

Library Management Phase 10: Final Presentation & Demo Day

Objective:

Demonstrate the end-to-end functionality of the **Library Management**, highlight outcomes, and provide documentation for teachers, students, librarians, and library admin.

1. Project Pitch

Goal: Showcase how Smart Library CRM simplifies resource tracking, improves student–teacher interaction, and increases library efficiency.

Key Points to Present:

1. Streamlined student/teacher record management.
 2. Automated book issue/return tracking & fine calculation.
 3. Real-time updates for teachers, students, and librarians.
 4. Analytics through reports and dashboards (usage trends, overdue items, popular books).
-

2. Demo Flow (Click-by-Click Navigation)

Step 1 – Login & Access App

- Open Salesforce → Go to **Library Management Lightning App**.

Step 2 – Add/Manage Student or Teacher

- Navigate to **Students or Teachers Tab** → **New**.
- Fill in details (Name, DOB, Contact, Class/Department).
- Save → System applies any automation (e.g., auto-assign class or default library card number).

Step 3 – Book Selection / Issue

- Navigate to **Books Tab** → Select Book.
- Create **Borrow Request** by selecting book, issue date, and due date.

Step 4 – Return & Fine Handling

- Navigate to **Borrow Requests Tab** → Mark as **Returned**.

- System auto-calculates Fine Amount (if overdue).
- Librarian records payment of fine if applicable.

Step 5 – Notifications & Feedback

- Confirm book issue/return → Student receives notification (demo simulated without actual emails).
- Navigate to **Feedback Tab** → **New Feedback** to record user feedback on library services.

Step 6 – Dashboard Update

- Navigate to **Dashboards** → **Library Overview Dashboard**.
 - Verify that new borrow requests, returns, fines, and feedback appear correctly.
-

3. Handoff Documentation

- **Setup Guide:** Step-by-step instructions to deploy the Library CRM in Salesforce Org.
 - **User Manuals:**
 - **Librarians:** How to manage books, issues, returns, and fines.
 - **Teachers/Students:** How their borrow requests, returns, and feedback are managed in the system.
-

4. LinkedIn Showcase

- Tag the project as **Smart Library Management** .
- Highlight Salesforce implementation skills, including:
 - Custom Objects & Relationships (Books, Borrow Requests, Students, Teachers).
 - Apex Triggers (auto-calculate fines, send reminders).
 - Reports & Dashboards (overdue books, most borrowed resources).
 - Data Management & Deployment (imports, backups, permissions).

PHASE 1 Project Title: Library Management

Problem Statement

Managing and tracking books can become cumbersome when done manually or on paper. Small libraries, students, or hobbyists often need a lightweight system to record book details, view them quickly, and make edits without installing heavy software or maintaining a server. The Simple Library Management website addresses this by offering a browser-based CRUD tool using only HTML, CSS, and JavaScript with localStorage.

Requirement Gathering

- Identify what core actions the user needs (add, view, edit, delete books).
- Decide which fields are mandatory (Title, Author, Year, ISBN).
- Determine storage mechanism (localStorage vs. server).
- Collect UI expectations (form + table layout, buttons).
- Plan file structure (index.html, styles.css, script.js).
- Document non-functional needs (works offline, simple design).

Stakeholder Analysis

- Define primary users (students, small libraries, demo users).
- Identify secondary users (developers learning CRUD apps).
- Note stakeholders' technical skills (beginner-friendly, no server).
- Capture feedback on desired features (extra fields, pop-ups).
- Understand constraints (browser-only, small data sets).
- Set expectations for future enhancements (server API, multi-user).

Business Process Mapping

- Map the flow of adding a book (form input → JS → localStorage → table).

- Describe how data persists after refresh (JSON in localStorage).
- Show the edit flow (click edit → prompt/update → save back).
- Map the delete flow (click delete → confirm → remove from storage).
- Define initial data state (empty table).
- Document error handling or confirmations (validation, clearing storage).

Industry-Specific Use Case Analysis

- Compare with small-scale library or personal book lists.
- Explore educational uses (teaching CRUD in classes).
- Consider adaptation for inventory or asset tracking.
- Assess scalability limits (localStorage size, single user).
- Identify privacy and data safety considerations (local only).
- Evaluate user interface needs across devices (desktop, mobile).

AppExchange / Expansion Exploration

- Plan how to extend fields (Genre, Publisher, Ratings).
- Replace prompts with inline edit forms or modals.
- Investigate moving from localStorage to a simple backend or API.
- Consider packaging as a small open-source app or template.
- Experiment with new UI styles (Tailwind, Bootstrap, animations).
- Explore publishing on platforms like Salesforce AppExchange or GitHub Pages as a free tool.

PHASE 2: Org Setup & Configuration

1. Introduction

This phase sets up a Salesforce Developer Org to model a small library for cataloguing, lending, and returns. It builds on Phase 1 by configuring book records, librarian/member roles, and

security so the system reflects real library workflows for adding, editing, deleting, and viewing books.

2. Objective

Create a clean Salesforce Developer Org named “Library Connect” with sample users (Head Librarian, Librarian, Member).

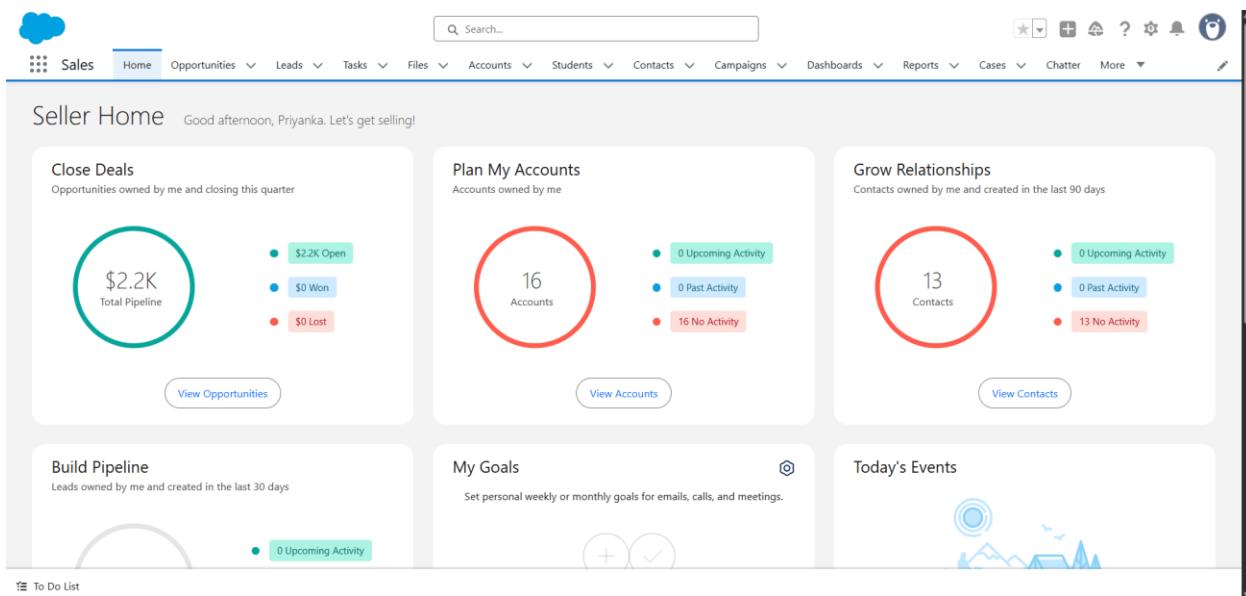
Use OWD and sharing rules to protect book data.

Showcase book entry, editing, and role-based visibility, and lay the groundwork for future fields and external integrations.

Step 1:- • Salesforce Editions

Sign up & Login

Signed up for a Salesforce Developer Edition and logged into the Lightning Experience. Confirmed access to the Setup area sing the gear icon.



Step 2:- •Company Profile Setup

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

- Search Bar:** Search Setup
- Header:** Setup, Home, Object Manager
- Left Sidebar:**
 - Company Settings
 - Business Hours
 - Calendar Settings
 - Public Calendars and Resources
 - Company Information
 - Data Protection and Privacy
 - Fiscal Year
 - Holidays
 - Language Settings
 - My Domain
 - Didn't find what you're looking for? Try using Global Search.
- Company Information Page:**
 - Section:** Company Information, Gyan Ganga College Of Technology
 - Organization Detail:**

Organization Name	Gyan Ganga College Of Technology
Primary Contact	OrgFarm EPIC
Division	
Address	United States
Fiscal Year Starts In	January
Activate Multiple Currencies	<input type="checkbox"/>
Enable Data Translation	<input type="checkbox"/>
Newsletter	<input checked="" type="checkbox"/>
Admin Newsletter	<input checked="" type="checkbox"/>
Hide Notices About System Maintenance	<input type="checkbox"/>
Hide Notices About System Downtime	<input type="checkbox"/>
Locale Formats	ICU
 - Phone:**
 - Default Locale: English (United States)
 - Default Language: English
 - Default Time Zone: (GMT-07:00) Pacific Daylight Time (America/Los_Angeles)
 - Currency Locale: English (United States) - USD
 - Used Data Space: 505 KB (10%) [View]
 - Used File Space: 17 KB (0%) [View]
 - API Requests, Last 24 Hours: 0 (15,000 max)
 - Streaming API Events, Last 24 Hours: 0 (10,000 max)
 - Restricted Logins, Current Month: 0 (0 max)
 - Salesforce.com Organization ID: 00GK000007Dvb
 - Organization Edition: Developer Edition
 - Instance: CAN96
 - Log:** Created By OrgFarm EPIC, 7/20/2025, 5:31 AM | Modified By Priyanka Pillai, 8/12/2025, 8:30 AM

Step 3:- • Business Hours & Holidays

Created " Library Management Hours" Mon-Sat 10 AM-5 PM.

