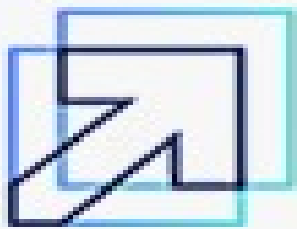


SF Project Ready - Online Training Management



**Smart
Internz**

By

B.Harika

INDEX:

Online Training Management - Developer
Create Your Salesforce Developer Org To Get Started
Custom Object Creation
Validation Rules Creation
Process Automation
Creating Custom Pages Using Visualforce
Triggers

Online Training Management - Developer

Category:

Salesforce

Skills Required:

Salesforce Admin,Salesforce Developer

Project Description:

We will be creating an online training management application that will store all the details of the training from courses offered, trainers, students enrolled, Meeting/training information, and many more. In the project, we will be creating objects to store these records and will also create various fields to store the values of these records. We will

establish relationships between various objects and create custom pages using visualforce.

Custom Object Creation:

Custom Objects:

Custom objects are the objects which are created manually by the users. Below is the video which shows how custom objects are created.

Create one Application Named as Online Training Management:

The custom objects are in online training management Application

- a) Courses
- b) Trainer
- c) Student
- d) Meeting
- e) Class Scheduling

Course:

Add fields in this course object:

Field Name	Data Type	Values
Course ID	Auto Number (Change the data type to AutoNumber While creating course object and changing record Type to course id)	Format(C-000001) Starting number=1
Course Name	Text	
Course Duration	Picklist	1 month, 2 months, 3 months
Course Fees	Currency	
Course Description	Text (Long)	
Course Category	Picklist	Data science, cloud computing, AI, ML
Course Trainer	Master-Detail (Trainer)	

Trainer:

Add fields in this Trainer object:

Field Name	Data Type	Values
Trainer ID	Autonumber	Format=T-00001,Starting number=1
Full Name	Text	
Email	Email	
PhoneNumber	Phone	
Whatsapp Number	Phone	
DateofBirth	Date	
Age	Formula	TODAY()-Date_of_Birth__c
Experience	Picklist	1year,2years,3years,.....
LinkedIN Profile	URL	
Expertise	Text	

Student:

Add fields in this Student object:

Field Name	Data Type	Values
Student ID	Autonumber	Format=S-00001,Starting Number=1
First Name	Text	
Last Name	Text	
Email	Email	
PhoneNumber	Phone	
Date Of Birth	Date	
Whatsapp Number	Phone	
Enrollment Date	Formula	TODAY()
Student Enrollment	Master Detail(Batch)	

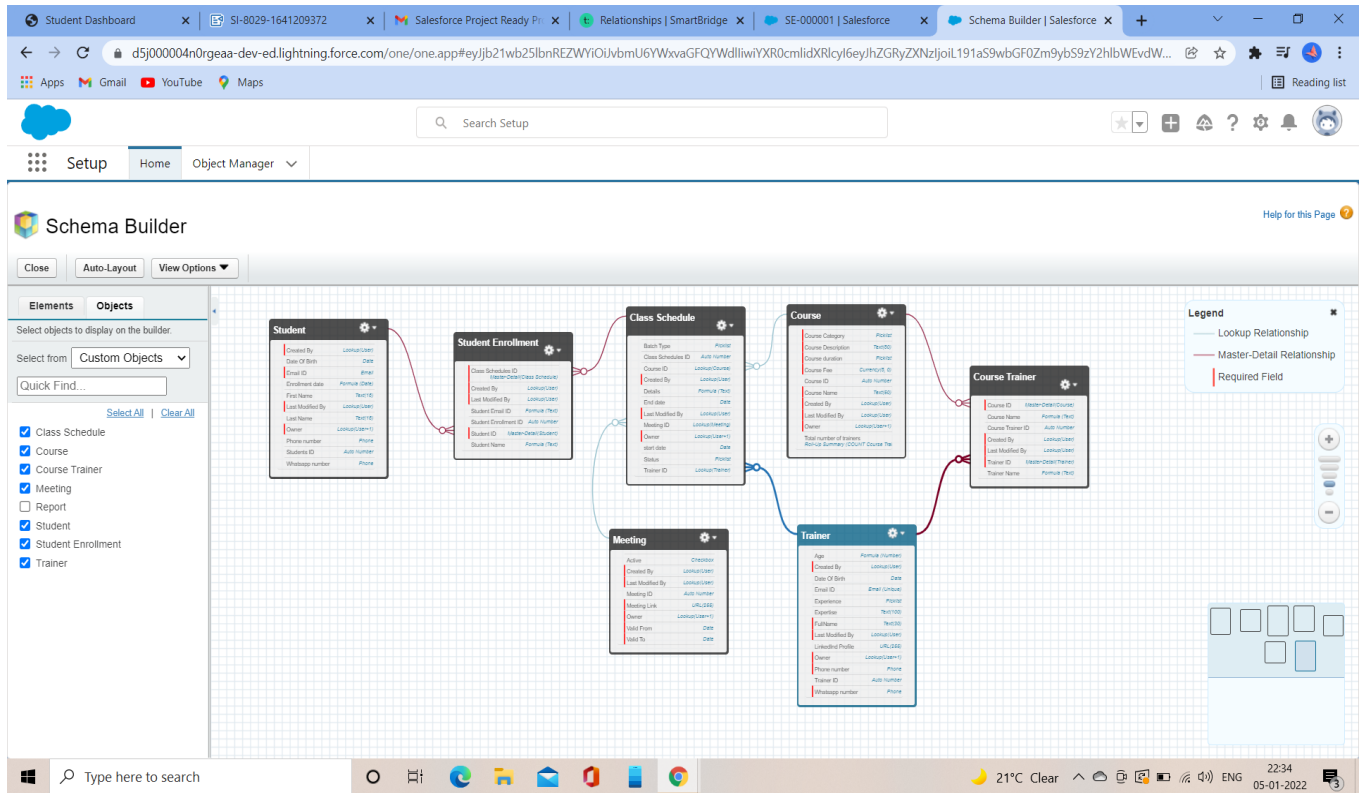
Class Schedule:

