



**Smart
Internz**



SALESFORCE PROJECT READY
ENTRY LEVEL TRAINING AND PROFESSIONAL DEVELOPMENT
PROGRAM

PROJECT TITLE: COLLEGE APPLICATION MANAGEMENT



DAY 1:

Topic: Milestone / Activities: Intro to Salesforce, Cloud Computing, Trail head Carr er Fair

Detailed Description:

Ram-ya Mam started the program of Salesforce Project ready and introduce the program and tells what this program is and introduce **Ms. Renita Ferreira** and **Mr. Nitin Jain** to us. Over 230 students join the program.

A screenshot of a video conference interface. In the top right corner, there is a small video thumbnail of a woman with the name "Ramya G" below it. The main screen shows three circular profile pictures of participants: Ms. Bhanu Dasgupta (Salesforce), Ms. Renita Ferreira (Salesforce), and Mr. Nitin Jain (SmartBridge). The background is a solid blue color. The Smart Internz logo is visible in the top left corner, and the hashtag "#Salesfor" is in the top right corner.

Later Renita Mam tells us the opportunities which are usable for us and the benefits of completing this project



Salesforce Customer 360

Trailblazer Success & Community • World's #1 CRM • Fast Time to Value • Scalable & Flexible • Work From Anywhere

The diagram illustrates the Salesforce Customer 360 ecosystem. At the center is a woman with dark hair and glasses, smiling. Surrounding her are various icons representing different business functions: Sales (chart), Service (heart), Marketing (magnifying glass), Commerce (shopping cart), Analytics (calculator), Integration (lightning bolt), Platform (building), Learning (person with laptop), Industries (factory), Employees (people), Partners (building with people), Success (trophy), and Service (handshake). To the right of the diagram is a legend for 'Experience' (Apps, Web, Channels, Bots), 'Einstein' (Automation, Workflow, Personalization, Low Code), and 'Single Source of Truth' (Data, Streams, Segments, Privacy), all under the heading 'Hyperforce'.

Later after Mr. Vinay help us to login in Smartinternz website by given credentials and explain each and every element in the website

The screenshot shows the Smartinternz Student Dashboard. On the left is a sidebar with links: Dashboard, Externships, Guided Projects, Challenges, Internships, Projects, Certificates, and Transactions. The main area has a header 'SMARTINTERNZ' and a 'SETTINGS' button. It features four cards: 'INTERNSHIPS' (1), 'CAREER READINESS FACTOR (CRF)' (0), and 'SKILL INDEX' (0). Below this is a table titled 'INTERNSHIPS' with one row:

Sno.	Application ID	Internship Title	Start Date	Internship Status
1.	SPS_API_20220014968	Externship	12/23/2020	Approved



Smart Internz

The screenshot shows the Smart Internz Student Dashboard. On the left, a sidebar lists navigation options: Dashboard, Externships, Guided Projects (selected), Challenges, Internships, Projects, Certificates, and Transactions. The main area displays a 'Guided Project' for 'College Management Application'. It shows project details: Duration 30 Days, Team VP, Overall Project Progress 90%, and Assigned Tasks Progress 90%. Below this is a 'MENTOR INSTRUCTION' section with 'Git Repo' and 'Project Doc' buttons. A note says 'Note: Use password xrY5c1R to access project Doc'. At the bottom, tabs for 'PROJECT DETAILS' (College Management Application), 'TASK & PROGRESS' (INTERMEDIATE), and 'MENTOR REVIEW' (College Management Application) are visible. The task progress bar indicates 90% completion. The status bar at the bottom right shows the date 07-01-2022 and time 10:46 AM.

Later Mr. Tushar started the class by starting the introduction of Cloud Computing

The screenshot shows a Microsoft Word document titled 'Configuration.docx'. The document contains a numbered list of 16 concepts related to cloud computing:

1. Environment
2. Objects
3. Fields
4. Relationship
5. Users
6. Profiles
7. Permission Set
8. Tab
9. Page Layouts
10. Record Types
11. Security Model
12. Validation Rules
13. Workflow
14. Workflow actions
15. Process builders
16. Flow Builder

The background of the slide features the SMARTBRIDGE and Smart Internz logos. The status bar at the bottom shows the date 03-01-2022 and time 17:31.



What is Salesforce?

Salesforce is a CRM Platform

Why Salesforce?

We have so many CRM Platforms in the market but Salesforce is best of them because salesforce provides us some functionalities which helps us our work to be easy

1. Scalability
2. Customization
3. Security
4. With best Features for market and salespersons
5. Cheaper

A screenshot of a Microsoft PowerPoint slide titled "Cloud computing". The slide has a dark purple header bar with the title. Below the title is a bulleted list:

- ▶ What is Cloud Computing?
Cloud computing is all about using a remote server, to store, manage, and process data, instead of a local server/ personal computer.
- ▶ Benefits:
 - ▶ eliminate headaches that come with storing your own data.
 - ▶ Not to worry about hardware and software.
 - ▶ upgrades are automatic, and scaling up or down is easy.

The left side of the screen shows a storyboard with five slides numbered 1 to 5. Slide 1 is titled "Introduction To Salesforce". Slide 3 is highlighted with an orange border and titled "Cloud computing". The PowerPoint ribbon at the top shows the "DESIGN" tab is selected. The taskbar at the bottom shows various application icons and the date/time "03-01-2022 17:34".

Create Your Salesforce Developer Organization To Get Started

Creating a developer org in salesforce.

1. Go to [developers.salesforce.com/](https://developer.salesforce.com/)
2. Click on sign up.
3. On the sign up form, enter the following details :
 1. First name & Last name



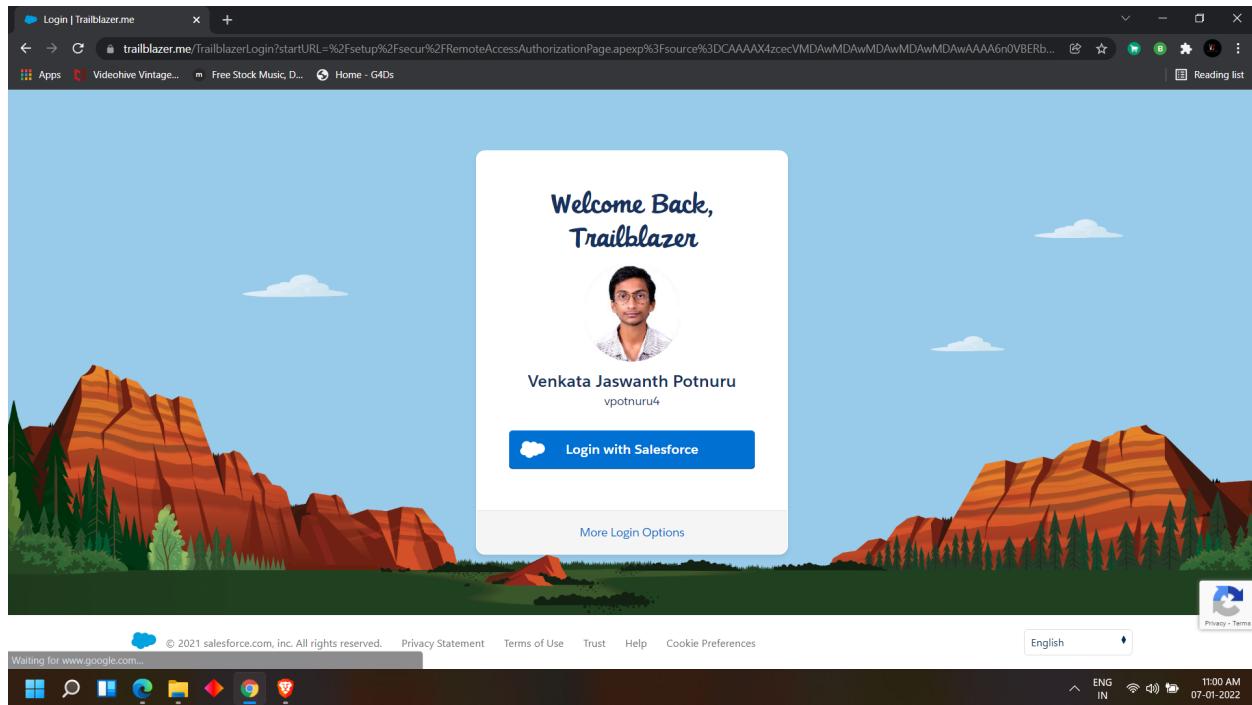
2. Email
3. Role : Developer
4. Company : College Name
5. County : India
6. Postal Code : pin code
7. Username : should be a combine of your name and company

This need not be an actual email id, you can give anything in the format :

username@system.com

Login into Trail head with salesforce account

A screenshot of a web browser displaying the Trailhead website at trailhead.salesforce.com/en. The page features a colorful, cartoonish illustration of a landscape with a waterfall, a character in a bear costume holding a map, and another character in a raccoon costume. The main heading "SKILL UP FOR THE FUTURE" is prominently displayed in large, bold, dark blue letters. Below it, the subtext "Learn new skills from anywhere" is shown. A central orange button with the text "Get Started for Free" is visible. The top navigation bar includes links for Home, Learn, Credentials, Community, For Companies, and COVID-19. On the right side of the header, there are "Sign Up" and "Log In" buttons. The bottom of the page features a dark footer bar with various icons and the text "TRAILHEAD IS THE FUN WAY TO LEARN". The browser's address bar shows the URL, and the taskbar at the bottom displays several open application icons.



Login To Your Salesforce Account

- 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.



Login | Salesforce

mvgrcollegeofengineering166-dev-ed.my.salesforce.com/?startURL=%2Fsetup%2Fsecur%2FRemoteAccessAuthorizationPage.apexp%3Fsource%3DCAAAAAX4zcf0PMDAwMDAwMD...

Apps Videohive Vintage... Free Stock Music, D... Home - G4Ds Reading list

Product news

Together, Safely with Dreampass.

Manage health and safety protocols and automate entry for safe event experiences.

[Read the release](#)

[Learn about Dreampass >](#)

salesforce

Username 1 Saved Username

venkatajaswanth10@gmail.com

Password

Log In

Remember me

Forgot Your Password?

© 2022 salesforce.com, Inc. All rights reserved.

Vaccine Check

Proof of vaccination
Rapid test scheduled
Rapid test results

A man in a suit, a woman in a yellow coat, and a brown bear-like character standing in front of a city skyline.

You've been verified

11:00 AM 07-01-2022 ENG IN

Trailhead | The fun way to learn

trailblazer.me/id?lang=en_US

Apps Videohive Vintage... Free Stock Music, D... Home - G4Ds Reading list

Trailhead

Venkata Jaswanth Potnuru

MVGR at MVGR College of engineering
Andhra Pradesh, India

Tell us about yourself! Add a short bio.

.me trailblazer.me/id/vpotnuru4

106 Badges

102,875 Points

11 Trails

0 Certifications

Go to Trailhead

11:00 AM 07-01-2022 ENG IN



DAY 2:

Topic: Milestone / Activities: Creating Objects, and creating tabs

Detailed Description:

Custom Object Creation:

1. College
2. Application Form
3. Students
4. Subjects

To create an object click on the gear button > click on set up

There you can find the object manager > create > custom object

A screenshot of the Salesforce Setup interface. The browser address bar shows 'empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/ObjectManager/new'. The page title is 'New Custom Object | Salesforce'. The main content area is titled 'New Custom Object' and contains fields for 'Label' (College), 'Plural Label' (Colleges), and 'Object Name' (College). There are sections for 'Description', 'Context-Sensitive Help Setting' (set to 'Open the standard Salesforce.com Help & Training window'), and 'Content Name' ('None'). Below these are sections for 'Enter Record Name Label and Format' (Record Name: College Name, Example: Account Name, Data Type: Text) and 'Optional Features' (Allow Reports checked). The bottom right corner shows the system status: ENG IN, 06:53 PM, 04-01-2022.

College object:

A screenshot of a Microsoft Edge browser window displaying the Salesforce Setup interface. The URL is empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j00000tFX6/Details/view. The page shows the "College" object details. The left sidebar lists various configuration tabs: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Restriction Rules, Triggers, and Validation Rules. The main content area is titled "Details" and contains sections for "Description" and "API Name" (set to "College__c"). It also includes sections for "Custom" (with "Singular Label" set to "College" and "Plural Label" set to "Colleges"), "Enable Reports", "Track Activities", "Track Field History", "Deployment Status" (set to "Deployed"), and "Help Settings". At the bottom right are "Edit" and "Delete" buttons. The status bar at the bottom of the browser shows system information: ENG IN, 06:55 PM, 04-01-2022.

Application Form:



Set Up Single Sign-On for Your... | Recently Viewed | Colleges | Sales | Application Form | Salesforce

empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000tG8w/Details/view

Cloud icon, Apps, Videohive Vintage..., Free Stock Music, D... Home - G4Ds Reading list

Setup Home Object Manager

Search Setup

SETUP > OBJECT MANAGER Application Form

Details

Description

API Name Application_Form_c

Custom ✓

Singular Label Application Form

Plural Label Application Forms

Enable Reports

Track Activities

Track Field History

Deployment Status Deployed

Help Settings Standard salesforce.com Help Window

Edit Delete

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

11:17 AM ENG IN 07-01-2022

Students:

Set Up Single Sign-On for Your... | Recently Viewed | Colleges | Sales | Student | Salesforce

empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000tGWB/view

Cloud icon, Apps, Videohive Vintage..., Free Stock Music, D... Home - G4Ds Reading list

Setup Home Object Manager

Search Setup

SETUP > OBJECT MANAGER Student

Details

Description

API Name Student_c

Custom ✓

Singular Label Student

Plural Label Students

Enable Reports

Track Activities

Track Field History

Deployment Status Deployed

Help Settings Standard salesforce.com Help Window

Edit Delete

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

11:17 AM ENG IN 07-01-2022

Subjects:



Screenshot of the Salesforce Setup interface showing the Object Manager for the "Subject" object.

The URL in the browser is empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j00000tGWG/Details/view.

The page title is "SETUP > OBJECT MANAGER Subject".

