

**PROJECT TITLE:** COLLEGE MANAGEMENT APPLICATION

**Day1**

**Topic: SalesForce Introduction and Creating a Salesforce Developer Org**

**Milestone / Activities:**

* What is salesforce and serveice nature of the salesforce
* Account creation of the salesforce
* Creation of the custom objects

**Detailed Description:**

**Salesforce** is a cloud-based Customer Relationship Management (CRM) software for managing customer relationships and integration with other systems. This SaaS tool helps to create custom solutions for marketing, sales, services and ecommerce as per business requirements. Salesforce has now expanded from just CRM to offer multiple products.

**Serveice Nature of the Salesforce**

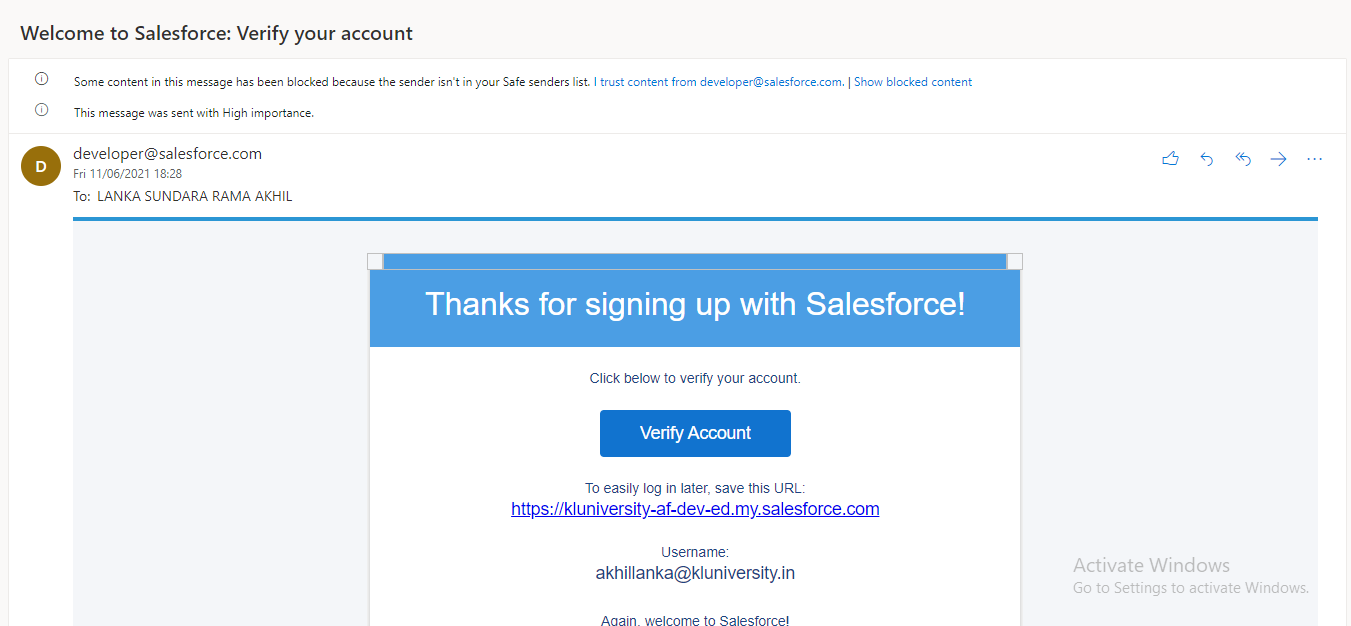
* **Salesforce as SaaS (Software as a Service):**No installations, setup or download required. Just Log in and use software across the Cloud.
* **Salesforce as PaaS (Platform as a service):** In this kind of software service, no separate platform is required. You can use the developer’s code to create and deploy applications.
* **Salesforce as laaS (Infrastructure as a service):**No hardware or software installations needed as your data and applications are stored securely on the Salesforce cloud.

**Creating a developer org in salesforce .**

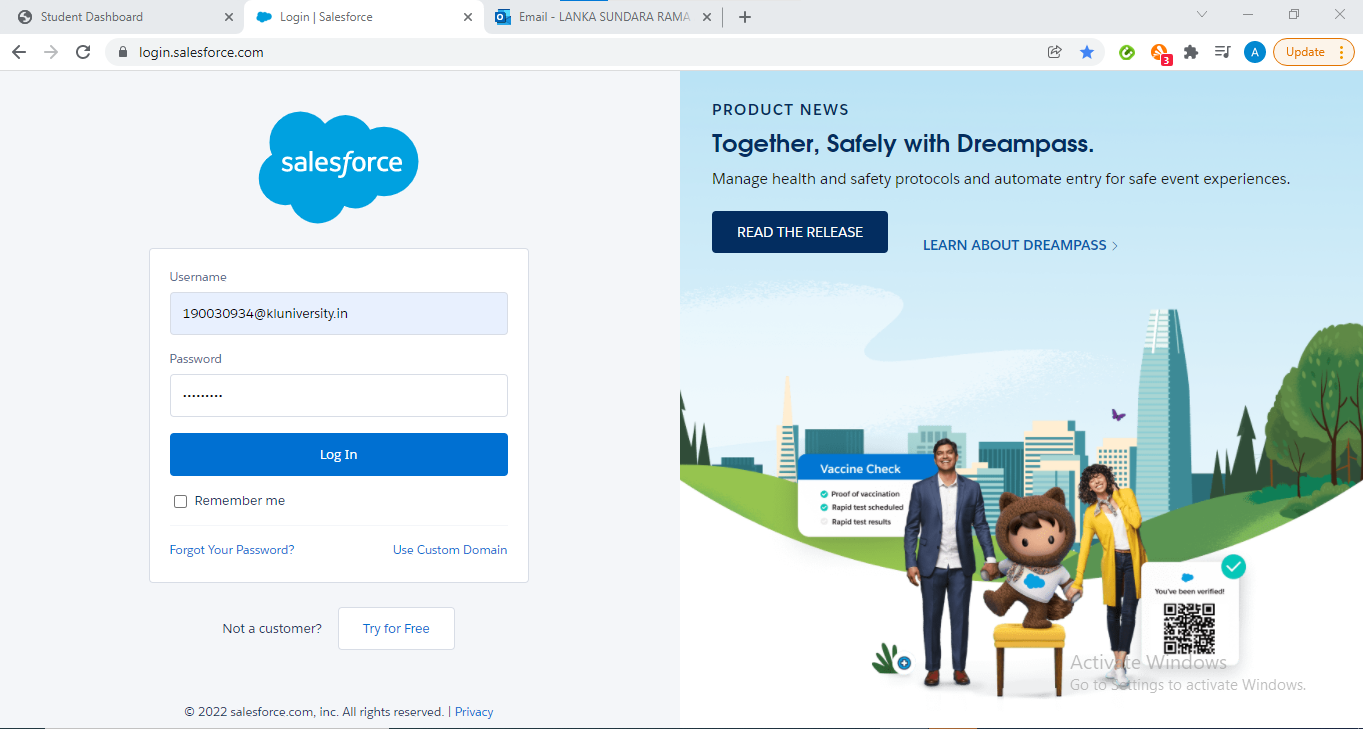
1. Go to [developers.salesforce.com/](https://developer.salesforce.com/)
2. Click on sign up.
3. On the sign up form, enter the following details :
   1. First name & Last name
   2. Email
   3. Role : Developer
   4. Company : College Name
   5. County : India
   6. Postal Code : pin code
   7. Username : should be a combination of your name and company
      1. This need not be an actual email id, you can give anything in the format :
      2. [username@organization.com](mailto:username@organization.com)

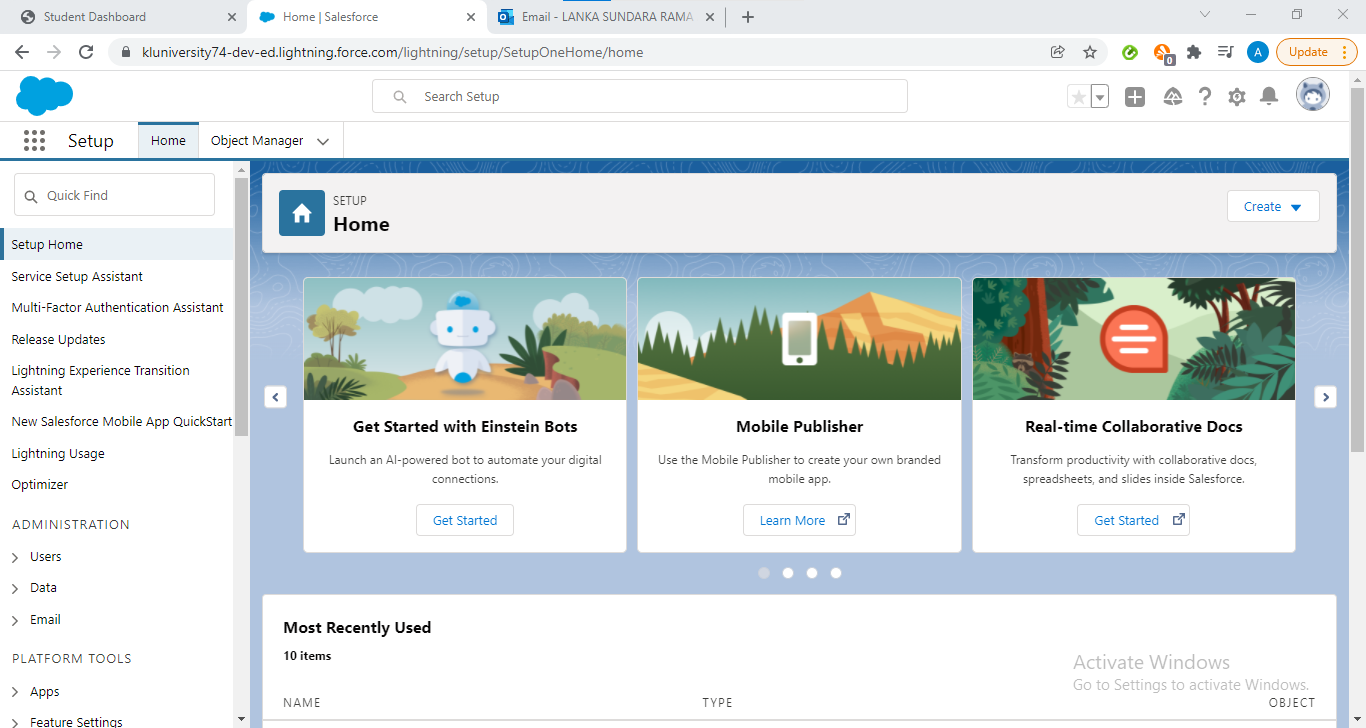
after filling Click on sign up these.

