

Day 1 :

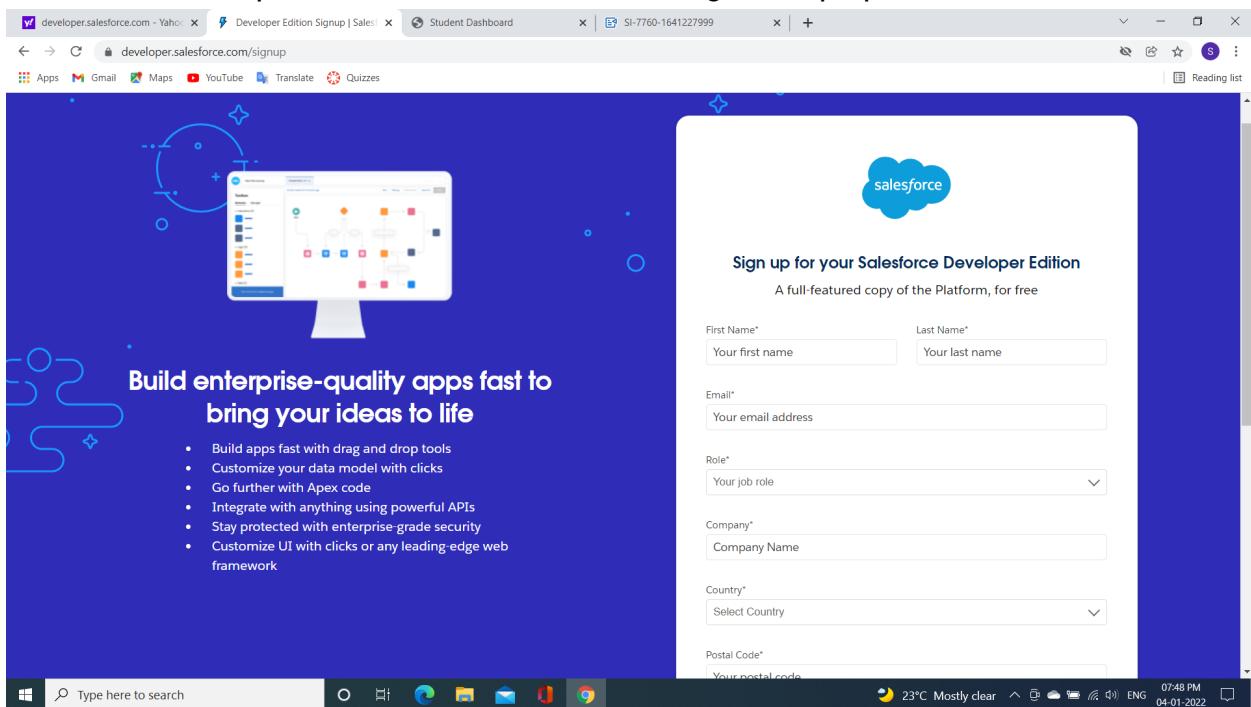
Topic: Create Salesforce developer org

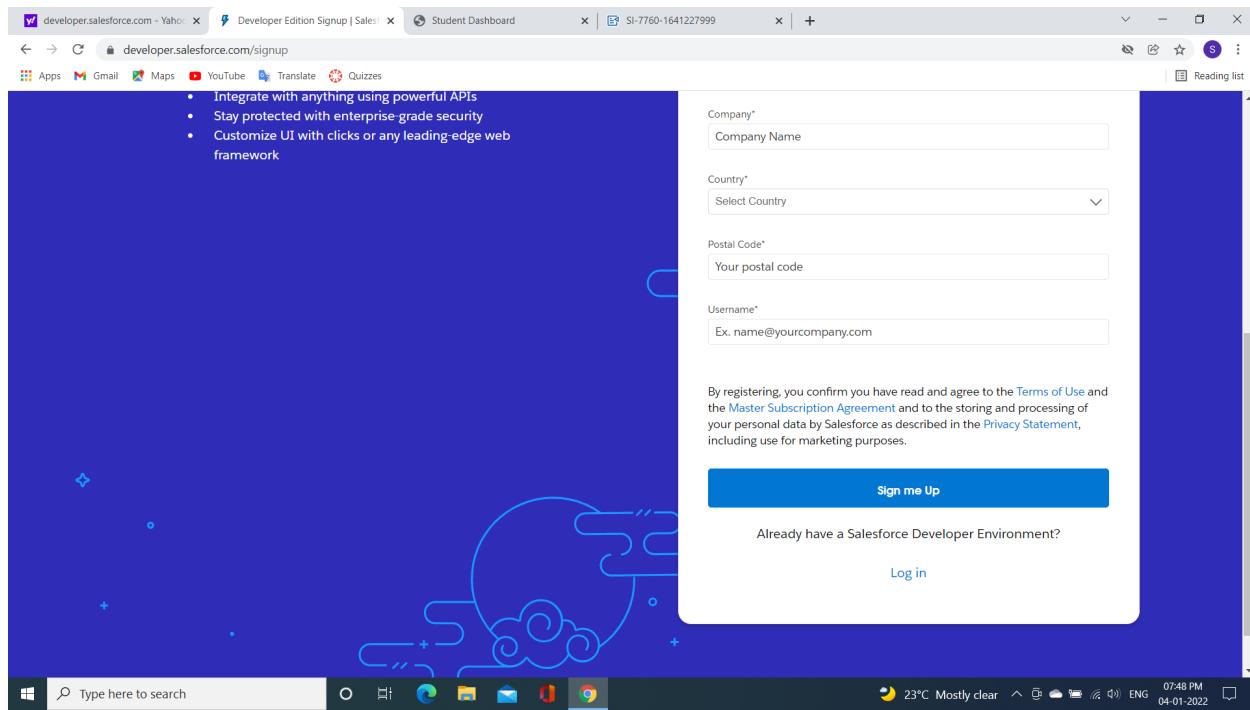
Milestone/Activities: create a developer account
account activation
login to salesforce account

Detailed Description:

1.create a developer account

- >first open browser and type the url:developer.salesforce.com
- >this take us to salesforce developer page
- >then we have to click on signup button for create an account
- >we fill required details then we click on sign me up option



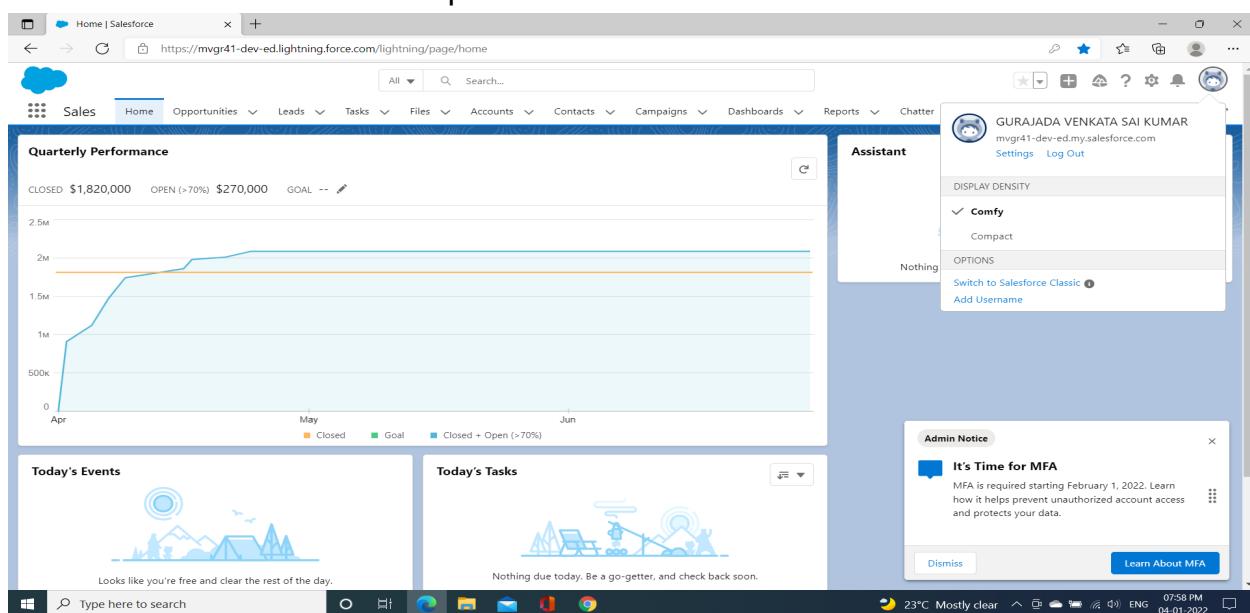


2.account activation:

- >we get a OTP to our mail when we submit
- >we need to enter that OTP to active an account

3.login to salesforce account:

- >open browser and type url as :salesforce.com
- >click on login button
- >enter valid mail id and password



Day - 2 :

Topic: Create The Custom Objects For The Application

Milestones/Activities:

1.Create the following Custom Objects For the Application.

1.1. College

1.2. Application Form.

1.3. Students.

1.4. Subjects.

2.Create Fields On College Object

Create the following fields on the college object.

Field Name	Data Type	Required	Values
Record Info	Text		
College Fees	Currency(7,2)	Yes	
Hostel Fees	Currency(6,2)	Yes	
College Name	Picklist(Refer Business Logic In Milestones)	Yes	
Email	Picklist		blr@mit.co.in hyd@mit.co.in mum@mit.co.in maa@mit.co.in ccu@mit.co.in del@mit.co.in
Capacity Of Students	Picklist		500-1000, 1000-2500, 2500-6000, 6000-10000

3.Create Fields On Application Form Object

Create the following fields on the application form object.

Field Name	Data Type	Required	Values
Application Form ID	Auto number		F-{000000} Starting Number=1
Address	Text(255)	Yes	
College	Master-Detail(College)	Yes	
College Fees	Formula(Currency)		
Hostel Fees	Formula(Currency)		

date of Birth	Date	Yes	
Email	Email(Unique)	Yes	
Guardian Name	Text(30)	Yes	
Looking For Hostel Stay	Checkbox(default=Uncheck)		
Ready To Join	Checkbox(default=Uncheck)		
Student Name	Text(30)	Yes	
Phone	Phone	Yes	

4.Create Fields On Student Object

Create the following fields on the student object.

Field Name	Data Type	Required	Values
Student Name	Text		
Address	Text(255)	Yes	
Application Form	Lookup(Application Form)		
College Name	Formula(Text)		
Date Of Birth	Date	Yes	
Guardian Name	Text(30)	Yes	
Phone	Phone	Yes	

5.Create Fields On Subject Object

Create the following fields On the Subject object.

Field Name	Data Type	Required	Values
Subject ID	AutoNumber		S-{000000} Starting Number=1
Paper1	Picklist(Refer Business Logic Milestone)		
Paper2	Picklist(Refer Business logic Milestone)		
Student	Lookup(Student)		

detail description:

1.1.college object creation:

- >click gear icon/setting symbol to navigate to sales force setup.
- >click on the object manager tab to navigate to object manager page.
- >click on new button to create object.
- >select custom object option.
- >enter the object name and api name.
- >click on next >click on next >click

The screenshot shows the Salesforce Setup interface with the 'Object Manager' tab selected. A new object named 'College' is being created. The 'Custom Object Information' section includes fields for 'Label' (College), 'Plural Label' (Colleges), and 'Object Name' (College). Other sections like 'Details' and 'Optional Features' are also visible.

on save and new button.

1.2:application form object creation

>enter the object name and api name.

- >click on next >click on next >click on save and new button.



Search Setup

Object Manager

Object Manager

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles. [Tell me more](#) [Don't show this message again](#)

Custom Object Definition Edit [Save](#) [Save & New](#) [Cancel](#)

Custom Object Information ! = Required information

The singular and plural labels are used in tabs, page layouts, and reports.

Label Example: Account

Plural Label Example: Accounts

Starts with vowel sound

The Object Name is used when referencing the object via the API

Object Name Example: Account

Description

Context-Sensitive Help Setting

Open the standard Salesforce.com Help & Training window

Open a window using a Visualforce page

Content Name

Enter Record Name Label and Format

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name Example: Account Name

Data Type

Optional Features

Allow Documents

1.3:student object creation

>enter the object name and api name.

>click on next >click on next >click on save and new button.