Add fields in this Class schedule object

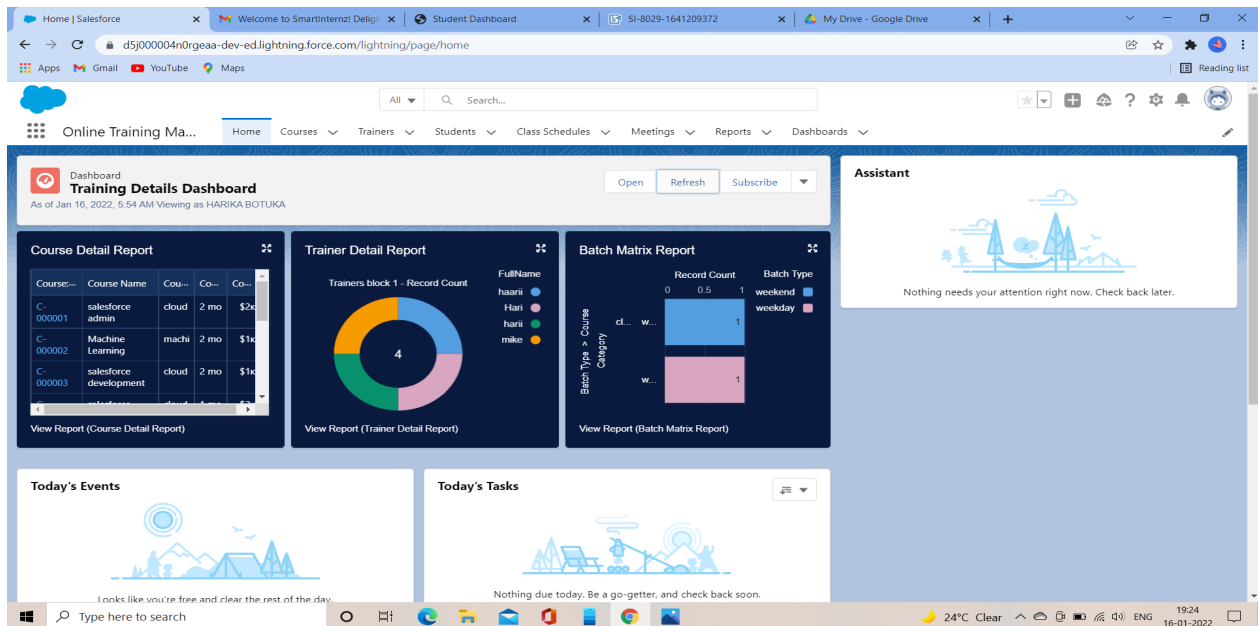
Field Name	Data Type	Values
Batch ID	Auto Number	Format=B-00001 Starting Number=1
Batch Type	Picklist	Weekday, Weekend
Start Date	Date	
End Date	Date	
Course ID	Lookup(Course)	
Meeting id	Lookup(Meeting)	
Trainer id	Lookup(Trainer)	
Status	Picklist	Scheduled/Started/Completed
Number Of Students Enrolled	Rollup Summary(Count)	
Details	Formula(Text)	Course_ID__r.Course_Name__c & Trainer_ID__r.FullName__c & Meeting_ID__r.Meeting_Link__c

Meeting:**Add fields in this Meeting object:**

Field Name	Data Type	Values
Meeting ID	Autonumber	Format=M-00001, Starting Number=1
Valid From	Date	
Valid To	Date	
Active	Checkbox	
Meeting Link	URL	



Add all objects in Online Training Management Application:



Create Validation Rules:

Validation rules verify that the data a user enters in a record meets the standards you specify before the user can save the record. A validation rule can contain a formula or expression that evaluates the data in one or more fields and returns a value of "True" or "False".

Create Validation Rules On Course Object:

1.Course Fees Cannot be Negative: From Object Manager select the course object and click on validation rules, give your rule a name and in error condition formula enter "Course_Fee__c<0", give error message according to the condition.

2.Course Fees Should Be Less than 3000\$: From Object Manager select the course object and click on validation rules, give your rule a name and in error condition formula enter "Course_Fee__c>=3000",give error message according to the condition.

The screenshot shows the Salesforce Setup interface. The breadcrumb trail is: SETUP > OBJECT MANAGER > Course. The left sidebar contains a list of setup options: 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, and Restriction Rules. The main content area is titled 'Validation Rules' and shows '1 Items, Sorted by Rule Name'. A table lists the validation rules:

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Course_fee_validation_rule	Course Fee	course fee cannot be less than zero or greater than 3000	✓	HARIKA BOTUKA, 1/4/2022, 5:15 AM

The bottom of the screen shows the Windows taskbar with the search bar, task icons, and system tray information including temperature (23°C), weather (Haze), and date/time (10:31, 18-01-2022).

Create Validation Rules On Trainer Object:

1.Full Name, Phone Number, Dateof Birth , Experience fields on the trainer object should not be blank: Search for Trainer Object from the object manager ,click on validation rules, click new and give your rule a name, in error condition formula enter "ISBLANK(Field Name)" and enter the error message according to the validation rule.

2. Age Should Not Be Less than 25 years: Enter "age__c<25" in error condition formula and enter the error message according to the rule.

3.Phone Number and Whatsapp Number should accept minimum 10 digits and maximum 15 digits: In error condition formula enter"NOT(Regex(Fieldname, "[0-9]{10,15}"))" and error message according to the validation rule.

