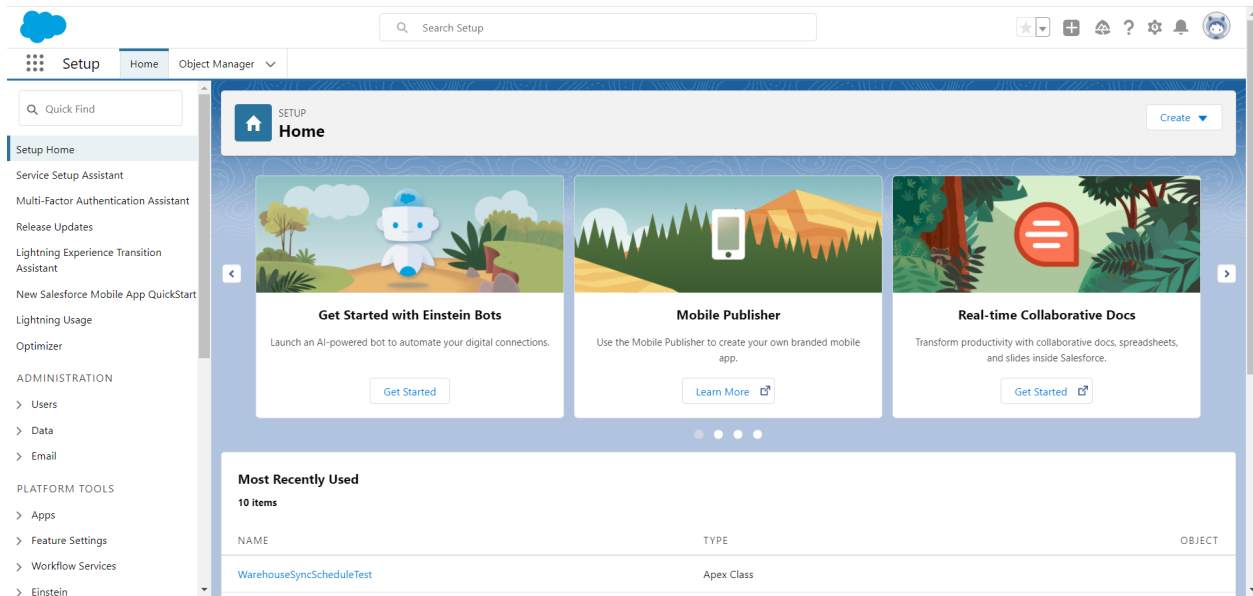
Day 2:

Topic : Custom Object Creation

Activities :
1.Create custom object for the applications
2.Create fields on college objects
3.Create fields on Application form object
4.Create field on student object
5.Create field on subject object

Detailed Description :

To create custom object in Salesforce, perform the following steps:

- Log in to your Salesforce account.

- Click **Setup** at the upper-right corner.

- Under the **Build** section, click **Create** and select **Objects**.

- To create a custom object, click **New Custom Object**.

- Enter the name of the Custom Object in **Label**, **Plural Label**, and **Object Name**.

Screen Shots :

College Object:

Search Setup

Setup | Home | Object Manager

Fields & Relationships
10 Items, Sorted by Field Label

Quick Find | New | Deleted Fields | Field Dependencies | Set History Tracking

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

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)	

Application Form:

Search Setup

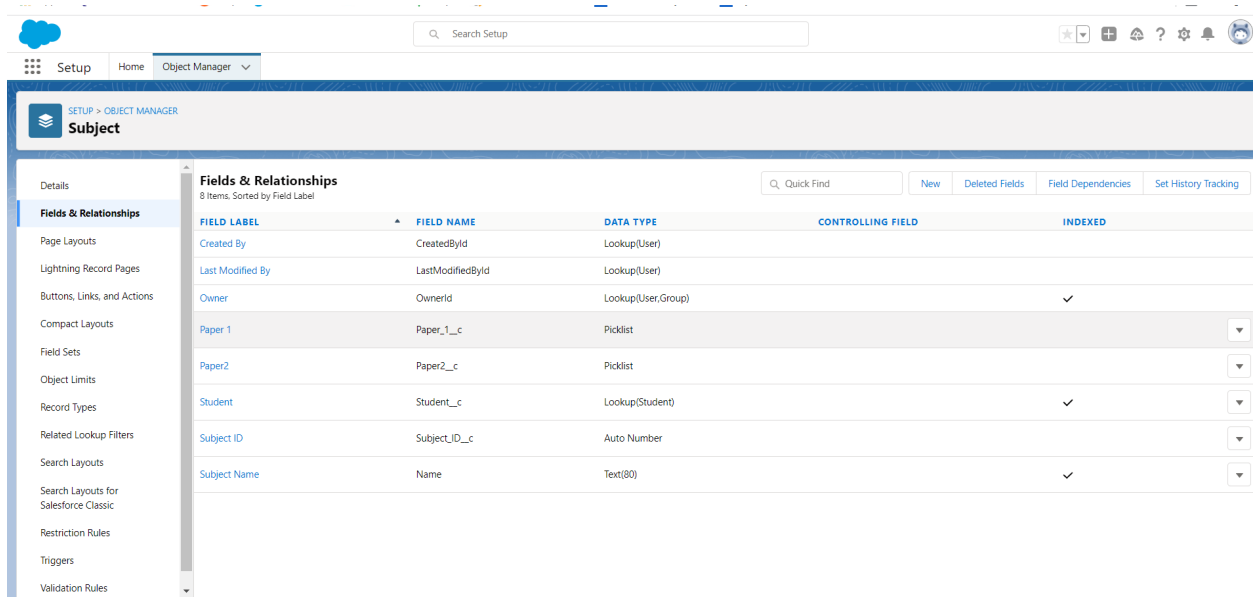Setup | Home | Object Manager

Fields & Relationships
15 Items, Sorted by Field Label

Quick Find | New | Deleted Fields | Field Dependencies | Set History Tracking

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

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 Name	Name	Text(80)		✓
College	College__c	Master-Detail(College)		✓
College Fees	College_Fees__c	Formula (Currency)		
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)		

Student Object:

Subject object:

Day 3:

Topic: Adding Bussiness logic for the application

Activities :

1.Creating Global Picklist Value Sets
2.Creating Field Dependencies
3.Creating Validation Rules
4.Process Automation
5.Create The Student Record Using Flow

Detailed Description :

Here are the steps in creating a global picklist:

1. Navigate to Setup, then click Create.
2. Select Picklists from the drop-down menu.
3. The Global Picklists page will pop out. Select New to start creating a new picklist.
4. Enter a name for the Label and the Developer Name. In the example above, the admin may use the name City for both fields. Note that Developer Name will become the picklist's actual name.
5. Enter a description for the picklist (optional).
6. Hit Save. A details page will then appear. The user will see three lists: Picklist

Value Set, Inactive Picklist Values, and Fields Where Used.

7. Click on the New button on the Picklist Value Set. Enter a new value for the list. In this case, enter the names of cities separately per line. Once done, hit Save.

Creating a field dependency fields

To create field dependency

– Go to setup menu and select the object you want to create field dependency.