Search Setup

Object Manager

Object Manager

Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles. [Tell me more](#) [Don't show this message again](#)

Custom Object Definition Edit [Save](#) [Save & New](#) [Cancel](#)

Custom Object Information ! = Required information

The singular and plural labels are used in tabs, page layouts, and reports.

Label Example: Account

Plural Label Example: Accounts

Starts with vowel sound

The Object Name is used when referencing the object via the API

Object Name Example: Account

Description

Context-Sensitive Help Setting

Open the standard Salesforce.com Help & Training window

Open a window using a Visualforce page

Content Name

Enter Record Name Label and Format

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name Example: Account Name

Data Type

Optional Features

Allow Documents

1.4:subject object creation:

>enter the object name and api name.

>click on next >click on next >click on save and new button.

The screenshot shows the Salesforce Setup interface with the 'Object Manager' tab selected. A message at the top states: 'Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles.' Below this, the 'Custom Object Definition Edit' section is displayed. The 'Label' field is set to 'Students' and the 'Plural Label' is also 'Students'. The 'Object Name' field is also set to 'Students'. There is a 'Description' text area and a 'Content Name' dropdown set to 'None'. Under 'Record Name Label and Format', the 'Record Name' is 'Students Name' and the 'Data Type' is 'Text'. At the bottom, there are optional features and a save/cancel button bar.

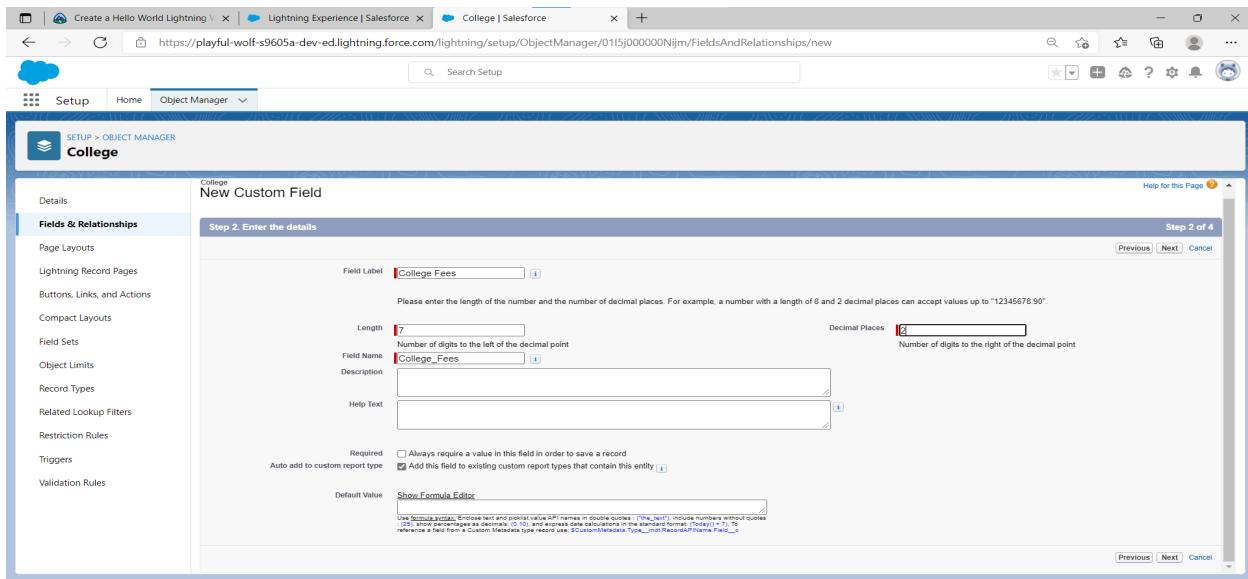
2.Create Fields On College Object:

2.1:Record Info field:

- >click on object manager tab on setup window
- >click on college object name College_c__
- >then click on fields and relationship and click on new
- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

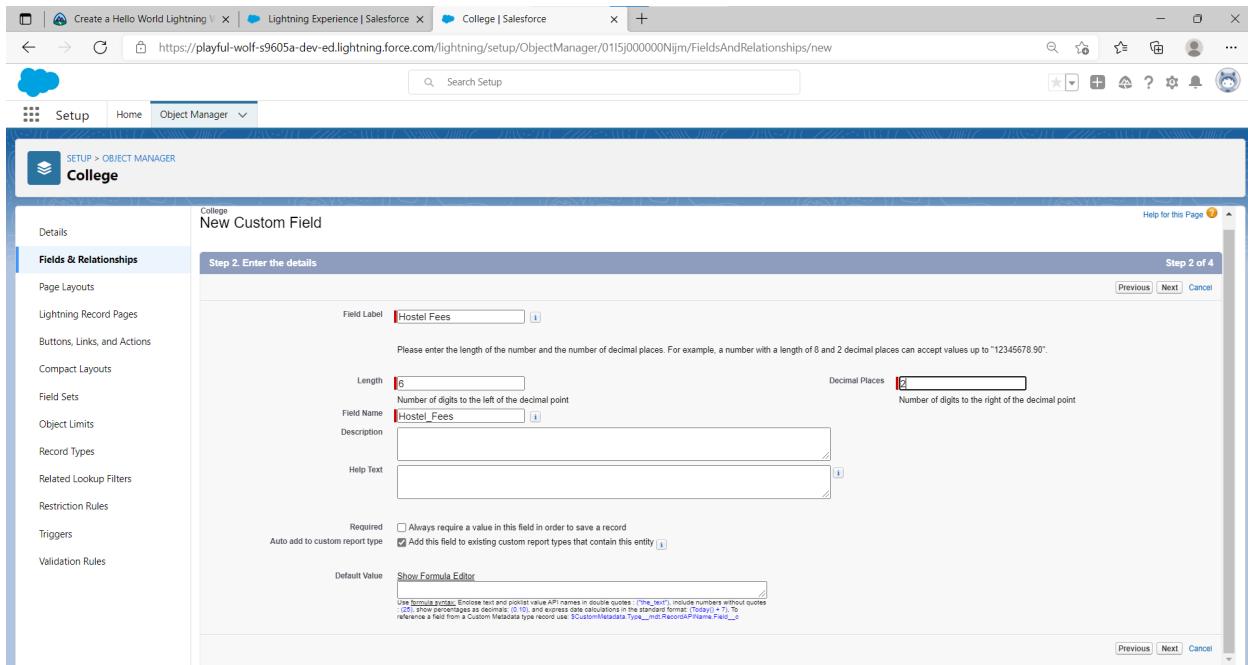
2.2:College Fees field

- >select the data type and click next
- >enter the valid label for field and click next
- >check the visibility and click next
- >click on save and new



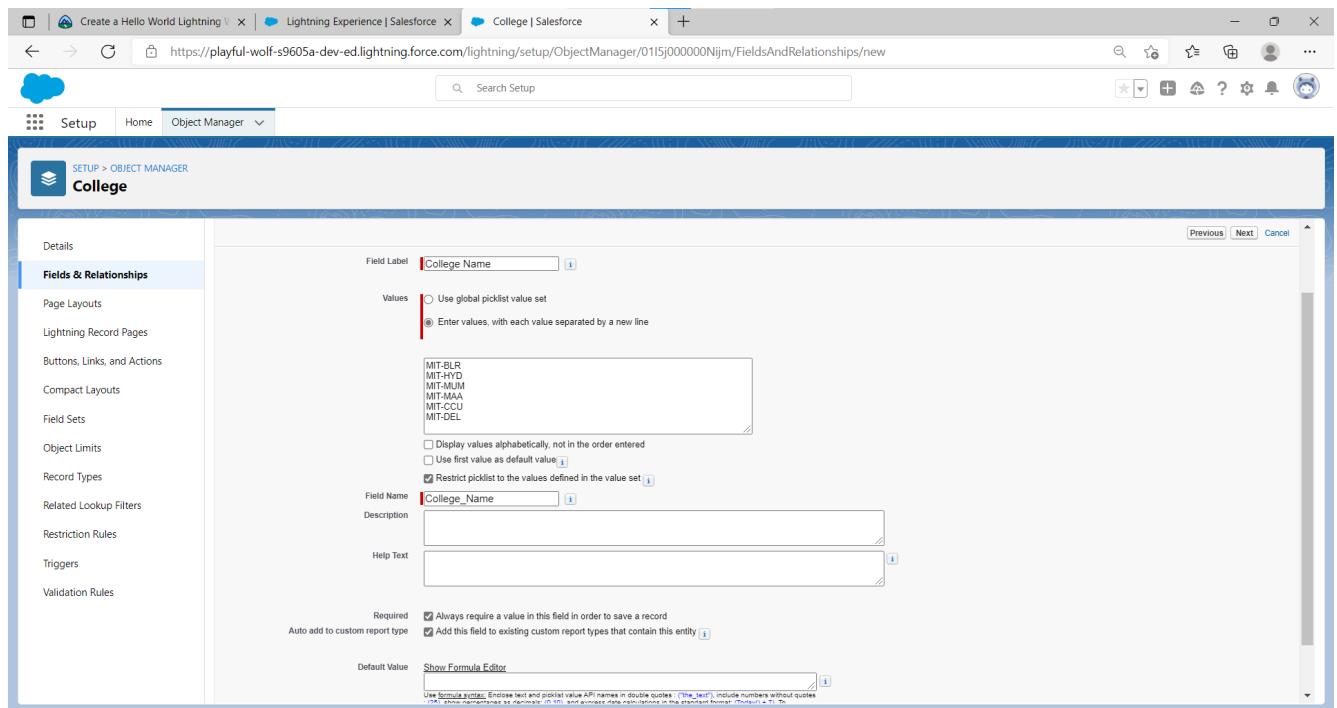
2.3:Hostel Fees field:

- >select the data type and click next
- >enter the valid label for field and click next
- > check the visibility and click next
- >click on save and new



2.4:College Name field:

- >select the data type and click next
- >enter the valid label for field and click next
- >check the visibility and click next
- >click on save and new



2.5:Email field:

- >select the data type and click next
- >enter the valid label for field and click next
- >check the visibility and click next
- >click on save and new

Step 2. Enter the details

Field Label: Email

Values:

- Use global picklist value set
- Enter values, with each value separated by a new line

```
sh@mit.co.in
hyd@mit.co.in
mumbai@mit.co.in
msa@mit.co.in
pune@mit.co.in
del@mit.co.in
```

Display values alphabetically, not in the order entered

Use first value as default value

Restrict picklist to the values defined in the value set

Field Name: Email

Description:

Help Text:

Required: Always require a value in this field in order to save a record

Auto add to custom report type: Add this field to existing custom report types that contain this entity

2.6:capacity of students field:

- >select the data type and click next
- >enter the valid label for field and click next
- >check the visibility and click next
- >click on save and new

Step 2. Enter the details

Field Label: Capacity Of Students

Values:

- Use global picklist value set
- Enter values, with each value separated by a new line

```
500-1000
1000-2500
2500-6000
6000-10000
```

Display values alphabetically, not in the order entered

Use first value as default value

Restrict picklist to the values defined in the value set

Field Name: Capacity_Of_Students

Description:

Help Text:

Required: Always require a value in this field in order to save a record

Auto add to custom report type: Add this field to existing custom report types that contain this entity

Default Value: Show Formula Editor

The screenshot shows the Salesforce Setup interface with the 'Object Manager' tab selected. Under the 'College' object, the 'Fields & Relationships' section is displayed. The table lists 10 fields, each with its field label, name, data type, controlling field, and indexed status. The fields listed are:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
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(20)		

3.Create Fields On College Object:

3.1:Application Form ID field:

- >click on object manager tab on setup window
- >click on college object name Application_Form_c__
- >then click on fields and relationship and click on new
- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

The screenshot shows the Salesforce Setup interface with the following details:

- Page Layouts**: Under Fields & Relationships.
- Custom Field Definition Detail** for the field **Application Form ID**.
- Field Information** section:
 - Field Label: Application Form ID
 - Field Name: Application_Form_ID
 - API Name: Application_Form_ID_c
 - Description: (empty)
 - Help Text: (empty)
 - Data Owner: (empty)
 - Field Usage: (empty)
 - Data Sensitivity Level: (empty)
 - Compliance Categorization: (empty)
- General Options** section:
 - External ID:
- Auto Number Options** section:
 - Display Format: F-{000000}
- Object Name**: Application Form
- Data Type**: Auto Number
- Created By**: GURAJADA VENKATA SAI KUMAR 1/5/2022, 7:42 AM
- Modified By**: GURAJADA VENKATA SAI KUMAR 1/5/2022, 7:42 AM

3.2:create address field:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

The screenshot shows the Salesforce Setup interface with the following details:

- Page Layouts**: Under Fields & Relationships.
- Custom Field Definition Detail** for the field **Address**.
- Field Information** section:
 - Field Label: Address
 - Field Name: Address
 - API Name: Address_c
 - Description: (empty)
 - Help Text: (empty)
 - Data Owner: (empty)
 - Field Usage: (empty)
 - Data Sensitivity Level: (empty)
 - Compliance Categorization: (empty)
- General Options** section:
 - Required:
 - Unique:
 - Case Sensitive:
 - External ID:
 - Encrypted:
 - Default Value: (empty)
- Text Options** section:
 - Length: 20
- Validation Rules** section: (empty)

3.3:create college field:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

The screenshot shows the Salesforce Setup interface under the Object Manager. A custom field named 'College' is being defined for the 'Application Form' object. The 'Field Information' section includes fields for Field Label ('College'), Field Name ('College'), API Name ('College__c'), Description, Help Text, Data Owner, Field Usage, Data Sensitivity Level, and Compliance Categorization. The 'Master-Detail Options' section shows 'Related To' set to 'College' and 'Sharing Setting' set to 'ReadWrite'. The 'Validation Rules' section indicates 'No validation rules defined.' The right side of the screen displays the 'Object Manager' sidebar with various setup categories like Details, Fields & Relationships, Page Layouts, and Record Types.