The screenshot shows the Salesforce Setup interface for the 'Trainer' object. The 'Validation Rules' section is active, displaying a list of 6 rules. The rules are sorted by Rule Name. The table below represents the data shown in the screenshot:

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Trainer_Age_Validation_Rule	Date Of Birth	Age Should Not Be Less than 25 years	✓	HARIKA BOTUKA, 1/4/2022, 8:00 AM
Trainer_Dateof_Birth_Validation_Rule	Date Of Birth	Date of Birth on Trainer object should not be blank		HARIKA BOTUKA, 1/10/2022, 10:18 PM
Trainer_details_validation_rule	Phone number	Phone Number should accept minimum 10 digits and maximum 15 digits		HARIKA BOTUKA, 1/10/2022, 10:19 PM
Trainer_details_validation_rule_2	Whatsapp number	Whatsapp Number should accept minimum 10 digits and maximum 15 digits		HARIKA BOTUKA, 1/10/2022, 10:18 PM
Trainer_Full_Name_Validation_Rule	FullName	Full Name on the trainer object should not be blank	✓	HARIKA BOTUKA, 1/6/2022, 5:04 AM
Trainer_Phone_Number_Validation_Rule	Phone number	Phone Number on the trainer object should not be blank		HARIKA BOTUKA, 1/10/2022, 10:18 PM

Create Validation Rules On Student Object:

1. Lastname should not be Blank: enter the following error condition formula "ISBLANK(Field name)" and give the error message according to the rule.

2. Minimum 16 years of age: student should be minimum 16 years of age, in error condition formula enter "age__c<16" and give the error message according to the rule.

3.Phone and whatsappNumber should be minimum 10 digits : in error condition formula enter "NOT(Regex(Fieldname, "[0-9]{10}"))" and give the error message according to the

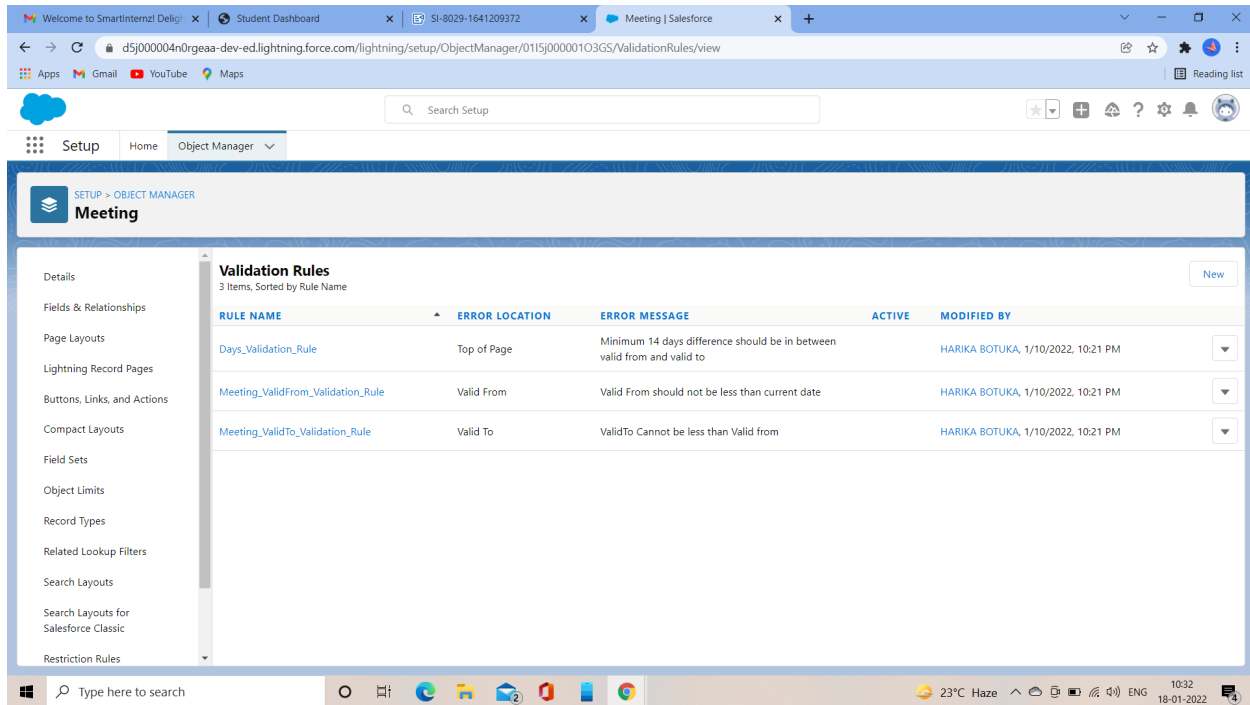
validation rule.

The screenshot shows the Salesforce Setup interface for the 'Student' 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, and Restriction Rules. The main content area is titled 'Validation Rules' and shows a list of 4 items, sorted by Rule Name. The table below lists the validation rules:

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Student_Age_Validation_Rule	Top of Page	Minimum 16 years of age: student should be minimum 16 years of age	✓	HARIKA BOTUKA, 1/6/2022, 6:36 AM
Student_details_validation_rule	Phone number	Phone Number should be minimum 10 digits		HARIKA BOTUKA, 1/10/2022, 10:09 PM
Student_details_validation_rule_2	Whatsapp number	Whatsapp Number should be minimum 10 digits		HARIKA BOTUKA, 1/10/2022, 10:09 PM
Student_Last_Name_Validation_Rule	Last Name	Lastname should not be Blank	✓	HARIKA BOTUKA, 1/6/2022, 5:16 AM

Create Validation Rules On Meeting Object:

1. Valid From should not be less than current date: Click on validation rules from meeting object, give your rule a name and in error condition formula enter "Valid_from__c < TODAY()" and enter the error message according to the rule.
2. Valid To Cannot be less than Valid from : Click on validation rules from meeting object, give your rule a name and in error condition formula enter "Valid_to__c < Valid_from__c " and enter the error message according to the rule.
3. Minimum 14 days difference should be in between valid from and valid to : Click on validation rules from meeting object, give your rule a name and in error condition formula enter "Valid_to__c - Valid_from__c < 14 " and enter the error message according to the rule.