1. After filling the details.In few Minutes you will recieve a mail in order to verify the account.
2. With the help of the link you can verify your account and then you can see your salesforce developer org is created for you.



After clicking Verify Account Your org will get activated and login with your own crendiatls if the crenditals are correct your will be directed to your home page of the salesforce





**Day2**

**Topic : Custom Object Creation:**

1. Custom Objects allow Salesforce to extend functionality past the native CRM suite (Leads, Accounts, Opportunities, etc).
2. Custom Objects are simply tables you create to store records that categorically do not belong in any of the existing tabs or objects.
3. For example Human Resources is a department that does not fit well inside the native Salesforce.com tabs, with maybe the exception of Accounts and Contacts.

**Detailed Description:**

Objects are the way you store your records in Salesforce. They are basically a container for spreadsheet data, account details, customer contacts, location and regional info, tracking status, and more, but are easier on the eyes and more customizable than a database. The idea is that you track everything related to your business in one place

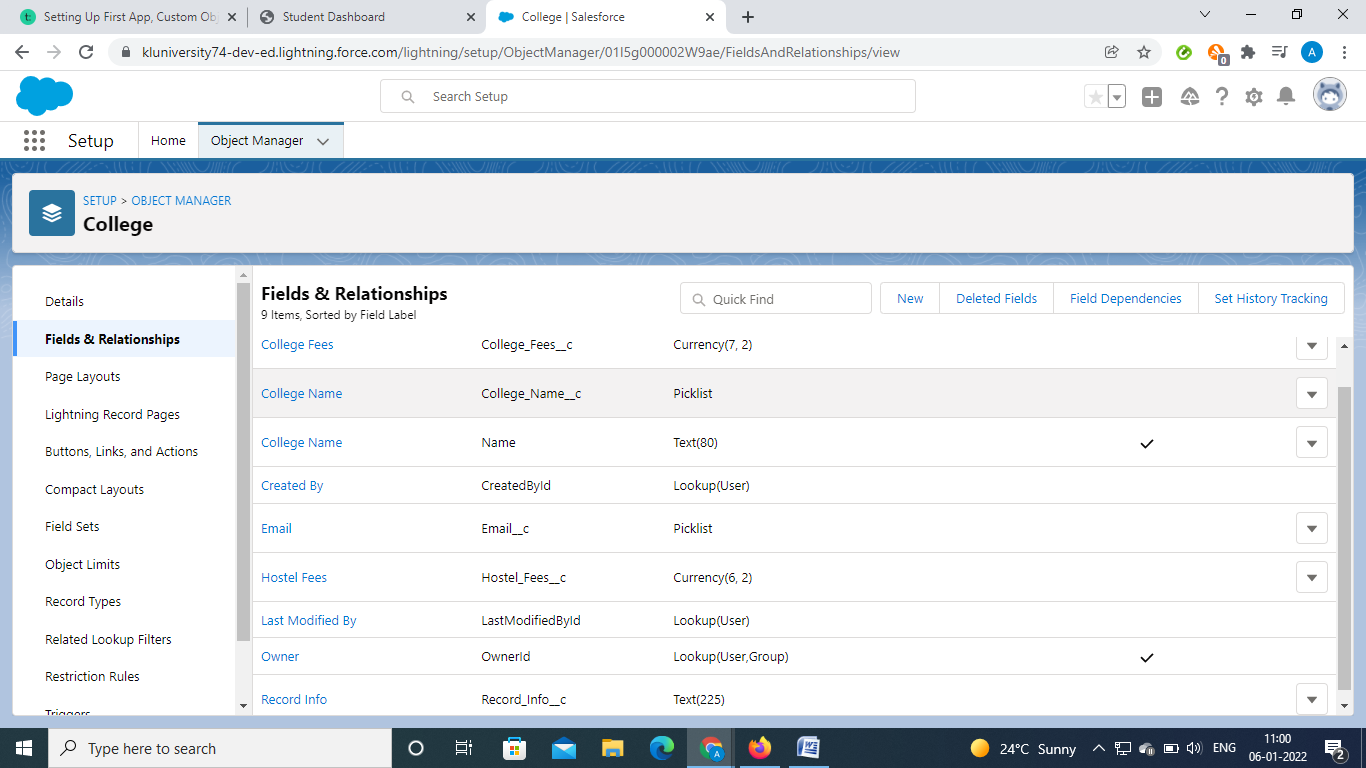
1. Log in to your Salesforce account
2. Click Setup at the upper-right corner.
3. Under the Build section, click Create and select Objects
4. To create a custom object, click New Custom Object.
5. Enter the name of the Custom Object in Label, Plural Label, and Object Name.
6. Select the**Launch New Custom Tab Wizard after saving this custom object** check box and click **Save**
7. To select an icon for the tab, click the **Search** icon and click **Save.**
8. To make the Custom Object available to Profiles, select the appropriate option and click **Next**.
9. Choose the custom apps for which the new Custom tab is required and click **Save**.