3.4:College fees field creation:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

College Fees

Custom Field Definition Detail

Field Information

- Field Label: College Fees
- Field Name: College_Fees
- API Name: College_Fees__c
- Description:
- Help Text:
- Data Owner:
- Field Usage:
- Data Sensitivity Level:
- Compliance Categorization:

General Options

- Required:
- Default Value: 0

Currency Options

- Length: 10
- Decimal Places: 2

Validation Rules

No validation rules defined.

3.5:create Hostel fees field:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

Hostel Fees

Custom Field Definition Detail

Field Information

- Field Label: Hostel Fees
- Field Name: Hostel_Fees
- API Name: Hostel_Fees__c
- Description:
- Help Text:
- Data Owner:
- Field Usage:
- Data Sensitivity Level:
- Compliance Categorization:

General Options

- Required:
- Default Value: 0

Currency Options

- Length: 10
- Decimal Places: 2

Validation Rules

No validation rules defined.

3.6:create date of birth field:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

The screenshot shows the Salesforce Setup interface for creating a custom field. The left sidebar shows various setup categories like Details, Fields & Relationships, Page Layouts, etc. The main area is titled 'Custom Field Definition Detail' for the 'date of Birth' field on the 'Application Form' object. The field is defined as a Date type with a required checkbox checked. It has a description and is visible on the application form.

3.7:create email field:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

3.8:create guardian name field:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

3.9:create looking for hostel field:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

The screenshot shows the Salesforce Setup interface for creating a custom field. The URL in the browser is <https://playful-wolf-s9605a-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01Ij000000Njjw/FieldsAndRelationships/00N5j000008SSAP/view>. The page title is "Application Form | Salesforce". The main content area displays the "Custom Field Definition Detail" for a field named "Looking For Hostel". The field information includes:

- Field Label: Looking For Hostel
- Field Name: Looking_For_Hostel
- API Name: Looking_For_Hostel_c
- Description: (empty)
- Help Text: (empty)
- Data Owner: (empty)
- Field Usage: (empty)
- Data Sensitivity Level: (empty)
- Compliance Categorization: (empty)
- Created By: GURAJADA VENKATA SAI KUMAR 1/5/2022, 7:50 AM
- Modified By: GURAJADA VENKATA SAI KUMAR 1/5/2022, 7:50 AM

Below the field information, there are sections for "General Options" (Default Value: Unchecked) and "Field Dependencies" (No dependencies defined). A "Validation Rules" section is also present (No validation rules defined). The left sidebar lists various setup categories like Details, Fields & Relationships, Page Layouts, Lightning Record Pages, etc.

3.10:create ready to join field:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

SETUP > OBJECT MANAGER

Application Form

Fields & Relationships

Details

Application Form Custom Field
Ready To Join

Custom Field Definition Detail

Field Information

Field Label	Ready To Join	Object Name	Application Form
Field Name	Ready_To_Join	Data Type	Checkbox
API Name	Ready_To_Join__c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			

Created By: GURAJADA VENKATA SAI KUMAR 1/5/2022, 7:51 AM Modified By: GURAJADA VENKATA SAI KUMAR 1/5/2022, 7:51 AM

General Options

Default Value: Unchecked

Field Dependencies: No dependencies defined.

Validation Rules: No validation rules defined.

[Back To Top](#) Always show me ▾ more records per related list

3.10:create student name field:

- >select the data type and click next
- >enter the valid label for field and check required check box and click next
- >check the visibility and click next
- >click on save and new

SETUP > OBJECT MANAGER

Application Form

Fields & Relationships

Details

Edit Application Form Custom Field
Student Name

Custom Field Definition Edit

Field Information

Field Label	Student Name	Data Type	Text
Field Name	Student_Name		
Description			
Help Text			
Data Owner	User		
Field Usage	--None--		
Data Sensitivity Level	--None--		
Compliance Categorization	Available: PII, HIPAA, GDPR, PCI Chosen:		

General Options

Required: Always require a value in this field in order to save a record

Unique: Do not allow duplicate values

- Treat "ABC" and "abc" as duplicate values (case insensitive)
- Treat "ABC" and "abc" as different values (case sensitive)

External ID: Set this field as the unique record identifier from an external system

Encrypted: Encrypt the contents of this field

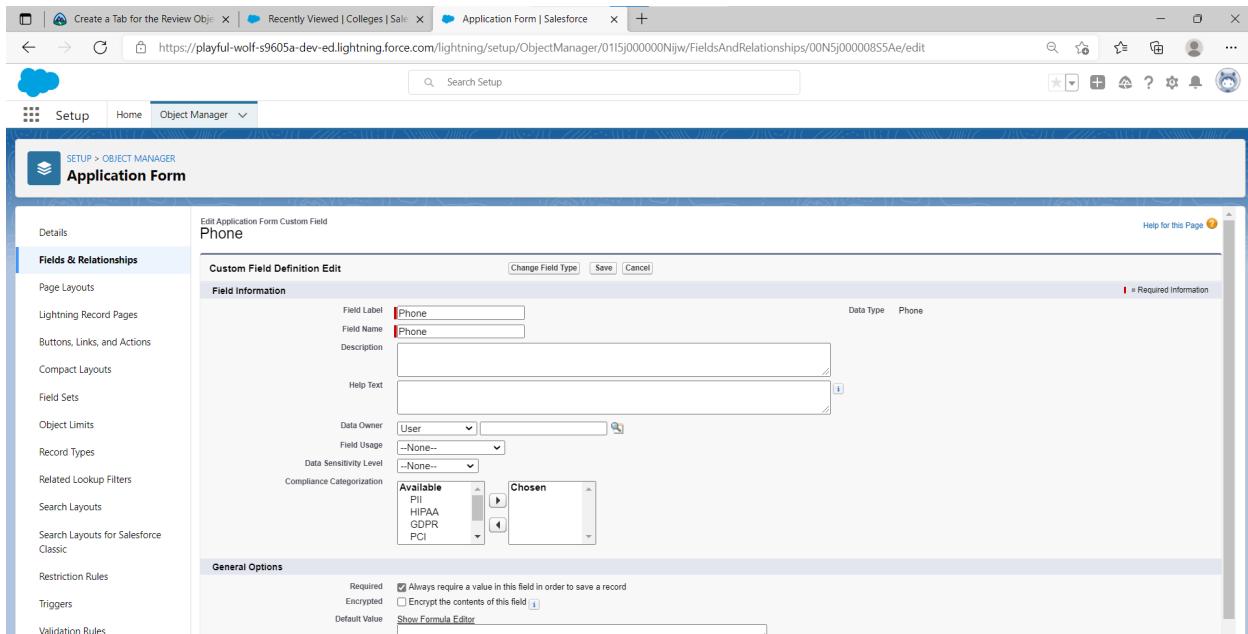
3.11:create phone field:

>select the data type and click next

>enter the valid label for field and check required check box and click next

>check the visibility and click next

>click on save and new



The screenshot shows the 'Custom Field Definition Edit' page for creating a new field named 'Phone'. The 'Field Label' is set to 'Phone', 'Field Name' to 'Phone', and 'Data Type' to 'Phone'. The 'Required' checkbox is checked. Under 'General Options', there are fields for 'Required', 'Encrypted', and 'Default Value'. A 'Show Formula Editor' button is also present. The 'Available' section lists PII, HIPAA, GDPR, and PCI. The 'Chosen' section is empty. The 'Help Text' and 'Description' fields are empty.

Custom Field Definition Edit

Field Information

Field Label: Phone
Field Name: Phone
Description:
Help Text:
Data Owner: User
Field Usage: None
Data Sensitivity Level: None
Compliance Categorization:

General Options

Required: Always require a value in this field in order to save a record
Encrypted: Encrypt the contents of this field
Default Value:
Show Formula Editor

Available: PII, HIPAA, GDPR, PCI
Chosen:

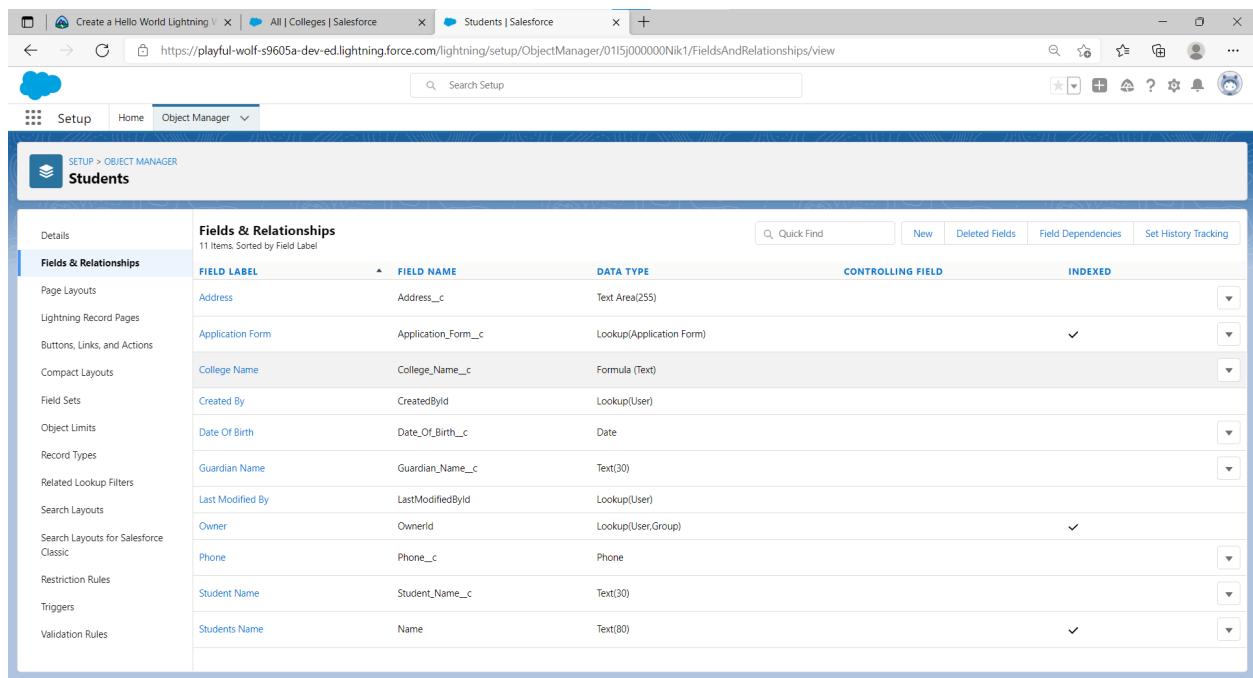
Fields & Relationships

15 items. Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text(20)		
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	Currency(10, 2)		
Created By	CreatedBy	Lookup(User)		
date of Birth	date_of_Birth__c	Date		
Email	Email__c	Email		
Guardian Name	Guardian_Name__c	Text(20)		
Hostel Fees	Hostel_Fees__c	Currency(10, 2)		
Last Modified By	LastModifiedBy	Lookup(User)		