Process Automation:

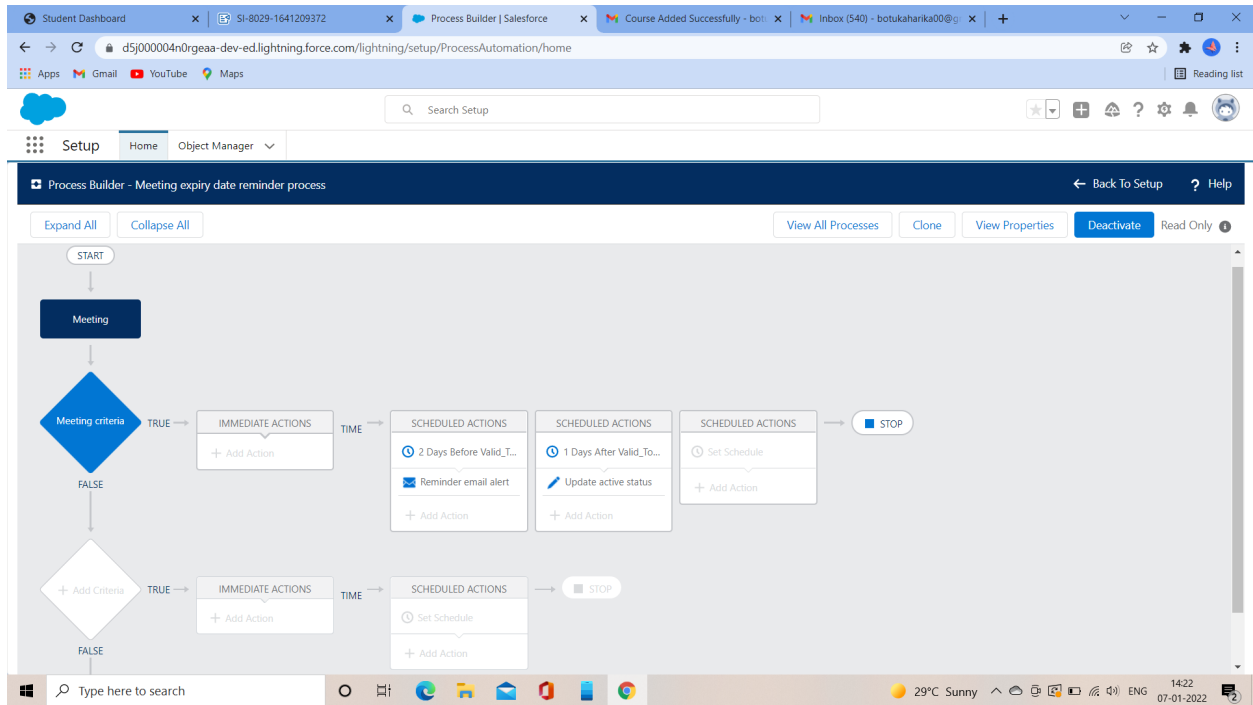
There are four automation tools in salesforce:

1. Approval Process
2. Process builder
3. Flows
4. Workflows

Creating Work Flow Rules On Meeting Object

1. create work flow rule:

1 day after valid to date the active check box will uncheck automatically and inform owner:



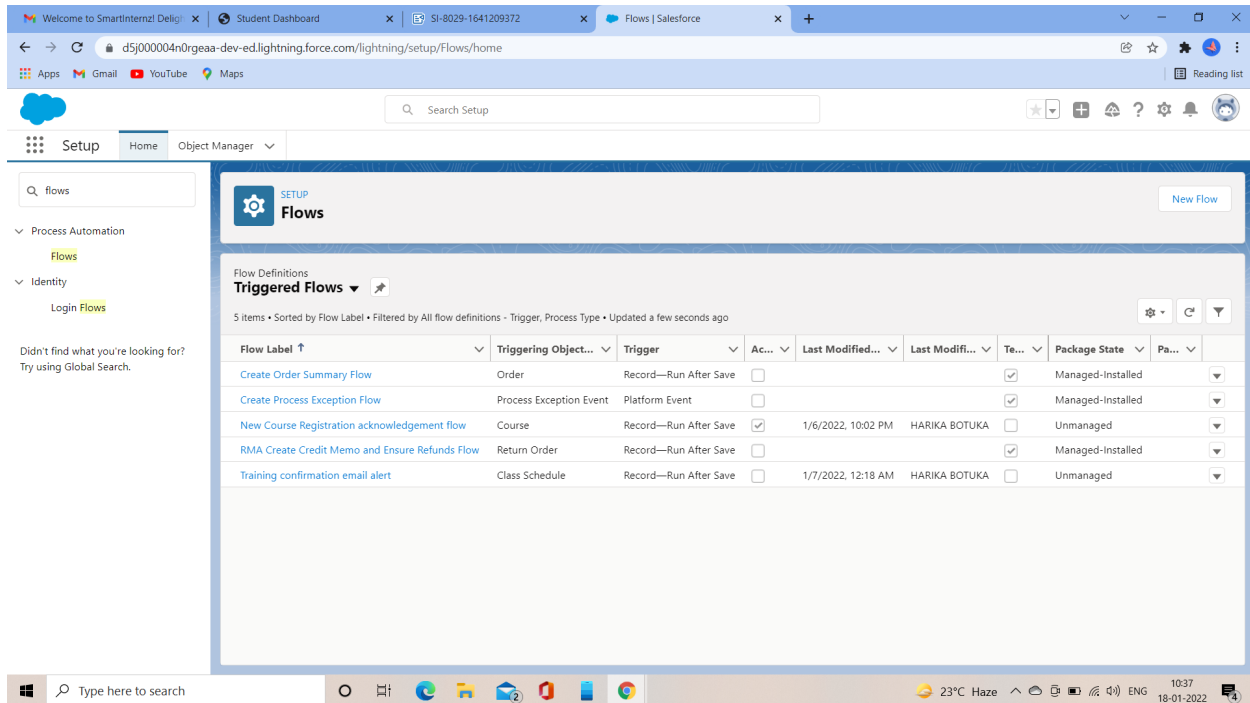
2.create work flow rule :

1 day before valid to date send a reminder email to owner:

Create Work Flow Rules On Class Object

1.create work flow rule:

2 days before valid to date send a reminder email to owner.