The left sidebar shows the following navigation 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
- Partition Rules

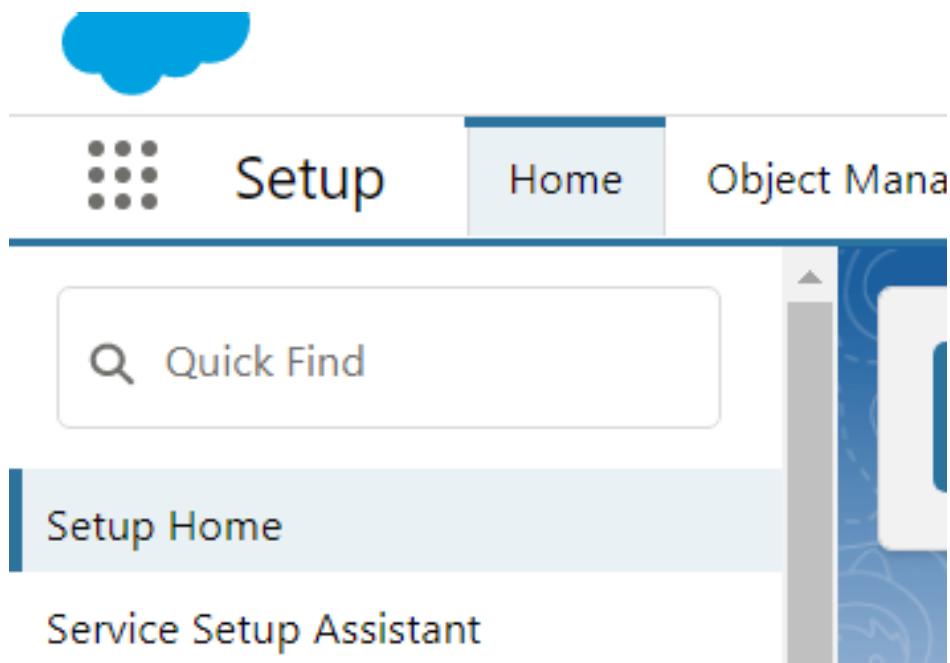
The main content area is titled "Details" and contains the following fields:

Description	Value
API Name	Subject_c
Custom	✓
Singular Label	Subject
Plural Label	Subjects
Enable Reports	✓
Track Activities	✓
Track Field History	✓
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

At the bottom right of the screen, there is a system status bar showing: ENG IN 11:17 AM 07-01-2022.

Now creating Tabs to the objects which we have created

click on the gear button > there we can find the quick find box >



search for tabs > click on tabs

here in tabs we can find different types like:

- Custom Object Tabs
- Visual force Tabs
- Lightning Componenet Tabs
- Lightning Page Tabs

we need to create in Custom Object Tabs because we have created custom objects for the app manager



The screenshot shows the Salesforce Setup interface with the 'Tabs' tab selected. In the 'Custom Tabs' section, there is a table titled 'Custom Object Tabs'. A context menu is open over the 'Properties' row, listing network adapter options: 'POTNURU'S WiFi Internet access', 'VMware Network Adapter VMnet8 No internet access', and 'VMware Network Adapter VMnet11 No internet access'. The table has columns for Action (Edit | Del), Label (Application Forms, Bot Commands, Brokers, Colleges, Properties), Tab Style (CD/DVD, Radar dish, Presenter, Sun, Real Estate Sign, Hands, Computer), and Description.

we need to create a new tab so click on new in custom object tabs> select the object form the given list>select a icon for the object >

The screenshot shows the 'New Custom Object Tab' wizard, Step 1. Enter the Details. It asks to choose a custom object or create a new one. The 'Object' dropdown is set to 'Favorite' and the 'Tab Style' dropdown is set to 'Guitar'. There is an optional field for a splash page custom link, which is currently empty. A description text area is present at the bottom.

click next> next> save

In the same way create tabs for all the 4 objects (College, Application form, Students and



Subjects)

A screenshot of the Salesforce Setup interface, specifically the "Tabs" page under "User Interface".

The page title is "SETUP Tabs".

The main content area displays a list of tabs categorized into "Application Icons" and "Web Tabs".

Application Icons:

Action	Label	Icon
Edit Del	CD/DVD	CD/DVD icon
Edit Del	Radar dish	Radar dish icon
Edit Del	Presenter	Presenter icon
Edit Del	Sun	Sun icon
Edit Del	Real Estate Sign	Real Estate Sign icon
Edit Del	Hands	Hands icon
Edit Del	Computer	Computer icon
Edit Del	Subjects	Subjects icon

Web Tabs:

Action	Label	Tab Style	Description
Edit Del	Data Import	Wrench	
Edit Del	Heat Map	Map	
Edit Del	Heat Map (Demo)	Map	

Visualforce Tabs:

Action	Label	Tab Style	Description
Edit Del	Data Import	Wrench	
Edit Del	Heat Map	Map	
Edit Del	Heat Map (Demo)	Map	

Lightning Component Tabs:

Action	Label	Tab Style	Description
Edit Del	Data Import	Wrench	
Edit Del	Heat Map	Map	
Edit Del	Heat Map (Demo)	Map	

The browser address bar shows the URL: <https://empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/CustomTabs/home>.

The bottom right corner of the screen shows system status: ENG IN 03:32 PM 07-01-2022.



DAY 3:

Topic: Milestone / Activities: Create an app manager and creating global picklist

Detailed Description:

Create an App manager for College Management Application:

we have to create an App for the college management application so

Navigate to Quick find box > search for App manager > click on app manager

A screenshot of the Salesforce Lightning App Manager interface. The top navigation bar shows 'Recently Viewed | Colleges | Sales' and 'Home | Salesforce'. The main search bar contains 'app manager'. On the left, a sidebar shows 'Setup' selected under 'Home' and 'Object Manager'. A message says 'Didn't find what you're looking for? Try using Global Search.' Below the search bar, there's a 'Create' button. The main content area features three cards: 'Get Started with Einstein Bots', 'Mobile Publisher', and 'Real-time Collaborative Docs'. At the bottom, there's a section titled 'Most Recently Used' with a table showing columns for NAME, TYPE, and OBJECT. The status bar at the bottom right shows 'ENG IN' and the date '07-01-2022'.

you can find the list of apps which are available now make sure we are going to create a new lightning builder so > click on New Lightning App



A screenshot of the Salesforce App Manager interface. The title bar says "Lightning Experience App Manager". On the left, there's a sidebar with "App Manager" selected. The main area shows a table with 29 items, sorted by App Name. The columns include App Name, Developer Name, Description, Last Modified, App Type, and Version. Some rows are partially cut off. A small screenshot of a mobile device displaying the app is shown in the center-left. The bottom right corner shows system status: ENG IN, 03:42 PM, 07-01-2022.

Select the name of the app and enter in the app details

A screenshot of the "New Lightning App" configuration page. The title bar says "App Details & Branding". It has two main sections: "App Details" and "App Branding". In the "App Details" section, there are fields for "App Name" (College Management Application) and "Developer Name" (College_Management_Application). There's also a "Description" field with placeholder text "Enter a description...". In the "App Branding" section, there's a "Primary Color Hex Value" set to "#0070D2", a "Upload" button for an image, and a checkbox for "Org Theme Options" with the sub-option "Use the app's image and color instead of the org's custom theme". At the bottom right is a "Next" button. The background shows the rest of the App Manager interface with other app details visible.

Later on fill details as per the Requirements

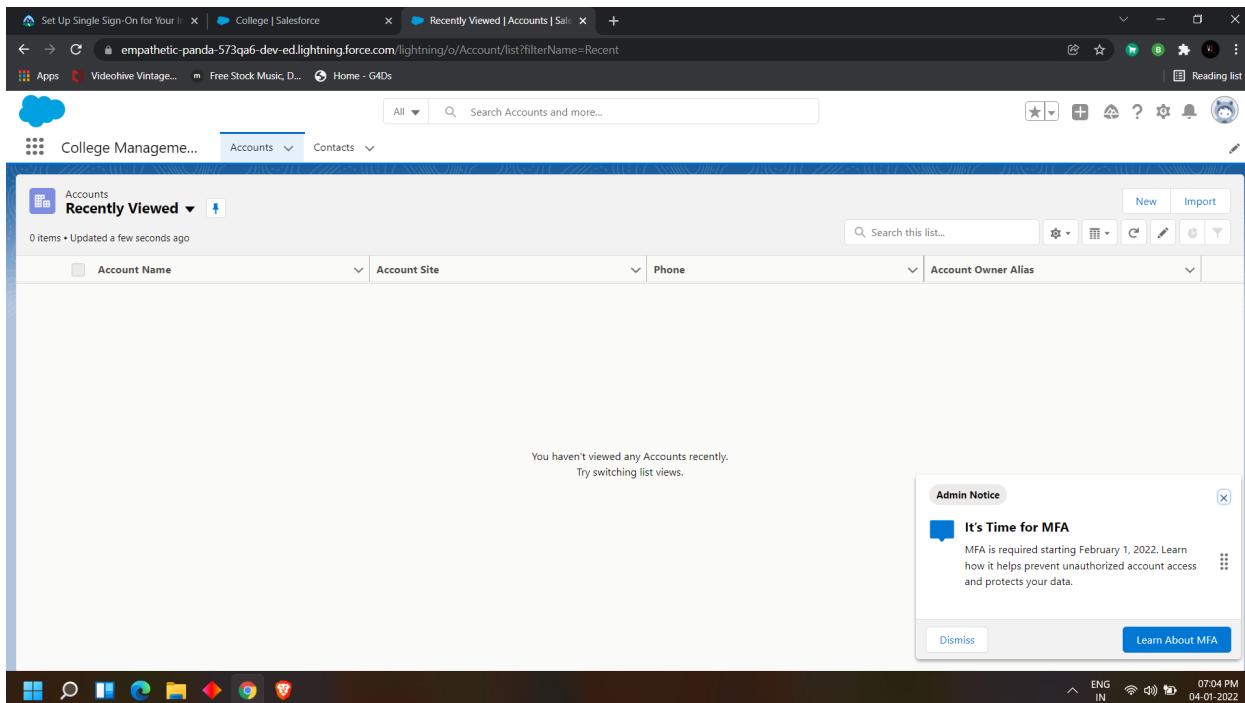


Screenshot of the Salesforce App Manager showing the "New Lightning App" setup screen. A callout box highlights the "Standard navigation shows items in a navigation bar at the top of the page. Console navigation shows a list of records in a sidebar and opens each record in a new workspace tab. Also, you can select the form factors where this app is available." section. The "Navigation and Form Factor" section includes options for "Navigation Style" (Standard navigation selected) and "Supported Form Factors" (Desktop and phone selected). The "Setup and Personalization" section includes "Setup Experience" (Setup selected) and "App Personalization Settings" checkboxes for "Disable end user personalization of nav items in this app" and "Disable temporary tabs for items outside of this app".

Screenshot of the Salesforce App Manager showing the "Utility Items (Desktop Only)" setup screen. A callout box highlights the "The utility bar is a fixed footer that opens components in docked panels. Available only when the app is viewed in Lightning Experience on a desktop." section. The "Add Utility Item" button is visible, along with "Utility Bar Alignment" set to "Default".



After creating the app manager navigate to App launcher and search for the app which you have created and open it



Creating global picklist values

when we declaring a field as picklist we can choose if we want custom picklist or global picklist

we need to create the global picklist values before using this
so head to quick find box in the setup>picklist value sets



A screenshot of a web browser displaying the Salesforce Setup interface. The URL in the address bar is "empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/Picklists/home". The page title is "Picklist Value Sets". The left sidebar shows navigation categories: Data (Picklist Settings, State and Country/Territory, Picklists), Objects and Fields (Picklist Value Sets). A search bar at the top has "pick" typed into it. The main content area is titled "Picklist Value Sets" and contains a sub-section "Global Value Sets". It shows a table with three rows: "Action" (Edit | Del), "Label" (College), and "Description" (empty). Below the table, there is a link "Deleted Global Value Sets (0)". At the bottom right of the page, there is a "Help for this Page" link. The browser's status bar at the bottom shows system information: ENG IN, 02:47 PM, 08-01-2022.

click on new>

enter the name of the global picklist>

enter the values which you want to give there>

save

college



Smart Internz

Trailhead | The fun way to learn | All | Colleges | Salesforce | Student | Salesforce | Picklist Value Sets | Salesforce

empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/Picklists/page?address=%2F0Nt5j0000004MJI

Apps Videohive Vintage... Free Stock Music, D... Home - G4Ds Reading list

Cloud Setup Home Object Manager

Search Setup

pick

Data

- Picklist Settings
- State and Country/Territory
- Picklists

Objects and Fields

- Picklist Value Sets

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

Picklist Value Sets

Global Value Set Detail

Information

Label	College
Name	College
Description	

Edit Delete

Picklist Values Used

Active and Inactive picklist values 6 (1,000 max)

Edit Delete

Values

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	MIT-HYD	MIT-HYD	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM
Edit Del Deactivate	MIT-BLR	MIT-BLR	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM
Edit Del Deactivate	MIT-MUM	MIT-MUM	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM
Edit Del Deactivate	MIT-MAA	MIT-MAA	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM
Edit Del Deactivate	MIT-DEL	MIT-DEL	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM
Edit Del Deactivate	MIT-CCU	MIT-CCU	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM

Inactive Values

ENG IN 02:50 PM 08-01-2022

Paper1

Trailhead | The fun way to learn | All | Colleges | Salesforce | Student | Salesforce | Picklist Value Sets | Salesforce

empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/Picklists/page?address=%2F0Nt5j0000004MJI

Apps Videohive Vintage... Free Stock Music, D... Home - G4Ds Reading list

Cloud Setup Home Object Manager

Search Setup

pick

Data

- Picklist Settings
- State and Country/Territory
- Picklists

Objects and Fields

- Picklist Value Sets

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

Picklist Value Sets

Global Value Set Detail

Information

Label	Paper1
Name	Paper1
Description	

Edit Delete

Picklist Values Used

Active and Inactive picklist values 4 (1,000 max)

Edit Delete

Values

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	APEX	APEX	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM
Edit Del Deactivate	JAVA	JAVA	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM
Edit Del Deactivate	C	C	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM
Edit Del Deactivate	C++	C++	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu 1/5/2022, 4:10 AM

Inactive Values

No Inactive Values values defined

Fields Where Used

ENG IN 02:50 PM 08-01-2022

Paper2



Screenshot of the Salesforce Setup interface showing the "Picklist Value Sets" page.