Added relevant Holidays and linked them to Business Hours.

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

- Search Bar:** Search Setup
- Header:** Setup, Home, Object Manager
- Left Sidebar:**
 - Company Settings
 - Business Hours**
 - Didn't find what you're looking for? Try using Global Search.
- Business Hours Page:**
 - Section:** Business Hours, Library Management Hours
 - Business Hours Detail:**

Business Hours Name	Library Management Hours	Time Zone														
Business Hours	<table border="1"> <tr><td>Sunday</td><td>No Hours</td></tr> <tr><td>Monday</td><td>10:00 AM to 5:00 PM</td></tr> <tr><td>Tuesday</td><td>10:00 AM to 5:00 PM</td></tr> <tr><td>Wednesday</td><td>10:00 AM to 5:00 PM</td></tr> <tr><td>Thursday</td><td>10:00 AM to 5:00 PM</td></tr> <tr><td>Friday</td><td>10:00 AM to 5:00 PM</td></tr> <tr><td>Saturday</td><td>10:00 AM to 5:00 PM</td></tr> </table>	Sunday	No Hours	Monday	10:00 AM to 5:00 PM	Tuesday	10:00 AM to 5:00 PM	Wednesday	10:00 AM to 5:00 PM	Thursday	10:00 AM to 5:00 PM	Friday	10:00 AM to 5:00 PM	Saturday	10:00 AM to 5:00 PM	(GMT+05:30) India Standard Time (Asia/Kolkata)
Sunday	No Hours															
Monday	10:00 AM to 5:00 PM															
Tuesday	10:00 AM to 5:00 PM															
Wednesday	10:00 AM to 5:00 PM															
Thursday	10:00 AM to 5:00 PM															
Friday	10:00 AM to 5:00 PM															
Saturday	10:00 AM to 5:00 PM															
 - Business Hours Detail (Continued):**
 - Active:
 - Created By: Priyanka Pillai 9/20/2025, 3:56 AM | Last Modified By: Priyanka Pillai 9/20/2025, 3:56 AM
 - Holidays:**

Holiday Name	Description	Date and Time
Diwali		10/20/2025 All Day
Holi		3/12/2025 All Day
 - Log:** Always show me [more] records per related list

Step 4 — Users

- Created 3 sample users:

Display Name	Username (example)	Profile	Role
Librarian Admin	admin_librarian1@library.com	Test Profile	Test Role
Teacher	Teacher.librarian@library.com	Identity User	SVP, Human Resources

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/> Edit Login	Ajay	ajay	ajay121@test.com	CEO	✓	Test Profile
<input type="checkbox"/> Edit	Chatter Expert	chatter	chatty_00dgk000007idvbuau_5igs2b2afqk@chatter.salesforce.com		✓	Chatter Free User
<input type="checkbox"/> Edit Login	EPIC_OrgFarm	QEPIIC	epic_5652db3a7c11@orgfarm.salesforce.com		✓	System Administrator
<input type="checkbox"/> Edit Login	Library Admin	admin	admin_librarian@library.com	Test Role	✓	Test Profile
<input type="checkbox"/> Edit	Pillai_Priyanka	cse	cse22_priyankapillai43@agentforce.com		✓	System Administrator
<input type="checkbox"/> Edit Login	Teacher	admin	teacher.librarian@library.com		✓	Identity User
<input type="checkbox"/> Edit	User_Integration	integ	integration@00dgk000007idvbuau.com	SVP_Human_Resources	✓	Analytics Cloud Integration User
<input type="checkbox"/> Edit	User_Security	sec	insightssecurity@00dgk000007idvbuau.com		✓	Analytics Cloud Security User

Step 5 Profiles

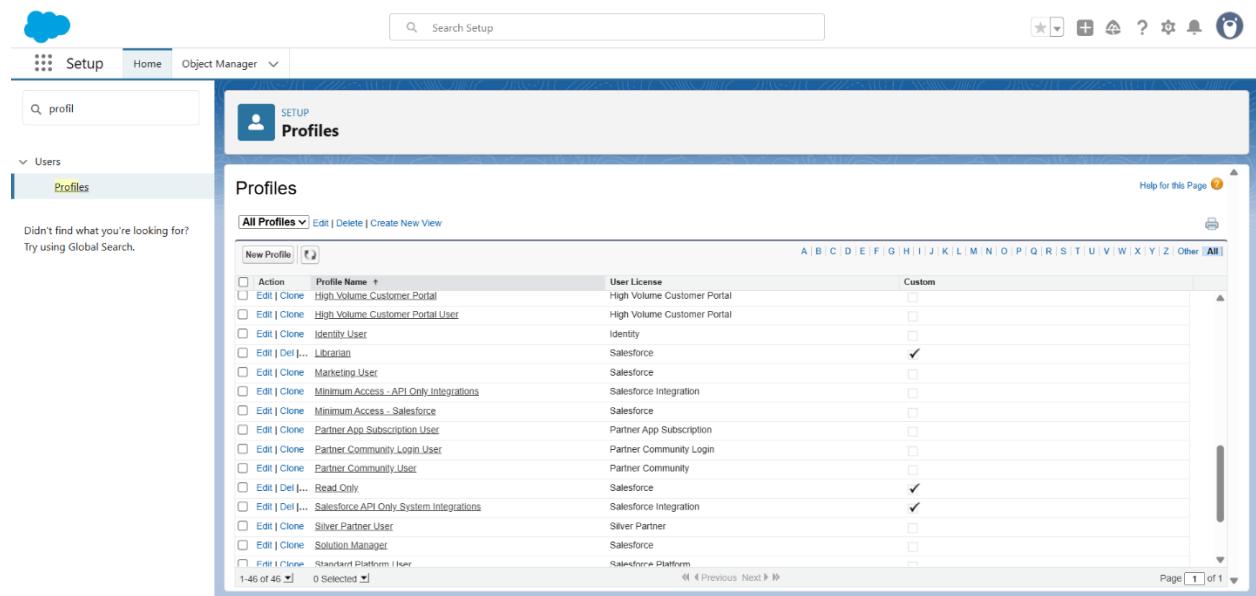
Cloned Standard User → created Librarian profile.

Cloned Standard User → created Teacher profile.

Adjusted Object Permissions:

Mentor = Read/Create/Edit/Delete

Student = Read/Edit/Create



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

- Header:** Includes the Salesforce logo, a search bar labeled "Search Setup", and various navigation icons.
- Left Navigation Bar:** Shows "Setup" selected, followed by "Home" and "Object Manager". Below that, under "Users", "Profiles" is selected. A global search bar shows the query "profil".
- Central Content Area:** The title is "Profiles". Below it is a table with columns: "Action", "Profile Name", "User License", and "Custom".
- Table Data:** The table lists 46 profiles. Some profiles have checkboxes in the "Action" column, while others do not. The "User License" column includes "High Volume Customer Portal", "Identity", "Salesforce", "Salesforce Integration", "Salesforce", "Partner App Subscription", "Partner Community Login", "Partner Community", "Salesforce", "Salesforce Integration", "Silver Partner", "Salesforce", and "Salesforce Platform". The "Custom" column has several checked boxes.
- Table Headers:** The first row of the table has headers: "Action", "Profile Name", "User License", and "Custom".
- Table Footer:** Shows "1-46 of 46" and "0 Selected".
- Page Navigation:** At the bottom right, it says "Page 1 of 1".

The screenshot shows the Salesforce Setup interface with the 'Profiles' page open. On the left, there's a sidebar with 'Users' and 'Profiles' selected. A search bar at the top says 'Search Setup'.