– Click on filed dependency button
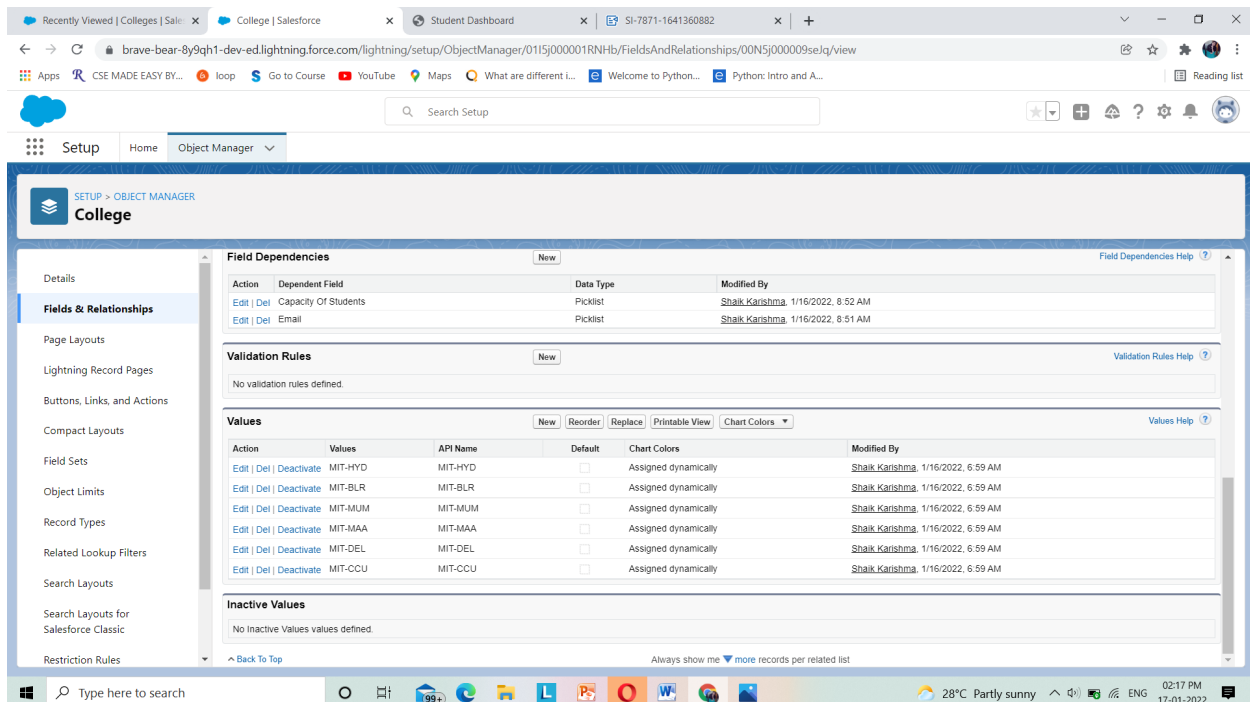
Creating validation rules

1. From Setup, go to Object Manager and click **Account**.
2. In the left sidebar, click **Validation Rules**.
3. Click **New**.

Screensots :

global picklist:
college:

paper1:

paper 2:

field dependencies:

College Field Dependencies

« Back to Custom Object: College

This page allows you to define dependencies between fields (e.g., dependent picklists).

Field Dependencies			New
Action	**Controlling Field**	**Dependent Field**	**Modified By**
Edit	Del	College Name	Email	Shaik Karishma, 1/16/2022, 8:51 AM
Edit	Del	College Name	Capacity Of Students	Shaik Karishma, 1/16/2022, 8:52 AM

validation values:

college object:

Validation Rules

2 Items, Sorted by Rule Name

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
college_name	Top of Page	else false	✓	Shaik Karishma, 1/16/2022, 8:56 AM
record_info	Top of Page	else false	✓	Shaik Karishma, 1/16/2022, 8:57 AM

Application form object:

Process Builder :
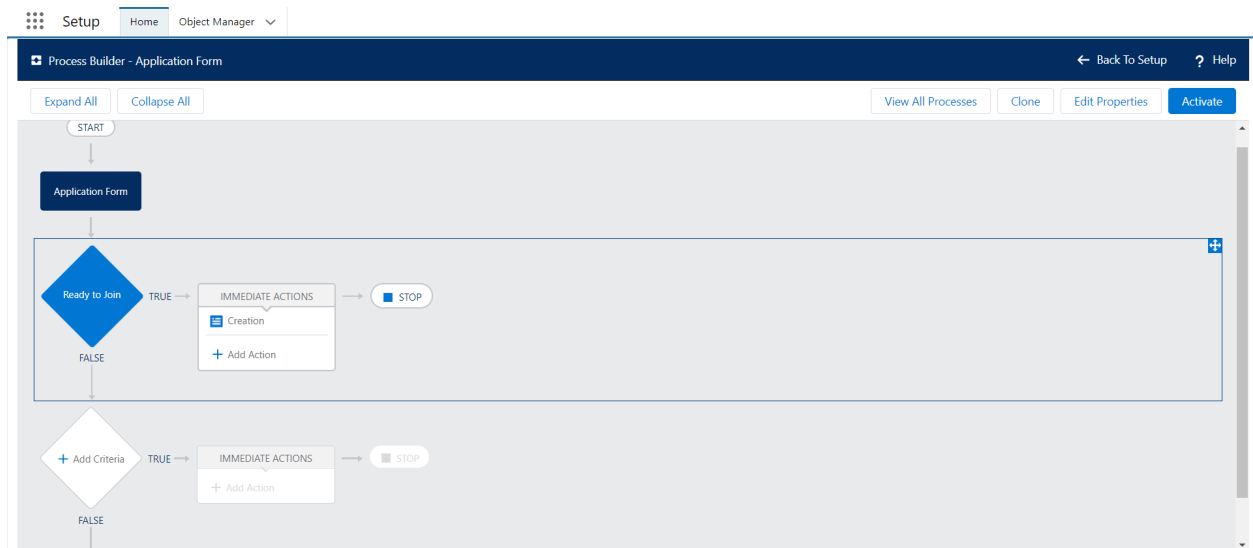
Flow builder :
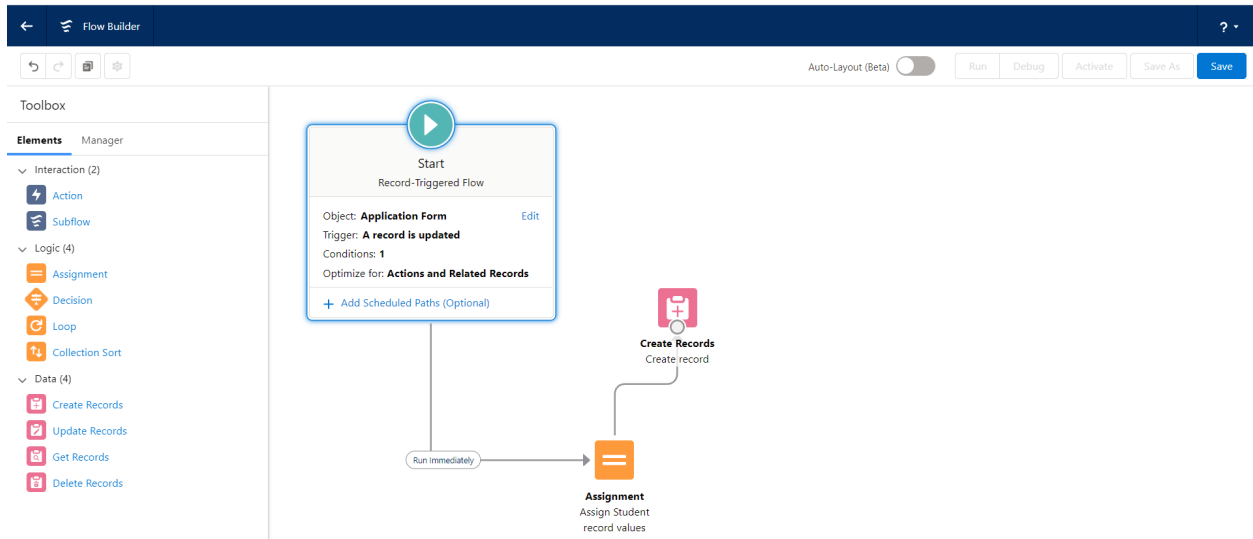
Day 4

Topics: Batch Apex

Activites:

1.Create A Batchapex For Application Form
2.Create A Schedular Class

Detailed Description :

1. From the developer console create a new apex class and enter the code.

Batch Apex:

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 (hence the name, "Batch Apex") to stay within platform limits. If you have a lot of records to process, for example, data cleansing or archiving, Batch Apex is probably your best solution.

To write a Batch Apex class, your class must implement the Database.Batchable

interface and include the following three methods:

Start
Execute
Finish

ScreenShots :