The URL in the browser is `empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/Picklists/page?address=%2F0Nt5j0000004MJS`.

The left sidebar shows the navigation path: Data > Objects and Fields > Picklist Value Sets.

The main content area displays the "Global Value Set" details:

- Global Value Set Detail:** Label: Paper2, Name: Paper2, Description: (empty)
- Picklist Values Used:** Active and Inactive picklist values: 3 (1,000 max). A table lists three values:

Action	Values	API Name	Default	Chart Colors	Modified By
Edit Del Deactivate	MATHEMATICS	MATHEMATICS	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu, 1/5/2022, 4:11 AM
Edit Del Deactivate	ENGLISH	ENGLISH	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu, 1/5/2022, 4:11 AM
Edit Del Deactivate	STATISTICS	STATISTICS	<input type="checkbox"/>	Assigned dynamically	Venkata Jaswanth Poturu, 1/5/2022, 4:11 AM

The bottom status bar shows system information: javascriptsrcUp(%27%2F0Nt5j0000004MJS%3fstdp%3dp1%27); ENG IN 02:51 PM 08-01-2022



DAY 4:

Topic: Milestone / Activities: Creating fields for objects, and creating field dependencies

Detailed Description:

Creating fields for the objects which we have created (College, students, Subjects and Application form)

to create the fields in the object

click on the gear button> select setup> here we can see the object manager here we find the all objects which are present

A screenshot of a web browser displaying the Salesforce Object Manager. The URL in the address bar is 'https://empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/ObjectManager/home'. The page title is 'Object Manager | Salesforce'. The main content area shows a table of objects with columns: LABEL, API NAME, TYPE, DESCRIPTION, LAST MODIFIED, and DEPLOYED. The table lists various objects including Account, Activity, Alternative Payment Method, API Anomaly Event Store, Application Form, Asset, Asset Action, Asset Action Source, Asset Relationship, Asset State Period, Associated Location, and Authorization Form. The 'Application Form' row is highlighted. The bottom status bar shows the URL again and system information like 'ENG IN' and the date '08-01-2022'.

select a object



The screenshot shows the Salesforce Object Manager interface for the 'College' object. On the left, a sidebar lists various setup options like Page Layouts, Lightning Record Pages, and Field Sets. The main area is titled 'Fields & Relationships' and displays a table of fields. The columns are: FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The fields listed are: Capacity Of Students (Capacity_of_Students__c, Picklist, College Name), College Fees (College_Fees__c, Currency(7, 2)), College Name (College_Name__c, Picklist), College Name (Name, Text(80)), Created By (CreatedBy, Lookup(User)), Email (Email__c, Picklist, College Name), Hostel Fees (Hostel_Fees__c, Currency(6, 2)), Last Modified By (LastModifiedBy, Lookup(User)), and Owner (OwnerId, Lookup(User, Group)).

click on the fields and relationships

there we can see all the fields which are present in the object before.

So we can create new fields as per our requirements by clicking on new

The screenshot shows the 'New Custom Field' creation page for the 'College' object. The sidebar on the left is identical to the previous screenshot. The main form is titled 'Step 1. Choose the field type'. It asks 'Specify the type of information that the custom field will contain.' Under 'Data Type', 'None Selected' is selected. Other options include: Auto Number (A system-generated sequence number that uses a display format you define. The number is automatically incremented for each new record.), Formula (A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.), Roll-Up Summary (A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.), Lookup Relationship (Creates a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other object is the source of the values in the list.), Master-Detail Relationship (Creates a special type of parent-child relationship between this object (the child, or "detail") and another object (the parent, or "master") where: • The relationship field is required on all detail records. • The ownership and sharing of a master record are determined by the master record. • When a user edits this master record, all detail records are updated. You can create roll-up summary fields on the master record to summarize the detail records.), External Lookup Relationship (Creates a relationship that links this object to an external object whose data is stored outside the Salesforce org.), Checkbox (Allows users to select a True (checked) or False (unchecked) value.), and Text (Allows users to enter a dollar or other currency amount and automatically formats the field as a currency amount. This can be useful if you export data to Excel or another spreadsheet.).

choose the field type

there we can find different data types



None Selected

Select one of the data types below.

VAuto Number

A system-generated sequence number that uses a display format you define. The number is automatically incremented for each new record.

ZFormula

A read-only field that derives its value from a formula expression you define. The formula field is updated when any of the source fields change.

1Roll-Up Summary

A read-only field that displays the sum, minimum, or maximum value of a field in a related list or the record count of all records listed in a related list.

YLookup Relationship

Creates a relationship that links this object to another object. The relationship field allows users to click on a lookup icon to select a value from a popup list. The other object is the source of the values in the list.

Y_MDMaster-Detail Relationship

Creates a special type of parent-child relationship between this object (the child, or "detail") and another object (the parent, or "master") where:

- The relationship field is required on all detail records.
- The ownership and sharing of a detail record are determined by the master record.
- When a user deletes the master record, all detail records are deleted.
- You can create roll up summary fields on the master record to summarize the detail records.

The relationship field allows users to click on a lookup icon to select a value from a popup list. The master object is the source of the values in the list.

xExternal Lookup Relationship

Creates a relationship that links this object to an external object whose data is stored outside the Salesforce org.

BCheckbox

Allows users to select a True (checked) or False (unchecked) value.

CCurrency

Allows users to enter a dollar or other currency amount and automatically formats the field as a currency amount. This can be useful if you export data to Excel or another spreadsheet.

DDate

Allows users to enter a date or pick a date from a popup calendar.

FDate/Time

Allows users to enter a date and time, or pick a date from a popup calendar. When users click a date in the pop-up, that date and the current time are entered into the Date/Time field.



EEmail	Allows users to enter an email address, which is validated to ensure proper format. If this field is specified for a contact or lead, users can choose the address when clicking Send an Email. Note that custom email addresses cannot be used for mass emails.
gGeolocation	Allows users to define locations. Includes latitude and longitude components, and can be used to calculate distance.
NNumber	Allows users to enter any number. Leading zeros are removed.
PPercent	Allows users to enter a percentage number, for example, '10' and automatically adds the percent sign to the number.
HPhone	Allows users to enter any phone number. Automatically formats it as a phone number.
LPicklist	Allows users to select a value from a list you define.
QPicklist (Multi-Select)	Allows users to select multiple values from a list you define.
SText	Allows users to enter any combination of letters and numbers.
XText Area	Allows users to enter up to 255 characters on separate lines.
JText Area (Long)	Allows users to enter up to 131,072 characters on separate lines.
zText Area (Rich)	Allows users to enter formatted text, add images and links. Up to 131,072 characters on separate lines.
6Text (Encrypted)	Allows users to enter any combination of letters and numbers and store them in encrypted form.
fTime	Allows users to enter a local time. For example, "2:40 PM", "14:40", "14:40:00", and "14:40:50.600" are all valid times for this field.
UURL	Allows users to enter any valid website address. When users click on the field, the URL will open in a separate browser window.

choose as per the requirement

fill the details like
field name and all

college object fields

Field Name	Data Type	Requir	Values
------------	-----------	--------	--------



		ed	
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

Application form object fields

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	



student object fields

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	

subjects object fields

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

after creating the values are

college object



Screenshot of the Salesforce Object Manager for the "College" object.

The page shows the "Fields & Relationships" section with 10 items. The table includes columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Capacity Of Students	Capacity__Of_Students__c	Picklist	College Name	
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	College Name	
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(10)		

Details sidebar:

- 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
- Triggers
- Validation Rules

application form

Screenshot of the Salesforce Object Manager for the "Application Form" object.

The page shows the "Fields & Relationships" section with 15 items. The table includes columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address__c	Text(255)		
Application Form ID	Application_Form__D__c	Auto Number		
Application Form Name	Name	Text(50)		✓
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)		
Hostel Fees	Hostel_Fees__c	Formula(Currency)		
Last Modified By	LastModifiedById	Lookup(User)		
Looking For Hostel Stay	Looking_For_Hostel_Stay__c	Checkbox		
Phone	Phone__c	Phone		
Ready To Join	Ready_To_Join__c	Checkbox		
Student Name	Student_Name__c	Text(30)		

Details sidebar:

- 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
- Triggers
- Validation Rules

student object



This screenshot shows the Salesforce Object Manager interface for the "Student" object. The left sidebar lists various setup options like Page Layouts, Lightning Record Pages, and Field Sets. The main content area is titled "Fields & Relationships" and displays 11 items sorted by Field Label. The table includes columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. Key fields shown include Address, Application Form, College Name, Created By, Date Of Birth, Guardian Name, Last Modified By, Owner, Phone, Student Name, and Student Name. The "Owner" field is highlighted with a red border.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Address	Address_c	Text(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(10)		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		
Phone	Phone_c	Phone		
Student Name	Student_Name__c	Text(10)		
Student Name	Name	Text(80)		

subjects object

This screenshot shows the Salesforce Object Manager interface for the "Subject" object. The left sidebar lists various setup options. The main content area is titled "Fields & Relationships" and displays 8 items sorted by Field Label. The table includes columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. Key fields shown include Created By, Last Modified By, Owner, Paper 1, Paper2, Student, Subject ID, and Subject Name. The "Owner" field is highlighted with a red border.

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)		
Paper 1	Paper_1__c	Picklist		
Paper2	Paper2__c	Picklist		
Student	Student__c	Lookup(Student)		
Subject ID	Subject_ID__c	Auto Number		
Subject Name	Name	Text(80)		

Creating Field Dependencies

We have created some fields in every object so now we need to give some field dependencies



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. 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.

To assign field dependencies

go to setup>

object manager>

Go to the object which we have to assign the field dependencies>

fields and relationships

A screenshot of the Salesforce Object Manager interface. At the top, there is a search bar labeled 'Find' and several navigation tabs: 'New', 'Deleted Fields', 'Field Dependencies' (which is highlighted in blue), and 'Set History Tracking'. Below the tabs, there are two columns: 'CONTROLLING FIELD' and 'INDEXED'. A small arrow icon is located at the bottom right of the table area.

Click on field dependencies



This screenshot shows the Salesforce Object Manager interface for the 'College' object. The left sidebar has 'Fields & Relationships' selected. The main content area is titled 'College Field Dependencies' and shows a table of existing field dependencies:

Action	Controlling Field	Dependent Field	Modified By
Edit Del	College Name	Email	Venkata Jswanth Potnuru, 1/5/2022, 4:44 AM
Edit Del	College Name	Capacity Of Students	Venkata Jswanth Potnuru, 1/5/2022, 4:44 AM

The status bar at the bottom indicates the date and time as 08-01-2022 08:57 PM.

Click on new

This screenshot shows the 'New Field Dependency' setup page. The left sidebar has 'Fields & Relationships' selected. The main content area has instructions for creating a dependency:

Create a dependent relationship that causes the values in a picklist or multi-select picklist to be dynamically filtered based on the value selected by the user in another field.
• The field that drives filtering is called the "controlling field." Standard and custom checkboxes and picklists with at least one and less than 300 values can be controlling fields.
• The field that has its values filtered is called the "dependent field." Custom picklists and multi-select picklists can be dependent fields.

Step 1. Select a controlling field and a dependent field. Click Continue when finished.
Step 2. On the following page, edit the filter rules that control the values that appear in the dependent field for each value in the controlling field.

Controlling Field: None
Dependent Field: None

The status bar at the bottom indicates the date and time as 08-01-2022 03:00 PM.

Fill the details and continue



Screenshot of the Salesforce Setup Object Manager interface for the 'College' object.

Fields & Relationships

Controlling Field: College Name
Dependent Field: Email

Instructions

- Double-click on a cell to toggle its visibility for the Controlling Field value shown in the column heading.
- To change multiple cells at once, select multiple cells and then click the Include Values or Exclude Values button to change the visibility of all selected cells at once.
- Use SHIFT + click to select a range of adjacent cells. Use CTRL + click to select multiple cells that are not adjacent.
- Use the Preview button to test the results.

College Name:	MIT-HYD	MIT-BLR	MIT-MUM	MIT-MAA	MIT-DEL
Email:	bir@mit.co.in hyd@mit.co.in	bir@mit.co.in hyd@mit.co.in	bir@mit.co.in hyd@mit.co.in	bir@mit.co.in hyd@mit.co.in	bir@mit.co.in hyd@mit.co.in
	mum@mit.co.in	mum@mit.co.in	mum@mit.co.in	mum@mit.co.in	mum@mit.co.in
	maa@mit.co.in	maa@mit.co.in	maa@mit.co.in	maa@mit.co.in	maa@mit.co.in
	cov@mit.co.in	cov@mit.co.in	cov@mit.co.in	cov@mit.co.in	cov@mit.co.in
	del@mit.co.in	del@mit.co.in	del@mit.co.in	del@mit.co.in	del@mit.co.in

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

Select the values and click on include values

Click on save

Screenshot of the Salesforce Setup Object Manager interface for the 'College' object.

Fields & Relationships

Controlling Field: College Name
Dependent Field: Capacity Of Students

Instructions

- Double-click on a cell to toggle its visibility for the Controlling Field value shown in the column heading.
- To change multiple cells at once, select multiple cells and then click the Include Values or Exclude Values button to change the visibility of all selected cells at once.
- Use SHIFT + click to select a range of adjacent cells. Use CTRL + click to select multiple cells that are not adjacent.
- Use the Preview button to test the results.

College Name:	MIT-HYD	MIT-BLR	MIT-MUM	MIT-MAA	MIT-DEL
Capacity Of Students:	500-1000 1000-2500 2500-6000 6000-10000	500-1000 1000-2500 2500-6000 6000-10000	500-1000 1000-2500 2500-6000 6000-10000	500-1000 1000-2500 2500-6000 6000-10000	500-1000 1000-2500 2500-6000 6000-10000

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

We can see that it was created in the field dependencies

lets test this



Navigate to the app launcher > select the app (college management application)

Click on create

A screenshot of a web browser displaying the Salesforce Lightning interface. The title bar shows three tabs: 'Trailhead | The fun way to learn', 'New College | Salesforce', and 'College | Salesforce'. The main content area is titled 'New College' and contains a form with several fields. The 'College Name' field is highlighted with a red border, indicating it is required. Other fields like 'Owner', 'Record Info', 'Capacity Of Students', 'College Fees', 'Email', and 'Capacity Of Students' dropdowns are shown but appear disabled or grayed out. The status bar at the bottom right shows '03:03 PM' and '08-01-2022'.