Custom Object Permissions

	Mentors	Basic Access				Data Administration				Students	Basic Access				Data Administration			
		Read	Create	Edit	Delete	View All Records	Modify All Records	View All Fields	Read		Create	Edit	Delete	View All Records	Modify All Records	View All Fields		
	Mentors	✓	✓	✓	□	□	□	✓	□	✓	□	□	□	□	□			
	Mentors	□	□	□	□	□	□	□	□	□	□	□	□	□	□			
	Mentors	□	□	□	□	□	□	□	□	□	□	□	□	□	□			

Session Settings

Session Times Out After: 2 hours of inactivity

Session Security Level Required at Login

Password Policies

- User passwords expire in: 90 days
- Enforce password history: 3 passwords remembered
- Minimum password length: 8
- Password complexity requirement: Must include alpha and numeric characters
- Password question requirement: Cannot contain password
- Maximum invalid login attempts: 10
- Lockout effective period: 15 minutes
- Obfuscate secret answer for password resets:
- Require a minimum 1 day password lifetime:
- Don't immediately expire links in forgot password emails:

Buttons: Edit, Clone, Delete, View Users

Step 6 — Role Hierarchy

Created simple hierarchy

- Librarian Admin
 - Librarian
 - Teacher

The screenshot shows the Salesforce Setup interface with the 'Roles' page open. On the left, there's a sidebar with 'Users' and 'Roles' selected. A search bar at the top says 'Search Setup'.

Creating the Role Hierarchy

You can build on the existing role hierarchy shown on this page. To insert a new role, click Add Role.

Your Organization's Role Hierarchy

Help for this Page ⓘ Show in tree view ▾

Tree view of roles:

- Gyan Ganga College Of Technology
 - CEO
 - CFO
 - COO
 - Librarian admin
 - SVP_Customer Service & Support
 - SVP_Human Resources
 - SVP_Sales & Marketing
 - Teacher

Step 7 Org:- Wide Defaults (OWD)

Setup → Sharing Settings

Mentor_c = Private.

Student_c = Controlled by Parent .

The screenshot shows the Salesforce Sharing Settings page. The left sidebar has a search bar and navigation links for Setup, Home, Object Manager, Security, and Sharing Settings. The main area is titled "Sharing Settings" and lists various objects with their sharing defaults:

Object	Sharing Default	Action
Shipping Configuration Set	Public Read Only	✓
Streaming Channel	Public ReadWrite	✓
Tableau Host Mapping	Public Read Only	✓
User Presence	Public Read Only	✓
User Provisioning Request	Private	✓
Waitlist	Private	✓
Web Cart Document	Private	✓
Work Order	Private	✓
Work Plan	Private	✓
Work Plan Template	Private	✓
Work Step Template	Private	✓
Work Type	Private	✓
Work Type Group	Public Read/Write	✓
Mentor	Public Read/Write	✓
Mentor	Private	✓
Student	Public Read/Write	✓
Student	Controlled by Parent	✓

At the bottom, there is an "Other Settings" section and a "Manager Groups" button.

Step 8 : Sharing Rules

- Created Public Group “Librarian” containing all Library users.
- Created Owner-based sharing rule:
 - Owned by: Teacher admin
 - Share with: Librarian admin
 - Access Level: Read/Write.

The screenshot shows the 'Sharing Settings' page in the Salesforce Setup interface. The left sidebar has 'Sharing Settings' selected under 'Security'. The main area displays various sharing rule sections: 'Work Type Sharing Rules', 'Work Type Group Sharing Rules', 'Mentor Sharing Rules', and 'Student Sharing Rules'. Each section includes a 'New' button and a 'Recalculate' button. Under 'Mentor Sharing Rules', there is a detailed table showing sharing criteria, shared with roles, and access levels:

Action	Criteria	Shared With	Access Level
Edit Del	Mentor: M_Email NOT EQUAL TO	Role: Test Role	Read Only
Edit Del	Owner in Role: Librarian admin	Role: Librarian admin	ReadWrite
Edit Del	Owner in Role: Teacher	Role: Teacher	ReadWrite

Step 9:- Login Access Policy

- Enabled Administrators Can Log in as Any User.**

The screenshot shows the 'Login Access Policies' page in the Salesforce Setup interface. The left sidebar has 'Login Access Policies' selected under 'Security'. The main area displays the 'Manage Support Options' section, which includes a table for controlling login access:

Setting	Enabled
Administrators Can Log in as Any User	<input checked="" type="checkbox"/>

Below this, there is a table for managing support organizations, packages, and availability:

Support Organization	Packages	Available to Users	Available to Administrators Only
Salesforce.com Support	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Step 10 — Library Management Lightning App

- Setup → App Manager → New Lightning App “Library Management”**

This screenshot shows the Salesforce Accounts page with the title "Recently Viewed". It displays a list of 15 recently viewed accounts, each with a checkbox, account name, account site, phone number, and account owner alias. The accounts listed are: Acme234, Acme234, Acme234, Acme234, Acme, Acme, Acme, HP, Sunrise Apartment, test_dashboard, prasad limma, Test Approval Process, pramod lee, Priyanka, and Thesmartbridge.

<input type="checkbox"/> Account Name	Account Site	Phone	Account Owner Alias
1 <input type="checkbox"/> Acme234			cse
2 <input type="checkbox"/> Acme234			cse
3 <input type="checkbox"/> Acme234			cse
4 <input type="checkbox"/> Acme234			cse
5 <input type="checkbox"/> Acme			cse
6 <input type="checkbox"/> Acme			cse
7 <input type="checkbox"/> Acme			cse
8 <input type="checkbox"/> HP		23445688	cse
9 <input type="checkbox"/> Sunrise Apartment			cse
10 <input type="checkbox"/> test_dashboard			cse
11 <input type="checkbox"/> prasad limma		56788989334	cse
12 <input type="checkbox"/> Test Approval Process			cse
13 <input type="checkbox"/> pramod lee		67687899896	cse
14 <input type="checkbox"/> Priyanka			cse
15 <input type="checkbox"/> Thesmartbridge		89786789557	cse

Step 11 : - Testing

This screenshot shows the Salesforce Account page for account "HP". The account details include: Type (Account), Phone (23445688), Website, Account Owner (Priyanka Pillai), Account Site, and Industry (Electronics). The "Details" tab is selected. A modal window titled "New Settings for Record Pages" is open, titled "Personalize your Activities view". It contains two sections: "Open Activities" and "Activity History". Both sections show a list of activities with checkboxes for selecting them. At the bottom of the modal, there is a message: "You can choose to keep track of current and past tasks and events in the Open Activities and Activity History related lists or the Activity timeline." There are also "Tell Me More" and "Show Me How" buttons.

Library Management : Phase 3

Introduction

Define and implement the data structure of Library Connect to link Members, Book Loans and Books. Configure record types, page layouts, navigation tabs and security so that each role (Admin, Librarian, Member) can perform its tasks.

1. Standard & Custom Objects

- ❖ Contact
 - Used to store student information such as Date of Birth, Roll Number/Student ID, Contact Details.
- ❖ Teacher
 - Captures teacher information and tracks their classes/sessions. Includes fields for Subjects Taught, Qualifications, Availability/Timings, and lookup to Student (Contact) for assigned students or mentoring relationships.
- ❖ Student
 - Tracks student academic cases like Performance, Issues, or Counseling. Includes fields for Description, Plan/Action, Status, and lookups to Student (Contact) and Teacher (User).

2. Fields

Student – Stores key student details (name, DOB, ID, contact, class, status) and links them to book loans, courses, and mentors to track borrowing and enrollment.

Teacher – Stores main teacher details (name, DOB, ID, subject, contact, status) and links them to students, classes, and book loans to show who teaches or approves activities.

SETUP > OBJECT MANAGER

Student

Details

Fields & Relationships

4 Items, Sorted by Field Label

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)		✓
Student Name	Name	Text(80)		✓

Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Restriction Rules
Scoping Rules
Object Access
Triggers

Student

SETUP > OBJECT MANAGER

Teacher

Details

Fields & Relationships

10 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Contact	Contact__c	Lookup(Contact)		✓
Created By	CreatedById	Lookup(User)		
Date Of Birth	DateOfBirth__c	Date		
Email	Email__c	Email (Unique)		✓
Employee ID	Employee_ID__c	Text(10)		
Joining Date	JoiningDate__c	Date		
Last Modified By	LastModifiedById	Lookup(User)		
Mentor	Mentor__c	Master-Detail(Mentor)		✓
Subject Specialization	Subject_Specialization__c	Picklist		
Teacher Name	Name	Text(80)		✓

Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules

Teacher

3. Record Types

- Teacher record types: Full-Time Teacher and Part-Time Teacher
- Separate page layouts exist for each record type to show relevant fields.

The screenshot shows the Salesforce Object Manager interface for the 'Teacher' object. The left sidebar has 'Record Types' selected under the 'Page Layouts' category. The main area displays a table titled 'Record Types' with two items: 'Full-Time Teacher' and 'Part-Time Teacher'. The table includes columns for 'Record Type Label', 'Description', 'Active', and 'Modified By'. Both records were modified by Priyanka Pillai on 9/21/2025 at 6:33 AM and 6:36 AM respectively.