Creating Custom Pages Using Visualforce:

Visualforce is a framework that allows developers to build sophisticated, custom user interfaces that can be hosted natively on the Lightning platform. The Visualforce framework includes a tag-based markup language, similar to HTML, and a set of server-side “standard controllers” that make basic database operations, such as queries and saves, very simple to perform.

Creating New Course Using Custom Controller:

course visualforce page code:

```
<apex:page lightningStylesheets="true" controller="course">
  <apex:form >
    <apex:sectionHeader title="New Course"/>
    <apex:pageMessages ></apex:pageMessages>
    <apex:pageBlock >
      <apex:pageBlockSection >
        <apex:inputfield value="{!cr.Course_Name__c}"/>
        <apex:inputfield value="{!cr.Course_Category__c}"/>
        <apex:inputfield value="{!cr.Course_Fee__c}"/>
        <apex:inputfield value="{!cr.Course_Duration__c}"/>
        <apex:inputfield value="{!cr.Course_Description__c}"/>
      </apex:pageBlockSection>
    </apex:pageBlock>
  </apex:form>
</apex:page>
```

```

<center><apex:commandButton value="Save" action="{!save}"/></center>
</apex:pageBlock>
</apex:form>
</apex:page>

```

course custom controller code:

```

public with sharing class course

```

```

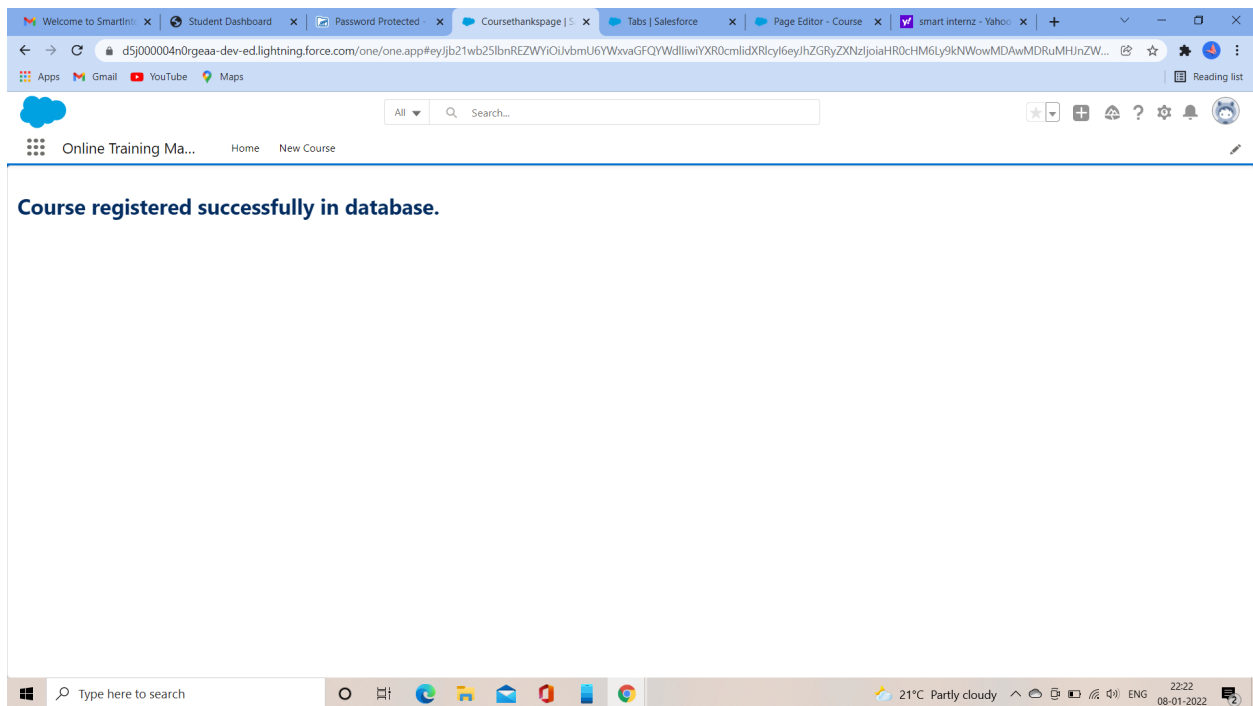
{
    public Course__c cr{get;set;}
    public course()
    {
        cr=new Course__c();
    }

```

```

    public void save() {
        if(cr.Course_Fee__c <0 || cr.Course_Fee__c>3000 )
        {
            ApexPages.addmessage(new ApexPages.message(ApexPages.severity.Error, 'Course Fee Should be less Than 3000 and Cannot Be Negative'));
        }
        else{
            insert cr;
            ApexPages.addmessage(new ApexPages.message(ApexPages.severity.Confirm, 'Record Inserted Successsfully'));
            cr=new Course__c();
        }
    }
}