There we can see that email and capacity of students are disabled

A screenshot of the same Salesforce interface, but now with the 'College Name' field populated with 'MIT-HYD'. This action has triggered dependencies, causing the 'Email' and 'Capacity Of Students' fields to become active and no longer grayed out. The status bar at the bottom right shows '03:04 PM' and '08-01-2022'.

When I enter the college name it enables both the fields according to the dependencies



A screenshot of a computer monitor displaying the Salesforce Lightning interface. The main title bar shows three tabs: "Trailhead | The fun way to learn", "New College | Salesforce", and "College | Salesforce". Below the tabs, the address bar indicates the URL is "empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/o/College__c/new?count=1&nooverride=1&useRecordTypeCheck=1&navigationLocation=LIST_VIEW&uid=1641634418...". The browser's toolbar includes icons for Apps, Videhive Vintage..., Free Stock Music, Home - G4Ds, and a Reading list. The main content area shows a "College Management..." page with a "Colleges" tab selected. A modal window titled "New College" is open, prompting for "Information". The "College Name" field is required and highlighted with a red border. The "Owner" field shows "Venkata Jaswanth Potnuru". Under "Record Info", there are fields for "Hostel Fees" (empty), "Capacity Of Students" (dropdown menu showing "--None--" and "500-1000" with a checkmark), "College Fees" (empty), and "Email" (dropdown menu showing "hyd@mit.co.in" with a checkmark). At the bottom of the modal are "Cancel", "Save & New", and a large blue "Save" button. The system status bar at the bottom right shows "ENG IN" and the date "08-01-2022" along with a timestamp of "03:05 PM".

we have only 1 value because we include only one value there in the field dependencies before

DAY 5:

Topic: Milestone / Activities: Creating Validation rules, and creating process automation

Detailed Description:

Creating Validation Rules

1. 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.
2. Create a validation rule on the application form object to stop any modification on the application form once a student record is created.

```
AND( Ready_To_Join__c == true,  
OR(  
ISCHANGED( Address__c ),  
ISCHANGED( College__c ),  
ISCHANGED( Date_Of_Birth__c ),  
ISCHANGED( Email__c ),  
ISCHANGED( Guardian_Name__c ),  
ISCHANGED( Phone__c )  
)  
)
```

to create a validation rule

In object manager go to the object >

We can find the validation rules there



Screenshot of the Salesforce Setup interface showing the Validation Rules page for the College object.

The Validation Rules table displays one item:

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Rule1	Top of Page	Values Mismatch	✓	Venkata Jaswanth Potnuru, 1/5/2022, 4:47 AM

The sidebar on the left lists various setup categories, and the status bar at the bottom shows the date as 08-01-2022.

Click on new>

Enter the name

Enter the formula based on the validation rule

Screenshot of the Salesforce Setup interface showing the creation of a new Validation Rule for the Application Form object.

The validation rule is named "Rule1" and has the formula:

```
AND( Ready_To_Join__c == true,  
OR( ISCHANGED( Address__c ) ,  
ISCHANGED( College__c ) ,  
ISCHANGED( Date_of_Birth__c ) ,  
ISCHANGED( Email__c ) ,  
ISCHANGED( Guardian_Name__c ) ,  
ISCHANGED( Phone__c ) ))
```

The error message is: "Discount percent cannot exceed 30%"

The status bar at the bottom shows the date as 08-01-2022.

Click on check syntax to check whether the syntax is correct or not



If there is no error click on save

The screenshot shows the Salesforce Object Manager page for the Application Form Validation Rule. The validation rule detail is displayed, with the rule name set to 'Rule1'. The error condition formula is:

```
AND( Ready__To_Join__c == true,  
ISCHANGED( Address__c ),  
ISCHANGED( College__c ),  
ISCHANGED( Date_of_Birth__c ),  
ISCHANGED( Email__c ),  
ISCHANGED( Guardian_Name__c ),  
ISCHANGED( Phone__c )  
)
```

The error message is 'Can't change the application form'. The rule is active and was created by Venkata Jaswanth Potnuru on 1/8/2022, 1:43 AM. The modified by field also shows Venkata Jaswanth Potnuru on the same date and time. The page includes standard Salesforce navigation and search tools.

Once the validation rule is created you can see the validation rule in the validation rules

The screenshot shows the Validation Rules section for the Application Form object. A single validation rule, 'Rule1', is listed. The rule has the following details:

Rule Name	Error Location	Error Message	Active	Modified By
Rule1	Top of Page	Can't change the application form	✓	Venkata Jaswanth Potnuru, 1/8/2022, 1:43 AM

The left sidebar lists various setup options like Details, Fields & Relationships, Page Layouts, etc. The bottom of the screen shows the Windows taskbar with the date and time as 08-01-2022 and 03:13 PM.



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. 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".
 - a. 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."

B Once the node is set up, click on the adjacent box called "Immediate action". And select create a record on the student object. Please follow the below screenshot.

To create a Process Automation

In quick find box search for process builder

A screenshot of a web browser displaying the Salesforce Process Builder interface. The URL in the address bar is "empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/ProcessAutomation/home". The page title is "Process Builder | Salesforce". The main content area shows a table titled "My Processes" with 5 items. The columns are: PROCESS, DESCRIPTION, OBJECT, PROCESS TYPE, LAST MODIFIED, STATUS, and ACTIONS. The processes listed are:

PROCESS	DESCRIPTION	OBJECT	PROCESS TYPE	LAST MODIFIED	STATUS	ACTIONS
> Advertise New Property	When a new property is added, post it to chatter	Property	Record Change	1/4/2022	Active	
> College Application		Application Form	Record Change	1/7/2022	Active	
> Opportunity Status Change	When the status of an opportunity changes, post the ... Opportunity	Opportunity	Record Change	1/4/2022	Active	
> Post Price Changes to Slack	When the price of a property changes, post it to Slack	Property	Record Change	1/4/2022	Active	
> Price Change Push Notification	When the price of a property changes, send Push Not...Property	Property	Record Change	1/4/2022	Active	

The browser's taskbar at the bottom shows various icons for Microsoft Office applications and the system tray indicates the date and time as 08-01-2022.



Click on new

Give the name of the process

The screenshot shows the Salesforce Process Builder interface. A modal window titled "New Process" is open. It has fields for "Name" and "Description". Below these, a dropdown menu is open under "The process starts when *", showing options: "Select One", "A record changes", "A platform event message is received", and "It's invoked by another process". At the bottom right of the modal are "Cancel" and "Save" buttons. In the background, the main "My Processes" list is visible, showing several existing processes. The system status bar at the bottom right indicates "ENG IN" and the date "08-01-2022".

The process starts when is

The screenshot shows the "Process Builder - College Application" screen. On the left, a process flow diagram for "Application Form" is displayed. It starts with a decision diamond "Ready to join". If "TRUE", it leads to an "IMMEDIATE ACTIONS" section with a "Student" entry and a "STOP" action. If "FALSE", it leads to another decision diamond "+ Add Criteria". If this second diamond is "TRUE", it leads to another "IMMEDIATE ACTIONS" section with a "+ Add Action" entry and a "STOP" action. If "FALSE", it leads to a "STOP" action. On the right, a sidebar titled "Define Criteria for this Action Group" is open. It shows a "Criteria Name" field with "Ready to join" and a "Conditions for Executing Actions" section with three radio button options: "Conditions are met", "Formula evaluates to true", and "No criteria—just execute the actions!". Below this is a "Set Conditions" table with one row: "Field * [Application_For...Q] Operator * Equals Type * Boolean Value * True". At the bottom of the sidebar are "Conditions" radio buttons for "All of the conditions are met (AND)" and "Any of the conditions are met (OR)", along with "Done" and "Cancel" buttons. The system status bar at the bottom right indicates "ENG IN" and the date "08-01-2022".

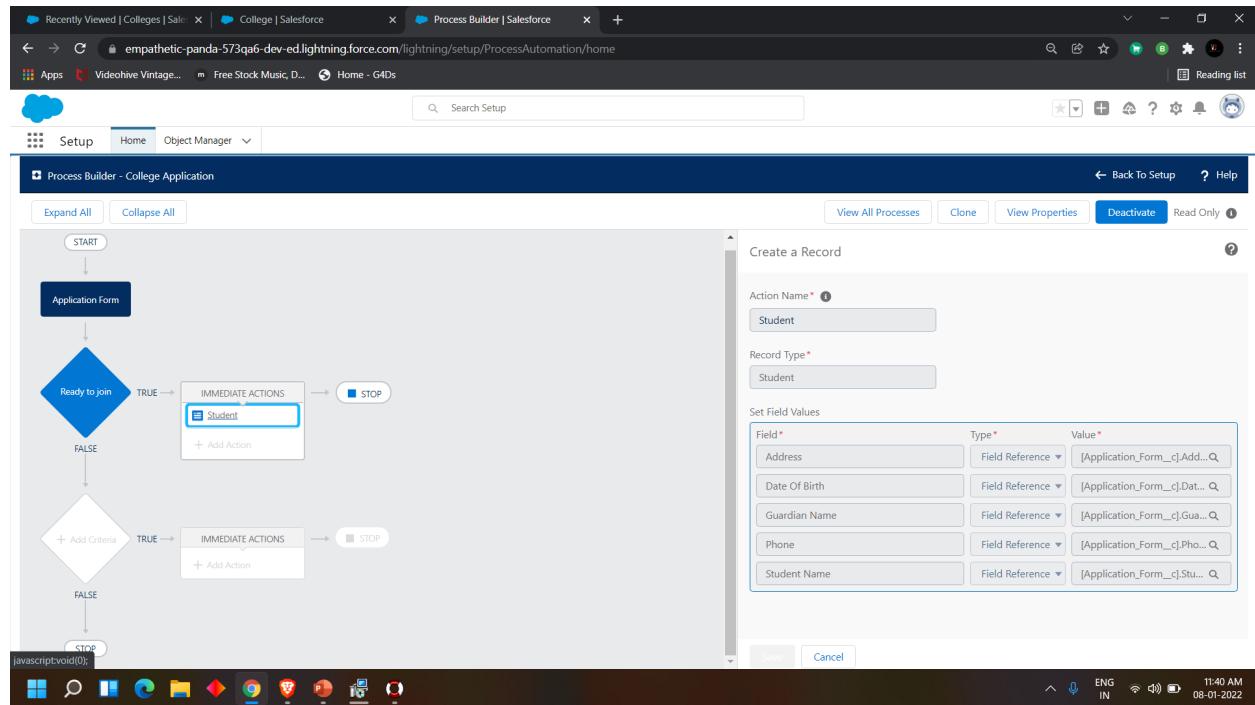


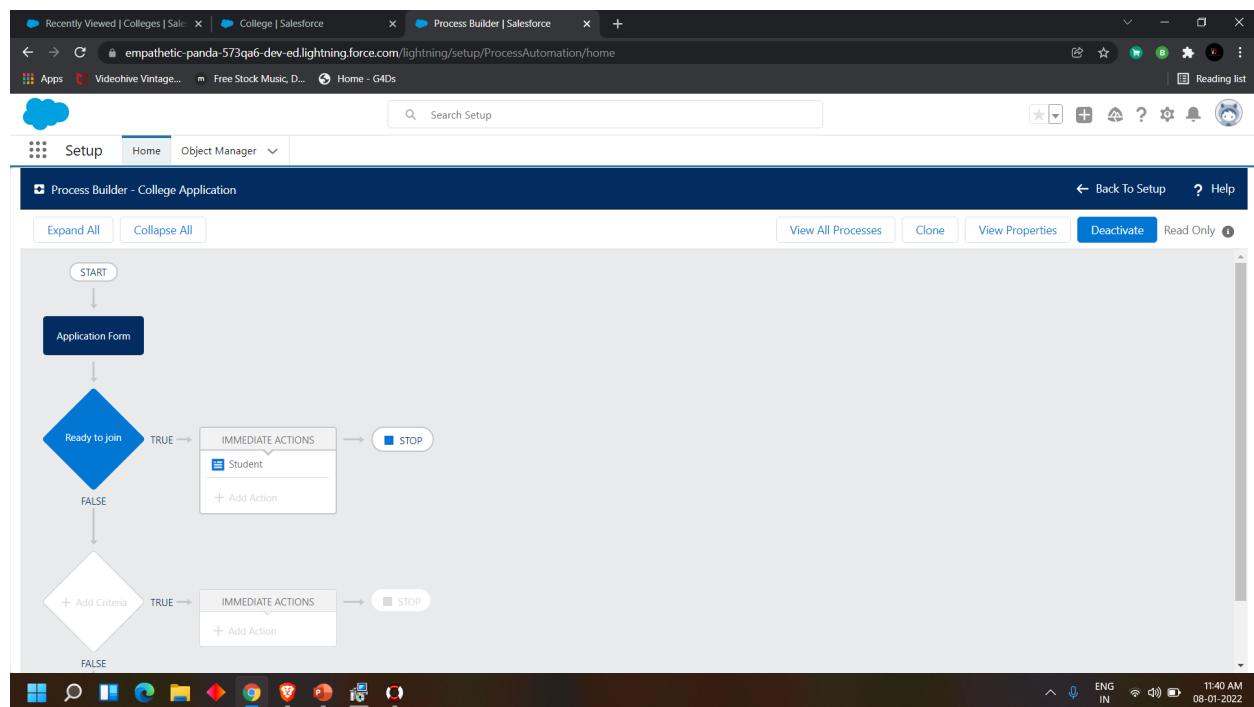
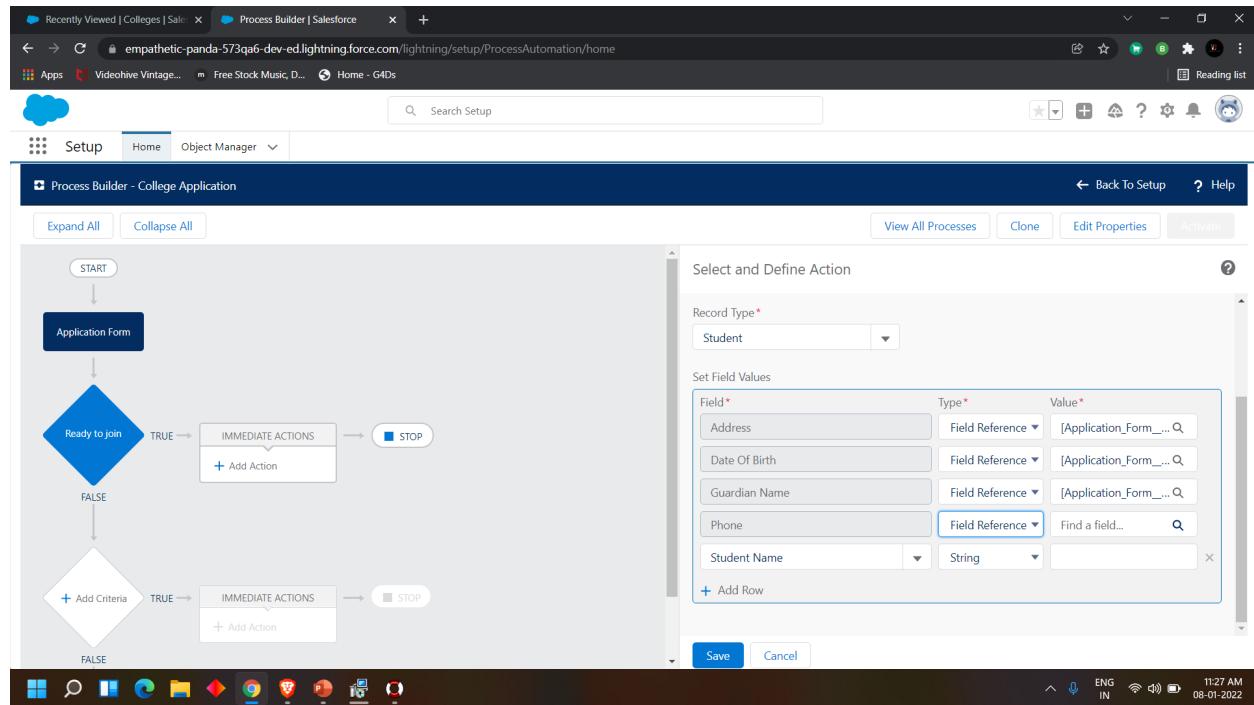
to be selected as per the process intention