Record Type Label	Description	Active	Modified By
Full-Time Teacher		✓	Priyanka Pillai, 9/21/2025, 6:33 AM
Part-Time Teacher		✓	Priyanka Pillai, 9/21/2025, 6:36 AM

Teacher Record Types

The screenshot shows the Salesforce Object Manager interface for the 'Teacher' object. The left sidebar has 'Page Layouts' selected under the 'Page Layouts' category. The main area displays a table titled 'Page Layouts' with three items: 'Full-Time Teacher Layout', 'Part-Time Teacher Layout', and 'Teacher Layout'. The table includes columns for 'Page Layout Name', 'Created By', and 'Modified By'. All layouts were created and modified by Priyanka Pillai on 9/21/2025 at various times between 6:54 AM and 5:21 AM.

Page Layout Name	Created By	Modified By
Full-Time Teacher Layout	Priyanka Pillai, 9/21/2025, 6:54 AM	Priyanka Pillai, 9/21/2025, 6:54 AM
Part-Time Teacher Layout	Priyanka Pillai, 9/21/2025, 6:55 AM	Priyanka Pillai, 9/21/2025, 6:55 AM
Teacher Layout	Priyanka Pillai, 9/21/2025, 5:08 AM	Priyanka Pillai, 9/21/2025, 5:21 AM

Teacher page layout

4. Page Layouts & Related Lists

SETUP > OBJECT MANAGER
Student

Save ▾ Quick Save Preview As... ▾ Cancel Undo Redo Layout Properties

Quick Find Related List Name *

Activity History Groups Student History

Approval History Notes & Attachments

Content Deliveries Open Activities

Files Related Content

Mobile Cards (Salesforce mobile only) Drag expanded lookups and mobile-enabled Visualforce pages here to display them as mobile cards.

Related Lists

Open Activities

Subject	Name	Task	Due Date	Status	Priority	Assigned To
Sample Text	Sample Text	✓	9/21/2025, 7:03 AM	Sample Text	Sample Text	Sarah Sample

Activity History

Subject	Name	Task	Due Date	Assigned To	Last Modified Date/Time
Sample Text	Sample Text	✓	9/21/2025, 7:03 AM	Sarah Sample	9/21/2025, 7:03 AM

5. Lightning Record Pages & Compact Layouts

SETUP

Lightning App Builder

The Lightning App Builder provides an easy to use graphical interface for creating custom Lightning pages for Salesforce Lightning Experience and mobile app. Lightning pages are built using Lightning components—compact, configurable, and reusable elements that you can drag and drop into regions of the page in the Lightning App Builder.

View: All ▾ Create New View

Lightning Pages

Action	Label	Name	Namespace Prefix	Description	Type	Created By	Last Modified By
Edit Clone Del	Home_Page_Default	Home_Page_Default			Home Page	CSE_ 8/14/2025, 6:07 AM	CSE_ 8/14/2025, 6:07 AM
Edit Clone Del	Home_Page_Default1	Home_Page_Default1			Home Page	CSE_ 8/14/2025, 6:21 AM	CSE_ 8/14/2025, 6:25 AM
Edit Clone Del	Library	Library			Record Page	CSE_ 9/21/2025, 7:44 AM	CSE_ 9/21/2025, 7:44 AM
Edit Clone Del	Teacher_Record_Page_Full_Time	Teacher_Record_Page_Full_Time			Record Page	CSE_ 9/21/2025, 7:35 AM	CSE_ 9/21/2025, 7:35 AM

➤ Library

- **Custom Lightning Record Pages created for Library to give Librarians and Students tailored views.**
- **Compact Layouts show key Library fields (Book Title, Author, Availability, Genre) for quick glance.**

The screenshot shows the Salesforce Lightning App Builder interface. On the left is a sidebar with navigation links like Setup Home, Service Setup Assistant, and Administration. The main area is titled "Lightning App Builder" and shows a "Lightning Page Detail" for "Library_Record_Page_Full_Time". The page has sections for "Information" (Name: Library_Record_Page_Full_Time, Label: Library Record Page - Full Time), "Assignments By App" (No assignments), and "Assignments By App, Record Type, and Profile". This section lists assignments for various profiles (Identity User, Librarian, Standard User, System Administrator, Teacher, Test Profile) across different apps (Library Management). At the bottom right of the main area, there's a note: "Always show me fewer ▲ / more records per related list".

➤ Teacher

- **Custom Lightning Record Pages created for Teacher to give Admins and Staff tailored views.**
- **Compact Layouts show key Teacher fields (Name, Subject, Class, Contact) for quick glance.**

This screenshot is identical to the one above, showing the "Lightning App Builder" interface for a "Teacher_Record_Page_Full_Time" page. It includes the same sections: "Information" (Name: Teacher_Record_Page_Full_Time, Label: Teacher Record Page - Full-Time), "Assignments By App" (No assignments), and "Assignments By App, Record Type, and Profile". The "Assignments By App, Record Type, and Profile" table lists assignments for profiles (Librarian, Standard User, System Administrator, Teacher, Test Profile) across various apps (Library Management) for the "Full-Time Teacher" record type. The bottom right of the main area also features the note: "Always show me fewer ▲ / more records per related list".

6. Tabs & App Navigation

- Teacher
- **Tabs & Navigation: Teachers, Students, Classes, Attendance, Reports. Custom tab for Teachers.**
- Library
- **Tabs & Navigation: Library, Books, Students, Issued Books, Reports. Custom tab for Library.**

The screenshot shows the Salesforce Setup interface. The left sidebar has a tree view with 'Feature Settings' expanded, showing 'Analytics' and 'Tableau' (with 'Tableau Embedding' and 'Tableau UAF Claims Definition' children). Under 'User Interface', 'Console Settings' and 'Rename Tabs and Labels' are listed, with 'Tabs' selected. The main content area is titled 'Custom Tabs' and contains three sections: 'Custom Object Tabs', 'Web Tabs', and 'Visualforce Tabs'. The 'Custom Object Tabs' section lists four tabs: 'Libraries' (purple), 'Mentors' (green), 'Students' (blue), and 'Teachers' (red). Each tab has an 'Edit | Del' link and a 'Tab Style' icon.

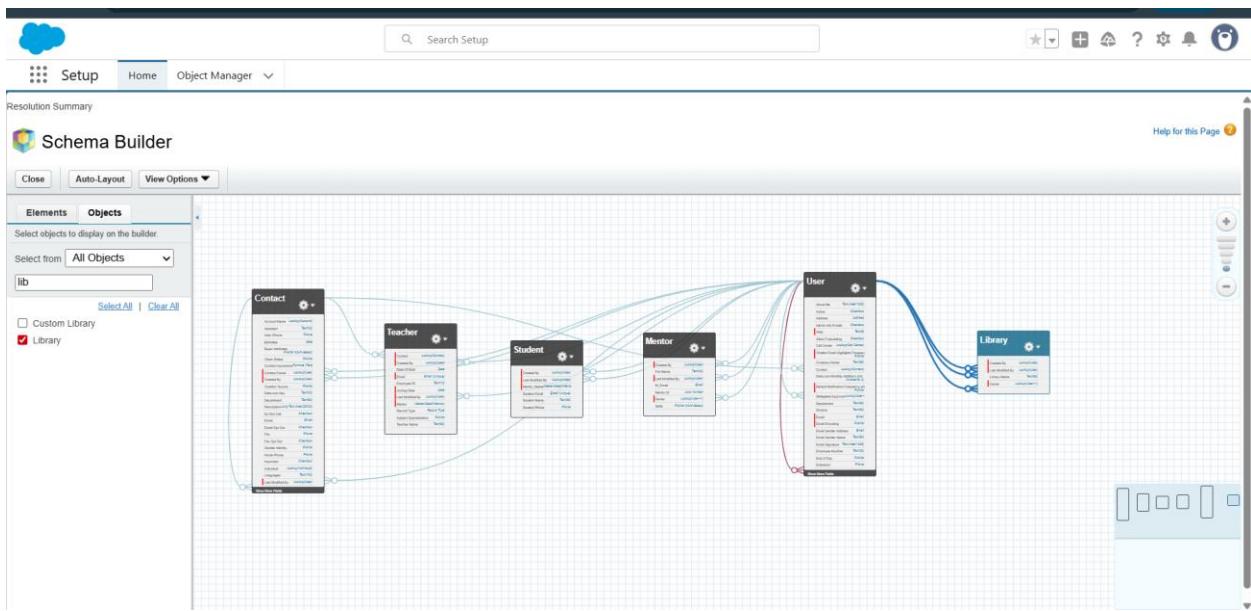
7. Schema Builder (ERD)

Teacher

Schema Builder (ERD): Displays Teacher, Student, Class, and User objects with lookup relationships. Teacher is central, linked to Classes and Students.

Library

Schema Builder (ERD): Displays Library, Book, Student, and User objects with lookup relationships. Library is central, linked to Books and Students.



8. Security & Field-Level Visibility :

- Teacher profile has Read/Create/Edit on Student and Class records.
- Librarian profile has Read/Create/Edit on Library and Book records, but read-only on Student records.
- Sensitive fields like Salary or Private Notes hidden from Librarian profile.
- Mentor profile can view assigned Students and Classes but cannot edit Library records.