```



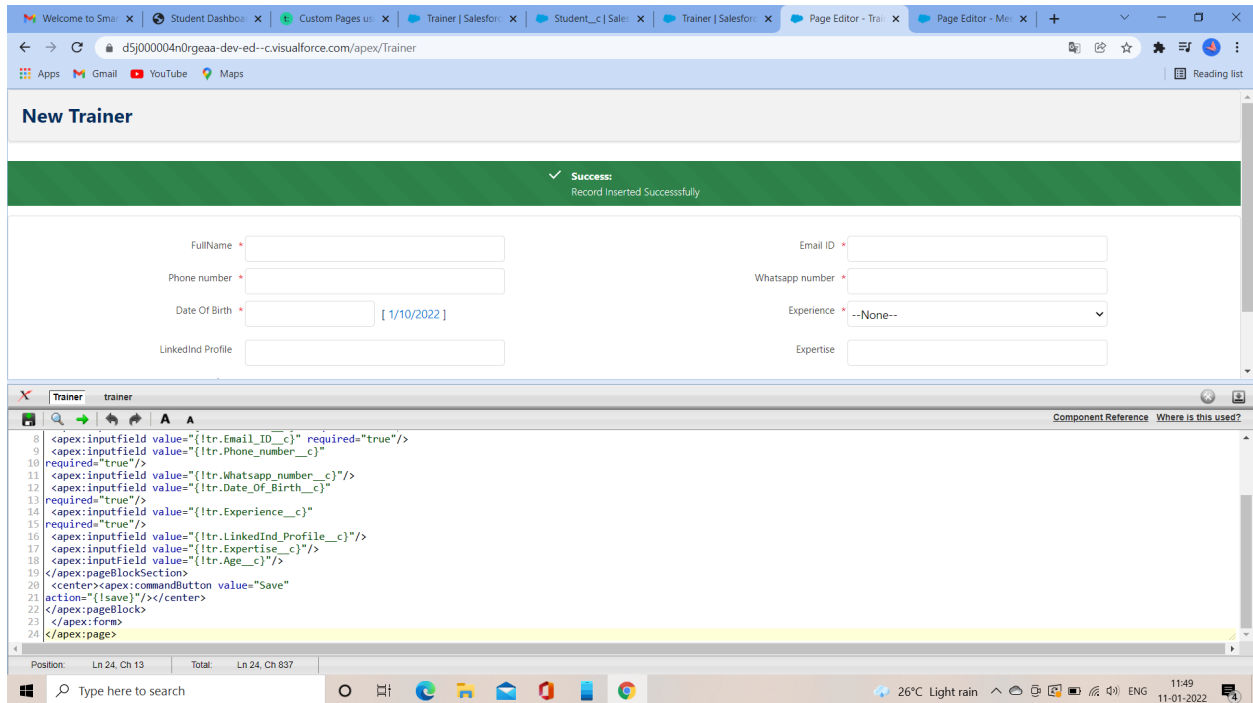
Create New Trainer Using Custom Controller

Using Trainer visualforce code:

```
<apex:page lightningStylesheets="true" controller="trainer">
  <apex:form >
    <apex:sectionHeader title="New Trainer"/>
    <apex:pageMessages ></apex:pageMessages>
    <apex:pageBlock >
      <apex:pageBlockSection >
        <apex:inputfield value="{!tr.FullName__c}" required="true"/>
        <apex:inputfield value="{!tr.Email__c}" required="true"/>
        <apex:inputfield value="{!tr.Phone_Number__c}" required="true"/>
        <apex:inputfield value="{!tr.Whatsapp_Number__c}" />
        <apex:inputfield value="{!tr.Date_Of_Birth__c}" required="true"/>
        <apex:inputfield value="{!tr.Experience__c}" required="true"/>
        <apex:inputfield value="{!tr.Linkedin_Profile__c}" />
        <apex:inputfield value="{!tr.Expertise__c}" />
        <apex:inputField value="{!tr.Age__c}" />
      </apex:pageBlockSection>
      <center><apex:commandButton value="Save" action="{!save}" /></center>
    </apex:pageBlock>
  </apex:form>
</apex:page>
```

Trainer controller code:

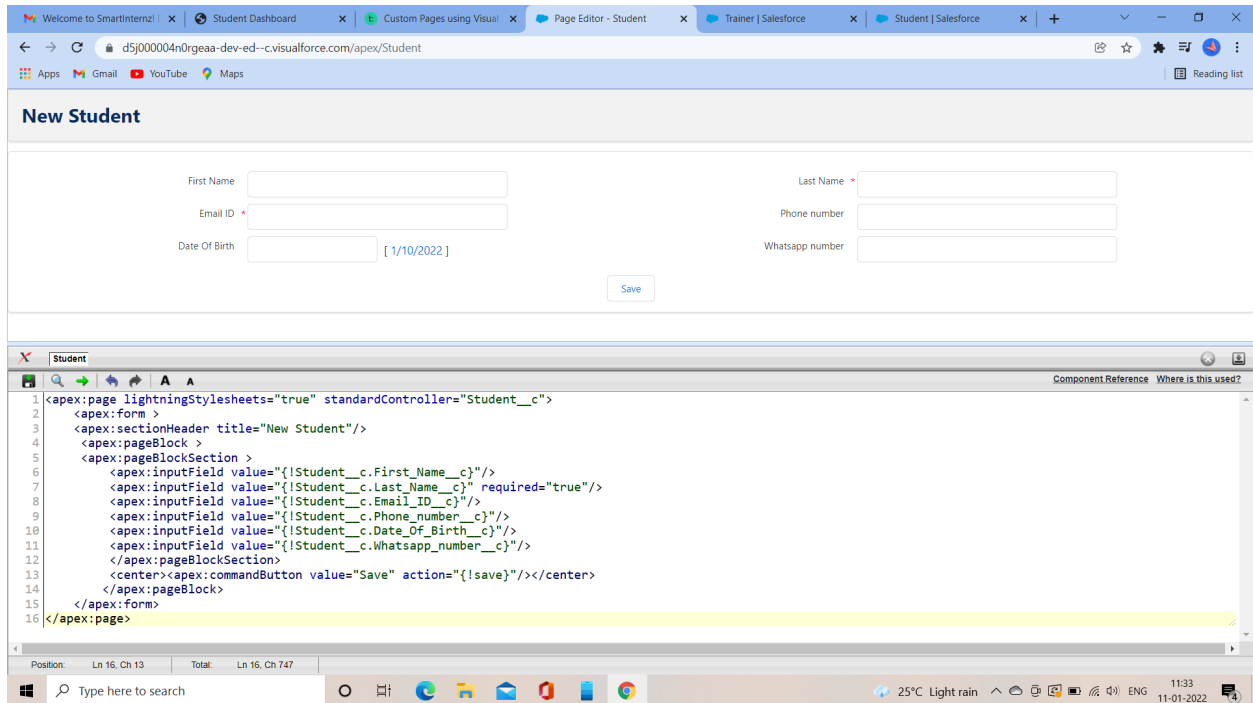
```
public with sharing class trainer {
  public Trainer__c tr{get; set;}
  public trainer()
  {
    tr=new Trainer__c();
  }
  public void save()
  {
    insert tr;
    ApexPages.addmessage(new ApexPages.message(ApexPages.severity.Confirm, 'Record Inserted Successsfully'));
    tr=new Trainer__c();
  }
}
```