then next

select object name > Application form

add criteria >







DAY 6:

Topic: Milestone / Activities: Create the student record using flow

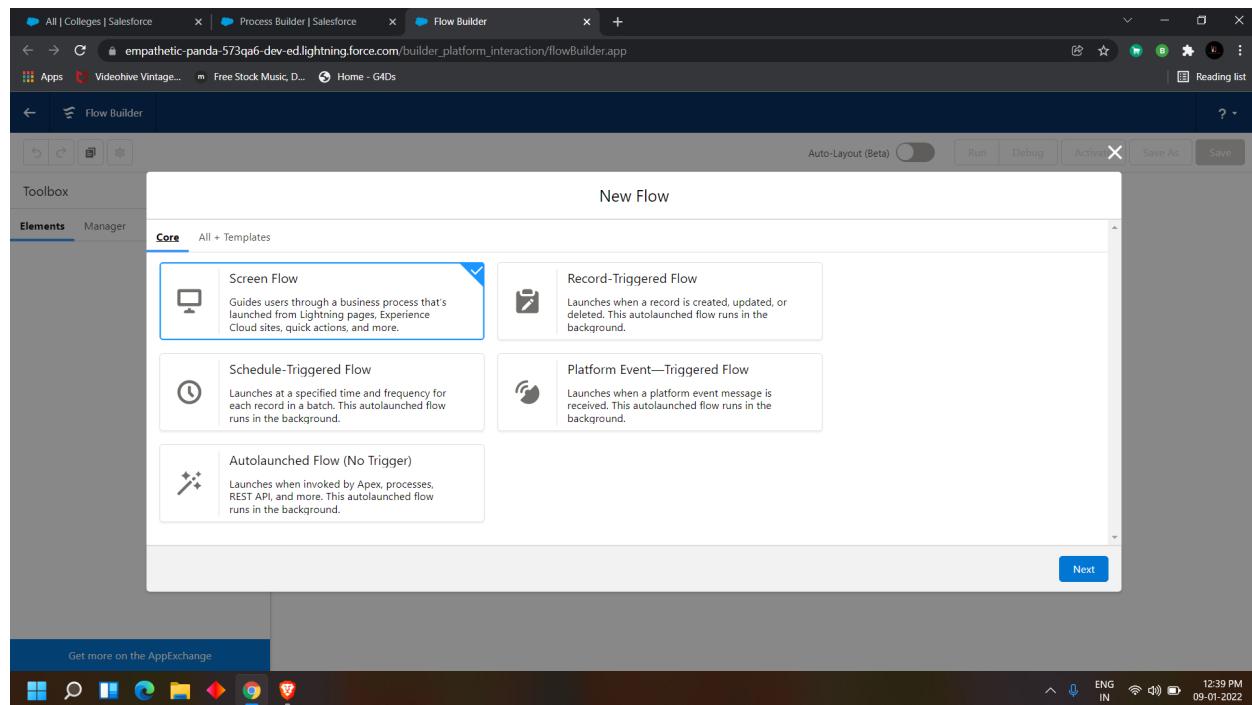
Detailed Description:

Create the Student Record using Flow

Skill Tags:

1. First deactivate the process builder which we created earlier.
2. Now search for flows and select new flow ->record triggered flow
3. 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.
4. Now create a variable named student in the resource section.
5. Now add the assignment as follows.

A screenshot of a web browser displaying the Salesforce Setup interface. The URL in the address bar is "https://empathetic-panda-573qa6-dev-ed.lightning.force.com/lightning/setup/Flows/home". The page title is "Flows". On the left, there's a sidebar with "Process Automation" expanded, showing "Flows" selected. Below it is "Identity" with "Login Flows" listed. A search bar at the top says "flows". The main content area shows a table titled "Flow Definitions All Flows". The table has columns: Flow Label, Process Type, Active, Template, Package State, Package N..., Last Modifi..., and Last Modifi.... The table lists several flows: "Cancel Item Flow" (Screen Flow, Active, Template checked, Managed-Installed), "Create a Case" (Screen Flow, Active, Template checked, Managed-Installed), "Create Order Summary Flow" (Autolaunched Flow, Active, Template checked, Managed-Installed), "Create Process Exception Flow" (Autolaunched Flow, Active, Template checked, Managed-Installed), "Reset Password" (Screen Flow, Active, Template checked, Managed-Installed), "Return Item Flow" (Screen Flow, Active, Template checked, Managed-Installed), and "Verify Identity" (Screen Flow, Active, Template checked, Managed-Installed). At the bottom right of the page, there's a footer with icons for Microsoft Edge, a magnifying glass, a play button, a red diamond, a globe, and a shield, along with the text "https://empathetic-panda-573qa6-dev-ed.lightning.force.com/one/one.app#/setup/Flows/h...". The system status bar at the bottom right shows "ENG IN", "12:38 PM", and the date "09-01-2022".



The screenshot shows the "New Flow" wizard in the Salesforce Flow Builder. The title bar reads "All | Colleges | Salesforce" and "Process Builder | Salesforce". The main area is titled "New Flow" and displays a "Core" category under "All + Templates".

The "Core" section contains four flow types:

- Screen Flow**: Guided users through a business process that's launched from Lightning pages, Experience Cloud sites, quick actions, and more.
- Record-Triggered Flow**: Launches when a record is created, updated, or deleted. This autolaunched flow runs in the background.
- Schedule-Triggered Flow**: Launches at a specified time and frequency for each record in a batch. This autolaunched flow runs in the background.
- Platform Event—Triggered Flow**: Launches when a platform event message is received. This autolaunched flow runs in the background.

Below the list is a "Next" button. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray showing the date and time as "09-01-2022 12:39 PM".



All | Colleges | Salesforce | Process Builder | Salesforce | Flow Builder | empathe... | Apps | Videohive Vintage... | Free Stock Music, D... | Home - G4Ds

Flow Builder

Toolbox

Elements Manager

How do you want to start building?

Freeform

You control the placement of all elements and connectors on the canvas.

Auto-Layout Beta

We automatically position all elements and connectors for you.

Get more on the AppExchange

12:40 PM 09-01-2022 ENG IN

This screenshot shows the initial interface of the Flow Builder. It features a central canvas with two examples of flow diagrams: one labeled 'Freeform' where the user controls element placement, and one labeled 'Auto-Layout Beta' where the platform automatically positions them. The interface includes a 'Toolbox' on the left, tabs for 'Elements' and 'Manager', and standard browser navigation and status bar elements at the bottom.

All | Colleges | Salesforce | Process Builder | Salesforce | Flow Builder | Home | Salesforce

empathetic-panda-573qa6-dev-ed.lightning.force.com/builder_platform_interaction/flowBuilder.app

Apps | Videohive Vintage... | Free Stock Music, D... | Home - G4Ds

Flow Builder

Select Elements

Toolbox

Manager

Search this flow...

New Resource

Configure Start

Select Object

Select the object whose records trigger the flow when they're created, updated, or deleted.

* Object: Application Form

Configure Trigger

* Trigger the Flow When:

- A record is created
- A record is updated
- A record is created or updated
- A record is deleted

Set Entry Conditions

Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Condition Requirements

Cancel Done

12:40 PM 09-01-2022 ENG IN

This screenshot shows the 'Configure Start' step of a flow. It includes sections for selecting the object ('Application Form') and configuring the trigger ('A record is created'). Below these, there's a detailed section for 'Set Entry Conditions' with instructions and a 'Condition Requirements' table. The interface is consistent with the top screenshot, including the 'Toolbox' and 'Manager' tabs.



Screenshot of the Salesforce Flow Builder interface showing the creation of a new Record-Triggered Flow named "New Resource".

The flow configuration screen includes the following fields:

- *Resource Type: Variable
- *API Name: NewStudent
- Description: (empty)
- *Data Type: Select... (dropdown menu)

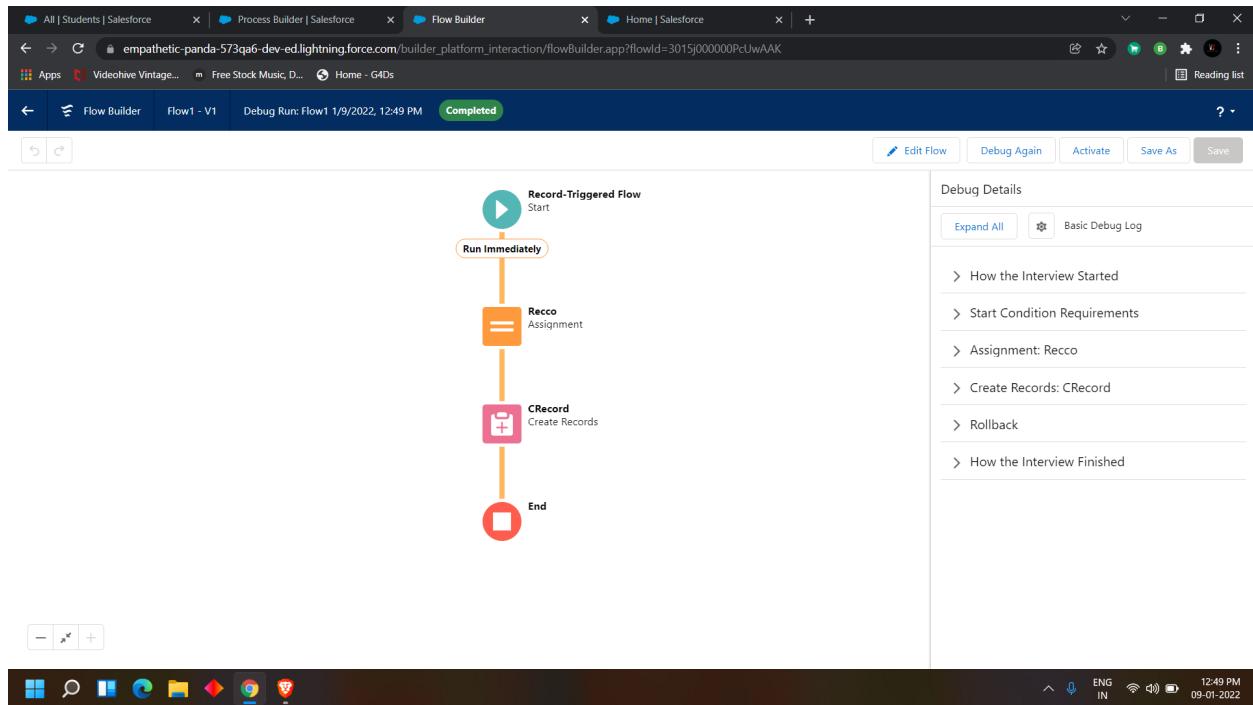
Buttons at the bottom right include "Cancel" and "Done".

Screenshot of the Salesforce Flow Builder interface showing the configuration of a new assignment for the "New Student" resource.

The "Set Variable Values" section contains the following assignments:

Variable	Operator	Value
A_a NewStudent > Address	Equals	A_a \$Record > Address
A_a NewStudent > College Name	Equals	A_a \$Record > College
A_a NewStudent > Student Name	Equals	A_a \$Record > Student Name
A_a NewStudent > Guardian Name	Equals	A_a \$Record > Guardian Name
A_a NewStudent > Phone	Equals	A_a \$Record > Phone
NewStudent > Date Of Birth	Equals	\$Record > Date Of Birth

Buttons at the bottom right include "Cancel" and "Done".





DAY 7:

Topic: Milestone / Activities: intro to Apex Programming

Detailed Description:

What is Apex programming language?

