Custom application using salesforce

You are expected to:

* Use Salesforce to create a **custom application** (like your own mini-software inside Salesforce)
* Use **tools** (like objects, fields, tabs) and optionally Apex/Lightning components
* Demonstrate that you understand **how apps are built and deployed** in the Salesforce platform

**Final Goal (What You’ll Build):**

You will build a small custom application — let’s say a **“Student Management App”** — inside Salesforce Cloud.

The app can include:

* A custom object called "Student"
* Custom fields like Name, Age, Grade
* A tab to access it from the main menu
* (Optional) Logic in Apex or validation rules

**What Is Salesforce?**

🔹 Salesforce is a cloud-based Customer Relationship Management (CRM) platform.  
🔹 It helps businesses manage leads, customers, sales, and services all in one place.  
🔹 It’s highly customizable using built-in tools and code (like Apex).  
🔹 Companies use Salesforce to automate workflows, store customer data, and build apps — all in the cloud.

**What Is Apex?**

🔹 Apex is a programming language developed by Salesforce.  
🔹 It’s similar to Java and is used to write custom logic on the Salesforce platform

Step1: Log in at <https://login.salesforce.com>

1. Click the ⚙️ (Setup) icon → Click “Setup”

Setup is where you create apps, objects, fields, and automation

**Step 2: Create a New Custom App**

🧩 What you're doing: Creating the shell of your application

1. In Setup, search "App Manager"
2. Click “App Manager”
3. Click “New Lightning App”

You land on app details : This is where you set up the **basic identity of your app** — like its name, logo, and colors. It makes your app look clean and branded inside Salesforce.

Name:Student Data

**description** *(Optional)*:  
📌 Example: An app to manage students' personal and academic information.  
**Why?** So that when someone opens App Manager, they can quickly see what this app is about.

Branding logo: It personalizes your app on the Salesforce header

**App Options**

**👉 What is this page for?**  
This is where you decide **how your app will be accessed** and displayed inside Salesforce.

**App Settings**

* ☑️ Keep **App is visible in Lightning Experience**
* ☑️ Keep **Setup option in App Launcher**

**STEP 6: Navigation Items**

**👉 What is this page for?**  
This is where you decide **which tabs (objects)** your app should have in its top navigation bar.

**✳️ Add Navigation Items:**

From **Available Items** on the left side — search for and select:

 Home

*  Students
*  Academic Records
*  Student Attendance
*  Reports

Assign to User Profiles: This is to decide **which users can see this app** based on their Salesforce profile.( System Administrator)

=----------------------------------------------------------

Add custom obejcts

Open Your New App

Go to the **App Launcher (9-dots icon top-left)**  
Search for: Student Data  
Click on it.

Now your app opens with tabs:  
**Home | Students | Academic Records | Student Attendance | Reports**

**Why?**  
This is your app’s main work area. From here, you can start adding records, customizing pages, and building list views.

**Adding custom objects**

**Steps to Create a Custom Object:**

**📍 Step 1: Go to Setup**

* **In Salesforce, click the Gear (⚙️) icon on top-right**
* **Select Setup**

**📍 Step 2: Find Object Manager**

* **In Setup, on the top menu bar — click Object Manager**

**📍 Step 3: Click Create | Custom Object**

* **In Object Manager, click the Create drop-down**
* **Choose Custom Object**

**📍 Step 4: Enter Object Details**

**Example for Books:**

* **Label: Books**
* **Plural Label: Books**
* **Object Name: Books (auto-filled)**
* **Check Launch New Custom Tab Wizard after saving this custom object ✅**

**👉 Leave other options as they are**

**Click Save**

**📍 Step 5: Create a Custom Tab**

* **As soon as you hit Save — it will ask you to create a Custom Tab for this object**
* **Choose a Tab Style (any book icon, etc.)**
* **Click Next, Next, then Save**

**✅ Now your Books object is created and visible in your app tabs**

**📍 Step 6: Add Custom Fields in Your Object**

**Now you need to add fields like Book Name, Author, ISBN.**

* **Go to Object Manager**
* **Find your object (like Books)**
* **Click on it**
* **Go to Fields & Relationships**
* **Click New**
* **Choose a field type (Text, Number, Date, etc.)**
* **Add details (like Field Label: Book Name, Data Type: Text)**
* **Click Next, set visibility for profiles, then Save**

**Repeat for other fields like:**

* **Author (Text)**
* **ISBN (Text)**
* **Genre (Picklist — Fiction, Non-Fiction, Sci-Fi, etc.)**
* **Availability (Picklist — Available, Issued)**

**📍 Step 7: Add the Object Tab to Your App**

* **Go to App Manager**
* **Find your app (Student Data or whatever)**
* **Click Edit**
* **Go to Navigation Items**
* **Click Add More Items**
* **Search for your new object (Books)**
* **Add it to Selected Items and Save**

**✅ Now it appears in your App’s navigation bar!**

**📢 That’s It — You Created Your Own Object!**

**Library Management System**

**Objects you can create:**

* **Books**
  + Book Name, Author, ISBN, Genre, Availability Status
* **Students**
  + Name, Roll No, Department, Year
* **Issue Records**
  + Book, Student, Issue Date, Return Date, Fine if any

**Bonus Idea:** Add a custom report for "Books Not Returned" or "Most Issued Books"

----------------------------------------------------------------------

**Event Registration System**

**Objects:**

* **Events**
  + Name, Date, Location, Capacity, Description
* **Participants**
  + Name, Email, College, Year, Department
* **Registrations**
  + Event, Participant, Registration Date, Payment Status

**Add On:** Send Email Notifications using Salesforce Flows

**Faculty & Lecture Scheduling System**

**Objects:**

* **Faculty**
  + Name, Department, Specialization, Contact
* **Lectures**
  + Subject, Date, Time, Faculty, Room No.
* **Attendance**
  + Faculty, Date, Status (Present/Absent)

**Bonus:** View faculty schedules in Calendar view