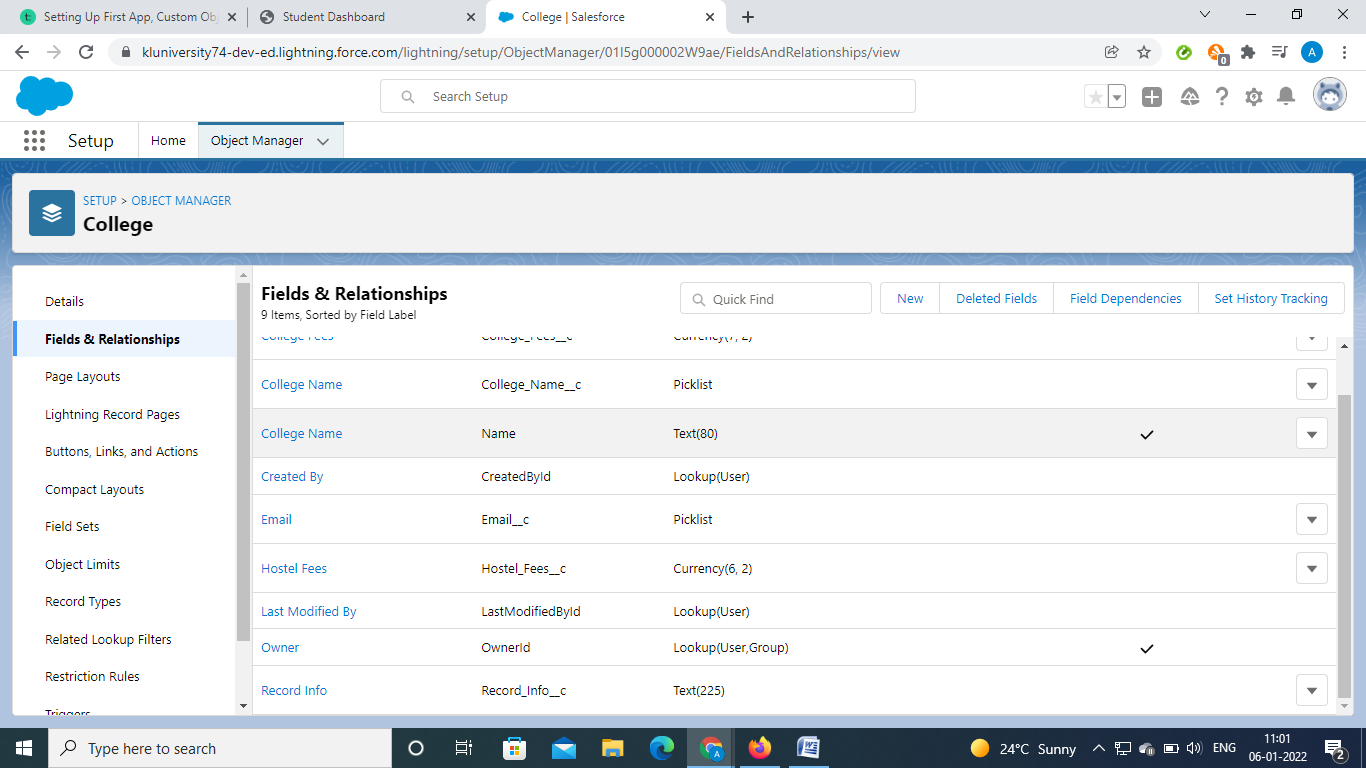
### Create Fields For The Object

1. **Create the following fields on the college object.**

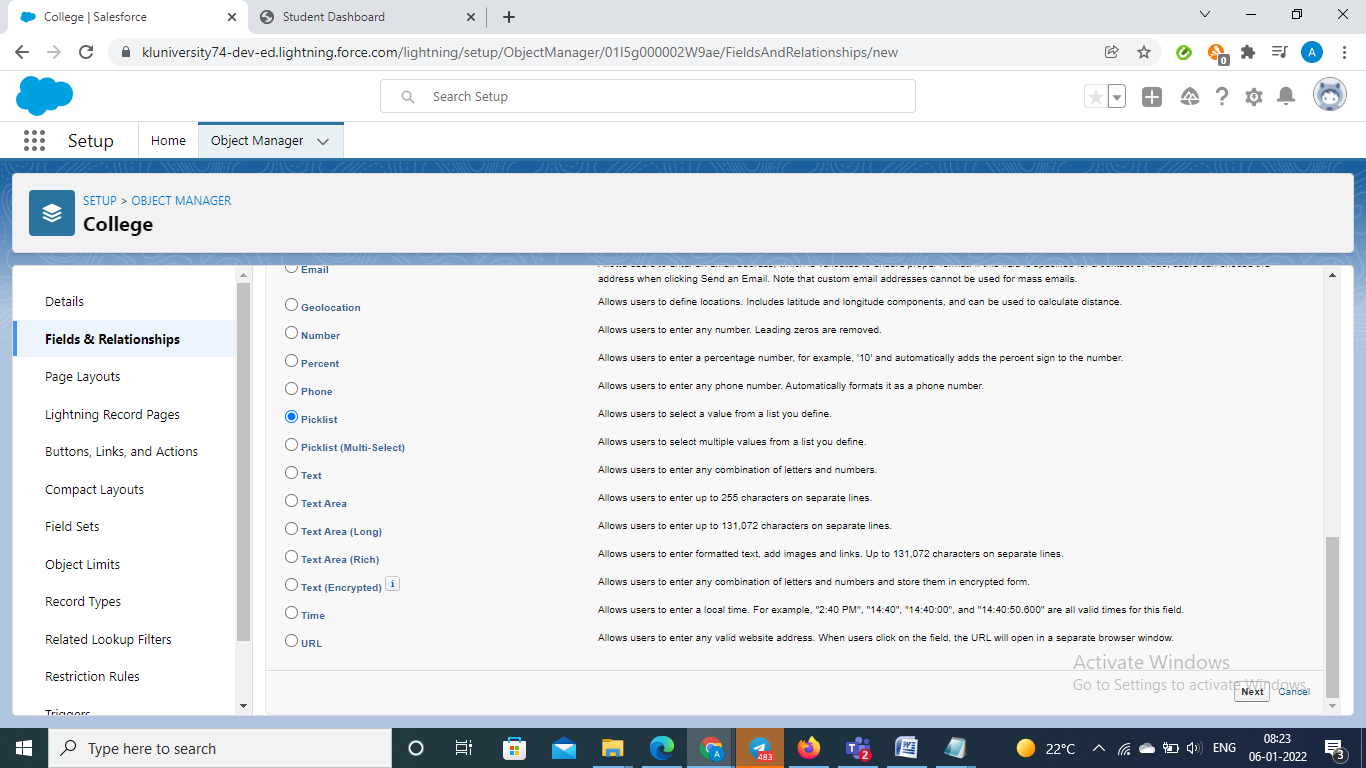
Steps:

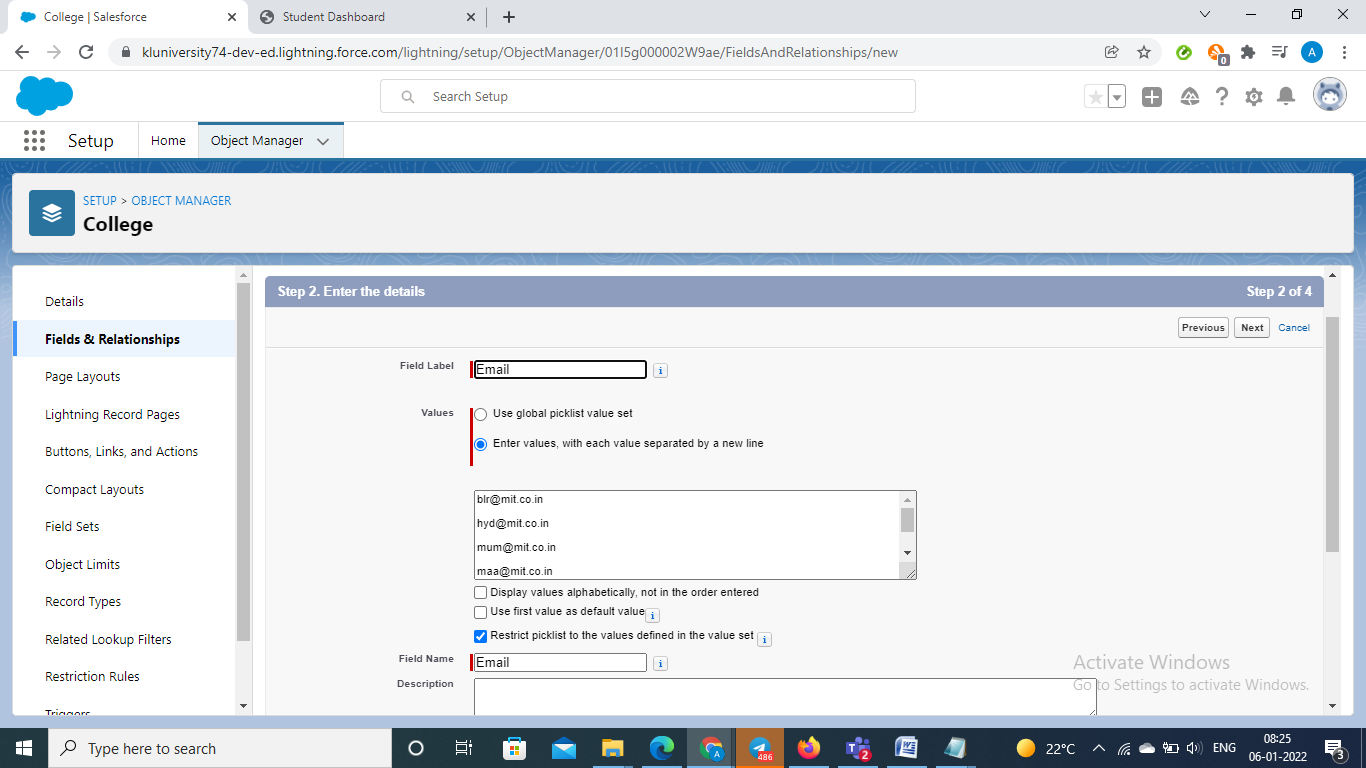
1. From Setup, click the Object Manager tab
2. On The top right side where you can find the search bar on the search bar type college and select the name of the object
3. On the left side select the fields and relationships
4. On the fields and relationships click new
5. Select the data type of the custom fields and then select next
6. Enter the field label and field name and then select next
7. On the Establish field level security leave as it is and then click net
8. Leave the page layouts as it is and then click save and new