9. Sample Data & Testing :

- Sample Students and Teachers created with details like Name, Class, and Contact info.
- Library records (Books) created by Librarian, linked to Students and Teachers with record type “Issued” or “Available.”
- Mentor records created and linked to Students for guidance tracking.
- Teacher login view shows Classes and Students; Student record shows related lists (Books issued, Mentor info).

10. Current System Behaviour :

- Library records link both Students and Teachers for issuing/tracking books.
- Mentors hold guidance/notes information linked to the same Students.
- Record types like Issued Book and Available Book display appropriate layouts.
- Student records show all related Classes, Library records, and Mentor assignments.
- Teachers can view Students and Classes assigned to them.
- Librarians can create Library records and view limited Student info.
- The Schema Builder ERD reflects the current relationships between Teacher, Library, Mentor, Student, and User.

Library Management:- Phase 4: Process Automation (Admin)

1. Validation Rules

- A validation rule created on Teacher object HireDate_Not_Future Date/ Time Hire Date cannot be blank or in the future.
- A validation rule created on Teacher object Email_Bank Please enter a valid email address.

SETUP > OBJECT MANAGER
Teacher

Validation Rules

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Email_Bank	Email	Please enter a valid email address.	✓	Priyanka Pillai, 9/22/2025, 10:39 AM
HireDate_Not_Future	Hire Date	Hire Date cannot be blank or in the future.	✓	Priyanka Pillai, 9/22/2025, 9:34 AM

Validation Rules

Employee ID: [redacted]
Subject Specialization: Mathematics
*Email: e@gmail.com
*Joining Date: 9/7/2017
*Contact: Rakesh sharma
*Mentor: M.ID-001235
Hire Date: [redacted]

We hit a snag.

Review the following fields

- Hire Date

Cancel Save & New Save

Validation Rules Testing

2. Email Templates for Automation

Two classic email templates were created and stored under Private Email Templates:

Teacher_HireDate_Confirmation : – sends confirmation to the teacher when a date is created or confirmed.

The screenshot shows the 'Teacher_HireDate_Confirmation' email template in a software application. The interface includes a top navigation bar with various menu items like Home, Accounts, Contacts, Custom Libraries, Calendar, Dashboards, Change Requests, Students, and more. Below the navigation is a toolbar with icons for Edit, Clone, Delete, and other actions. The main content area is titled 'Email Template Teacher_HireDate_Confirmation'. It has two tabs: 'Details' (selected) and 'Related'. The 'Information' section contains fields for Email Template Name (Teacher_HireDate_Confirmation), Description, Related Entity Type (Teacher), Folder (Private Email Templates), and a note that it was made in Email Template Builder. The 'Message Content' section includes a Subject field with the value 'Confirmation: Record Created for {{Teacher_c.Name}}', an Enhanced Letterhead placeholder, and an HTML Value field containing a welcome message and a note about successful creation. There are also sections for 'If you have any questions, please contact us.' and 'Thank you, Admin Team'.

Teacher_HireDate_Confirmation

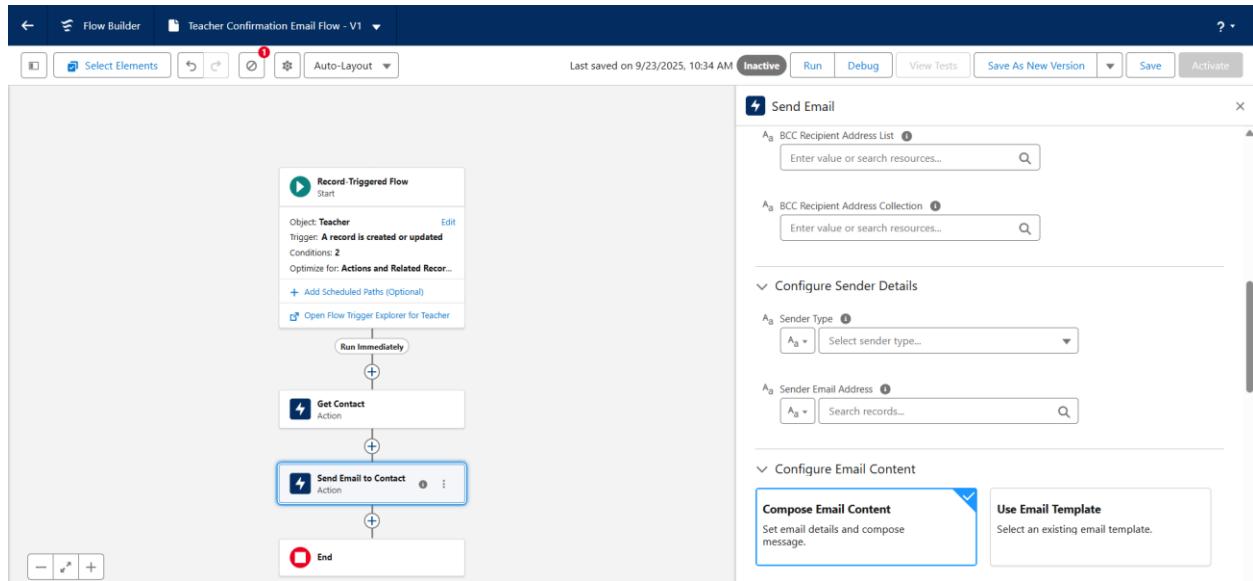
Teacher_HireDate_Reminder :-

The screenshot shows the 'Teacher_HireDate_Reminder' email template in a software application. The interface is similar to the previous one, with a top navigation bar and a toolbar. The main content area is titled 'Email Template Teacher_HireDate_Reminder'. It has 'Details' and 'Related' tabs. The 'Information' section contains fields for Email Template Name (Teacher_HireDate_Reminder), Description, Related Entity Type (Teacher), Folder (Private Email Templates), and a note that it was made in Email Template Builder. The 'Message Content' section includes a Subject field with the value 'Reminder: Your Hire Date is {{Teacher_c.Hire_Date_c}}', an Enhanced Letterhead placeholder, and an HTML Value field containing a reminder message and a note about the hire date. There are also sections for 'Thank you, HR Team' and an 'Additional Information' section at the bottom.

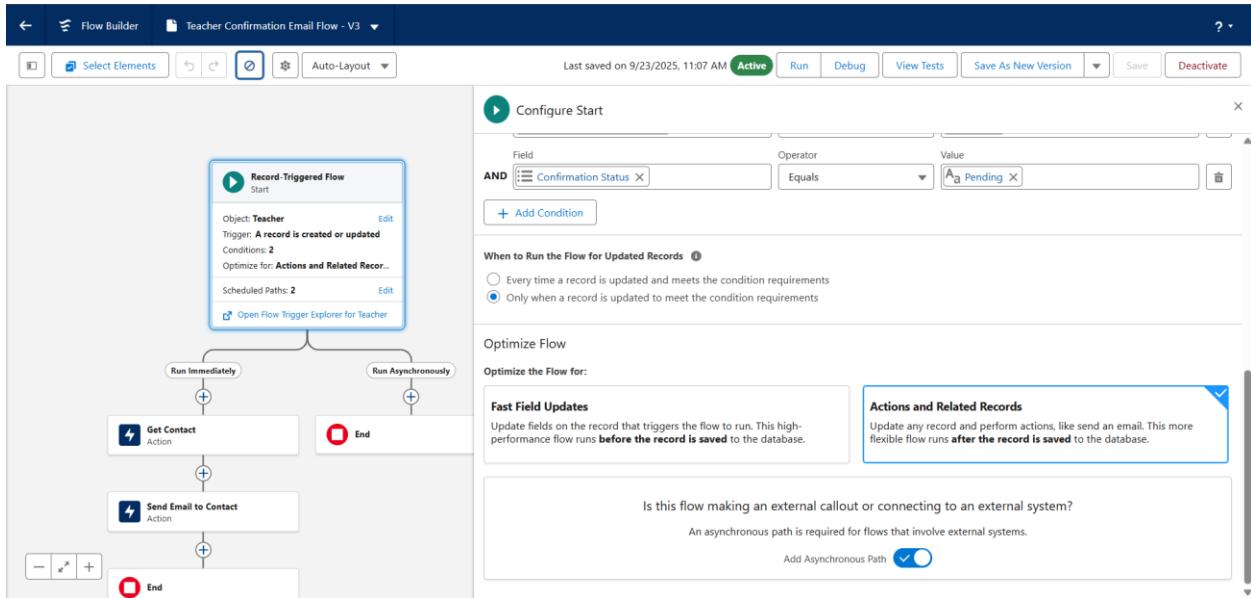
Teacher_HireDate_Reminder

3. Flow for TeacherConfirmation :-

- A record-triggered flow “Teacher_Confirmation_Flow” built on Teacher object:
- Trigger: When record is created or when Status changes to “Confirmed.”
- Action:
- Update “Confirmation Sent” checkbox on the Teacher.