Scheduler Class:


```
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  }
```
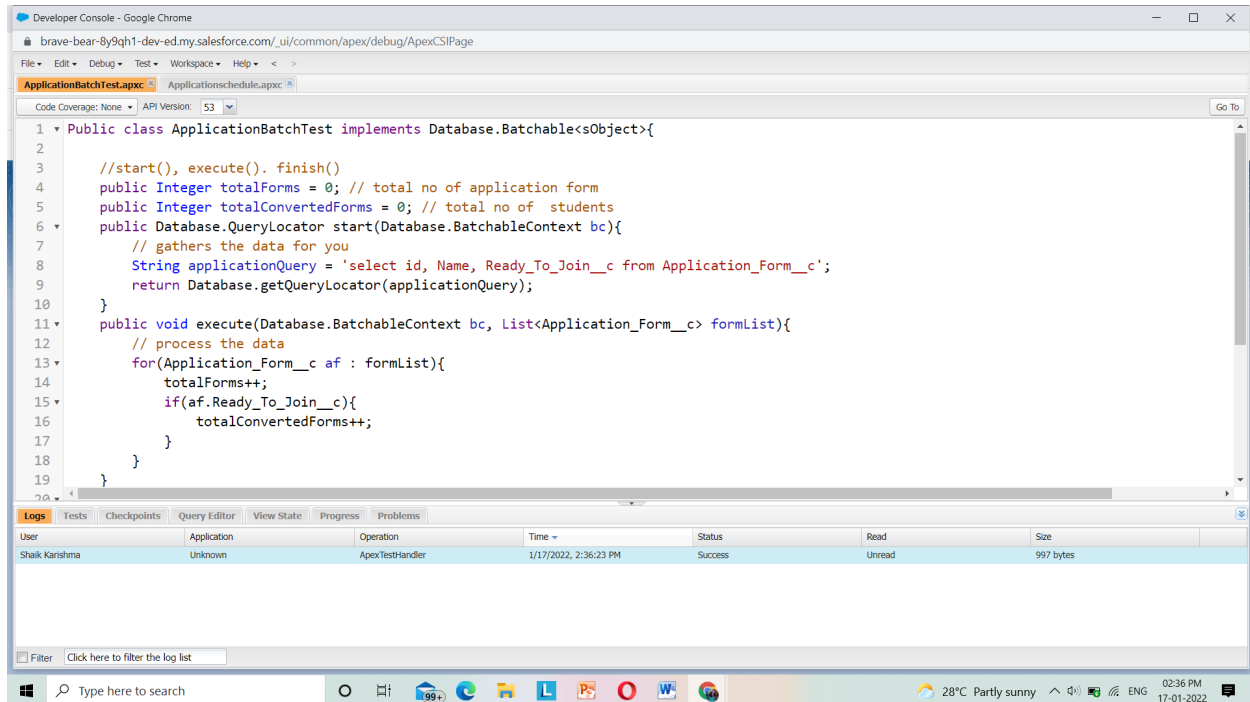