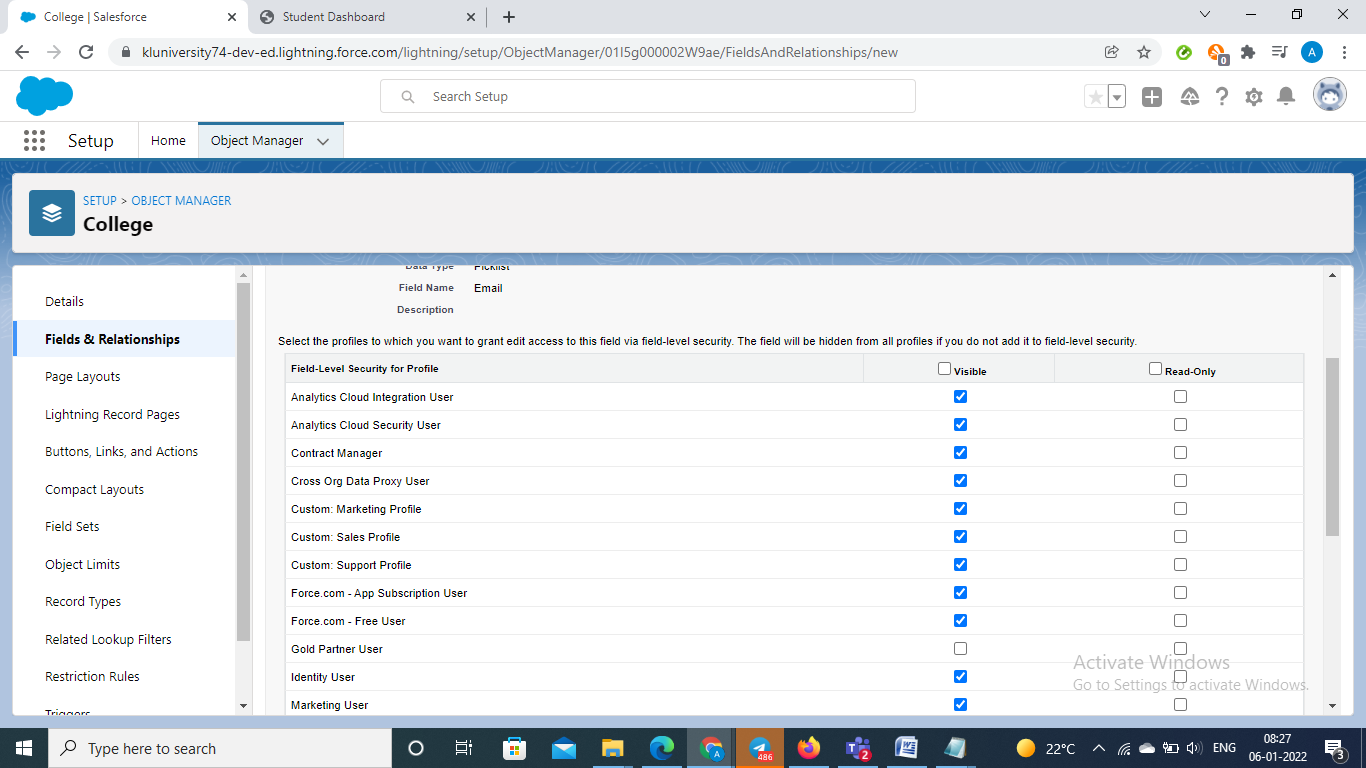


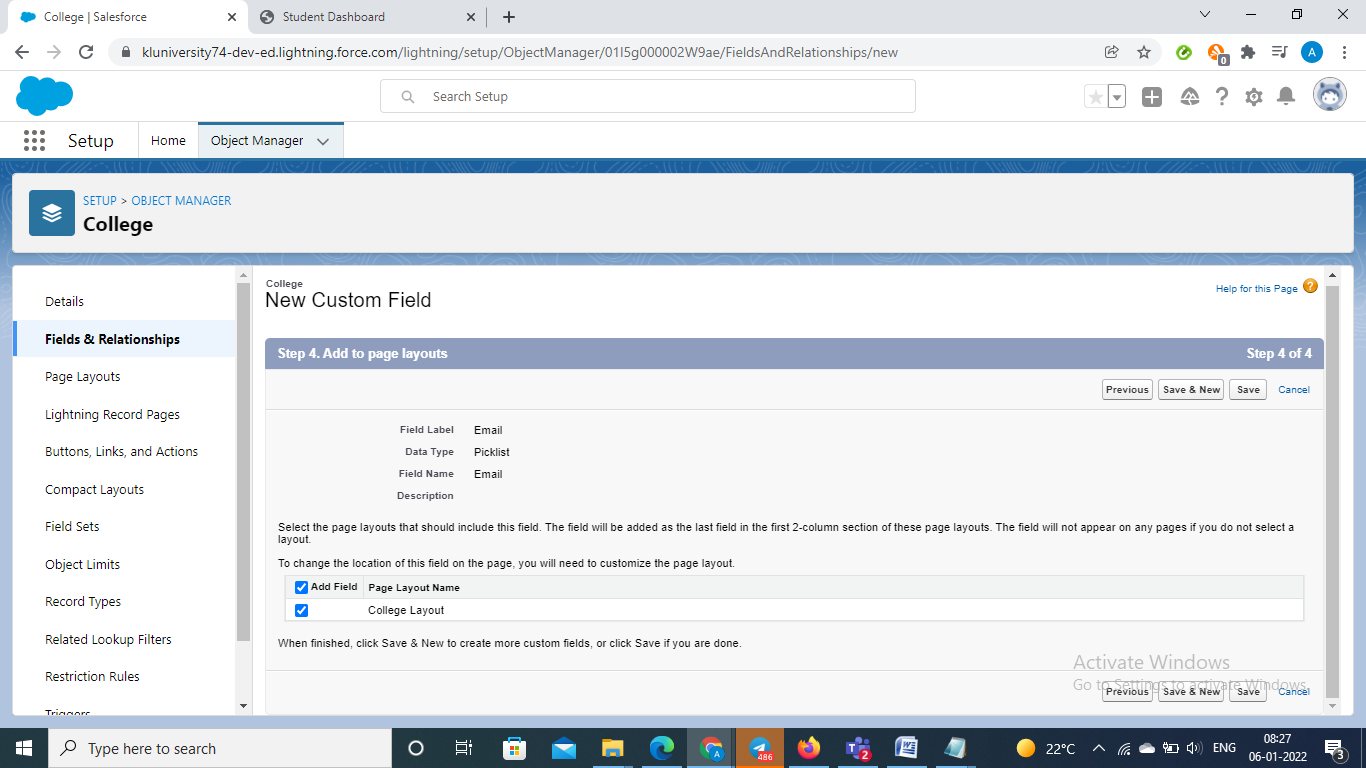


**Step By Step For The Creation of the Fields inside the standard object**









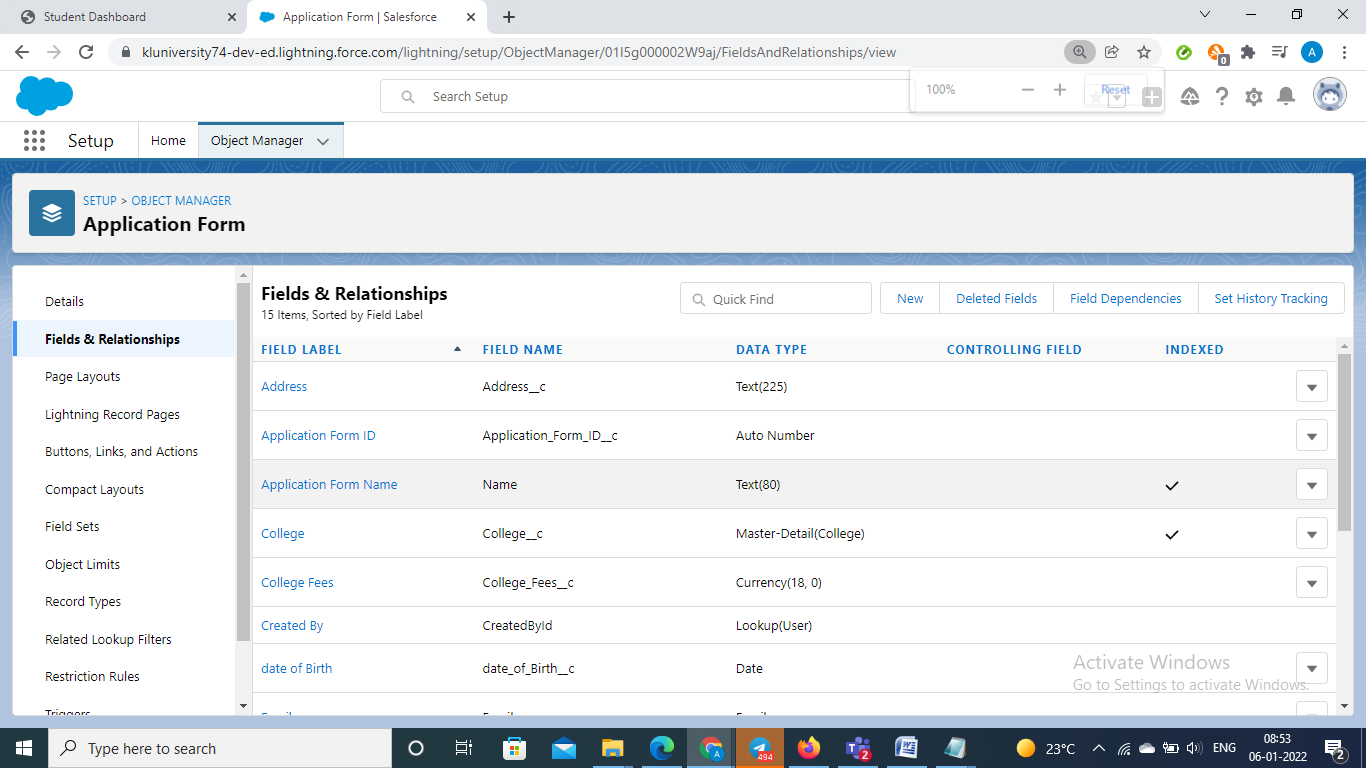
### Create Fields On Application Form Object