New Student Using Standard Controller

Using Student visualforce code:

```
<apex:page lightningStylesheets="true" standardController="Students__c">
  <apex:form >
    <apex:sectionHeader title="New Student"/>
    <apex:pageBlock >
      <apex:pageBlockSection >
        <apex:inputField value="{!Students__c.First_Name__c}"/>
        <apex:inputField value="{!Students__c.Last_Name__c}" required="true"/>
        <apex:inputField value="{!Students__c.Email__c}"/>
        <apex:inputField value="{!Students__c.Phone_Number__c}"/>
        <apex:inputField value="{!Students__c.Date_Of_Birth__c}"/>
        <apex:inputField value="{!Students__c.Whatsapp_Number__c}"/>
      </apex:pageBlockSection>
      <center><apex:commandButton value="Save" action="{!save}"/></center>
    </apex:pageBlock>
  </apex:form>
</apex:page>
```



Create New Meeting Using Standard Controller

Meeting visualforce code:

```

<apex:page lightningStylesheets="true" standardController="Meeting__c">
  <apex:form >
    <apex:sectionHeader title="New Meeting"/>
    <apex:PageBlock >
      <apex:PageBlockSection >
        <apex:inputField value="{!Meeting__c.Valid_From__c}"/>
        <apex:inputField value="{!Meeting__c.Valid_To__c}"/>
        <apex:inputField value="{!Meeting__c.Meeting_Link__c}"/>
        <apex:inputField value="{!Meeting__c.Active__c}"/>
      </apex:PageBlockSection>
      <center><apex:commandButton value="Save" action="{!save}"/></center>
    </apex:PageBlock>
  </apex:form>
</apex:page>

```


The screenshot shows a web application interface for creating a new meeting. The top navigation bar includes links for Home, New Course, New Trainer, New Student, New Meeting (selected), and New Batch. The 'New Meeting' form is displayed with the following fields:

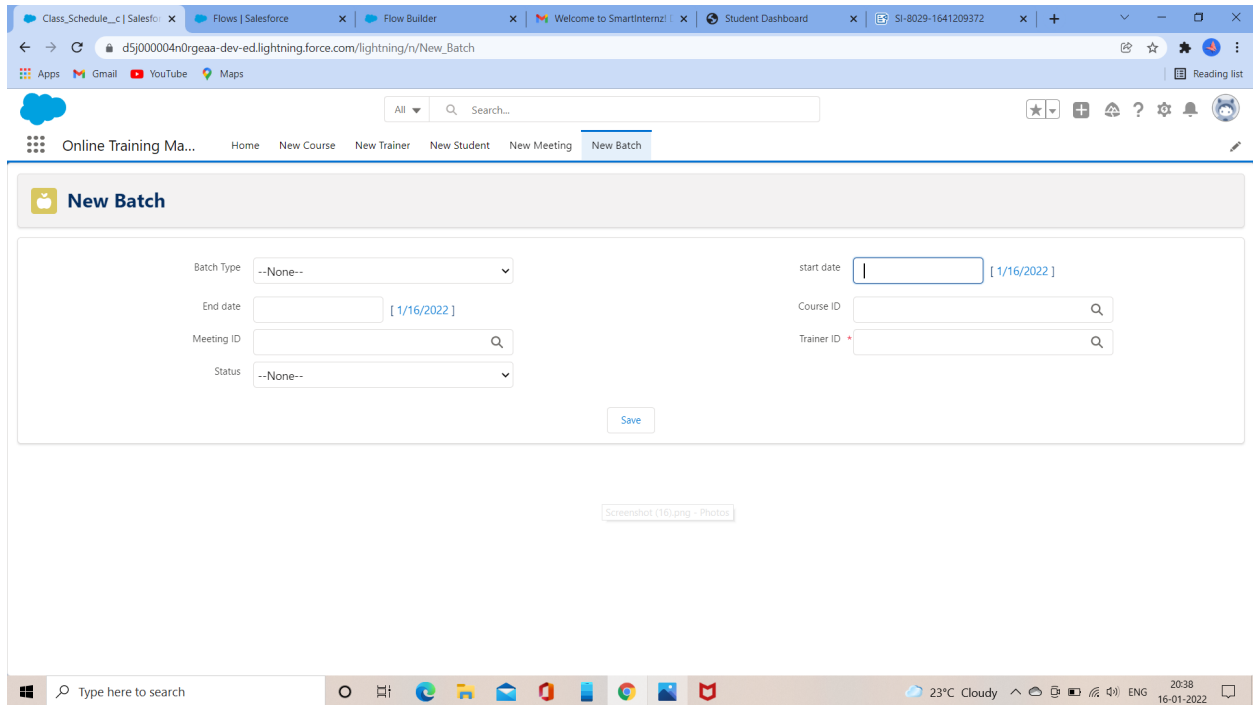
- Valid From:** A date picker set to 1/16/2022.
- Valid To:** A date picker set to 2/15/2022.
- Meeting Link:** A text input field.
- Active:** A checkbox that is checked.

A 'Save' button is located at the bottom right of the form. The bottom of the image shows a Windows taskbar with various application icons and system information like '23°C Cloudy' and the date '16-01-2022'.