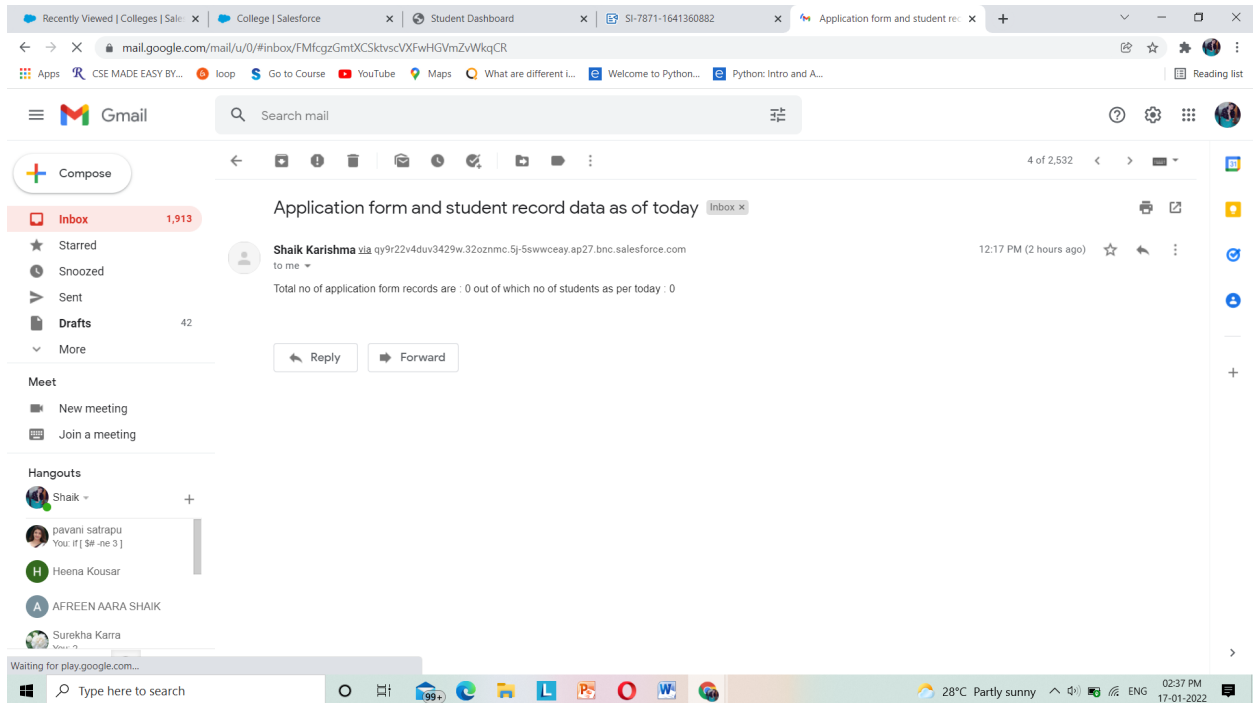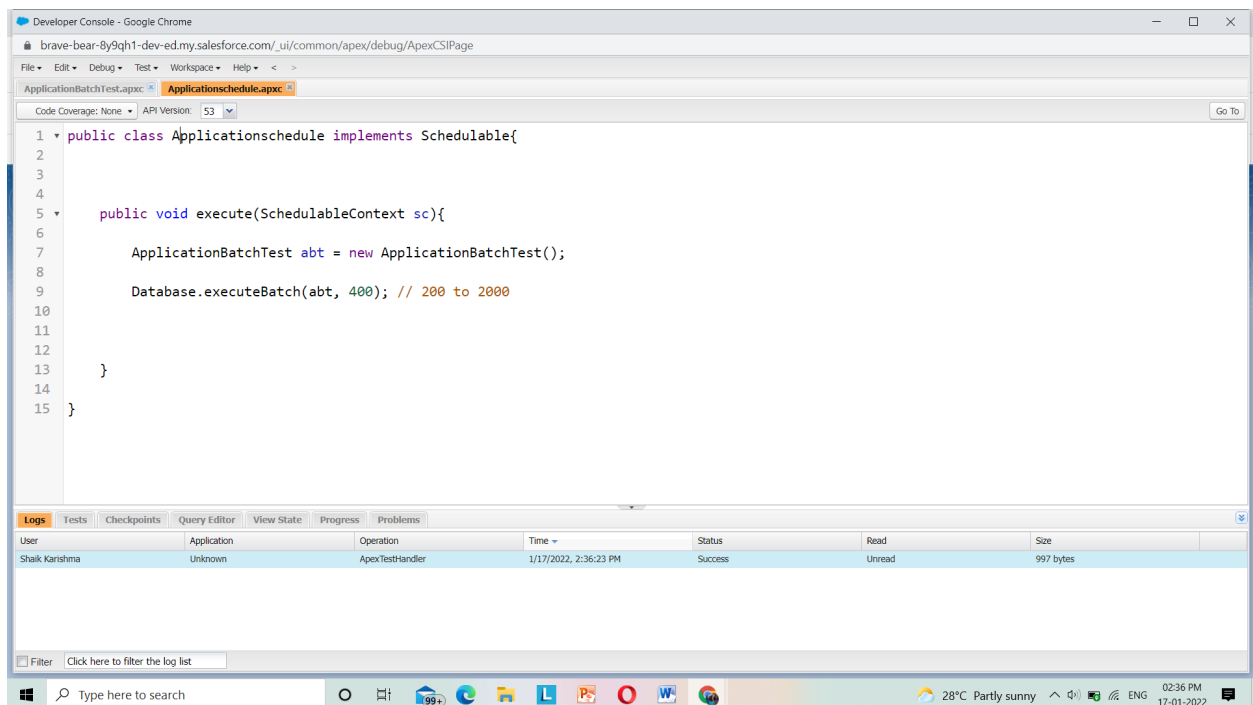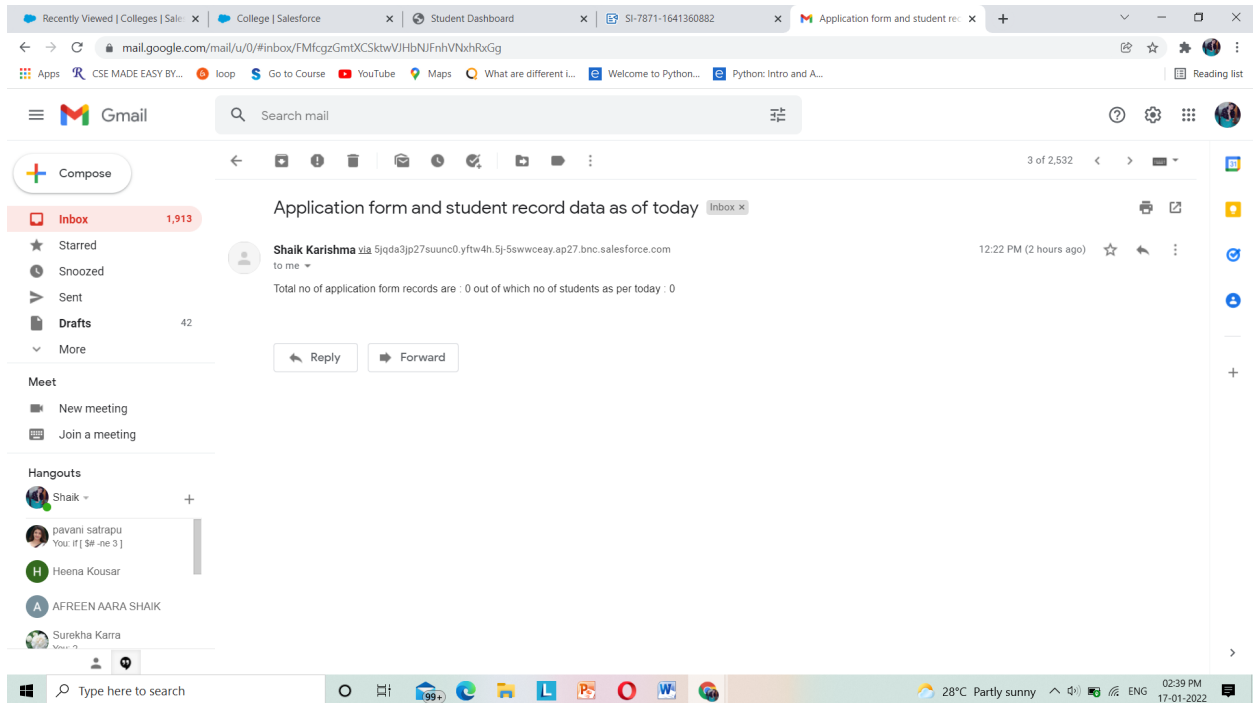
Day 5-