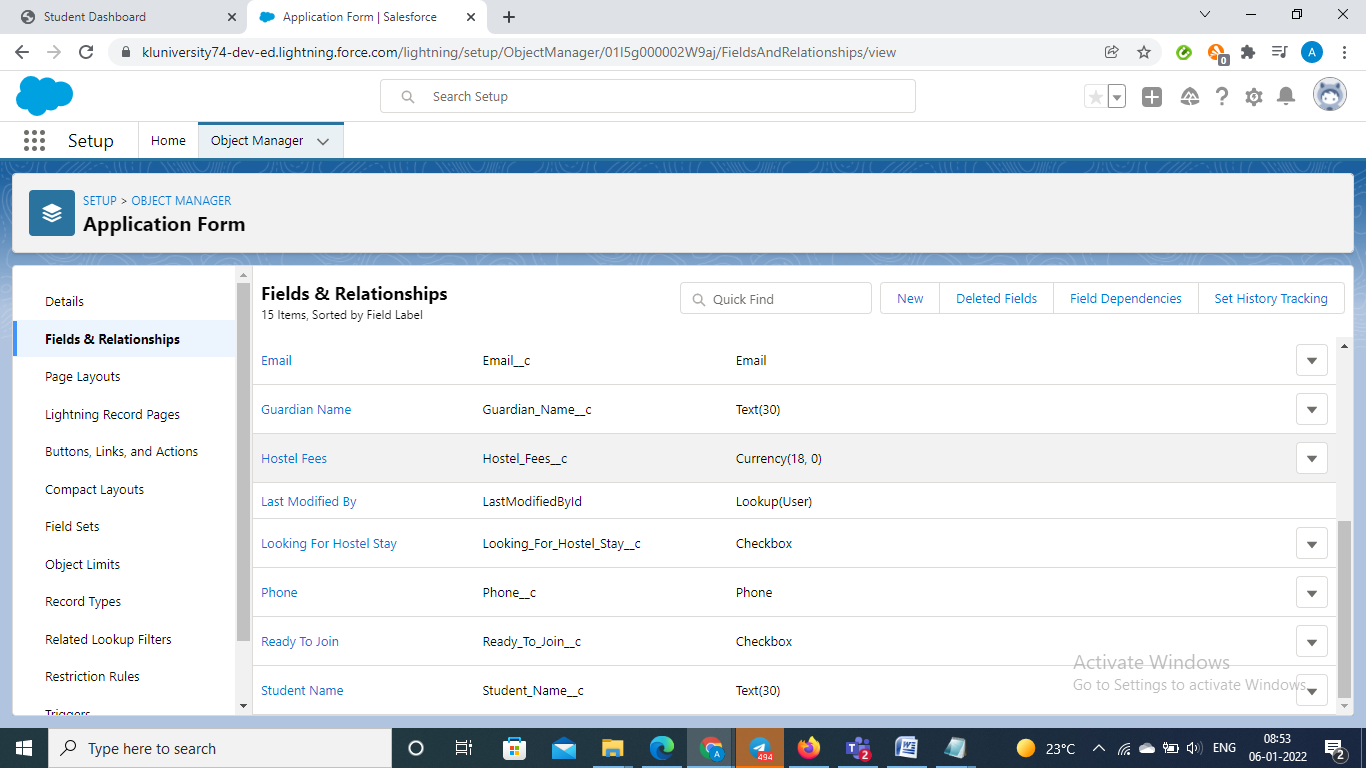
1. From Setup, click the Object Manager tab
2. On The top right side where you can find the search bar on the search bar type college and select the name of the object
3. On the left side select the fields and relationships
4. On the fields and relationships click new
5. Select the data type of the custom fields and then select next
6. Enter the field label and field name and then select next
7. On the Establish field level security leave as it is and then click net
8. Leave the page layouts as it is and then click save and new

Same we fill follow the above steps:

In this we are going to create the fields in the application form object where the fields are

* Application Form ID
* Address
* College
* Hostel Fees
* College Fees
* date of Birth
* Email
* Guardian Name
* Looking For
* Hostel Stay
* Ready To Join
* Student Name
* Phone

****

****

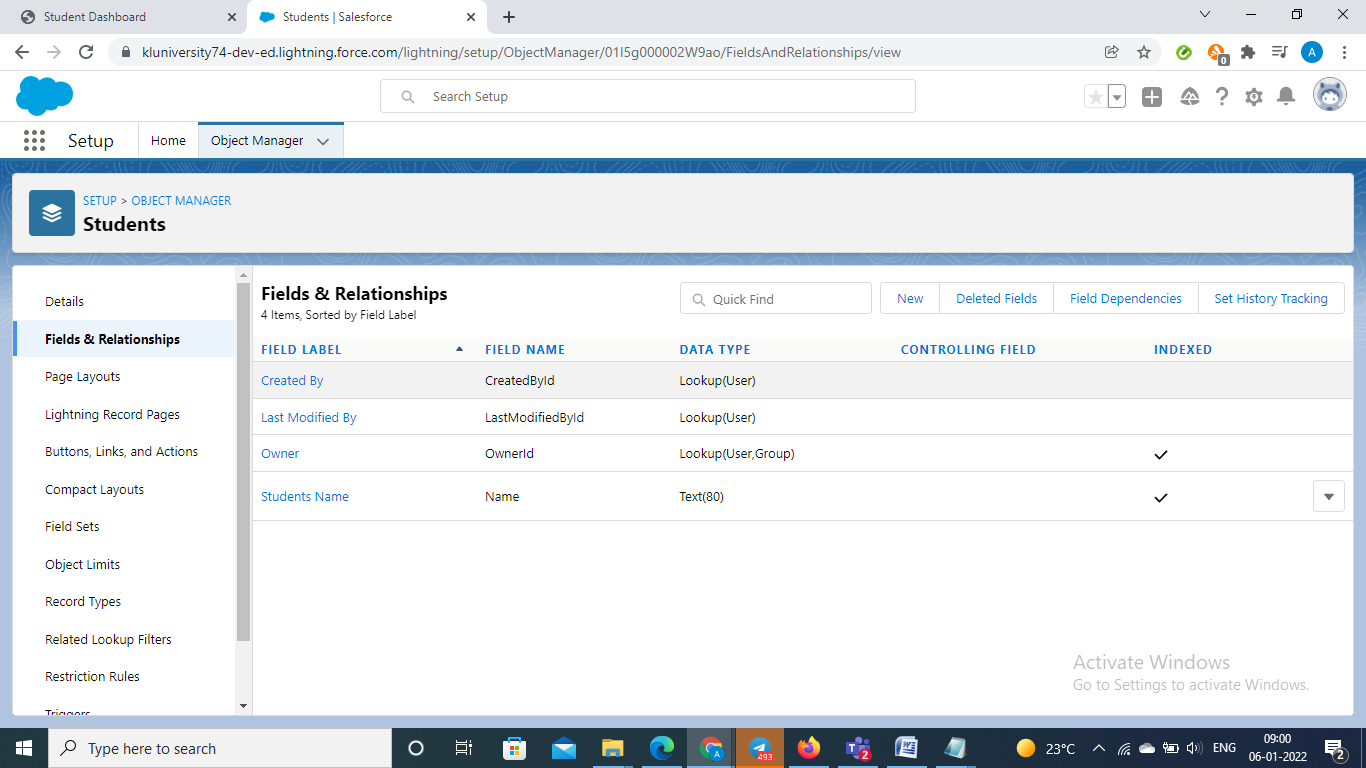
### Create Fields On Student Object

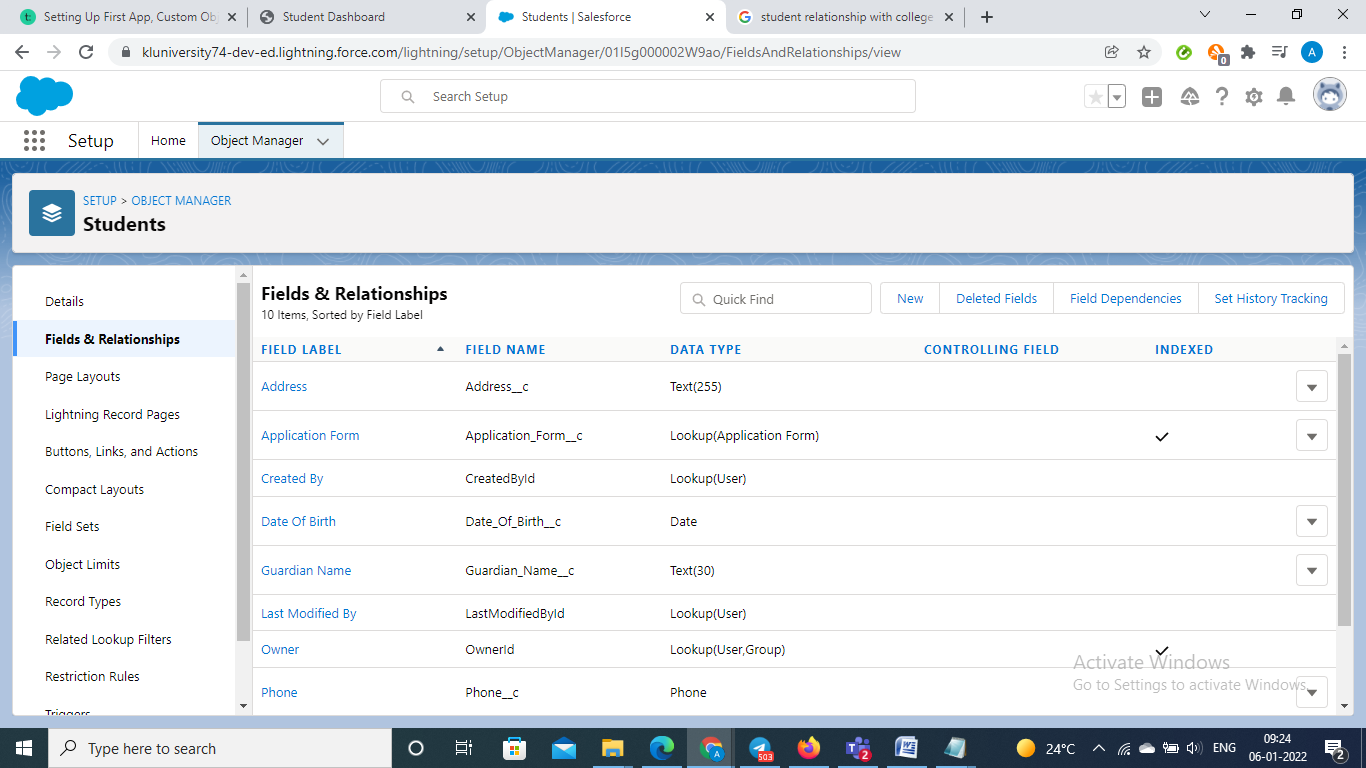
1. From Setup, click the Object Manager tab
2. On The top right side where you can find the search bar on the search bar type college and select the name of the object
3. On the left side select the fields and relationships
4. On the fields and relationships click new
5. Select the data type of the custom fields and then select next
6. Enter the field label and field name and then select next
7. On the Establish field level security leave as it is and then click net
8. Leave the page layouts as it is and then click save and new