- Apex is strongly typed object-oriented, on-demand programming language. It is compiled, stored, and run entirely on the Force.com platform (multi-tenant environment and is very controlled in its invocations and limits).
- Apex syntax looks mostly like Java and acts like stored procedures.
- Apex allows developers to attach business logic to the record save process
- Apex has built-in support for unit test creation and execution.

As a language apex is Integrated, Easy to use, Data focused, Rigorous, Hosted, Multi-tenant aware, automatically up-gradable, easy to test and versioned.

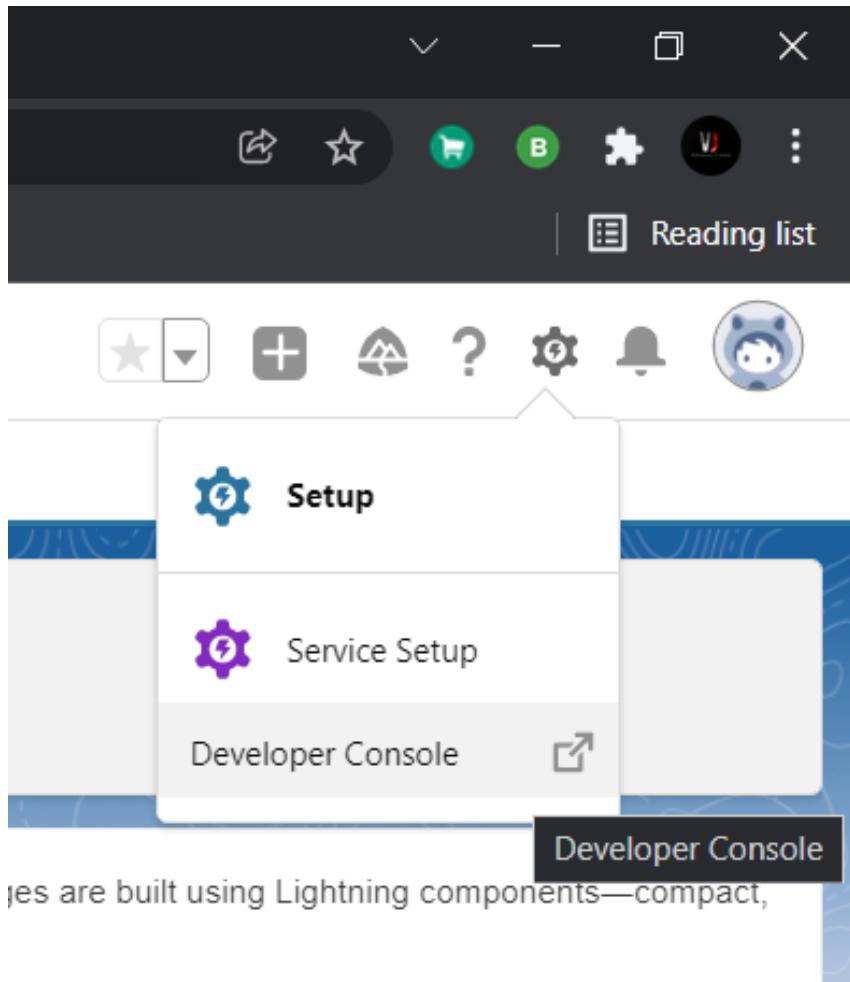
There are only two places you can write code in Salesforce:

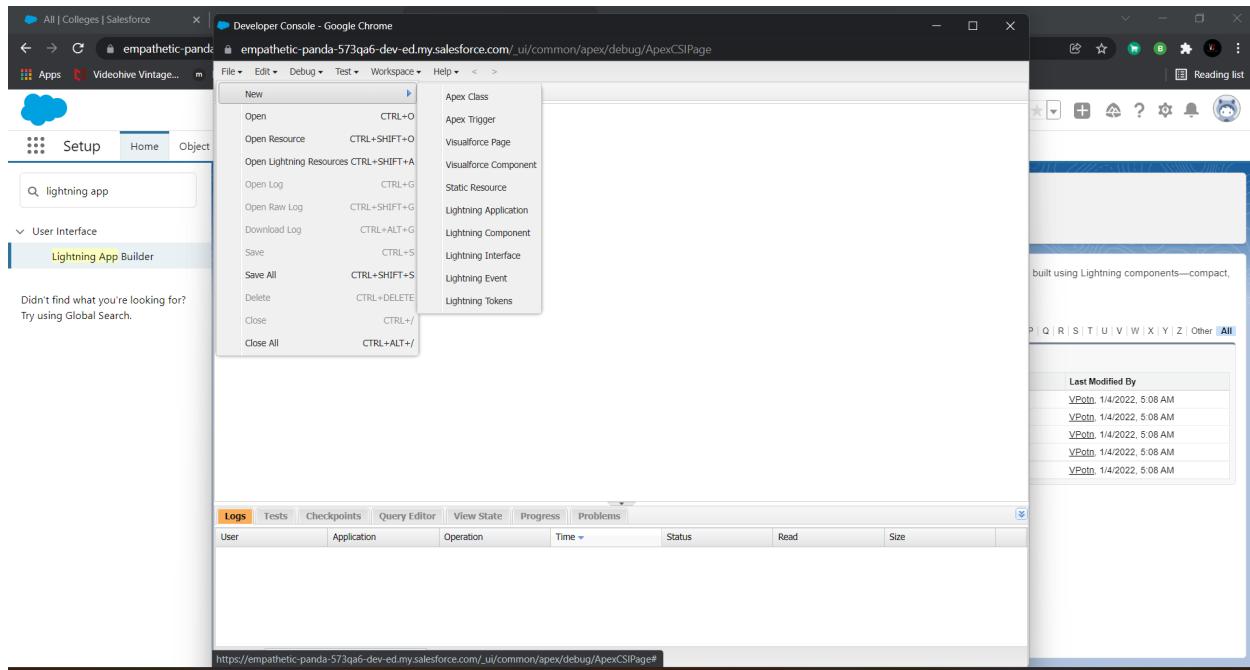
1. Sandbox – most people will want to code here! Sandboxes are developer-friendly copies of your normal org and they're easy to create.
2. Developer Edition – if you don't have an instance of Salesforce to call home, you can practice here.

You can follow along in either version! If you want to eventually deploy into your normal “production” org though, you'll want to use a sandbox.

To open up the trigger coding UI in either version, navigate here:

Setup >> Customize >> Users >> Triggers >> New





What is the apex class and Triggers?

Apex Classes: It is a collection of variables and a library of methods that can be reused.

Apex Trigger: It is a script that executes before or after a specific data manipulation language (DML) event on a particular Salesforce object.

Apex should be used as a solution when:

- You need to apply complex business logic to rows of data being saved by any means.
- You need to create additional web services API functionality for exposing logic either within Salesforce or to external applications.



- You need to call out to external Web service and process the results.
- You need to handle incoming or outgoing emails in ways more complex than the declarative functionality.

Apex triggers execute no matter how the triggering data is being saved.

Apex executes regardless of whether the action originates in the user interface, through AJAX toolkit, or from web services API.

Primitive Type Keyword

Below are the primitive data type keywords used in apex language to declare variables.

1. Blob

This data type represents binary data stored as a single object.

Example

```
Blob b1 = Blob.valueof('idea');
```

2. Boolean

This data type represents Value that can only be assigned true, false, or null.

Example:

```
Boolean isvar = true;
```

3. Date

This data type represents particular day.



Example:

```
Date myDateVar = Date.today();  
Date weekStart = myDateVar.toStartofWeek();
```

4. Datetime

This data type represents particular time and date.

Example:

```
Datetime myDateTimeVar = Datetime.now();  
Datetime newd = myDateTimeVar.addMonths(5);
```

5. Decimal

This data type represents Number that includes a decimal point. Decimal is an arbitrary precision number.

Example:

```
Decimal myVar = 12.4567;  
Decimal divDecVar = myVar.divide(7, 2, System.RoundingMode.UP);  
System.assertEquals(divDecVar, 1.78);
```

6. Double

Represents 64-bit number that includes a decimal point. Minimum value -2^63.

Maximum value of 2^63-1.

Example:

```
Double d=3.14159;
```



7. ID

Represents 18-character Force.com record identifier.

Example:

```
ID id='00300000003T2PGAA0';
```

8. Integer

32-bit number that doesn't include a decimal point. Minimum value

-2,147,483,648 – maximum value of 2,147,483,647

Example:

```
Integer i = 1;
```

9. Long

Represents 64-bit number that doesn't include a decimal point. minimum value of -

263 – maximum value of 263-1.

Example:

```
Long l = 2147483648L;
```

10. String

Represents Set of characters surrounded by single quotes.

Example:

```
String s1 = 'Hello';
```



11. Time

Represents particular time.

Example:

```
Time myTimeVar = Time.newInstance(18, 30, 2, 20);
Integer myMinutes = myTimeVar.
minute();
```

List Type Keywords

Below are the Collection Type keyword used in Apex:

1. List

Ordered collection of typed primitives, sObjects, objects, or collections that are distinguished by their indices.

Example:

```
// Create an empty list of String
List<String> my_list = new List<String>();
My_list.add('first');
String x = my_list.get(0);
// Create list of records from a query
List<Account> accs = [SELECT Id, Name FROM Account LIMIT 1000];
```

2. Map

Collection of key-value pairs where each unique key maps to a single value. Keys can be any primitive data type, while values can be a primitive, sObject,



collection type, or an object.

Example:

```
Map<String, String> mys = new Map<String, String>();  
Map<String, String> mys = new Map<String,  
String>{'a' => 'b', 'c' => 'd'.toUpperCase()};  
Account myAcct = new Account();  
Map<Integer, Account> m = new  
Map<Integer, Account>();  
m.put(1, myAcct);
```

3. Set

Unordered collection that doesn't contain any duplicate elements.

Example:

```
Set<Integer> s = new Set<Integer>();  
s.add(1);  
s.add(1);  
System.assert(s.size() == 1);
```



DAY 8:

Topic: Milestone / Activities: Create a apex batch class

Detailed Description:

1--> Created A Batchapex For Application Form

In developer console we created apex class with class name as ApplicationBatchTest. Apex Batch Class consists three methods called

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

```
Public class ApplicationBatchTest implements Database.Batchable<sObject>{

    //start(), execute(). finish()
    public Integer totalForms = 0; // total no of application form
    public Integer totalConvertedForms = 0; // total no of students
    public Database.QueryLocator start(Database.BatchableContext bc){
        // gathers the data for you
        String applicationQuery = 'select id, Name, Ready_To_Join__c from ApplicationForm__c';
        return Database.getQueryLocator(applicationQuery);
    }
    public void execute(Database.BatchableContext bc, List<ApplicationForm__c> formList){
        // process the data
        for(ApplicationForm__c af : formList){
            totalForms++;
            if(af.Ready_To_Join__c){
                totalConvertedForms++;
            }
        }
    }
    public void finish(Database.BatchableContext bc){
        // emails ,
        Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
        // address, subject, content( data to sent to admins)
        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[] emailAddess = new String[]{'your email address'};  
mail.setToAddresses(emailAddess);  
Messaging.sendEmail(new Messaging.SingleEmailMessage(){ mail } );  
}  
}
```

1-->start()
2-->execute()
3-->finish()

The screenshot shows the Salesforce IDE interface. The top part displays the code for `ApplicationBatchTest.apxc`. The code implements the `Database.Batchable<sObject>` interface. It defines two variables: `totalForms` and `totalConvertedForms`. The `start` method sets up a query to select records from `Application_Form_c`. The `execute` method iterates through the results, incrementing counters for each record found. The bottom part of the screenshot shows a table titled "Logs" with columns: User, Application, Operation, Time, Status, Read, and Size. There are no entries in the table.

```
1 * Public class ApplicationBatchTest implements Database.Batchable<sObject>{  
2  
3  
4     public Integer totalForms = 0; // total no of application form  
5     public Integer totalConvertedForms = 0; // total no of students  
6     public Database.QueryLocator start(Database.BatchableContext bc){  
7         String applicationQuery = 'select id, Name, Ready_To_Join__c from Application_Form__c';  
8         return Database.getQueryLocator(applicationQuery);  
9     }  
10    public void execute(Database.BatchableContext bc, List<Application_Form__c> formList){  
11        for(Application_Form__c af : formList){  
12            totalForms++;  
13            if(af.Ready_To_Join__c){  
14                totalConvertedForms++;  
15            }  
16        }  
17    }  
18 }  
19  
20
```

User	Application	Operation	Time	Status	Read	Size



2--> Created a Scheduler Class in the developer class

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

```
public class applicationschedule implements Schedulable{  
  
    public void execute(SchedulableContext sc){  
        ApplicationBatchTest abt = new ApplicationBatchTest();  
        Database.executeBatch(abt, 400); // 200 to 2000  
  
    }  
}
```

From setup search for apex class and click on schedule jobs and fill the details as per your requirements.

A screenshot of the Salesforce Developer Console. The top navigation bar includes File, Edit, Debug, Test, Workspace, Help, and a Go To button. The tabs at the top show ApplicationBatchTest.apxc and applicationschedule.apxc. The applicationschedule.apxc tab is active. The code editor shows the Apex code for the applicationschedule class. Below the code editor is a logs table with columns for Log ID, User, Application, Operation, Time, Status, Read, and Size. The first log entry is for a user named 'Read' with a status of 'Success'.

Then went to Setup and searched for Apex Classes and click on Schedule Apex



to schedule a Apex Class according to our requirements. After scheduling this scheduled class will be visible in Apex Jobs.