Topics : Lightening Web Components

Activites :

Create College DataTable Component(APEX CLASS)

Create College DataTable Component (HTML FILE)

Create College DataTable Component(JAVA SCRIPT FILE)

Create College DataTable Component(META FILE)

Description :

Apex is a **strongly typed, object-oriented programming language** that allows developers to execute flow and transaction control statements on the Salesforce platform. If you're used to Java or . NET development, you'll find the programming in

Apex fairly straightforward.

We will copy all the files(HTML,Js,Meta) to vs and then link the vseditor to salesforce org

ScreenShots:

All ▼

College Management | College ▼ | Application Forms ▼ | Stud

College
MIT-MAA

DataTable

College and Application form list table

	Application Form ▼	College Fees ▼	date of Birth ▼
1	abcd	$500,000.00	Jan 7, 2022
2	stunt	$500,000.00	Jan 7, 2022
3	klaus	$500,000.00	Jan 5, 2005
4	elaja	$500,000.00	Jan 11, 2022

Related | **Details**

College Name
MIT-MAA

Ow

Record Info
MIT-MAA

College Fees
$500,000.00

Hostel Fees
$100,000.00

Type here to search

```javascript
import { LightningElement, api, wire } from 'lwc';
import getapplicationvalues
from '@salesforce/apex/GetapplicationDetails.getapplicationvalues';
export default class DataTable 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'})
    wiredCollegeData({data, error}){
        if(data){
            this.recordList = data;
        }
        else if(error){
            this.error = error;
            this.recordList = undefined;
        }
    }
}
```



```xml
<?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>
```

force-app > main > default > lwc > DataTable.html > template > h1

```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> </template>
```