4. Reminder Notification & Email (Scheduled Path)



6. Quick Action – “Mark as Completed”

A Quick Action added Teacher object:

- Action Type: Update a Record.**
- Predefined Field Values: Status c = “Completed.”**

Action	Field Name	API Name	Field Type	Value
Confirmation Status	Confirmation_Status_c	Confirmation_Status_c	Picklist	Email Sent

Mark as Completed

8. Testing & Results

Sample Appointments were created to test the automation:

Library Management :- Phase 5: Apex Programming (Developer)

➤ Classes & Objects:

- Created an Apex utility class **TeacherService**:
- Inner Comparator: **TeacherHireDateComparator** – sorts teachers by **Hire_Date__c**.

➤ Static Method:

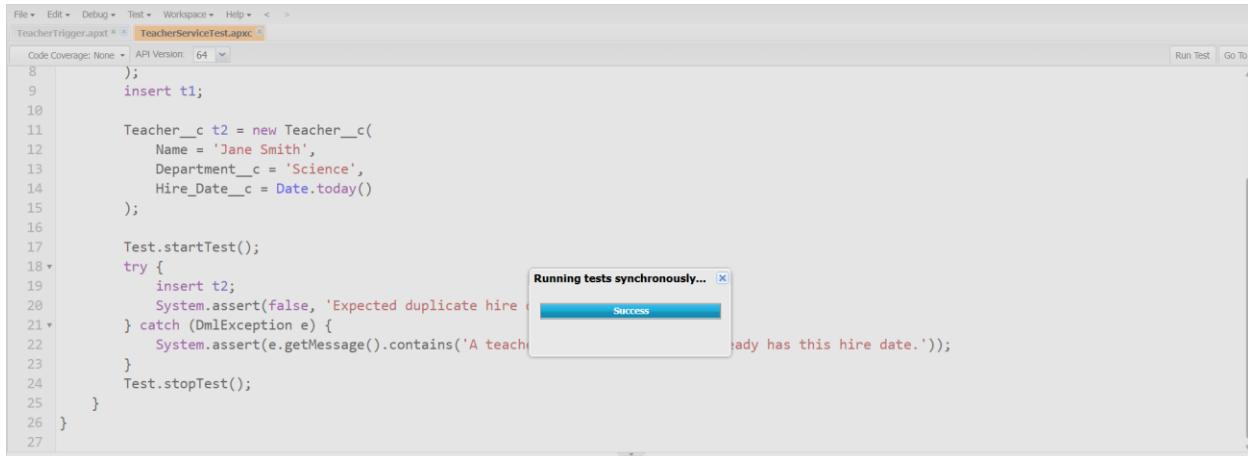
- **preventDuplicateHireDates(List<Teacher_c> incoming, Map<Id, Teacher_c> oldMap)** – checks new/updated teacher records against existing teacher records to ensure no conflicting hire dates for the same department (or other criteria) and uses **addError()** to block duplicates.

File → New → Apex Class.

Name it TeacherServiceTest → click OK.

Paste test code:

Save.



```
File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < ▾ >
TeacherTrigger.apxt ▾ TeacherServiceTest.apxt
Code Coverage: None ▾ API Version: 64 ▾ Run Test Go To
8     );
9     insert t1;
10
11    Teacher_c t2 = new Teacher_c(
12        Name = 'Jane Smith',
13        Department__c = 'Science',
14        Hire_Date__c = Date.today()
15    );
16
17    Test.startTest();
18    try {
19        insert t2;
20        System.assert(false, 'Expected duplicate hire date error');
21    } catch (DmlException e) {
22        System.assert(e.getMessage().contains('A teacher already has this hire date.'));
23    }
24    Test.stopTest();
25
26 }
```

2.Apex Trigger

File → New → Apex Trigger.

Enter:

- Name: TeacherTrigger
- sObject: Teacher__c

Click Submit.

Save.

The screenshot shows the Salesforce IDE interface. At the top, there's a menu bar with File, Edit, Debug, Test, Workspace, Help, and a Go To button. Below the menu is a toolbar with Code Coverage: None, API Version: 64, and a dropdown arrow. The main area contains the code for TeacherTrigger.apxt:

```
1 trigger TeacherTrigger on Teacher__c (before insert) {
! 2 trigger TeacherTrigger on Teacher__c (before insert, before update) {
3 if (Trigger.isBefore && (Trigger.isInsert || Trigger.isUpdate)) {
4     TeacherService.preventDuplicateHireDates(Trigger.new, Trigger.oldMap);
5 }
6 }
7
8 }
```

Below the code editor is a navigation bar with Logs, Tests, Checkpoints, Query Editor, View State, Progress, and Problems. The Problems tab is selected, showing one error:

Name	Line	Problem
TeacherTrigger	2	Expecting ')' but was: 'trigger'

3.Trigger Design Pattern:

Business logic is kept in the TeacherService class; the trigger on Teacher__c simply calls the class method.

4. SOQL & Collections

Inside preventOverlap:

- SOQL queries existing Teacher for relevant new Teacher.

```

1  @isTest
2  public class TeacherServiceHandleTest {
3
4      @isTest
5      static void testPreventDuplicateHireDates() {
6          // Insert first teacher
7          Teacher_c t1 = new Teacher_c(
8              Name = 'John Doe',
9              Department__c = 'Science',
10             Hire_Date__c = Date.today()
11         );
12         insert t1;
13
14         // Insert second teacher with same hire date in same department
15         Teacher_c t2 = new Teacher_c(
16             Name = 'Jane Smith',
17             Department__c = 'Science',
18             Hire_Date__c = Date.today()
19         );
20
21         Test.startTest();
22         try {
23             insert t2;
24             System.assert(false, 'Expected duplicate hire date error.');
25         } catch (DmlException e) {
26             System.assert(e.getMessage().contains('A teacher in this department already has this hire date.'));
27         }
}

```

5. Control Statements

for loops, if conditions and addError() to prevent overlaps.

6. Test Classes (Teacher Example)

A test class **TeacherTriggerTest** was created to:

- Insert a valid teacher (no duplicate hire date in the same department) and verify it saves successfully.
- Attempt to insert a teacher with a duplicate hire date in the same department and verify a **DmlException** with “already has this hire date” in the message is thrown.

This test indirectly executes the logic in

TeacherServiceHandle.preventDuplicateHireDates and gives code coverage for both the trigger (**TeacherTrigger**) and the service class (**TeacherServiceHandle**).

```

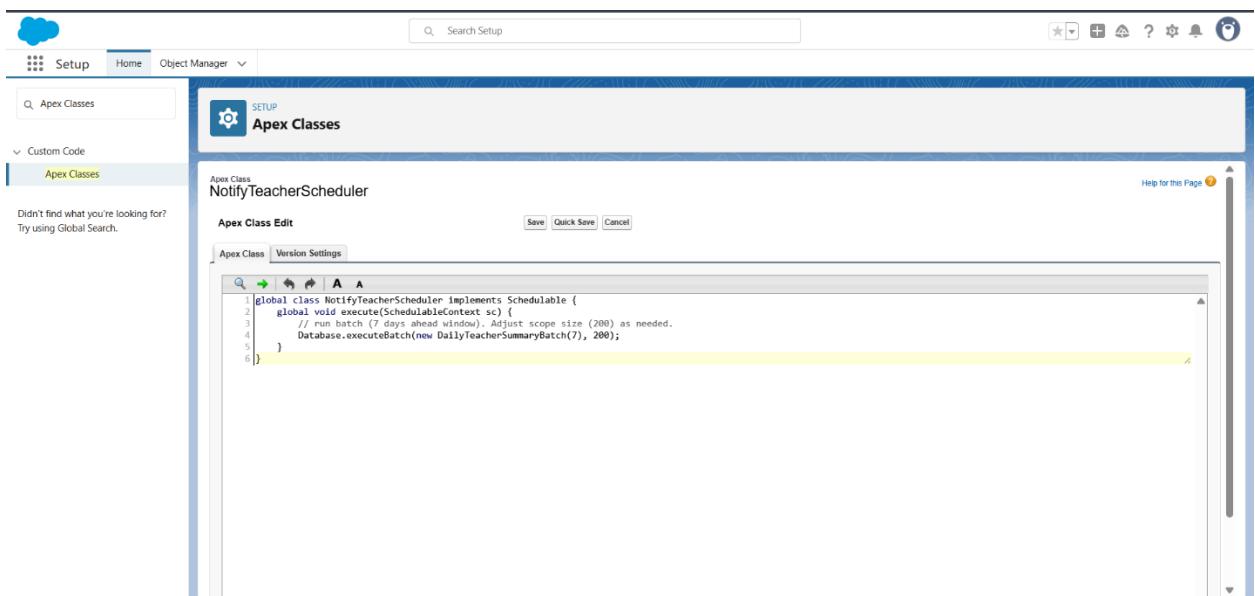
1 @isTest
2 public class TeacherTriggerTest {
3
4     @isTest
5     static void validateDuplicateHireDatePrevention() {
6         // Insert first teacher (valid record)
7         Teacher_c firstTeacher = new Teacher_c(
8             Name = 'Alice Johnson',
9             Department__c = 'Mathematics',
10            Hire_Date__c = Date.today()
11        );
12        insert firstTeacher;
13
14        // Prepare second teacher with same department and hire date (should fail)
15        Teacher_c duplicateTeacher = new Teacher_c(
16            Name = 'Bob Smith',
17            Department__c = 'Mathematics',
18            Hire_Date__c = Date.today()
19        );
20
21        Test.startTest();
22        Boolean exceptionThrown = false;
23        try {
24            insert duplicateTeacher;
25        } catch (DmlException e) {
26            exceptionThrown = true;
27            System.assert(