DAY 9:

Topic: Milestone / Activities: overview of LWC and installation of visual studio

Detailed Description:

Lightning Web Components

Lightning Web Components (LWC) is a stack of modern lightweight frameworks built on the latest web standards. It is a DOM (Document Object Model), element created through reusable code and is used to generate a dynamic interface without using JavaScript or building a Library. This feasibility makes it quick and seamless, saving the developers a ton of time and effort on the Web Stack. Let's look at some of its remarkable features:

- Improved performance of the component as most of the code is recognized by the native web browser engine and web stack
- Ability to compose applications using smaller chunks of code since the crucial elements that are required to create a component is part of the native web browser engine and web stack
- Increase in the robustness of the applications built using LWCs as they are inclusive of the said modern web standards.
- Parallel interoperability and feasibility to use both Lightning Web Components and Aura components together in the applications with no visible differentiation to the end-users



- 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)

Before going into this
we need to install

Visual Studio

The screenshot shows the Microsoft Visual Studio landing page. It features three main download cards:

- Visual Studio**: Version 17.0. Description: The best comprehensive IDE for .NET and C++ developers on Windows. Fully packed with a sweet array of tools and features to elevate and enhance every stage of software development. [Download Visual Studio](#)
- Visual Studio for Mac**: Version 8.10. Description: A comprehensive IDE for .NET developers that's native to macOS. Includes top-notch support for web, cloud, and game development —plus ridiculously good tools for making cross-platform mobile apps. [Download Visual Studio for Mac](#)
- Visual Studio Code**: Version 1.62. Description: A standalone source code editor that runs on Windows, macOS, and Linux. The top pick for JavaScript and web developers, with tons of extensions to support just about any programming language. [Download Visual Studio Code](#)

Below the cards, there's a section for "Developer machine OS" with links for Windows, macOS, and Linux. Another section for "Languages, platforms, and workflows" lists .NET, C#, C++, Web, Cloud, Desktop, Python, Java, JavaScript, TypeScript, Node.js, Games, and Mobile (Android/iOS). At the bottom, there's a navigation bar with icons for Data Science, Machine Learning/AI, and a feedback link.

<https://visualstudio.microsoft.com/>



Salesforce LWS Editor

The screenshot shows the extension page for the "Salesforce LWC Editor" on the Chrome Web Store. The page includes the extension's icon, name, developer information (Offered by: nisarahmad.ajmer@gmail.com), rating (★★★★★ 28), user count (8,000+ users), and an "Add to Brave" button. Below this, there are tabs for Overview, Privacy practices, Reviews, Support, and Related. A large preview window displays the LWC Editor interface, showing a component tree on the left and a code editor on the right with some template code. The status bar at the bottom shows standard Windows icons and the date/time (17-01-2022, 09:02 PM).

WhatsApp Student Dashboard LWC | SmartBridge SI-7769-1641358131 Telegram Web

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

Videohive Vintage... Home - G4Ds

chrome web store dcmesklm@gmail.com

Home > Extensions > Salesforce LWC Editor

Salesforce LWC Editor
Offered by: nisarahmad.ajmer@gmail.com
★★★★★ 28 | Developer Tools | 8,000+ users

Add to Brave

Overview Privacy practices Reviews Support Related

Salesforce LWC Editor (Detail)

Search Component

- Basic
- Basic2
- Basic3
- Basic4
- Basic5
- Basic6
- Basic7
- Basic8
- Basic9
- Basic10
- Basic11
- Basic12
- Basic13
- Basic14
- Basic15
- Basic16
- Basic17
- Basic18
- Basic19
- Basic20
- Basic21
- Basic22
- Basic23
- Basic24
- Basic25
- Basic26
- Basic27
- Basic28
- Basic29
- Basic30
- Basic31
- Basic32
- Basic33
- Basic34
- Basic35
- Basic36
- Basic37
- Basic38
- Basic39
- Basic40
- Basic41
- Basic42
- Basic43
- Basic44
- Basic45
- Basic46
- Basic47
- Basic48
- Basic49
- Basic50
- Basic51
- Basic52
- Basic53
- Basic54
- Basic55
- Basic56
- Basic57
- Basic58
- Basic59
- Basic60
- Basic61
- Basic62
- Basic63
- Basic64
- Basic65
- Basic66
- Basic67
- Basic68
- Basic69
- Basic70
- Basic71
- Basic72
- Basic73
- Basic74
- Basic75
- Basic76
- Basic77
- Basic78
- Basic79
- Basic80
- Basic81
- Basic82
- Basic83
- Basic84
- Basic85
- Basic86
- Basic87
- Basic88
- Basic89
- Basic90
- Basic91
- Basic92
- Basic93
- Basic94
- Basic95
- Basic96
- Basic97
- Basic98
- Basic99
- Basic100
- Basic101
- Basic102
- Basic103
- Basic104
- Basic105
- Basic106
- Basic107
- Basic108
- Basic109
- Basic110
- Basic111
- Basic112
- Basic113
- Basic114
- Basic115
- Basic116
- Basic117
- Basic118
- Basic119
- Basic120
- Basic121
- Basic122
- Basic123
- Basic124
- Basic125
- Basic126
- Basic127
- Basic128
- Basic129
- Basic130
- Basic131
- Basic132
- Basic133
- Basic134
- Basic135
- Basic136
- Basic137
- Basic138
- Basic139
- Basic140
- Basic141
- Basic142
- Basic143
- Basic144
- Basic145
- Basic146
- Basic147
- Basic148
- Basic149
- Basic150
- Basic151
- Basic152
- Basic153
- Basic154
- Basic155
- Basic156
- Basic157
- Basic158
- Basic159
- Basic160
- Basic161
- Basic162
- Basic163
- Basic164
- Basic165
- Basic166
- Basic167
- Basic168
- Basic169
- Basic170
- Basic171
- Basic172
- Basic173
- Basic174
- Basic175
- Basic176
- Basic177
- Basic178
- Basic179
- Basic180
- Basic181
- Basic182
- Basic183
- Basic184
- Basic185
- Basic186
- Basic187
- Basic188
- Basic189
- Basic190
- Basic191
- Basic192
- Basic193
- Basic194
- Basic195
- Basic196
- Basic197
- Basic198
- Basic199
- Basic200
- Basic201
- Basic202
- Basic203
- Basic204
- Basic205
- Basic206
- Basic207
- Basic208
- Basic209
- Basic210
- Basic211
- Basic212
- Basic213
- Basic214
- Basic215
- Basic216
- Basic217
- Basic218
- Basic219
- Basic220
- Basic221
- Basic222
- Basic223
- Basic224
- Basic225
- Basic226
- Basic227
- Basic228
- Basic229
- Basic230
- Basic231
- Basic232
- Basic233
- Basic234
- Basic235
- Basic236
- Basic237
- Basic238
- Basic239
- Basic240
- Basic241
- Basic242
- Basic243
- Basic244
- Basic245
- Basic246
- Basic247
- Basic248
- Basic249
- Basic250
- Basic251
- Basic252
- Basic253
- Basic254
- Basic255
- Basic256
- Basic257
- Basic258
- Basic259
- Basic260
- Basic261
- Basic262
- Basic263
- Basic264
- Basic265
- Basic266
- Basic267
- Basic268
- Basic269
- Basic270
- Basic271
- Basic272
- Basic273
- Basic274
- Basic275
- Basic276
- Basic277
- Basic278
- Basic279
- Basic280
- Basic281
- Basic282
- Basic283
- Basic284
- Basic285
- Basic286
- Basic287
- Basic288
- Basic289
- Basic290
- Basic291
- Basic292
- Basic293
- Basic294
- Basic295
- Basic296
- Basic297
- Basic298
- Basic299
- Basic300
- Basic301
- Basic302
- Basic303
- Basic304
- Basic305
- Basic306
- Basic307
- Basic308
- Basic309
- Basic310
- Basic311
- Basic312
- Basic313
- Basic314
- Basic315
- Basic316
- Basic317
- Basic318
- Basic319
- Basic320
- Basic321
- Basic322
- Basic323
- Basic324
- Basic325
- Basic326
- Basic327
- Basic328
- Basic329
- Basic330
- Basic331
- Basic332
- Basic333
- Basic334
- Basic335
- Basic336
- Basic337
- Basic338
- Basic339
- Basic340
- Basic341
- Basic342
- Basic343
- Basic344
- Basic345
- Basic346
- Basic347
- Basic348
- Basic349
- Basic350
- Basic351
- Basic352
- Basic353
- Basic354
- Basic355
- Basic356
- Basic357
- Basic358
- Basic359
- Basic360
- Basic361
- Basic362
- Basic363
- Basic364
- Basic365
- Basic366
- Basic367
- Basic368
- Basic369
- Basic370
- Basic371
- Basic372
- Basic373
- Basic374
- Basic375
- Basic376
- Basic377
- Basic378
- Basic379
- Basic380
- Basic381
- Basic382
- Basic383
- Basic384
- Basic385
- Basic386
- Basic387
- Basic388
- Basic389
- Basic390
- Basic391
- Basic392
- Basic393
- Basic394
- Basic395
- Basic396
- Basic397
- Basic398
- Basic399
- Basic400
- Basic401
- Basic402
- Basic403
- Basic404
- Basic405
- Basic406
- Basic407
- Basic408
- Basic409
- Basic410
- Basic411
- Basic412
- Basic413
- Basic414
- Basic415
- Basic416
- Basic417
- Basic418
- Basic419
- Basic420
- Basic421
- Basic422
- Basic423
- Basic424
- Basic425
- Basic426
- Basic427
- Basic428
- Basic429
- Basic430
- Basic431
- Basic432
- Basic433
- Basic434
- Basic435
- Basic436
- Basic437
- Basic438
- Basic439
- Basic440
- Basic441
- Basic442
- Basic443
- Basic444
- Basic445
- Basic446
- Basic447
- Basic448
- Basic449
- Basic450
- Basic451
- Basic452
- Basic453
- Basic454
- Basic455
- Basic456
- Basic457
- Basic458
- Basic459
- Basic460
- Basic461
- Basic462
- Basic463
- Basic464
- Basic465
- Basic466
- Basic467
- Basic468
- Basic469
- Basic470
- Basic471
- Basic472
- Basic473
- Basic474
- Basic475
- Basic476
- Basic477
- Basic478
- Basic479
- Basic480
- Basic481
- Basic482
- Basic483
- Basic484
- Basic485
- Basic486
- Basic487
- Basic488
- Basic489
- Basic490
- Basic491
- Basic492
- Basic493
- Basic494
- Basic495
- Basic496
- Basic497
- Basic498
- Basic499
- Basic500
- Basic501
- Basic502
- Basic503
- Basic504
- Basic505
- Basic506
- Basic507
- Basic508
- Basic509
- Basic510
- Basic511
- Basic512
- Basic513
- Basic514
- Basic515
- Basic516
- Basic517
- Basic518
- Basic519
- Basic520
- Basic521
- Basic522
- Basic523
- Basic524
- Basic525
- Basic526
- Basic527
- Basic528
- Basic529
- Basic530
- Basic531
- Basic532
- Basic533
- Basic534
- Basic535
- Basic536
- Basic537
- Basic538
- Basic539
- Basic540
- Basic541
- Basic542
- Basic543
- Basic544
- Basic545
- Basic546
- Basic547
- Basic548
- Basic549
- Basic550
- Basic551
- Basic552
- Basic553
- Basic554
- Basic555
- Basic556
- Basic557
- Basic558
- Basic559
- Basic560
- Basic561
- Basic562
- Basic563
- Basic564
- Basic565
- Basic566
- Basic567
- Basic568
- Basic569
- Basic570
- Basic571
- Basic572
- Basic573
- Basic574
- Basic575
- Basic576
- Basic577
- Basic578
- Basic579
- Basic580
- Basic581
- Basic582
- Basic583
- Basic584
- Basic585
- Basic586
- Basic587
- Basic588
- Basic589
- Basic590
- Basic591
- Basic592
- Basic593
- Basic594
- Basic595
- Basic596
- Basic597
- Basic598
- Basic599
- Basic600
- Basic601
- Basic602
- Basic603
- Basic604
- Basic605
- Basic606
- Basic607
- Basic608
- Basic609
- Basic610
- Basic611
- Basic612
- Basic613
- Basic614
- Basic615
- Basic616
- Basic617
- Basic618
- Basic619
- Basic620
- Basic621
- Basic622
- Basic623
- Basic624
- Basic625
- Basic626
- Basic627
- Basic628
- Basic629
- Basic630
- Basic631
- Basic632
- Basic633
- Basic634
- Basic635
- Basic636
- Basic637
- Basic638
- Basic639
- Basic640
- Basic641
- Basic642
- Basic643
- Basic644
- Basic645
- Basic646
- Basic647
- Basic648
- Basic649
- Basic650
- Basic651
- Basic652
- Basic653
- Basic654
- Basic655
- Basic656
- Basic657
- Basic658
- Basic659
- Basic660
- Basic661
- Basic662
- Basic663
- Basic664
- Basic665
- Basic666
- Basic667
- Basic668
- Basic669
- Basic670
- Basic671
- Basic672
- Basic673
- Basic674
- Basic675
- Basic676
- Basic677
- Basic678
- Basic679
- Basic680
- Basic681
- Basic682
- Basic683
- Basic684
- Basic685
- Basic686
- Basic687
- Basic688
- Basic689
- Basic690
- Basic691
- Basic692
- Basic693
- Basic694
- Basic695
- Basic696
- Basic697
- Basic698
- Basic699
- Basic700
- Basic701
- Basic702
- Basic703
- Basic704
- Basic705
- Basic706
- Basic707
- Basic708
- Basic709
- Basic710
- Basic711
- Basic712
- Basic713
- Basic714
- Basic715
- Basic716
- Basic717
- Basic718
- Basic719
- Basic720
- Basic721
- Basic722
- Basic723
- Basic724
- Basic725
- Basic726
- Basic727
- Basic728
- Basic729
- Basic730
- Basic731
- Basic732
- Basic733
- Basic734
- Basic735
- Basic736
- Basic737
- Basic738
- Basic739
- Basic740
- Basic741
- Basic742
- Basic743
- Basic744
- Basic745
- Basic746
- Basic747
- Basic748
- Basic749
- Basic750
- Basic751
- Basic752
- Basic753
- Basic754
- Basic755
- Basic756
- Basic757
- Basic758
- Basic759
- Basic760
- Basic761
- Basic762
- Basic763
- Basic764
- Basic765
- Basic766
- Basic767
- Basic768
- Basic769
- Basic770
- Basic771
- Basic772
- Basic773
- Basic774
- Basic775
- Basic776
- Basic777
- Basic778
- Basic779
- Basic780
- Basic781
- Basic782
- Basic783
- Basic784
- Basic785
- Basic786
- Basic787
- Basic788
- Basic789
- Basic790
- Basic791
- Basic792
- Basic793
- Basic794
- Basic795
- Basic796
- Basic797
- Basic798
- Basic799
- Basic800
- Basic801
- Basic802
- Basic803
- Basic804
- Basic805
- Basic806
- Basic807
- Basic808
- Basic809
- Basic810
- Basic811
- Basic812
- Basic813
- Basic814
- Basic815
- Basic816
- Basic817
- Basic818
- Basic819
- Basic820
- Basic821
- Basic822
- Basic823
- Basic824
- Basic825
- Basic826
- Basic827
- Basic828
- Basic829
- Basic830
- Basic831
- Basic832
- Basic833
- Basic834
- Basic835
- Basic836
- Basic837
- Basic838
- Basic839
- Basic840
- Basic841
- Basic842
- Basic843
- Basic844
- Basic845
- Basic846
- Basic847
- Basic848
- Basic849
- Basic850
- Basic851
- Basic852
- Basic853
- Basic854
- Basic855
- Basic856
- Basic857
- Basic858
- Basic859
- Basic860
- Basic861
- Basic862
- Basic863
- Basic864
- Basic865
- Basic866
- Basic867
- Basic868
- Basic869
- Basic870
- Basic871
- Basic872
- Basic873
- Basic874
- Basic875
- Basic876
- Basic877
- Basic878
- Basic879
- Basic880
- Basic881
- Basic882
- Basic883
- Basic884
- Basic885
- Basic886
- Basic887
- Basic888
- Basic889
- Basic890
- Basic891
- Basic892
- Basic893
- Basic894
- Basic895
- Basic896
- Basic897
- Basic898
- Basic899
- Basic900
- Basic901
- Basic902
- Basic903
- Basic904
- Basic905
- Basic906
- Basic907
- Basic908
- Basic909
- Basic910
- Basic911
- Basic912
- Basic913
- Basic914
- Basic915
- Basic916
- Basic917
- Basic918
- Basic919
- Basic920
- Basic921
- Basic922
- Basic923
- Basic924
- Basic925
- Basic926
- Basic927
- Basic928
- Basic929
- Basic930
- Basic931
- Basic932
- Basic933
- Basic934
- Basic935
- Basic936
- Basic937
- Basic938
- Basic939
- Basic940
- Basic941
- Basic942
- Basic943
- Basic944
- Basic945
- Basic946
- Basic947
- Basic948
- Basic949
- Basic950
- Basic951
- Basic952
- Basic953
- Basic954
- Basic955
- Basic956
- Basic957
- Basic958
- Basic959
- Basic960
- Basic961
- Basic962
- Basic963
- Basic964
- Basic965
- Basic966
- Basic967
- Basic968
- Basic969
- Basic970
- Basic971
- Basic972
- Basic973
- Basic974
- Basic975
- Basic976
- Basic977
- Basic978
- Basic979
- Basic980
- Basic981
- Basic982
- Basic983
- Basic984
- Basic985
- Basic986
- Basic987
- Basic988
- Basic989
- Basic990
- Basic991
- Basic992
- Basic993
- Basic994
- Basic995
- Basic996
- Basic997
- Basic998
- Basic999
- Basic1000