4.Create Fields On Student Object

>click on object manager tab on setup window
 >click on college object name Students__c__
 >then click on fields and relationship and click on new
 >select the data type and click next
 >enter the valid label for field and check required check box and click next
 >check the visibility and click next
 >click on save and new



The screenshot shows the Salesforce Setup interface with the URL https://playful-wolf-s9605a-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01i5j000000Nik1/FieldsAndRelationships/view. The page title is 'Students | Salesforce'. The main content area is titled 'Fields & Relationships' and lists 11 items sorted by Field Label. The table columns are FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The data includes:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address_c	Text Area(255)		
Application Form	Application_Form_c	Lookup(Application Form)		✓
College Name	College_Name_c	Formula (Text)		
Created By	CreatedById	Lookup(User)		
Date Of Birth	DateOfBirth_c	Date		
Guardian Name	Guardian_Name_c	Text(30)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone	Phone_c	Phone		
Student Name	Student_Name_c	Text(30)		
Students Name	Name	Text(80)		✓

5.Create Fields On Subject Object

>click on object manager tab on setup window
 >click on college object name Subject__c__
 >then click on fields and relationship and click on new
 >select the data type and click next
 >enter the valid label for field and check required check box and click next
 >check the visibility and click next
 >click on save and new

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Paper1	Paper1__c	Picklist		
Paper2	Paper2__c	Picklist		
Students	Students__c	Lookup(Students)		✓
Subject ID	Subject_ID__c	Auto Number		
Subjects Name	Name	Text(80)		✓

Day-3:

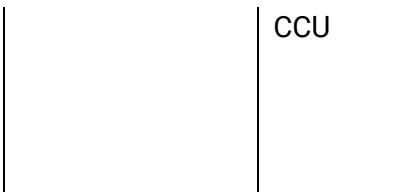
Topic : Adding Business Logic To Application

Milestones/Activities: 1.create global pick list values

Create the following global picklist value sets for the application.

a)College

Picklist Value Name	Values
College	MIT-HYD MIT-BLR MIT-MUM MIT-MAA MIT-DEL MIT-



b)Paper1

Picklist Value Name	Values
Paper 1	APEX JAVA C C++

c)Paper2

Picklist Value Name	Values
Paper2	MATHEMATICS ENGLISH STATISTICS

2.create field dependencies

- 2.1. Create field dependency between college Name and Email, where the controlling field is college Name and dependent field is Email. Select the email ids according to the college names.
- 2.2. Create field dependency between college Name and capacity of students, where the controlling field is college Name and dependent field is Capacity of Students. Select the values according to your wish.

3.create validation rules

Create a validation rule on the college object such that the college name and record info should have the same name.

TEXT(College_Name__c) <> Name

Description:

1.create global pick list values

- 1.1 college pick list values creation
>go to home in setup page

- >on quick finder type pick list value set and select
- >click on new option
- >enter the label and name as college
- >enter the values on value set
- >click on next>click on next >click on save and new

The screenshot shows the Salesforce Setup interface for creating a Global Value Set. The page title is "Picklist Value Sets". The "Information" section shows a single value "College". The "Values Used" section shows six active values: MIT-HYD, MIT-BLR, MIT-MUM, MIT-MAA, MIT-DEL, and MIT-CCU, each with a "Default" checkbox and a "Modified By" field showing "GURAJADA VENKATA SAI KUMAR" and the date "1/5/2022, 9:13 AM". The "Values" table has columns: Action, Values, API Name, Default, Chart Colors, and Modified By.

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	MIT-HYD	MIT-HYD	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:13 AM
Edit Del Deactivate	MIT-BLR	MIT-BLR	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:13 AM
Edit Del Deactivate	MIT-MUM	MIT-MUM	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:13 AM
Edit Del Deactivate	MIT-MAA	MIT-MAA	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:13 AM
Edit Del Deactivate	MIT-DEL	MIT-DEL	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:13 AM
Edit Del Deactivate	MIT-CCU	MIT-CCU	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:13 AM

1.2 paper1 pick list values creation

- >go to home in setup page
- >on quick finder type pick list value set and select
- >click on new option
- >enter the label and name as paper1
- >enter the values on value set
- >click on next>click on next >click on save and new

Global Value Set Detail

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	APEX	APEX	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:14 AM
Edit Del Deactivate	JAVA	JAVA	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:14 AM
Edit Del Deactivate	C	C	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:14 AM
Edit Del Deactivate	C++	C++	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:14 AM

Fields Where Used

Field Label	Object	Data Type	Controlling Field
Paper1			

1.1 paper2 pick list values creation

- >go to home in setup page
- >on quick finder type pick list value set and select
- >click on new option
- >enter the label and name as paper2
- >enter the values on value set
- >click on next>click on next >click on save

Global Value Set Detail

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	MATHEMATICS	MATHEMATICS	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:15 AM
Edit Del Deactivate	ENGLISH	ENGLISH	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:15 AM
Edit Del Deactivate	STATISTICS	STATISTICS	<input type="checkbox"/>	Assigned dynamically	GURAJADA VENKATA SAI KUMAR 1/5/2022, 9:15 AM

Fields Where Used

Field Label	Object	Data Type	Controlling Field
Paper2	Subjects	Picklist	

The image contains two screenshots of the Salesforce Lightning interface, specifically for the 'Subjects' object.

Screenshot 1: New Subject Creation

This screenshot shows the 'New Subjects' form. The 'Subjects Name' field is populated with 'APEX'. The 'Owner' is listed as 'GURAJADA VENKATA SAI KUMAR'. Under 'Subject ID', there are two entries: 'Paper1' with value 'APEX' and 'Paper2' with value 'MATHEMATICS'. The 'Students' field contains a user record 'g.sai'. At the bottom, there are 'Cancel', 'Save & New', and 'Save' buttons.

Screenshot 2: Subject Listing

This screenshot shows a list of subjects. There are two items: 'JAVA' and 'APEX'. The list includes a header 'Subjects Name' and a search bar at the top right. The interface includes standard Salesforce navigation and toolbars.

2.create field dependencies

2.1 field dependencies between college name and email

>click on object manager

>click on college object and click on fields and dependencies

>click on new

>enter the controlling field as college name

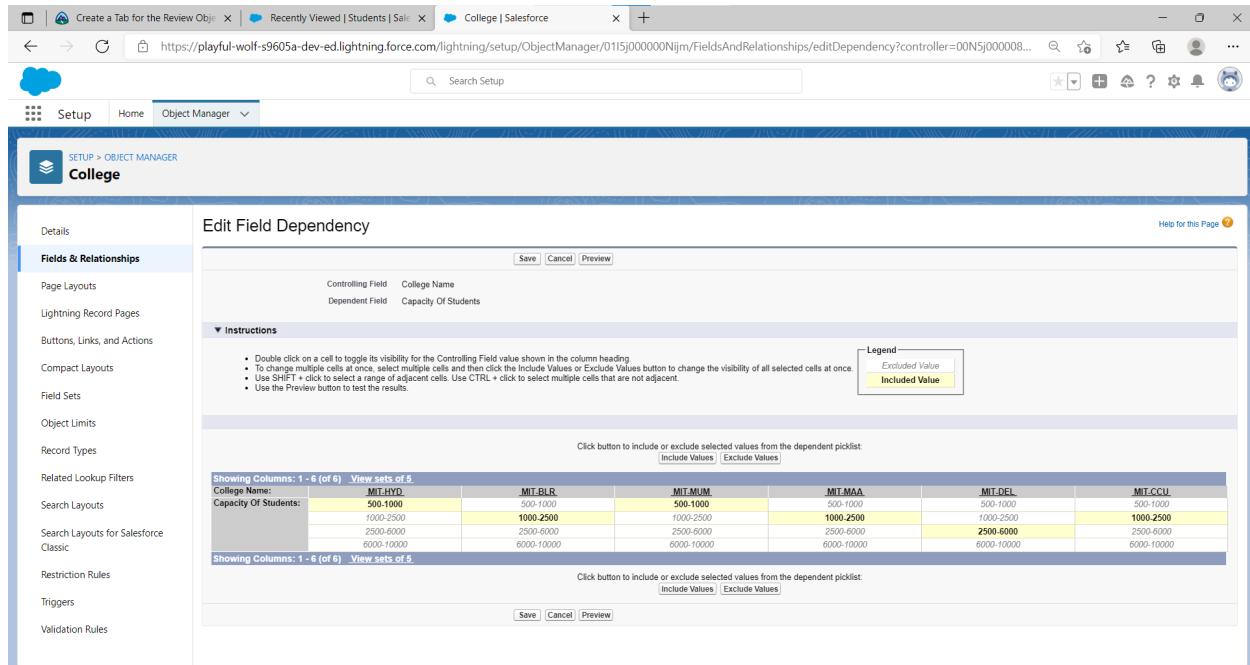
>enter the dependent field as email
 >now it display a table by select required field options between two field by
 >using include we can include it
 >click on save

The screenshot shows the 'Edit Field Dependency' page in the Salesforce Setup. The 'Dependent Field' is set to 'Email'. The page displays a grid of email addresses from various colleges. A legend indicates that yellow cells represent 'Included Value' and white cells represent 'Excluded Value'. The 'Dependent Field' column is set to 'Email'.

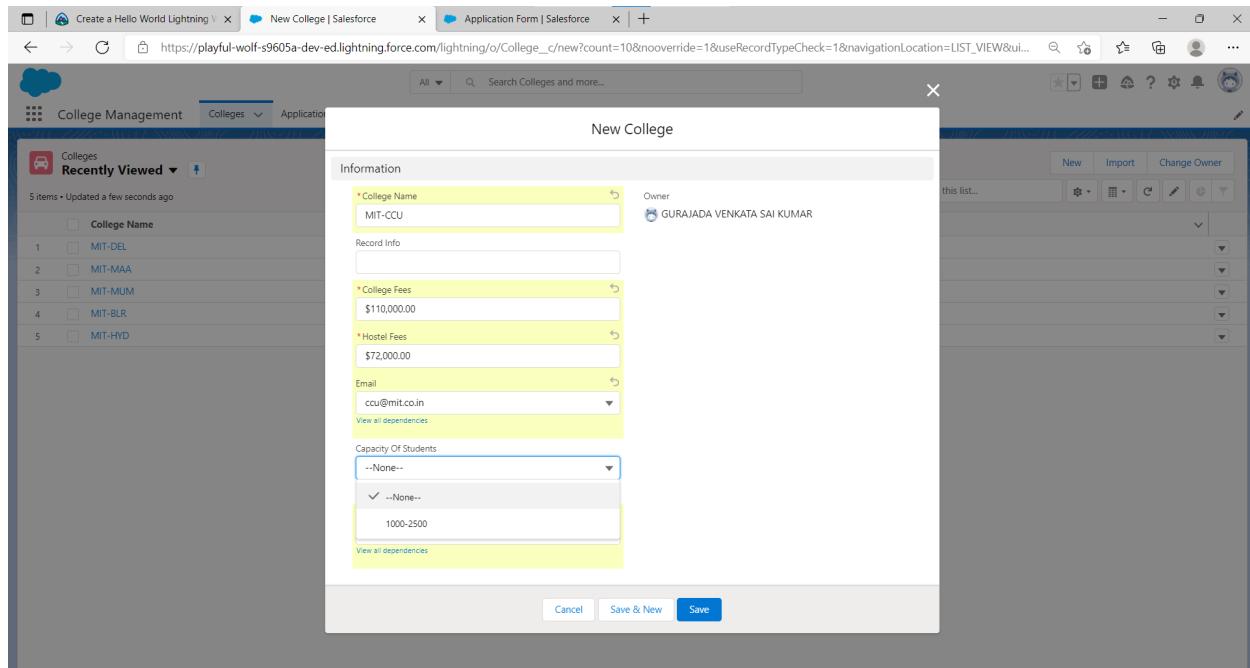
The screenshot shows the 'New College' form in the Salesforce application. The 'College Name' field is populated with 'MIT-DEL'. Other fields like 'Record Info', 'Email', and 'View all dependencies' also show 'MIT-DEL'.