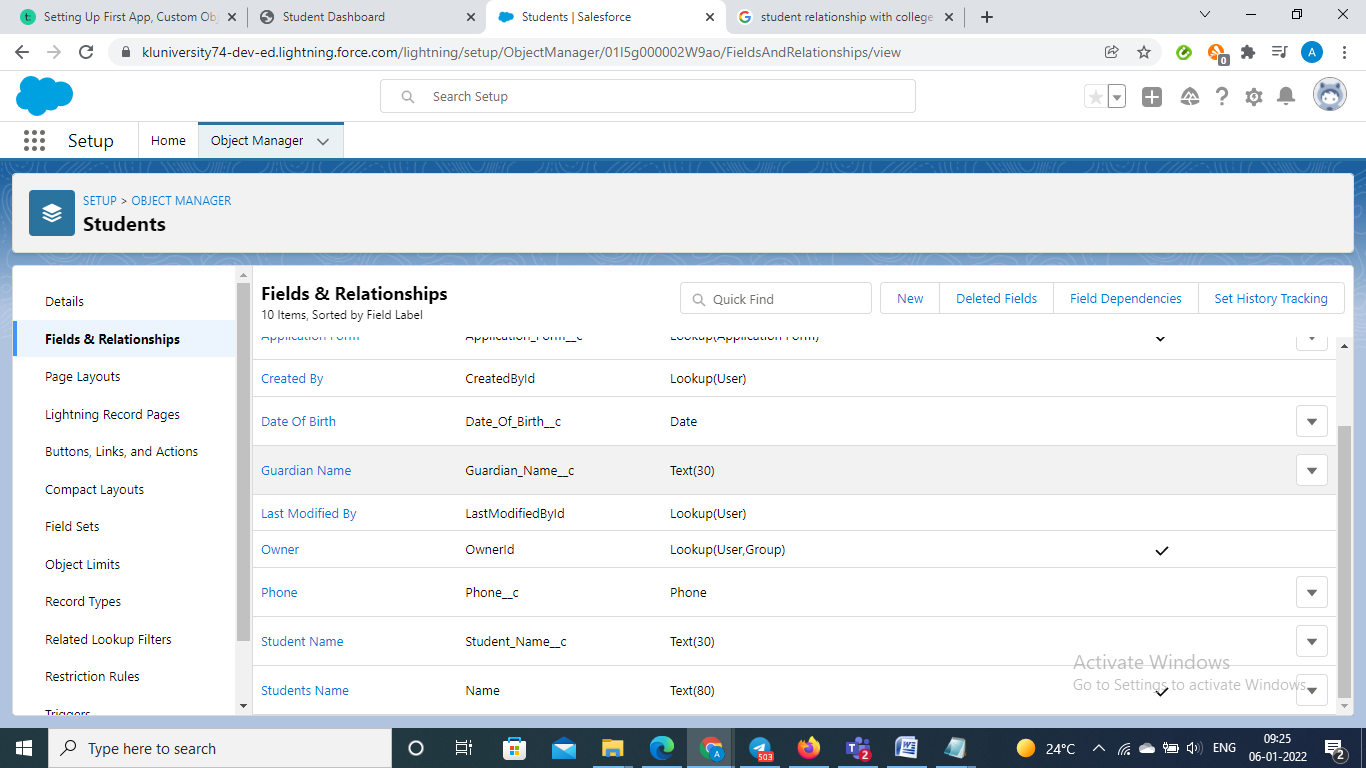
Same we fill follow the above steps:

In this we are going to create the fields in the Student object where the fields are

* Student Name
* Address
* Application Form
* College Name
* Date Of Birth
* Guardian Name
* Phone

****

****

****

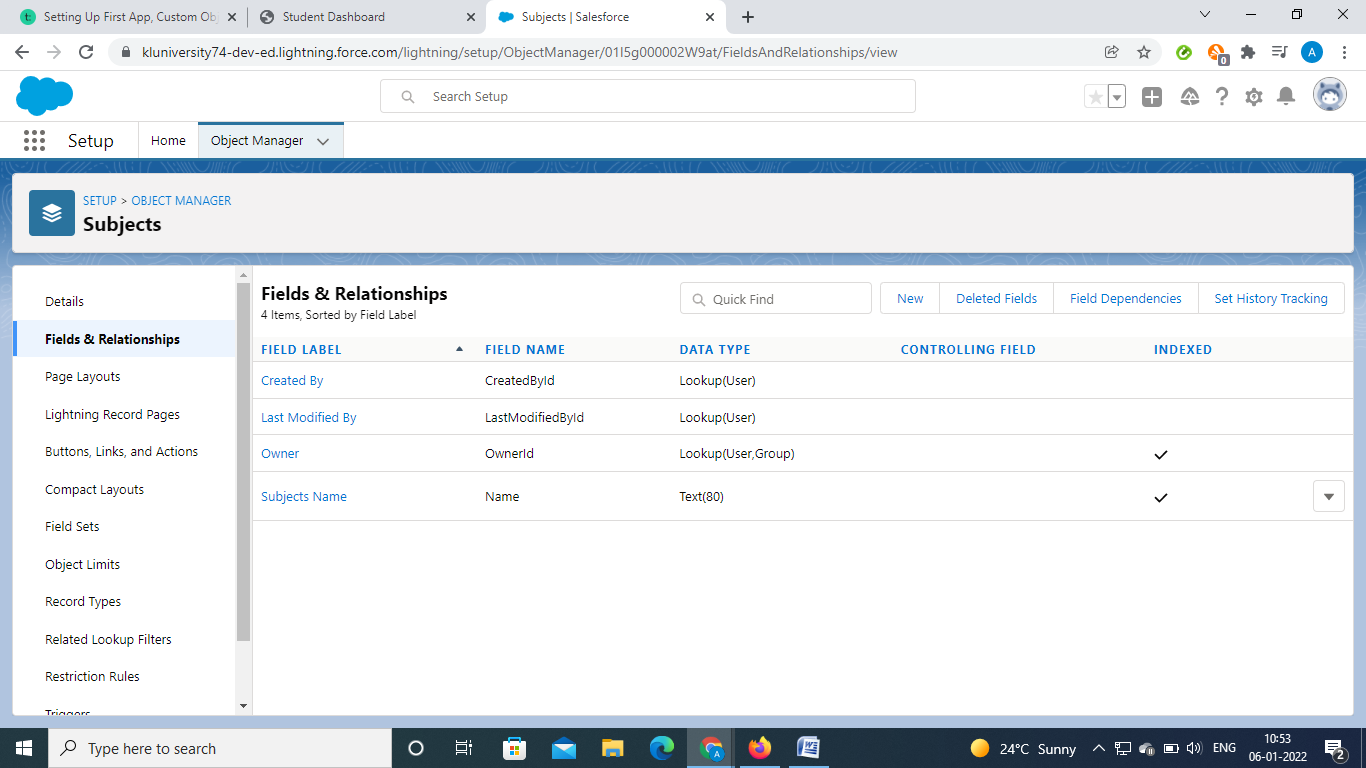
### Create Fields On Subject Object

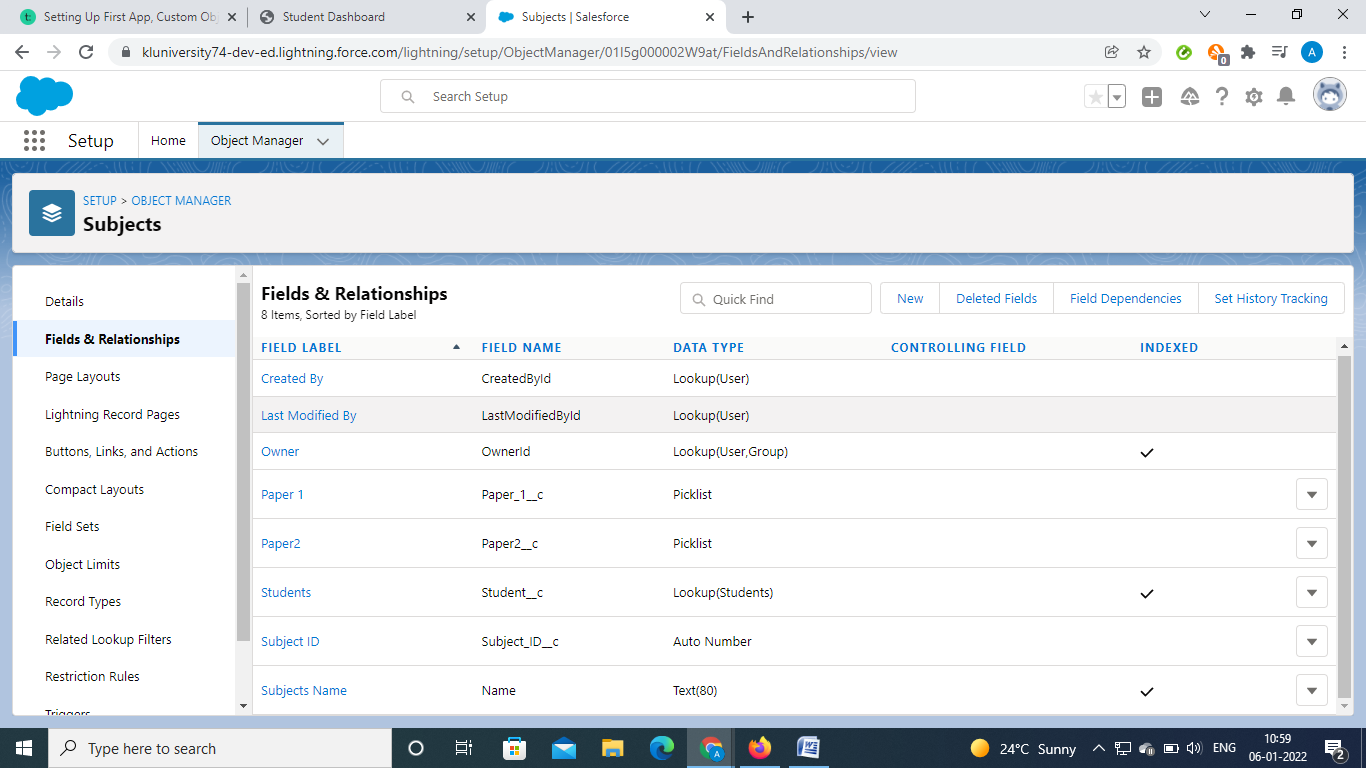
1. From Setup, click the Object Manager tab
2. On The top right side where you can find the search bar on the search bar type college and select the name of the object
3. On the left side select the fields and relationships
4. On the fields and relationships click new
5. Select the data type of the custom fields and then select next
6. Enter the field label and field name and then select next
7. On the Establish field level security leave as it is and then click net
8. Leave the page layouts as it is and then click save and new