<https://chrome.google.com/webstore/detail/salesforce-lwc-editor/ehkpneicmpbdejpoancidgkejlkahjgo>



Install CLI

A screenshot of a web browser window displaying the Salesforce developer documentation for installing the CLI on Windows. The URL in the address bar is https://developer.salesforce.com/docs/atlas.en-us.sfdx_setup.meta/sfdx_setup/sfdx_setup_install_cli.htm#sfdx_setup_install_cli_windows.

The page is titled "Install the CLI on Windows". It provides instructions for installing the CLI on Windows using an .exe file. A "Warning" box states: "Salesforce CLI works best within the native Windows command prompt (cmd.exe) and the Microsoft Powershell. We don't recommend using Salesforce CLI with a Linux terminal emulator, such as Windows 10 Subsystem for Linux, cygwin, or MinGW, because support for bugs is limited." Below this, a "See Also" section lists "Verify Your Installation" and "Disable Automatic Update of the CLI". Another section, "Install the CLI with a TAR File", explains that Salesforce CLI distributes TAR files for all supported operating systems, with Linux being the only exception where it uses .tar.gz or .tar.xz files. A table provides unversioned URLs for the TAR file for various operating systems, including manifest URLs for each file.

https://developer.salesforce.com/docs/atlas.en-us.sfdx_setup.meta/sfdx_setup/sfdx_setup_install_cli.htm#sfdx_setup_install_cli_windows



DAY 10:

Topic: Milestone / Activities: create college data table components(APEX CLASS,HTML FILE)

Detailed Description:

Create College DataTable Component(APEX CLASS)

Here we are going to create the college data table, for this we need to create an apex class, from which we are going to retrieve the data and Html, javascript files for UI .

Create an Apex Class With The Required Functionality

```
public class GetapplicationDetails
{
    @AuraEnabled(cacheable=true)
    public static List<ApplicationForm__c> getapplicationvalues(id Collegelid)
    {
        List<ApplicationForm__c> formlist = [SELECT ID, College_Fees__c, Name,
Date_of_Birth__c, Email__c, Hostel_Fees__c, Student_Name__c FROM
ApplicationForm__c WHERE College__c =:Collegelid];
        return formList;
    }
}
```



{}

Salesforce LWC Editor (Beta)

Search Apex or Component...

Apex > getApplicationDetails

```
1 public class getApplicationDetails {
2     @AuraEnabled(cacheable=true)
3     public static List<Application_Form__c> getapplicationvalues(id CollegeId)
4     {
5
6         List<Application_Form__c> formlist = [SELECT ID, College_Fees__c,Application_Form_ID__c,Name,
7 Date_of_Birth__c,Email__c,Hostel_Fees__c, Student_Name__c,Phone__c FROM
8 Application_Form__c WHERE College__c =:CollegeId];
9
10    return formList;
11 }
12
13 }
```

Result Panel

File	Deployment Time
1 collegeDataTable.html successfully deployed source	1/18/2022 7:03:18 PM
2 collegeDataTable.js successfully deployed source	1/18/2022 7:03:18 PM
3 collegeDataTable.js-meta.xml successfully deployed source	1/18/2022 7:03:18 PM

MVGR

Use LWC Standard Deployment?

Create College DataTable Component (HTML FILE)

Create the component with the following files.

HTML:

<template>

<h1> College and Application form list table </h1>



```
<template if:true={recordList}>
<lightning-data-table
    key-field="id"
    data={recordList}
    show-row-number-column
    hide-checkbox-column
    columns={columnsList}
>
</lightning-data-table>
</template>

<template if:true={error}>
    {error}
</template>
</template>
```

A screenshot of the Salesforce LWC Editor (Beta). The interface includes a sidebar with project navigation, a central code editor window, and a bottom toolbar. The code editor displays the template component code provided above. The file navigation pane on the left shows the project structure with files like collegeDataTable.html, getApplicationDetails.cls, collegeDataTable.js, and collegeDataTable.js-meta.xml.



DAY 11:

Topic: Milestone / Activities: create college data table components(JAVA SCRIPT FILE,XML FILE)

Detailed Description:

Create College DataTable Component(JAVA SCRIPT FILE)

After the creation of the Html file create the following js file

```
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, {Collegeld : '$recordId'})
wiredCollegeData({data, error}){
```



```
if(data){  
    this.recordList = data;  
}  
  
else if(error){  
    this.error = error;  
    this.recordList = undefined;  
}  
  
}  
}
```

Salesforce LWC Editor (Beta)

collegeDataTable > collegeOutatable.js

Search Apex or Component...

APEX CLASSES

- ApplicationBatchTest
- applicationschedule
- getApplicationDetails

LIGHTNING WEB COMPONENTS

collegeDataTable

- collegeDataTable.html
- collegeDataTable.js
- collegeDataTable.css
- collegeDataTable.js-meta.xml

```
1 import { LightningElement, api, wire } from 'lwc';  
2 import getApplicationValues from '@salesforce/apex/getApplicationDetails.getApplicationValues';  
3 export default class CollegeDataTable extends LightningElement {  
4     columnsList = [  
5         {label : 'Application Form' , fieldName : 'Application_Form_ID__c', type:'text' },  
6         {label : 'Application Form Name' , fieldName : 'Name', type:'text' },  
7         {label : 'College Fees' , fieldName : 'College_Fees__c', type:'currency' },  
8         {label : 'Date of Birth' , fieldName : 'date_of_birth__c', type:'date' },  
9         {label : 'Email' , fieldName : 'Email__c', type:'email' },  
10        {label : 'Hostel Fees' , fieldName : 'Hostel_Fees__c', type:'Currency' },  
11        {label : 'studentName' , fieldName : 'Student_Name__c', type:'text' },  
12        {label : 'Phone' , fieldName : 'Phone__c', type:Phone }  
13    ];  
14  
15    @api recordId;  
16    recordList;  
17    error;  
18  
19    @wire(getApplicationValues, {CollegeId : '$recordId'})  
20    wiredCollegeData({data, error}){  
21        if(data){  
22            this.recordList = data;  
23        }  
24        else if(error){  
25            this.error = error;  
26            this.recordList = undefined;  
27        }  
28    }  
29 }
```

MVGR

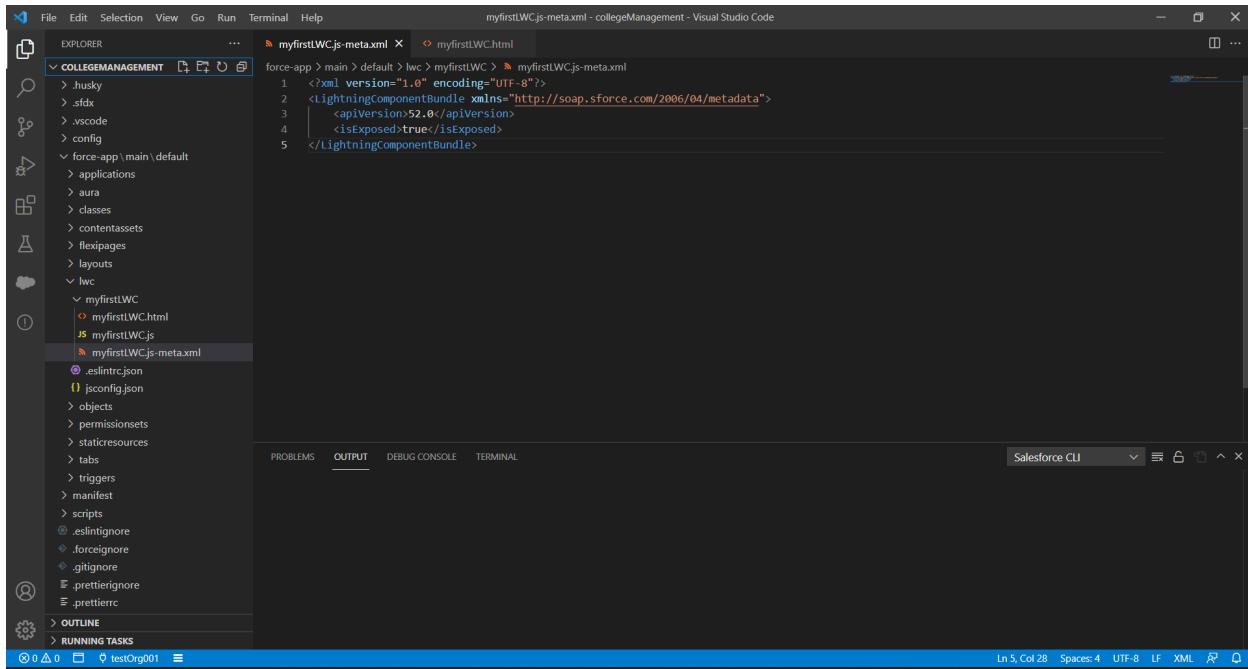
Use LWC Standard Deployment? [Twitter](#) [LinkedIn](#) [GitHub](#)



Create College DataTable Component(META FILE)

Change the Meta File as follows:

```
<?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>
```





DAY 12:

Topic: Milestone / Activities: checking all the data in the college management application

Detailed Description:

As we have done all the tasks now we need to check in the college management application app which we have created

go to the college management application from the app launcher

now-->

click edit page then we have to create another tab for data table

And then drag the college data table in that tab

now you can see the data table in the

A screenshot of the Salesforce Lightning App Builder interface. The top navigation bar shows 'Lightning App Builder', 'Pages', 'College Record Page', 'Help'. Below the navigation is a toolbar with icons for back, forward, search, and refresh, along with dropdowns for 'Desktop' and 'Shrink To View'. On the left, a sidebar displays 'Components' and 'Fields' sections, with a search bar and a 'Standard (0)' section. Under 'Custom (1)', there is a component for 'collegeDataTable'. Under 'Custom - Managed (0)', there is no content. The main workspace contains a component titled 'College MIT-BLR' with tabs for 'Related', 'Details', and 'Datatable'. The 'Datatable' tab shows a table with columns: Application..., College..., date of..., Email, Hostel F..., Student..., Phone. The right side of the workspace shows an 'Activity' section with tabs for 'New Event', 'New Task', 'Log a Call', and 'Email'. It includes a 'Set up an event...' button and a 'Filters' section with a 'Refresh' button. Below the activity section is a 'Upcoming & Overage' section with a note: 'No new tasks. To get things moving, add a task or set up a meeting.' At the bottom of the workspace is a footer with 'Get more on the AppExchange' and a 'Page Analysis' button labeled 'Moderate'. The status bar at the bottom right shows a progress bar and a close button.



A screenshot of the Smart Internz College Management application. At the top, there's a navigation bar with icons for Apps, Cloud, College Management, and various dropdown menus like Colleges, Application Forms, Students, and Subjects. Below the header, a banner for "MIT-MUM" is displayed. The main content area has two main sections: "Datatable" on the left showing a list of application forms with columns for Application ID, Name, Fee, Date of Birth, Email, Hostel Fees, Student ID, and Phone; and "Activity" on the right showing a summary of upcoming events, tasks, and calls. The Activity section includes a search bar, filters, and a note about no past activity.

here we can see the details of the applicant with all his personal details which are given when registered