2.2 field dependencies between college name and capacity of students
 >click on object manager

>click on college object and click on fields and dependencies
 >click on new
 >enter the controlling field as college name
 >enter the dependent field as capacity of student
 >now it display a table by select required field options between two field by
 >using include we can include it
 >click on save



The screenshot shows the 'Edit Field Dependency' page for the 'College' object. The 'Fields & Relationships' tab is selected. The 'Controlling Field' is set to 'College Name' and the 'Dependent Field' is set to 'Capacity Of Students'. The 'Instructions' section provides guidance on using the grid to toggle field visibility. The main area contains a grid where rows represent 'College Name' and columns represent 'Capacity Of Students'. Yellow cells indicate 'Included Value', while white cells indicate 'Excluded Value'. The grid shows various capacity ranges for different colleges like MIT-HYD, MIT-BLR, MIT-MUM, MIT-MAA, MIT-DEL, and MIT-CCU.



The screenshot shows the 'New College' form. The 'Information' section includes fields for 'College Name' (set to 'MIT-CCU'), 'Record Info', 'College Fees' (\$110,000.00), 'Hostel Fees' (\$72,000.00), 'Email' (ccu@mit.co.in), and 'Capacity Of Students' (dropdown menu showing options like -None-, ..None.., and 1000-2500). The 'Capacity Of Students' dropdown has the '1000-2500' option selected. The form also includes 'Save' and 'Save & New' buttons at the bottom.

3.create validation rule for name (text filed) and college name both as to be same in college object

- >click on object manager
- >click on college object and click on validation rules
- >click on new
- >enter name as college_name_same
- >enter formula as TEXT (College_Name__c) <> Name
- >click on save

The screenshot shows the Salesforce Setup Object Manager interface for the 'College' object. The 'Validation Rules' tab is selected. A new validation rule is being created with the following details:

- Rule Name:** college_name_same
- Active:** checked
- Description:** (empty)
- Error Condition Formula:** TEXT(College_Name__c) <> Name
- Functions:** A dropdown menu is open, showing options like ABS, ADDMONTHS, AND, BEGINS, BLANKVALUE, and BR.
- Error Message:** (empty)

The left sidebar lists various configuration tabs for the College object, including 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, and Triggers. The Validation Rules tab is currently active.

The screenshot shows a Salesforce Lightning Application Form. The form contains the following fields:

- *Application Form Name: g.Sai
- Application Form ID: vzm
- *College: MIT-HYD
- *date of Birth: 4/4/2001
- *Email: saikumarsk2001@gmail.com
- *Guardian Name: anantha rao
- Locking For Hostel: checked
- Ready To Join: checked
- *Student Name: g.sai
- *Phone: 1234567890

At the bottom of the form are three buttons: Cancel, Save & New, and Save.

Day-4:

Topic: Adding Business Logic To Application

Milestones/Activities: 1. Process Automation

1. Create an automation process such that when the "ready to join" field is checked on the application form object we need to create the student record automatically with the information specified in the application form record

2. Flow builder

create a student record using flow builder

Description:

1. Process Automation

> Go to Setup

> select "Process Builder" from quick find. Create a Process Builder on the "Application Form" object with a condition as "When a record change". And select "When a record is created or edited".

> In the diamond shape box(called nodes), select the criteria which trigger the Process builder to fire. In our example, it is "When Ready to Join field is checked."

> Once the node is setup, click on the adjacent box called "Immediate action". And select

create a record on the student object.

The screenshot shows the Salesforce Process Builder interface with two main panels.

Left Panel: Process Flow

- The process starts with an "Application Form" step.
- An "IF" decision step checks the condition `ready_to_join_fe = TRUE`.
- If TRUE, it performs an "IMMEDIATE ACTION" to create a "student_record".
- If FALSE, it proceeds to another "IF" decision step.
- The second "IF" decision step checks the condition `+ Add Criteria` (which is always TRUE).
- If TRUE, it performs an "IMMEDIATE ACTION" to create a "student_record".
- If FALSE, the process ends with a "STOP" node.

Right Panel: Action Group Configuration

Define Criteria for this Action Group

- Criteria Name:** ready_to_join_field
- Criteria for Executing Actions:**
 - Conditions are met (radio button selected)
 - Formula evaluates to true
 - No criteria—just execute the actions!
- Set Conditions:**

Field *	Operator *	Type *	Value *
[Application_Form__c]	Equals	Boolean	True
- Conditions:**
 - All of the conditions are met (AND)
 - Any of the conditions are met (OR)
 - Customize the logic
- Advanced:** (button)

Create a Record

Action Name: student_record

Record Type: Students

Set Field Values

Field *	Type *	Value *
Address	Field Reference	[Application_Form__c].Add... Q
Date Of Birth	Field Reference	[Application_Form__c].dat... Q
Guardian Name	Field Reference	[Application_Form__c].Gua... Q
Students Name	Field Reference	[Application_Form__c].Stu... Q
Application Form	Field Reference	[Application_Form__c].Id Q
Phone	Field Reference	[Application_Form__c].Pho... Q

Save Cancel

Screenshot 1: A screenshot of the Salesforce Lightning Experience showing a new application form record. The form fields are highlighted in yellow. The fields include:

- *Application Form Name: g.kumar
- Address: sklm
- *College: MIT-DEL
- *date of Birth: 5/4/2001
- *Email: kumar234@gmail.com
- *Guardian Name: g.rao
- Looking For Hostel:
- Ready To Join:
- *Student Name: g.kumar
- *Phone: 9554523651

The "Save" button is visible at the bottom right.

Screenshot 2: A screenshot of the Salesforce Lightning Experience showing a new application form record. The form fields are highlighted in yellow. The fields include:

- *Application Form Name: g.kumar
- Address: sklm
- *College: MIT-DEL
- *date of Birth: 5/4/2001
- *Email: kumar234@gmail.com
- *Guardian Name: g.rao
- Looking For Hostel:
- Ready To Join:
- *Student Name: g.kumar
- *Phone: 9554523651

The "Save" button is visible at the bottom right.

The screenshot shows a Salesforce Lightning Experience interface. At the top, there are three tabs: 'Create a Hello World Lightning' (active), 'g.kumar | Salesforce', and 'Process Builder | Salesforce'. The main content area is titled 'College Management' with a sub-section 'Students'. A search bar at the top right says 'Search Students and more...'. Below the search bar, there are navigation links: 'All', 'Students', and 'Subjects'. The main view displays a student record for 'g.kumar'. The record details include:

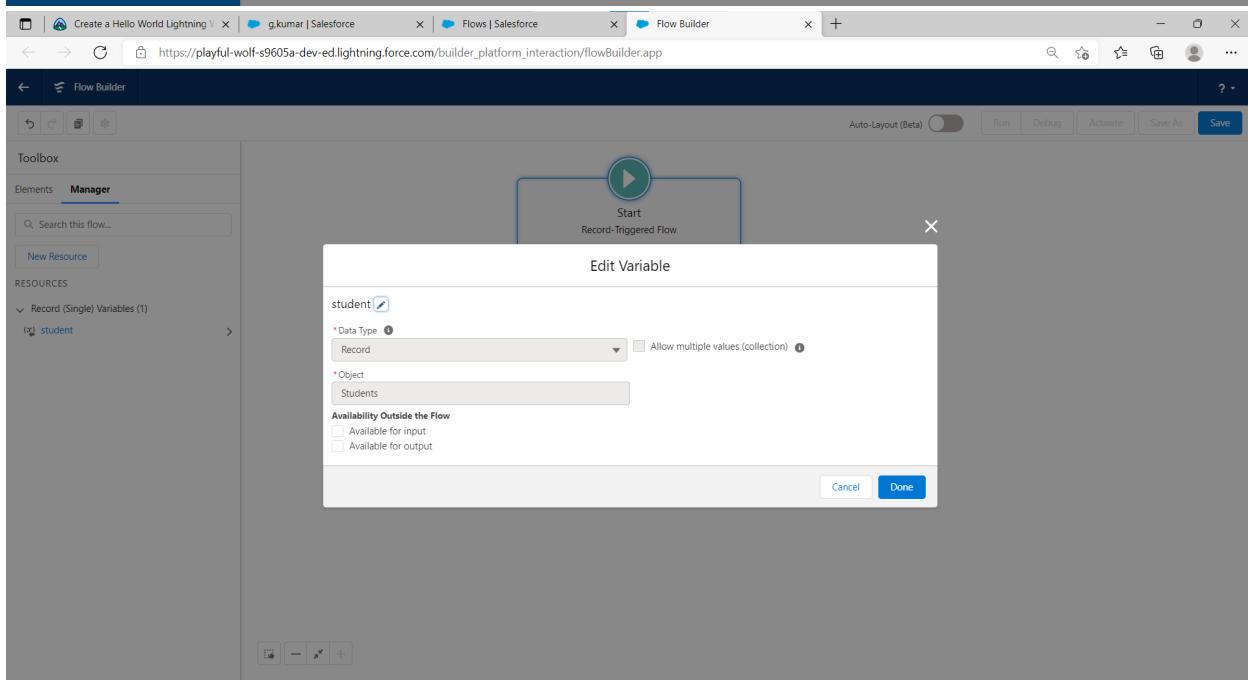
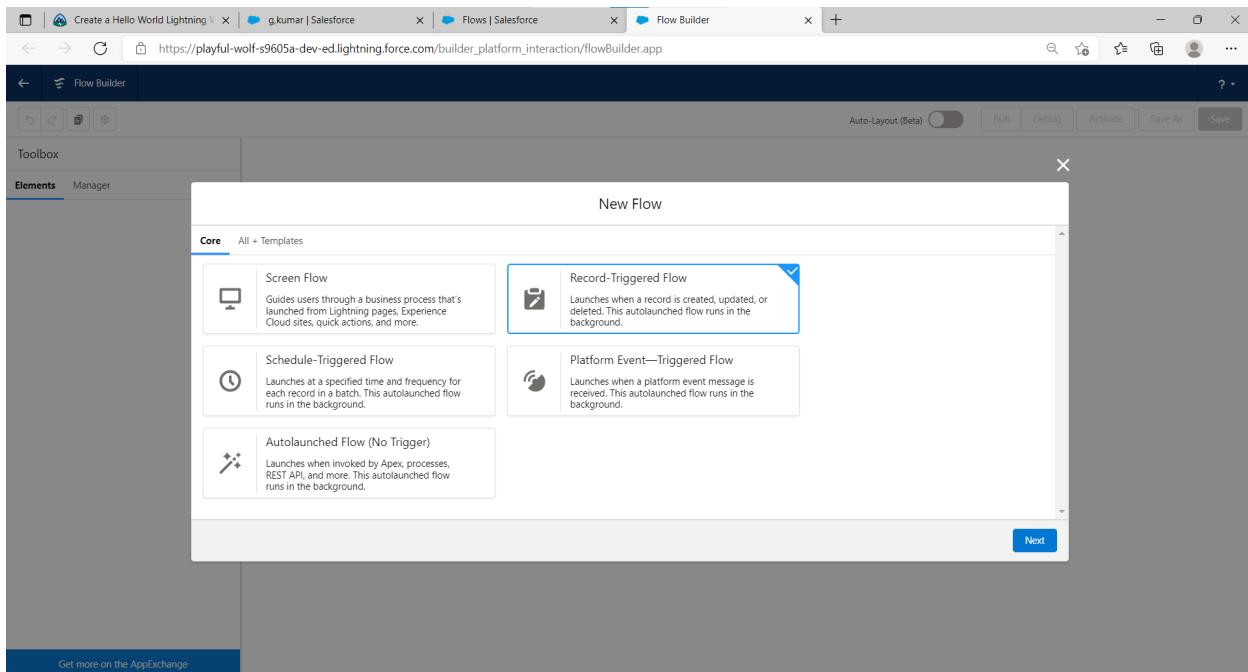
- Student Name:** g.kumar
- Address:** sklm
- Application Form:** g.kumar
- College Name:** MIT-DEL
- Date Of Birth:** 5/4/2001
- Guardian Name:** g.rao
- Phone:** (955) 452-3651
- Created By:** GURAJADA VENKATA SAI KUMAR, 1/8/2022, 6:23 AM
- Last Modified By:** GURAJADA VENKATA SAI KUMAR, 1/8/2022, 6:23 AM

2. Flow Builder

- >First deactivate the process builder which we created earlier.
- >Now search for flows and select new flow ->record triggered flow
- > For object select application form, in configure trigger select when the record is updated for entry criteria select as ready to join equals to true.
- >Now create a variable named student in the resource section.
- > Now add the assignment as follows.