```

7. Batch Apex and Scheduled Jobs

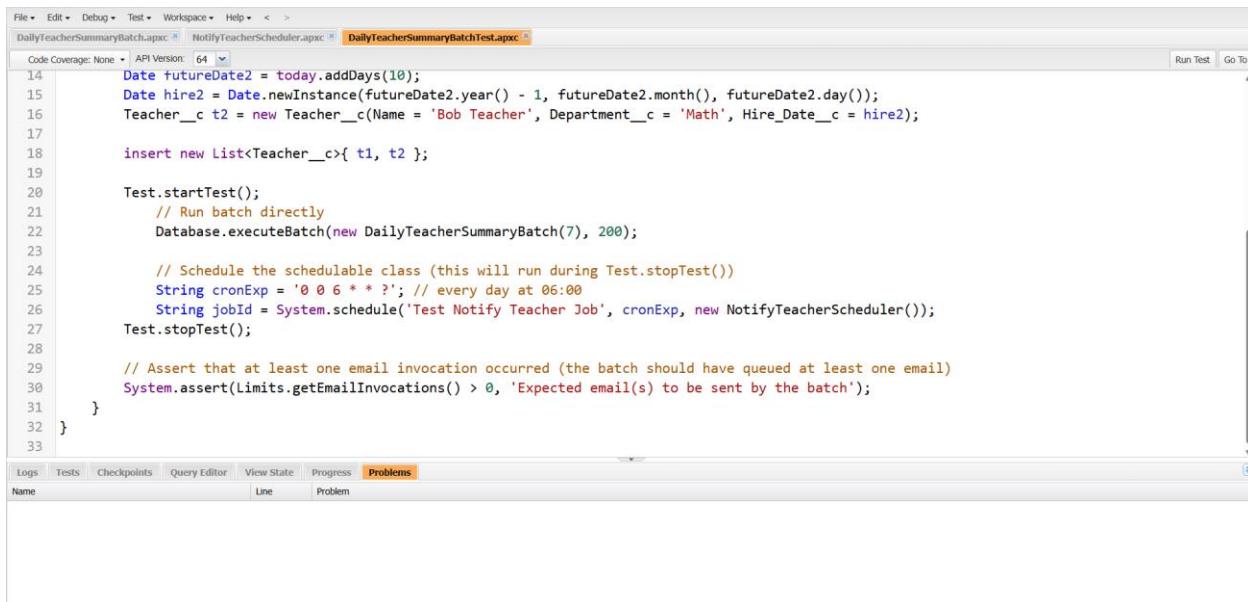
A Batch Apex class TeacherSummaryBatch was created to:

- Query recently hired teachers by department.
- Generate and log summary reports (or prepare reminders for admin review).
- Run daily at 6:00 AM using a Scheduled Apex job.



8.Exception Handling

Used **addError()** on SObjects to stop DML with a user-friendly message.

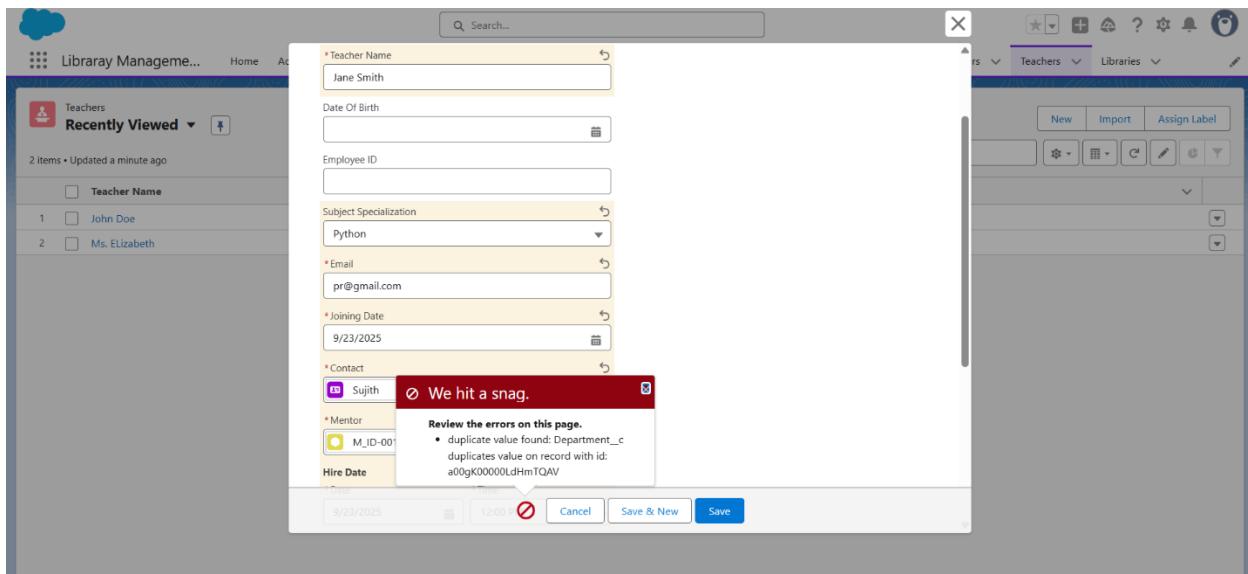


```
File -> Edit -> Debug -> Test -> Workspace -> Help -> <- >
DailyTeacherSummaryBatch.apxc NotifyTeacherScheduler.apxc DailyTeacherSummaryBatchTest.apxc
Code Coverage: None API Version: 64
14 Date futureDate2 = today.addDays(10);
15 Date hire2 = Date.newInstance(futureDate2.year() - 1, futureDate2.month(), futureDate2.day());
16 Teacher__c t2 = new Teacher__c(Name = 'Bob Teacher', Department__c = 'Math', Hire_Date__c = hire2);
17
18 insert new List<Teacher__c>{ t1, t2 };
19
20 Test.startTest();
21     // Run batch directly
22     Database.executeBatch(new DailyTeacherSummaryBatch(7), 200);
23
24     // Schedule the schedulable class (this will run during Test.stopTest())
25     String cronExp = '@ 0 6 * * ?'; // every day at 06:00
26     String jobId = System.schedule('Test Notify Teacher Job', cronExp, new NotifyTeacherScheduler());
27     Test.stopTest();
28
29     // Assert that at least one email invocation occurred (the batch should have queued at least one email)
30     System.assert(Limits.getEmailInvocations() > 0, 'Expected email(s) to be sent by the batch');
31 }
32 }
33 }
```

9.Skipped / Not Implemented

Queueable Apex, Future Methods, Asynchronous Processing beyond the daily batch job were not implemented at this phase.

10. Testing & Results



Library Management : Phase 6: User Interface Development

Introduction

This phase focuses on building an intuitive Lightning experience for different library roles (Teacher, Student, Librarian/Admin).

Using Lightning App Builder, Record Pages, Tabs, Home Page Layouts, Utility Bar, plus a custom LWC and Apex integration, a seamless UI was delivered to every user type.

1. Lightning App Builder

- Customized Pages for Each Role: Different Lightning Record Pages were designed for Book/Resource – Teacher View and Book/Resource – Student View (and optionally Librarian/Admin View).

Role-Based Components: Each page displays only the components relevant to that specific role, e.g.

- Teacher: can upload resources, create assignments, view students' borrowing history.
- Student: can search books, borrow/return, view personal borrowing history.
- Librarian/Admin: can add/edit/delete books, manage stock, approve requests.

- Tabs & Home Layouts: Tabs and Home Page Layouts were set up separately for Teacher, Student, and Librarian/Admin so each profile lands on the correct workspace.

Utility Bar: Added quick-access actions and shortcuts in the Utility Bar for commonly used features by each role, e.g.

Librarian/Admin: "Add New Book," "View Overdue Items," "Send Notifications."