Same we fill follow the above steps:

In this we are going to create the fields in the Subjects object where the fields are

* Subject ID
* Paper 1
* Paper2
* Student





**Day 3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Topic:** Adding Business Logic To the Application

**Milestone / Activities:**

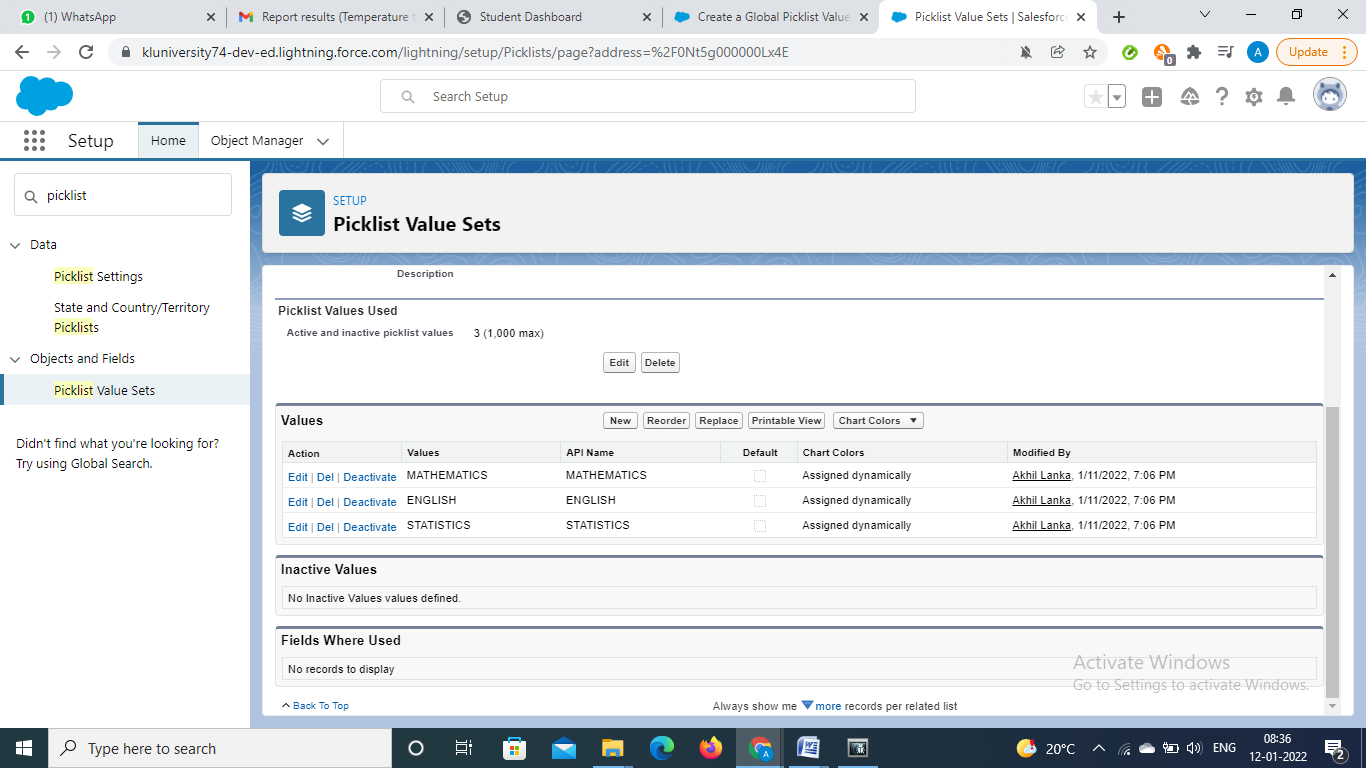
* Using global picklist value sets to share values across objects and custom picklist fields, and to restrict the picklists to only the values that you specify.
* Field dependencies are filters that allow us to change the contents of a picklist based on the value of another field. Rather than displaying every value for Region in a single picklist, you can limit the values that are displayed based on a value for another field, like Zone.
* Validation rules verify that the data a user enters in a record meets the standards you specify before the user can save the record. A validation rule can contain a formula or expression that evaluates the data in one or more fields and returns a value of “True” or “False”.
* Salesforce Flow provides declarative process automation for every Salesforce app, experience, and portal. Included in Salesforce Flow are two point-and-click automation tools:
* A flow is an application that automates complex business processes. Simply put, it collects data and then does something with that data. Flow Builder is the declarative interface used to build individual flows.

### Creating Global Picklist Value Sets

1. From Setup, enter Picklist in the Quick Find box, then select Picklist ValueSets.
2. Next to Global Value Sets, click New.
3. Enter a label for the global value set. This name appears in Setup, and when users create a picklist based on this global value set.
4. To tell users what these values are for, enter a specific description of the global value set. This text appears on the Picklist Value Sets list page in Setup.
5. Enter the values, one per line.
6. Optionally choose to sort the values alphabetically or to use the first value in the list as the default value, or both.
7. Click Save.

In this we are going to create the global picklist values where the picklist values are

* College
* Paper1
* Paper2



### Creating Field Dependencies

### In this we are going to create the field dependency between college Name and Email, where the controlling field is college Name and dependent field is Email

### &

### We also going to create field dependency between college Name and capacity of students, where the controlling field is college Name and dependent field is Capacity of Students

### While still viewing the College object in the Object Manager, select Fields & Relationships.

### Click Field Dependencies.

### Click New.

### Select College Name as the Controlling Field.

### Select Email / Capacity of The Students as the Dependent Field.

### Click Continue.

### Select the appropriate Email / Capacity in each column by double-clicking them

### Click Preview, then test the dependency.

### Click Close to close the preview window.

### Click Save.

### 

**Creating Validation Rules**

* **validation rule on the college object**

### From Setup, go to Object Manager and click College.

### In the left sidebar, click Validation Rules.

### Click New.

### Enter the following properties for your validation rule:

### a. Rule Name: College\_Name\_similar

### b. Error Condition Formula: TEXT(College\_Name\_\_c) <> Name.

### c. Error Message: Please use the College Name in Record info

### To check your formula for errors, click Check Syntax.

### Click Save to finish.

### 

### validation rule on the application form object

### From Setup, go to Object Manager and click application form .

### In the left sidebar, click Validation Rules.

### Click New.

### Enter the following properties for your validation rule:

### a. Rule Name: studentrecord

### b. Error Condition 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 ) ) ).

### Error Message: stop any modification on the application forn once a student record is created

### To check your formula for errors, click Check Syntax.

### Click Save to finish.

### 

### Create The Student Record Using Flow

### 1. Create Flow Navigate to Setup, find Flows menu and click New Flow button

### Add Variable

### Add Screen for Data Entry

### Add Record Create item

### Add Screens for success and error handling

### 2. Edit Account Lightning Page

### Navigate to  Setup | Object Manager | Student record  | Lightning Record Pages.

### Click the page, then click Edit button.

### 3. Add Component

### From Standard components available in the Component panel at the left, drag Flow component into Lightning Page.

### 4. Select the Flow

### Select component added and change the values, change Flow value with the Flow created in step-1

### 5. Showtime

### Now, open an Student record check the Flow component added. You can add Contact and the contact will automatically link to the Student record.

### 

### Process Automation

### Genrall Process for The Creation of the process Builder

### Click and select Setup. This launches Setup in a new tab.

### From Setup, enter Builder in the Quick Find box, and select Process Builder.

### Click New.

### For Process Name, type Application Form.

### For The process starts when, select A record changes, and click Save.

### Click + Add Object.

### In the right window, select Application Form from the Object drop-down list.

### For Start the process select when a record is created or edited.

### Click Save.

### 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

### &

1. **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 setup, click on the adjacent box called “Immediate action”. And select create a record on the student object. Please follow the below screenshot.**

### 

### 

**Day4**

**Topic:** Batch Apex

**Milestone / Activities:**

An Apex class is a template or blueprint from which Apex objects are created. Classes consist of other classes, user-defined methods, variables, exception types, and static initialization code.

An Apex class is a template or blueprint from which Apex objects are created. Classes consist of other classes, user-defined methods, variables, exception types, and static initialization code.