The screenshot shows the Salesforce Setup interface with the 'Flows' section selected. On the left, there is a sidebar with various setup categories like Apps, Workflow Services, Process Automation, and Workflow Actions. The main content area is titled 'Flows' and shows a table of 'Flow Definitions'.

Flow Label	Process Type	Active	Template	Package State	Package Name	Last Modified By	Last Modified
Cancel Item Flow	Screen Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Create a Case	Screen Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Create Order Summary Flow	Autolaunched Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Create Process Exception Flow	Autolaunched Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Reset Password	Screen Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Return Item Flow	Screen Flow	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			
Verify Identity	Screen Flow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Managed-Installed			



New Assignment

Set Variable Values

Each variable is modified by the operator and value combination.

Variable	Operator	Value
A _a student > Address	Equals	A _a \$Record > Address
A _a student > Application Form	Equals	A _a \$Record > Record ID
A _a student > College Name	Equals	A _a \$Record > College
A _a student > Date Of Birth	Equals	A _a \$Record > date of Birth
A _a student > Guardian Name	Equals	A _a \$Record > Guardian Name
A _a student > Phone	Equals	A _a \$Record > Phone
A _a student > Student Name	Equals	A _a student > Student Name

Add Assignment

Done

New Create Records

Create Salesforce records using values from the flow.

*Label: **create_student_record** *API Name: **create_student_record**

Description:

How Many Records to Create

- One
- Multiple

How to Set the Record Fields

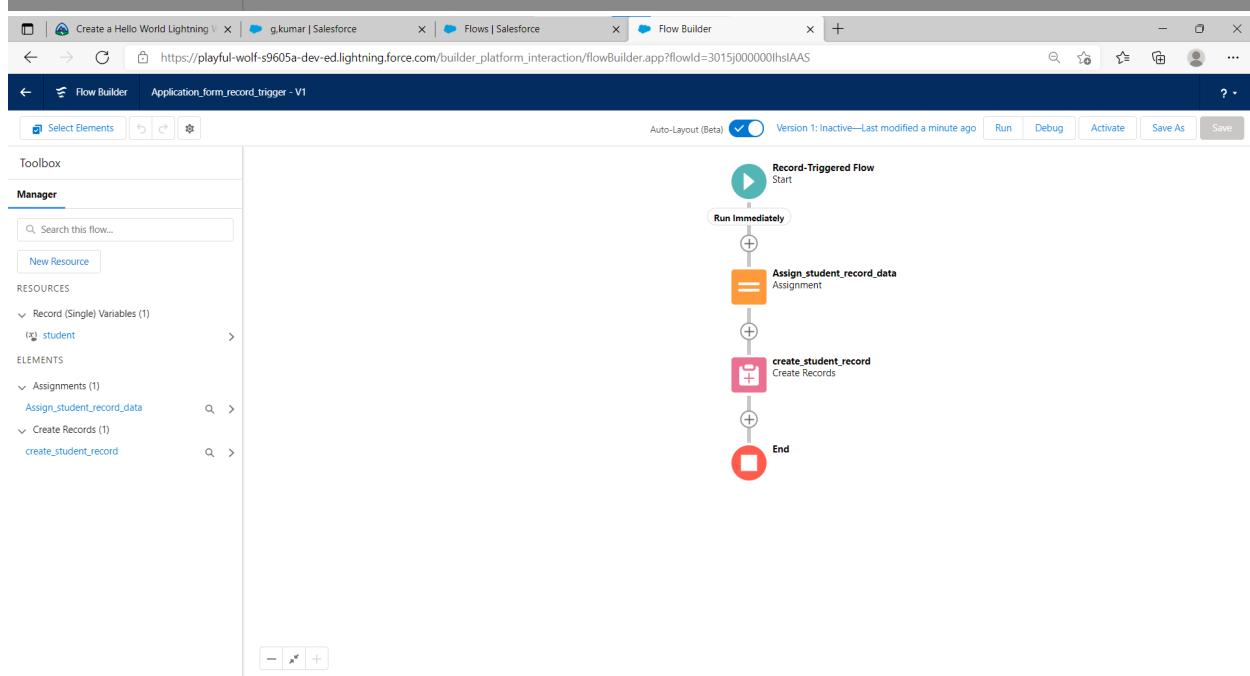
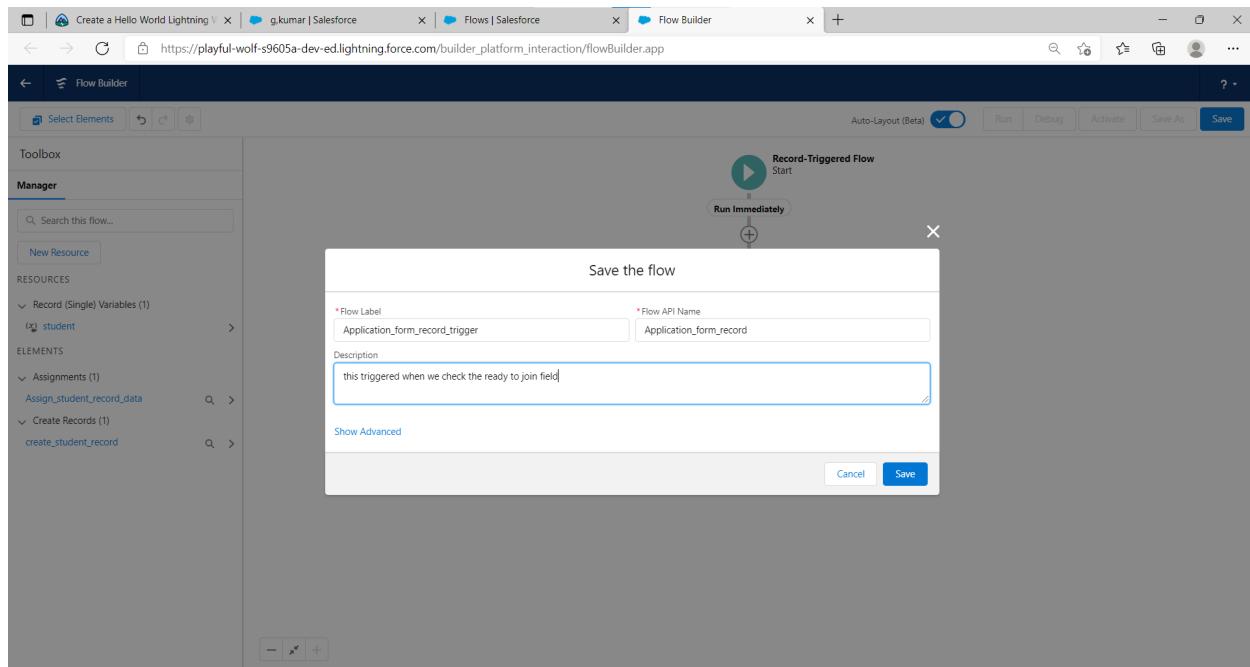
- Use all values from a record
- Use separate resources, and literal values

Create a Record from These Values

*Record: **(x) student**

Make sure that ID is blank. After the flow creates the records, ID is set to match the record that was created.

Done



Screenshot of the Salesforce Lightning Experience showing a new application form being created.

The application form fields filled are:

- *Application Form Name: g.karthik
- Address: hyd
- *College: MIT-CCU
- *date of Birth: 5/4/2001
- *Email: karthik725@gmail.com
- *Guardian Name: g.rao
- Locking For Hostel: checked
- Ready To Join: checked
- *Student Name: g.karthik
- *Phone: 8954645454

Buttons at the bottom: Cancel, Save & New, Save

Screenshot of the Salesforce Lightning Experience showing the details of the newly created student record.

The student details are:

- Students Name: g.karthik
- Owner: GURAJADA VENKATA SAI KUMAR
- Address: hyd
- Application Form: g.karthik
- College Name: MIT-CCU
- Date Of Birth: 5/4/2001
- Guardian Name: g.rao
- Phone: (865) 485-6565
- Created By: GURAJADA VENKATA SAI KUMAR, 1/8/2022, 7:25 AM
- Last Modified By: GURAJADA VENKATA SAI KUMAR, 1/8/2022, 7:25 AM

Buttons at the top right: New Contact, Edit, New Opportunity

The image contains two screenshots of the Salesforce Lightning interface, both titled "College Management".

Screenshot 1: Application Forms

This screenshot shows a list of application forms. The columns are: Application Form Name, Application Form ID, College, Address, and Phone. The data is as follows:

	Application Form Name	Application Form ID	College	Address	Phone
1	g.karthik	F-000007	MIT-CCU	hyd	(865) 485-6565
2	g.kumar	F-000003	MIT-DEL	sklm	(955) 452-3651
3	g.sai	F-000001	MIT-HYD	vzm	1234567890
4	g.venkata	F-000002	MIT-MUM	vizag	(956) 432-8542

Screenshot 2: Students

This screenshot shows a list of students. The columns are: Students Name, College Name, Address, Phone, and Application Form. The data is as follows:

	Students Name	College Name	Address	Phone	Application Form
1	g.kumar	MIT-DEL	sklm	(955) 452-3651	g.kumar
2	g.karthik	MIT-CCU	hyd	(865) 485-6565	g.karthik
3	g.sai	MIT-HYD	vzm	1234567890	g.sai
4	g.venkata	MIT-MUM	vizag	(954) 862-1564	g.venkata

Day-5:

Topic: Batch Apex

Milestones/Activities:

- From the developer console create a new apex class to retrieve number of application forms

count and send a mail of count of application forms

Description:

we create an apex class from developer console and write code and save it

```
public class ApplicationBatchTest implements Database.Batchable<sobject>, Database.stateful {
    public Integer totalForms = 0;
    public Integer totalConvertedForms = 0;
    public Database.QueryLocator start(Database.BatchableContext bc){
        String applicationQuery = 'select id, name, Ready_To_Join__c from Application_Form__c';
        return Database.getQueryLocator(applicationQuery);
    }
    public void execute(Database.BatchableContext bc, List<Application_Form__c> formList){
        // process the data
        for(Application_Form__c af : formList){
            totalForms++;
            if(af.Ready_To_Join__c){
                totalConvertedForms++;
            }
        }
    }
    public void finish(Database.BatchableContext bc){
        Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
        mail.setSubject(' Application form and student record data as of today ');
        mail.setPlainTextBody(' Total no of application form records are : '+totalForms+ ' out of which no of students as per today : '+totalConvertedForms );
        String[] emailAddress = new String[]{saikumarsk2001@gmail.com};
        mail.setToAddresses(emailAddress);
        Messaging.sendEmail(new Messaging.SingleEmailMessage[]{ mail });
    }
}
```

Logs, Tests, and Problems

then open a anonymous block execute test on that code

The screenshot shows the Developer Console with the following details:

- Execution Log:** Displays the execution of the ApplicationBatchTest class. It shows various system events like USER_INFO, EXECUTION_STARTED, and HEAP_ALLOCATE, along with specific Apex code being executed:

```
1 ApplicationBatchTest abt = new ApplicationBatchTest();
2
3     Database.executeBatch(abt, 400); // 200 to 2000
```
- Logs Tab:** Shows a table of logs with columns: User, Application, Operation, Time, Status, Read, and Size. The logs are as follows:

User	Application	Operation	Time	Status	Read	Size
GURAJADA VENKATA SAI KUMAR	Unknown	Batch Apex	1/18/2022, 9:32:35 PM	Success	Unread	3.85 KB
GURAJADA VENKATA SAI KUMAR	Unknown	Batch Apex	1/18/2022, 9:32:35 PM	Success	Unread	4.92 KB
GURAJADA VENKATA SAI KUMAR	Unknown	SerialBatchApexRangeChunkHandler	1/18/2022, 9:32:35 PM	Success	Unread	4.98 KB
GURAJADA VENKATA SAI KUMAR	Unknown	SerialBatchApexRangeChunkHandler	1/18/2022, 9:32:35 PM	Success	Unread	2.43 KB
GURAJADA VENKATA SAI KUMAR	Unknown	/services/data/v53.0/tooling/executeA...	1/18/2022, 9:32:34 PM	Success	Unread	3.4 KB
GURAJADA VENKATA SAI KUMAR	Unknown	Batch Apex	1/18/2022, 9:26:50 PM	sObject type 'ApplicationForm__c' is n...	Unread	4.61 KB

Developer Console - Google Chrome
 mvgr41-dev-ed.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >

ApplicationBatchTest.apxc [] applicationschedule.apxc [] Log executeAnonymous @1/18/2022, 9:26:49 PM [] Application_Form_c@9:32 PM [] Log executeAnonymous @1/18/2022, 9:32:35 PM []

select id, name, Ready_To_Join__c from Application_Form__c

Query Results - Total Rows: 4

Id	Name	Ready_To_Join__c
a015g00000b1p8gAAC	g.karthik	true
a015g00000b1p8RAAS	g.sal	true
a015g00000b1p8AAC	g.kumar	true
a015g00000b1p8WAAS	g.venkata	true

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 History

select id, name, Ready_To_Join__c from Application_Form__c

Any query errors will appear here...