Create New Batch Using Standard Controller

Using Batch visualforce code:

```
<apex:page lightningStylesheets="true" standardController="Batch__c">
  <apex:form >
    <apex:sectionHeader title="New Batch"/>
    <apex:pageBlock >
      <apex:pageBlockSection >
        <apex:inputField value="{!Batch__c.Batch_Type__c}"/>
        <apex:inputField value="{!Batch__c.Start_Date__c}"/>
        <apex:inputField value="{!Batch__c.End_Date__c}"/>
        <apex:inputField value="{!Batch__c.Course_ID__c}"/>
        <apex:inputField value="{!Batch__c.Meeting_ID__c}"/>
        <apex:inputField value="{!Batch__c.Trainer_ID__c}"/>
        <apex:inputField value="{!Batch__c.Status__c}"/>
      </apex:pageBlockSection>
      <center><apex:commandButton value="Save" action="{!save}"/></center>
    </apex:pageBlock>
  </apex:form>
</apex:page>
```



Triggers

A trigger is an Apex script that executes before or after data manipulation language (DML) events occur. Apex triggers enable you to perform custom actions before or after events to record in Salesforce, such as insertions, updates, or deletions. Just like database systems support triggers, Apex provides trigger support for managing records.

Trigger On Trainer Object

1. Create the following trigger on trainer object.

```
trigger age on Trainer__c (before insert,before update)
{
    for(Trainer__c tr :trigger.new)
    {
        date dob = tr.Date_of_Birth__c;
        Integer age = Integer.valueOf(dob.daysBetween(Date.Today()))/365;
        if(age<25){
            tr.Date_of_Birth__c.adderror('Age should be more than 25');
        }
    }
}
```

Trigger On Student Object

1.Create the following trigger on student object

```

trigger ageCheck on Students__c (before insert)
{
    for(Students__c std :trigger.new)
    {
        date dob = std.Date_of_Birth__c;
        Integer age = Integer.valueOf(dob.daysBetween(Date.Today()))/365;
        if(age<25){
            std.Date_of_Birth__c.adderror('Age should be more than 16');
        }
    }
}

```

Trigger On Meeting Object

1. Create the following trigger on meeting object.

```

trigger Datecheck on Meeting__c (before insert)
{
    for(Meeting__c mt :trigger.new)
    {
        if(mt.Valid_From__c<Date.today() )
        {
            mt.Valid_From__c.adderror('Valid From Date Should Be Greater Than Today');
        }
        if(mt.Valid_To__c < mt.Valid_From__c)
        {
            mt.Valid_To__c.adderror('Valid To Date shouldbe greater than valid from date');
        }

        Integer noOfDays = (mt.Valid_From__c). daysBetween(mt.Valid_To__c);
        if(noOfDays < 14)
        {
            mt.Valid_From__c.adderror('Minimum 14 days difference  should be present between one meeting');
        }
    }
}

```

Reports and Dashboards:

Reports:

Create reports on:

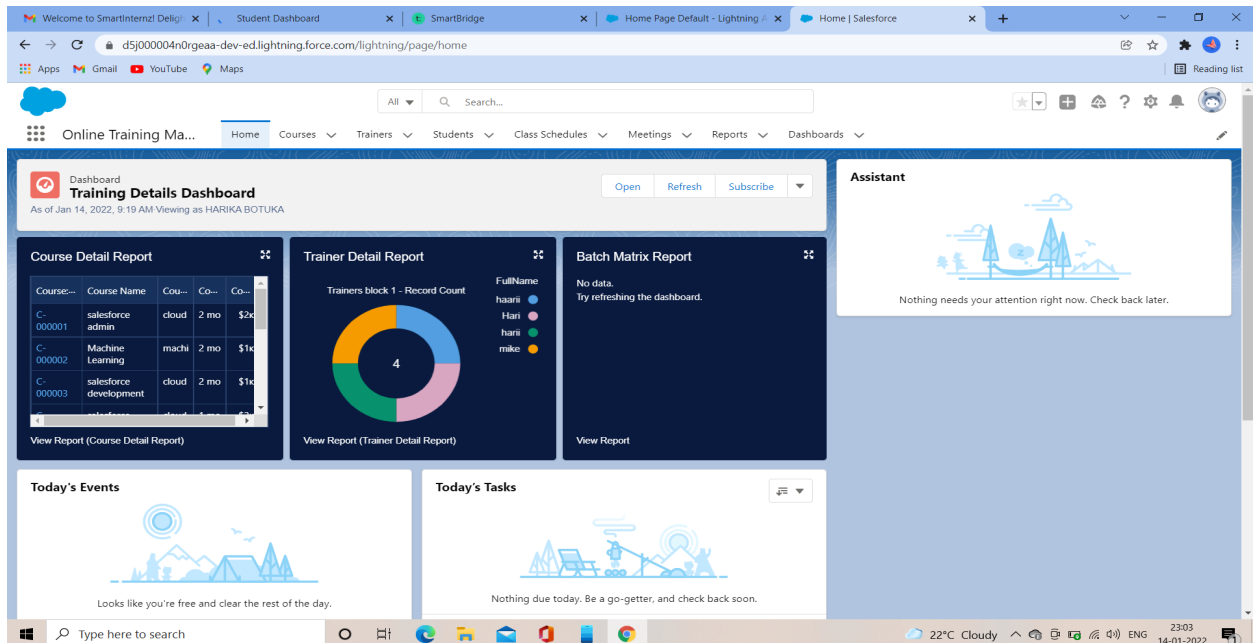
Course Detail Report

Trainer Detail Report

Batch Matrix Report

Create Dashboard Named as Training Details Dashboard:

Add Those 3 Reports in Training Details Dashboard And Add to Home Page



Conclusion:

In this project, I've learnt many things like how to create custom objects, validate rules on objects, work on process automations, learnt how to create custom pages using visual force and Triggers in Easiest way. While doing all these, I got many errors but all were solved by next session. Trainer, mentors were helped and supported to complete such a great project. I really enjoyed this short period of time in completing my daily tasks. Now, am able to solve my errors by my own and given me a real time experience.

THANK YOU.