**Create A Batchapex For Application Form**

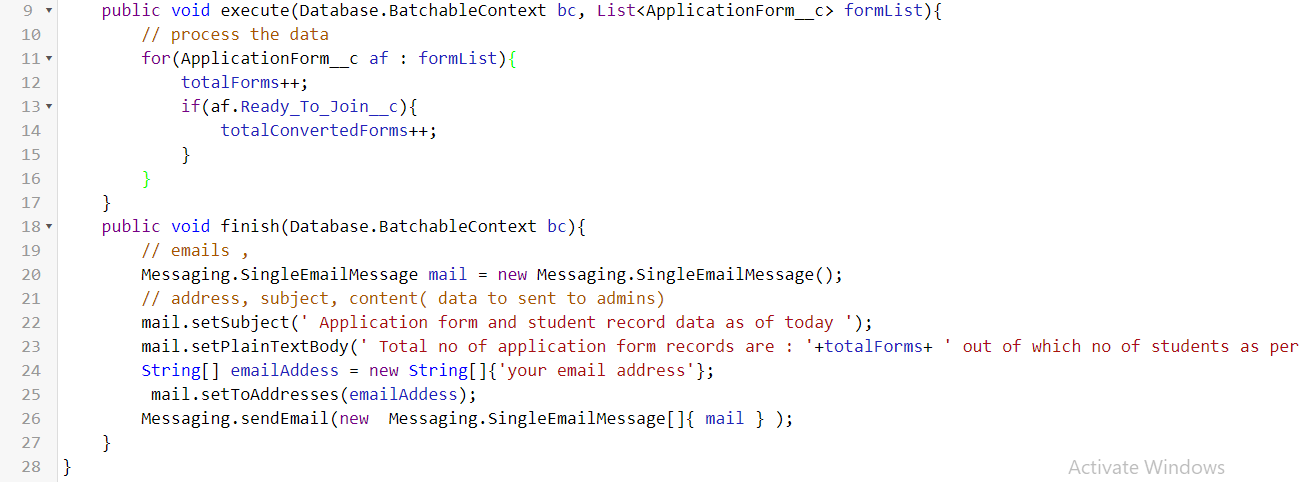
1. In the Developer Console, select **File** > **New** > **Apex Class**,

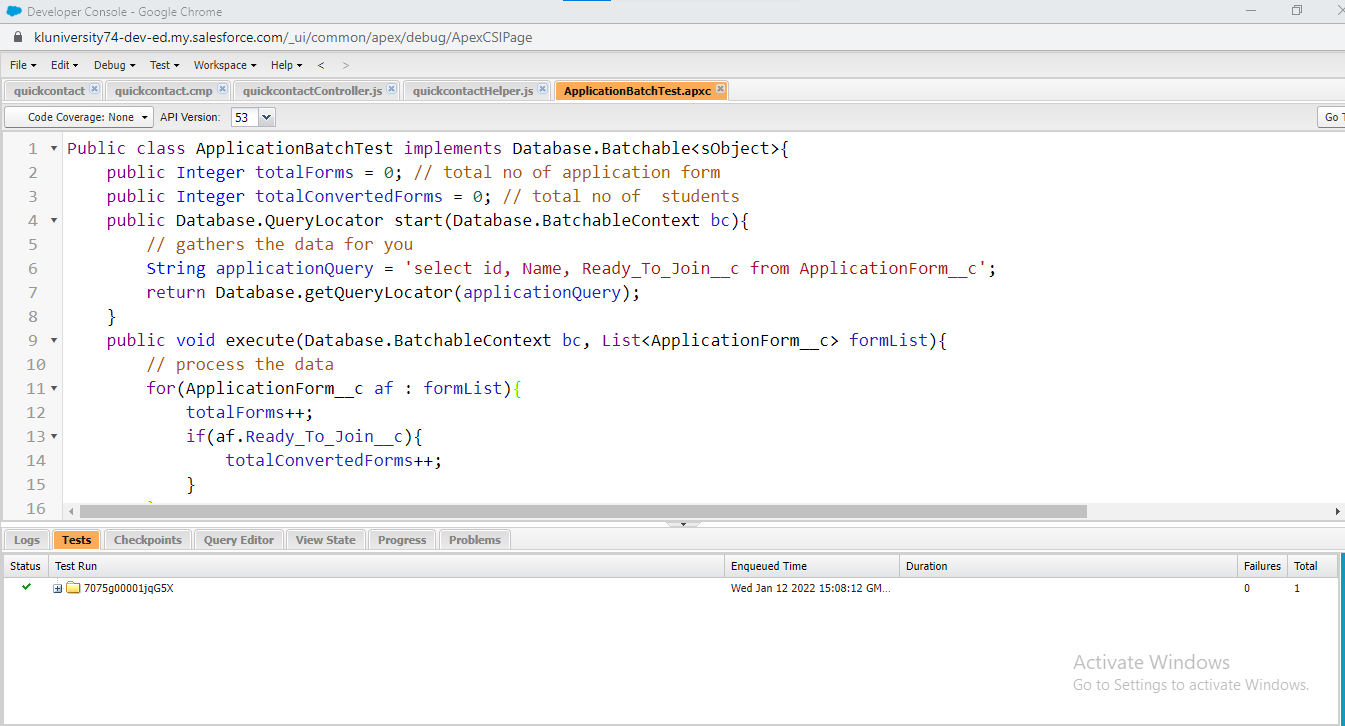
specify   the class name and click **OK**.

1. Make the class **global** and define the three methods of the interface.
2. Declare instance variables to store the query
3. Define a **constructor**
4. Implement the **start()** method
5. Implement the **execute()** method
6. Save the file

**Step 2: Run the Batch**

1. In the Developer Console, click **Debug** > **Open Execute Anonymous Window**.
2. Type the Sample Apex code for testing the code
3. Click **Execute** and check the output



****

**Create A Schedular Class**

1. From Setup, enter Apex Classes in the Quick Find box, select Apex Classes, and then click Schedule Apex.
2. Specify the name of a class that you want to schedule.
3. Specify how often the Apex class is to run.
4. Specify the start and end dates for the Apex scheduled class. If you specify a single day, the job only runs once.
5. Specify a preferred start time. The exact time the job starts depends on service availability.
6. Click Save

**Code:**

**public class applicationschedule implements Schedulable{**

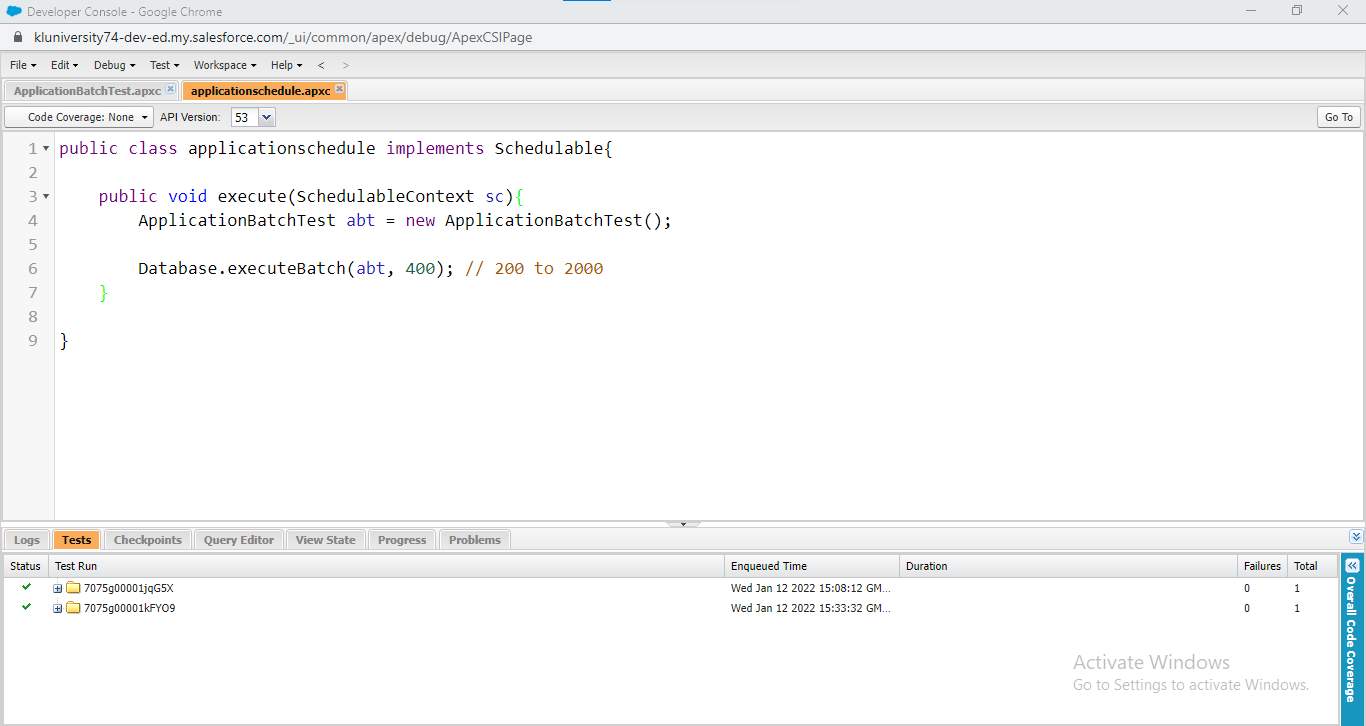
**public void execute(SchedulableContext sc){**

**ApplicationBatchTest abt = new ApplicationBatchTest();**

**Database.executeBatch(abt, 400); // 200 to 2000**

**}**

**}**

****