History

Executed

select id, name, Ready_To_Join__c from Application_Form__c

Execute Use Tooling API

Developer Console - Google Chrome
 mvgr41-dev-ed.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >

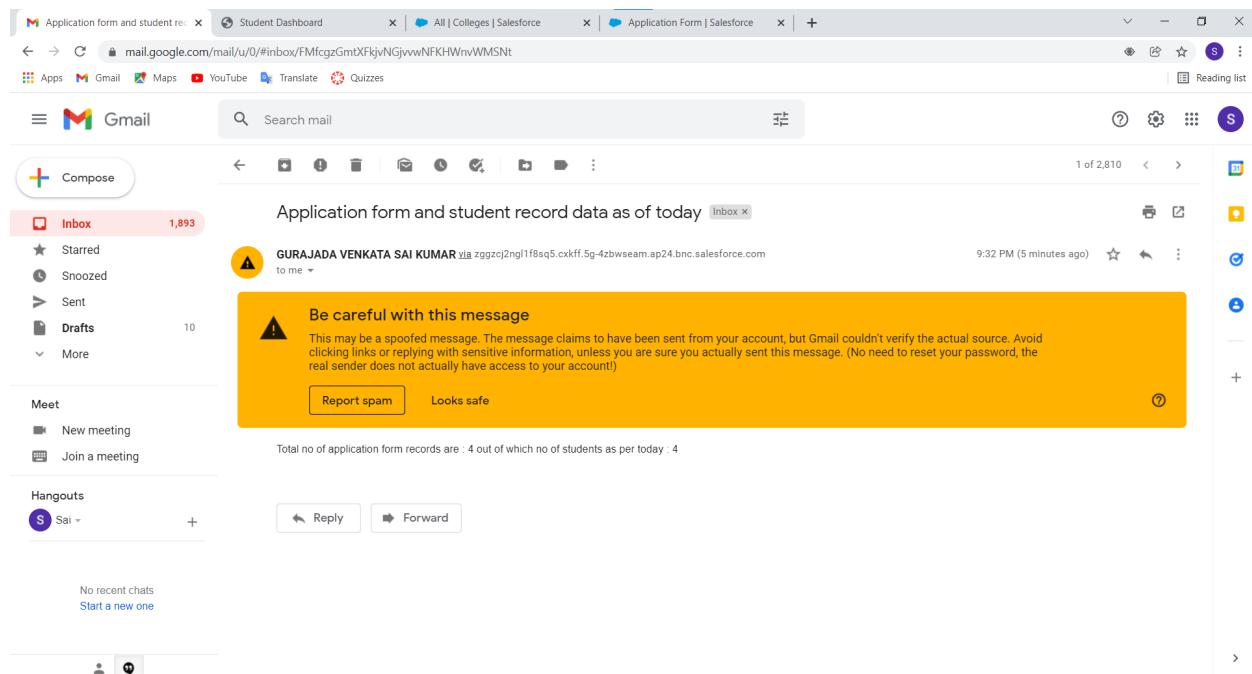
ApplicationBatchTest.apxc [] applicationschedule.apxc [] Log executeAnonymous @1/18/2022, 9:26:49 PM [] Application_Form_c@9:32 PM [] Log executeAnonymous @1/18/2022, 9:32:35 PM []

Execution Log

Timestamp	Event	Details
21:32:35:000	USER_INFO	[EXTERNAL]0055g000008nj5E saikumarsk2001@gmail.com (GMT-08:00) Pacific Standard Time (America/Los_Angeles) GMT-08:00
21:32:35:000	EXECUTION_ST...	
21:32:35:000	CODE_UNIT_ST...	[EXTERNAL]01p5g00000Xysz ApplicationBatchTest
21:32:35:002	HEAP_ALLOCATE	[79]Bytes:3
21:32:35:002	HEAP_ALLOCATE	[84]Bytes:152
21:32:35:002	HEAP_ALLOCATE	[399]Bytes:408
21:32:35:002	HEAP_ALLOCATE	[412]Bytes:40
21:32:35:002	HEAP_ALLOCATE	[520]Bytes:48
21:32:35:002	HEAP_ALLOCATE	[139]Bytes:6
21:32:35:002	HEAP_ALLOCATE	[EXTERNAL]Bytes:28
21:32:35:019	HEAP_ALLOCATE	[EXTERNAL]Bytes:11
21:32:35:019	SYSTEM_METH...	[15][BatchableContextImpl.BatchableContextImpl()]
21:32:35:019	STATEMENT_EX...	[15]
21:32:35:019	SYSTEM_METH...	[15][BatchableContextImpl]
21:32:35:019	HEAP_ALLOCATE	[EXTERNAL]Bytes:12
21:32:35:019	HEAP_ALLOCATE	[EXTERNAL]Bytes:4
21:32:35:019	VARIABLE_SO...	[31][this Database.BatchableContextImpl true false]
21:32:35:019	VARIABLE_ASIL...	[31][this]0x68f02283
21:32:35:019	VARIABLE_SO...	[31] jobId Id false false
21:32:35:019	VARIABLE_ASIL...	[31] jobId 70 5g000001oKQHRAA4"
21:32:35:019	VARIABLE_SO...	[31] childJobId Id false false
21:32:35:019	VARIABLE_ASIL...	[31] childJobId null
21:32:35:022	HEAP_ALLOCATE	[EXTERNAL]Bytes:75
21:32:35:022	HEAP_ALLOCATE	[EXTERNAL]Bytes:6
21:32:35:022	METHOD_ENTRY	[1]01p5g00000Xysz ApplicationBatchTest.ApplicationBatchTest()
21:32:35:022	STATEMENT_EX...	[1]
21:32:35:022	STATEMENT_EX...	[1]
21:32:35:022	METHOD_EXIT	[1]ApplicationBatchTest
21:32:35:022	HEAP_ALLOCATE	[EXTERNAL]Bytes:12

This Frame Executable Debug Only Filter Click here to filter the log

Logs, Tests, and Problems



Day-6:

Topic: Scheduling Apex

Milestones/Activities:

1. From the developer console create a new scheduler apex class and schedule a apex job

Description:

we create an apex class from developer console and write code and save it

Developer Console - Google Chrome
 mvgr41-dev-ed.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < ▾ >

ApplicationBatchTest.apxc [] applicationschedule.apxc [] Log executeAnonymous @1/18/2022, 9:26:49 PM [] Application Form c@9:32 PM [] Log executeAnonymous @1/18/2022, 9:32:35 PM []

Code Coverage: None ▾ API Version: 53 ▾ Go To

```

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 }
16
17

```

Logs, Tests, and Problems

and open home and at quick finder type apex class click on new schedule apex

Inbox (1,892) - saikumarsk2001@ | Student Dashboard | Apex Classes | Salesforce

Search Setup

Apex Classes

Apex Classes

Apex Code is an object oriented programming language that allows developers to develop on-demand business applications on the Lightning Platform.

Percent of Apex Used: 0.92%
 You are currently using 1,472 characters of Apex Code (excluding comments and @isTest annotated classes) in your organization, out of an allowed limit of 6,000,000 characters. Note that the amount in use includes both Apex Classes and Triggers defined in your organization.

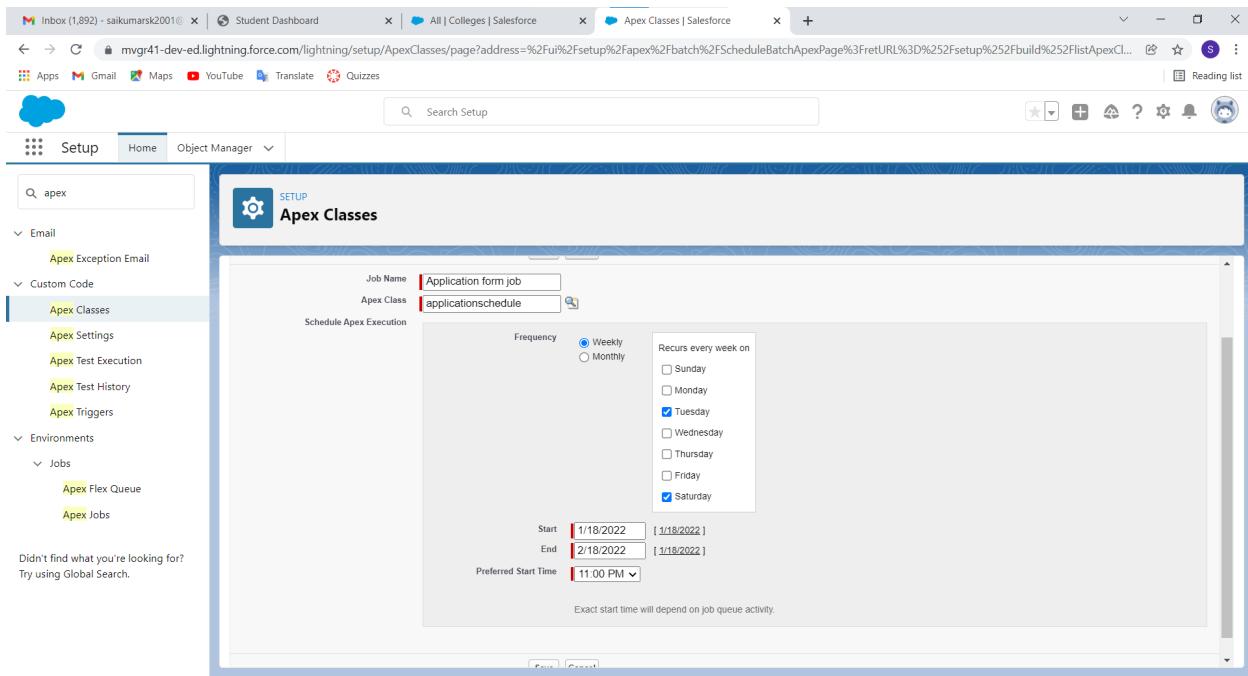
Estimate your organization's code coverage [\[i\]](#)
[Compile all classes](#) [\[i\]](#)

View: All [Create New View](#)

Action	Name +	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By	Has Trace Flags
Edit Del Security	ApplicationBatchTest		53.0	Active	1,254	GURAJADA VENKATA SAI KUMAR	1/18/2022, 8:02 AM
Edit Del Security	applicationschedule		53.0	Active	218	GURAJADA VENKATA SAI KUMAR	1/18/2022, 8:02 AM

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 [\[All\]](#)

Didn't find what you're looking for?
 Try using Global Search.



and click on save

Day-7:

Title : LWC COMPONENTS

Activities/Milestones:

add LWC chrome extension

Description:

open chrome browser and type salesforce LWC chrome extension in search bar

The image shows two screenshots of the Salesforce LWC Editor extension. The top screenshot is from the Chrome Web Store, displaying the extension's page with a 5-star rating of 28, developer tools, and over 8,000 users. It also shows related extensions like Salesforce Lightning Web Components, Salesforce Inspector, and Salesforce DevTools. The bottom screenshot is the extension's landing page, featuring a large title 'Salesforce LWC Editor' and the tagline 'Build Salesforce LWC development easy and fastest.' A 'How To Use' section provides instructions for installation, and a 'Features' section lists the ability to create new components.

Salesforce LWC Editor - Chrome

chrome.google.com/webstore/detail/salesforce-lwc-editor/ehkpneicmpbdejpoancidgkejlkahjgo/related

chrome web store

Home > Extensions > Salesforce LWC Editor

Salesforce LWC Editor

Offered by: nisarahmad.ajmer@gmail.com

★★★★★ 28 | Developer Tools | 8,000+ users

Remove from Chrome

Overview Privacy practices Reviews Support Related

Salesforce Lightning Web Components 291 users

Salesforce Inspector 224

Salesforce DevTools 36

Salesforce API Fieldname... 125

Salesforce Tool Suite (D...) 28

Salesforce Assistant 19

Apex Debugger 29

Code Coverage Calculat... 14

Salesforce LWC Editor - Chrome

sites.google.com/view/salesforce-lwc-editor

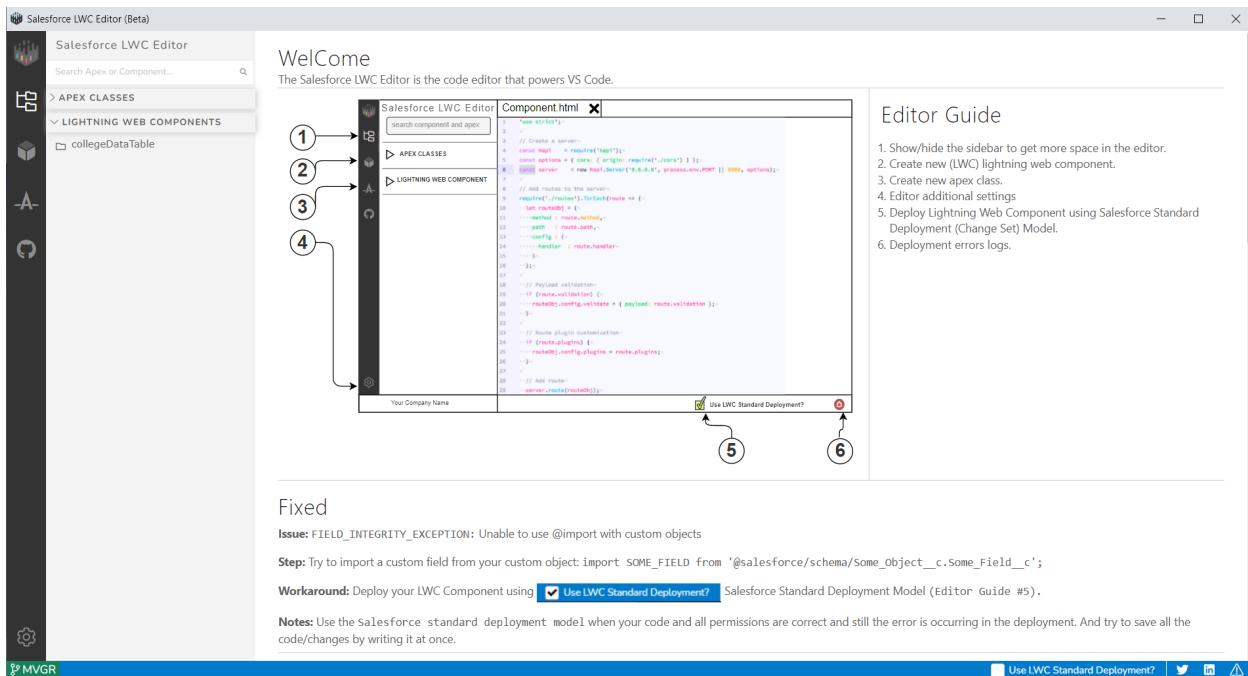
How To Use

- Just log in to salesforce and click the "Salesforce LWC Editor" icon (browser top-right corner [chrome extension standard icon-bar]) when the icon changes to blue color.
- If installed for the first time, refresh the page before using it.

Features

- Quick 'Create New Components' with additional configurations.

and click on extension button on sales forceloggedin page
it navigate to LWC editor



Day-8

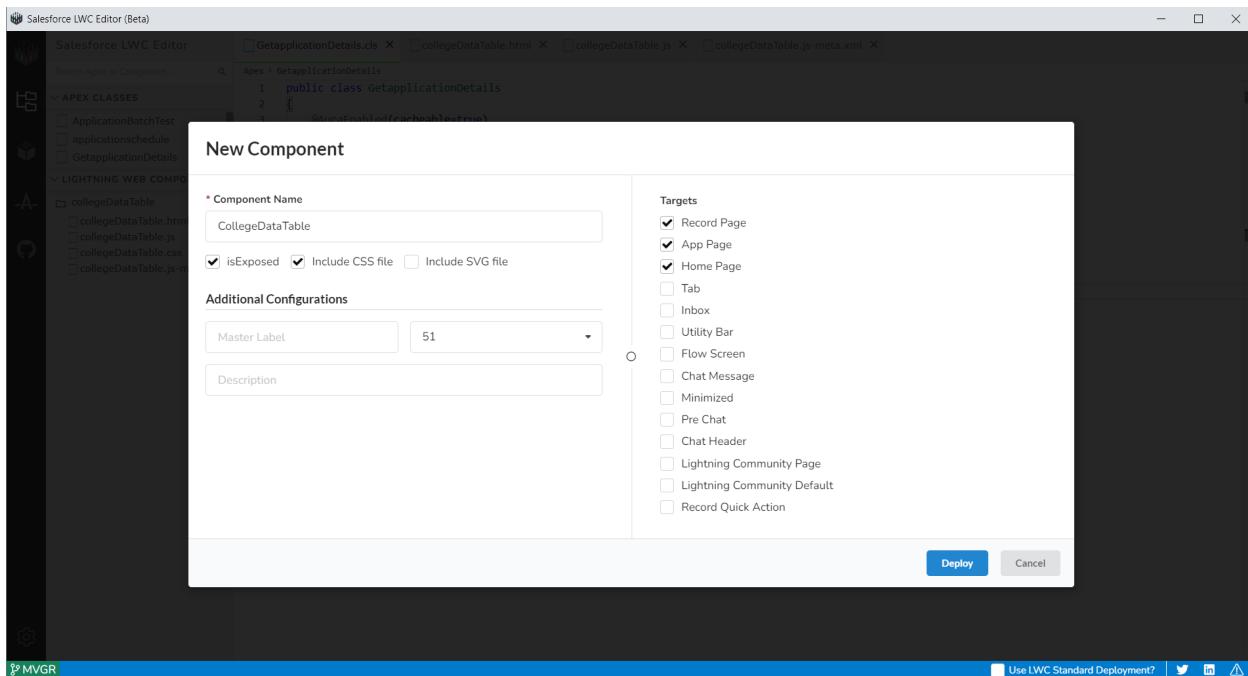
Title: LW Components

Milestones/Activities:

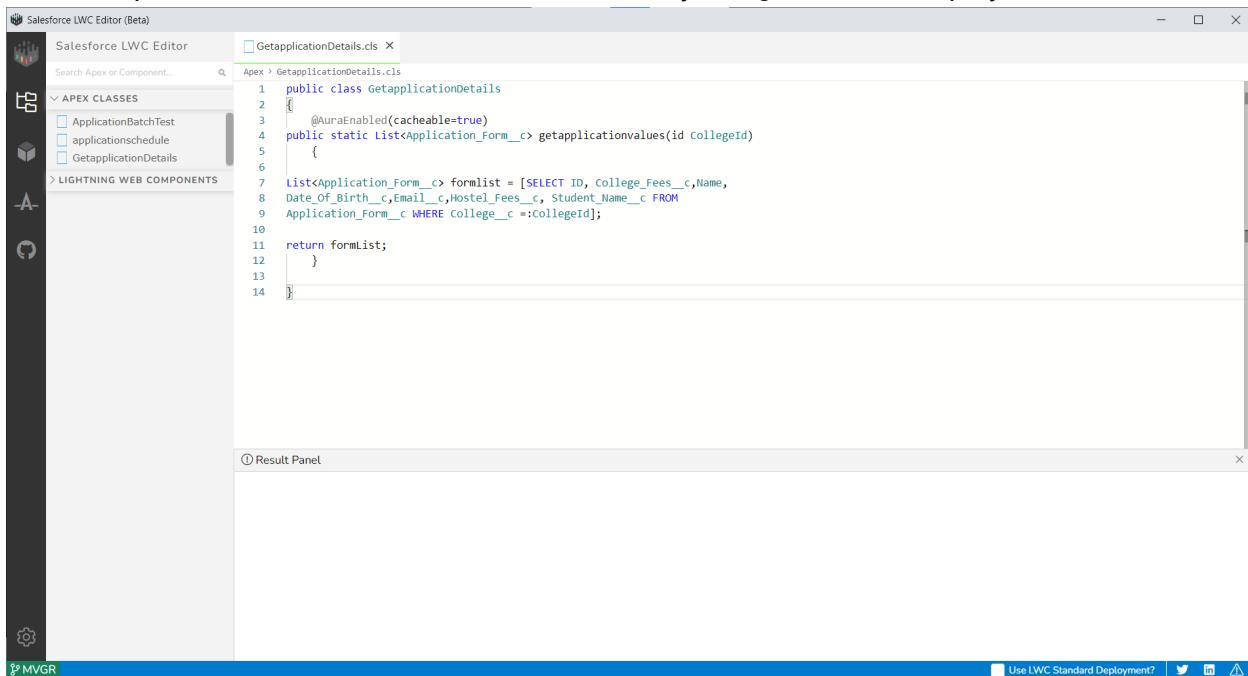
- 1.Create College DataTable Component(APEX CLASS)
- 2.Create College DataTable Component (HTML FILE)
- 3.Create College DataTable Component(JAVA SCRIPT FILE)
- 4.Create College DataTable Component(META FILE)

Description:

go LWC editor and create a component



create a apex class for retrieve the data from form by using SOQL and deploy



write code on html file for college data table component and deploy it

```

<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>

```

write code on js file for college data table component and deploy that code

```

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'})
    wiredCollegeData({data, error}){
        if(data){
            this.recordList = data;
        }
        else if(error){
            this.error = error;
            this.recordList = undefined;
        }
    }
}

```

write code on xml file for college data table component and deploy it

```

<?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>

```

and open college tab and open one college data at top right corner at setup symbol click on it and click on edit page option it will take to lightning builder for college record page

The screenshot shows the Lightning App Builder interface. On the left, the 'Components' sidebar lists various components like 'Related List Quick Links', 'Report Chart', etc. In the center, the 'College Record Page' component is displayed. A modal window is open for creating a new tab. The 'Tab Label' field contains 'Data Table'. A red box highlights this field with the error message 'This field is required.' At the bottom right of the modal, there is a 'Done' button.

create a custom tab as above name in picture
add component to that tab and click on save option

The screenshot shows the Lightning App Builder interface. On the left, there's a sidebar with 'Components' and 'Fields' tabs, and a search bar. Below it is a list of standard components. In the center, a 'Data Table' component is being edited. The table has columns: Application Form, College Fees, Date Of Birth, Email, Hostel Fees, and StudentName. Two rows of data are visible: one for 'g.radhika' and another for 'g.saai'. On the right, there are tabs for 'Related', 'Details', and 'Data Table'. A 'Page > Tabs' section shows three tabs: 'Related', 'Details', and 'Data Table'. Below that is a 'Set Component Visibility' section with a 'Filters' button. At the bottom right, there are 'Save' and 'Cancel' buttons.

at college data refresh it

The screenshot shows the Salesforce Lightning Experience. At the top, there's a navigation bar with links for 'College Management', 'Colleges', 'Application Forms', 'Students', and 'Subjects'. Below the navigation is a header with a cloud icon, the text 'College MIT-HYD', and buttons for 'New Contact', 'Edit', and 'New Opportunity'. The main content area displays a 'Data Table' component with the same structure as the one in the App Builder: Application Form, College Fees, Date Of Birth, Email, Hostel Fees, and StudentName. The two rows of data ('g.radhika' and 'g.saai') are also present here.