The screenshot shows the Salesforce Lightning App Builder interface. At the top, there's a navigation bar with icons for Home, Object Manager, and a search bar labeled "Search Setup". Below the navigation is a sidebar with links like Setup Home, Salesforce Go, Service Setup Assistant, Commerce Setup Assistant, Field Service Setup Home (Beta), Hyperforce Assistant, Release Updates, Salesforce Mobile App, Lightning Usage, Optimizer, and Sales Cloud Everywhere. Under Administration, there are links for Users, Data, and Email. Under Platform Tools, there's a link for Subscription Management. The main content area is titled "Lightning App Builder" and contains a sub-section titled "Lightning Pages". It includes a brief description: "The Lightning App Builder provides an easy to use graphical interface for creating custom Lightning pages for Salesforce Lightning Experience and mobile app. Lightning pages are built using Lightning components—compact, configurable, and reusable elements that you can drag and drop into regions of the page in the Lightning App Builder." Below this is a table titled "Lightning Pages" with columns: Action, Label +, Name, Namespace Prefix, Description, Type, Created By, and Last Modified By. The table lists several pages, including "Full_Time_Teacher", "Home_Page_Default", "Home_Page_Default1", "Library_Home", "Library_Record_Page_Full_Time", and "Teacher_Record_Page_Full_Time". Each row shows the action links (Edit | Clone | Del), the label, the name, the namespace prefix (e.g., "Full_Time_"), the type (Record Page or Home Page), and the details of the creator (CSE) and last modifier (CSE).

Action	Label +	Name	Namespace Prefix	Description	Type	Created By	Last Modified By
Edit Clone Del	Full-Time Teacher	Full_Time_Teacher			Record Page	CSE, 9/22/2025, 3:32 AM	CSE, 9/22/2025, 3:33 AM
Edit Clone Del	Home_Page_Default	Home_Page_Default			Home Page	CSE, 8/14/2025, 6:07 AM	CSE, 8/14/2025, 6:07 AM
Edit Clone Del	Home_Page_Default1	Home_Page_Default1			Home Page	CSE, 8/14/2025, 6:21 AM	CSE, 8/14/2025, 6:25 AM
Edit Clone Del	Library_Home	Library_Home			Record Page	CSE, 9/23/2025, 9:42 PM	CSE, 9/23/2025, 9:45 PM
Edit Clone Del	Library_Record_Page_Full_Time	Library_Record_Page_Full_Time			Record Page	CSE, 9/21/2025, 7:44 AM	CSE, 9/22/2025, 3:44 AM
Edit Clone Del	Teacher_Record_Page_Full_Time	Teacher_Record_Page_Full_Time			Record Page	CSE, 9/21/2025, 7:35 AM	CSE, 9/21/2025, 7:35 AM

2. Record Pages (Teacher & Student Views)

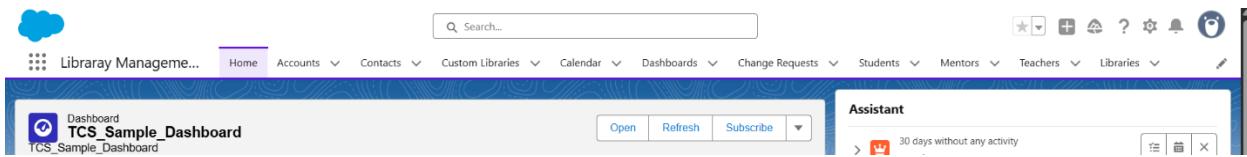
- Teachers need quick access to class resources and student borrowing history. Students need an easy way to browse and request library materials. Two record pages were created with appropriate components:
- Library Resource – Teacher View: shows resource summary, related student borrow/return history, and an upcoming assignments/resources LWC.
- Library Resource – Student View: optimized for quick browsing, book request/renewal, and viewing personal borrowing history.
- Activated per profile/record type so each user sees the correct page automatically.

Library(Record Page)

Teacher(Record Page)

3. Tabs

All major library objects (**Books/Resources, Borrow Requests, Students/Teachers, Classes**) were exposed as tabs in the **Library Management Lightning App** so Teachers, Students, and Librarians can navigate quickly to the areas they need.



Tabs in App

4. Home Page Layouts

A custom **Library Management Home Page** was built that displays dashboard components such as **Upcoming Book Returns**, **Recent Borrow Requests**, **Popular Books**, and **Overdue Items Report Chart** to give Teachers, Students, and Librarians a real-time view of library activity.

Library Home Page Layout

Phase 8: Library Management Data Management & Deployment

1. Create the Matching Rule

- **Setup → Quick Find → Matching Rules → New Rule**
- **Object:** Teacher
- **Rule Name:** LibraryMember_Email_Phone_Match
- **Add Matching Criteria:**
 - Field = **Email** → Matching Method = **Exact**

- Click **Add Row** → Field = **Phone** → Matching Method = **Exact**
- Click **Save**
- Click **Activate**

The screenshot shows the Salesforce Setup interface with the 'Matching Rules' page open. The page title is 'Matching Rules' and the specific rule is 'LibraryMember_Email_Phone_Match'. The rule details are as follows:

Matching Rule Detail	Object	Rule Name	Description	Status	Created By	Modified By
	Teacher	LibraryMember_Email_Phone_Match	(Teacher: Name EXACT MatchBlank = FALSE) AND (Teacher: Contact EXACT MatchBlank = FALSE) AND (Teacher: Email EXACT MatchBlank = FALSE)	Active	Priyanka Pillai, 9/24/2025, 8:44 AM	Priyanka Pillai, 9/24/2025, 8:45 AM

2) Create the Duplicate Rule

- **Setup** → **Quick Find** → **Duplicate Rules** → **New Rule**
- **Object: Teacher**
- **Rule Label: LibraryMember_DuplicateRule**
- **Under Matching Rules:** click **Add** and select the **LibraryMember_Email_Phone_Match** matching rule you activated in the previous step.
- **Action on Create:** choose **Alert** (recommended while testing so users are warned but not blocked).
- **Action on Edit:** choose **Alert**.

Step 2: Data Backup – Library Management

Goal: Regularly back up all key Library Management data (Teachers, Students, Books, Borrow Requests, etc.) for safety and compliance.

Steps

- **Go to Setup.**
- In **Quick Find**, type **Data Export** → click **Data Export**.
- **Choose one:**
 - **Export Now** → run a one-time backup.
 - **Schedule Export** → set weekly or monthly backups.
- **Select the objects you want:**
 - Library Members (Teachers/Students)
 - Books/Resources
 - Borrow Requests / Returns
 - Classes / Courses

- Librarians (Users)
- Any other standard objects you use (e.g., Users, Attachments).
- Click **Start Export** (for immediate) or **Save** (for scheduled).
- **Wait** — Salesforce will email you when the backup is ready.
- **Download** the .zip file from the export page → extract CSV files.
- **Store the backup securely** (encrypted drive, secure server, or trusted cloud storage).

The screenshot shows the Salesforce Data Export page. At the top, there's a navigation bar with links like Home, Chatter, Campaigns, Leads, Accounts, Contacts, Opportunities, Forecasts, Contracts, Orders, Cases, Solutions, Products, Reports, Dashboards, Students, Teachers, Libraries, and Sales. On the left, a sidebar titled "Salesforce Mobile Quick Start" lists various administrative and data management options under "Administer". The main content area displays a message about Data Export, stating it lets users prepare a copy of all their data. It shows a scheduled export entry for "Priyanka Pillai" on "9/24/2025" with the file encoding set to ISO-8859-1. A table below lists the export details:

Action	File Name	File Size
download	WE_00DgK000007DvBUAU_1.ZIP	2.0K

Your Organization Data Export has completed - Gyan Ganga College Of Technology External Inbox ×

 **Do not reply** <noreply@salesforce.com>
to me ▾ 23:29 (5 minutes ago) Star Forward More

The export of your organization's data has been completed. Please click on the following link within the next 48 hours to receive the export.

<https://orgfarm-a8bfd7c69d-dev-ed.my.salesforce.com/ui/setup/export/DataExportPage/d>

Thank you,
Salesforce

Reply Forward

Library Management Phase 9:

Reporting, Dashboards & Security Review

Step 1: Reporting

A **Report** is a list or summary of records in Salesforce that helps you analyze library data.

- Reports let you **filter, sort, group, and summarize** your Library Management data.
 - **Example:** You can create a report to see all books borrowed this month or the top books borrowed by Teachers vs. Students.
-

Steps

- Go to the App Launcher → search “Reports” → click Reports.
- Click “New Report” - Monthly Borrowed Books.
- Select the Report Type - Teacher
- Click Continue.
- Add Filters
- Add Columns
- Group Data (optional)
 - Example: Group Borrow Requests by Book or by Member (Teacher/Student).
- Summarize (optional)
 - Add totals, averages, or counts (like total borrowed books per Teacher).
- Click Save & Run → Give the report a name → Choose a folder to save it in.

Step 2: Dashboards

What it is:

A Dashboard is a visual board that displays charts, numbers, and tables built from your Reports. It helps Teachers, Librarians and Management instantly see performance trends (for example: most borrowed books, top students, pending returns).

Dashboard Components You Can Add

- **Charts** – Bar, Pie, Line charts to show trends.
 - **Gauges** – Show progress toward a target (e.g. % of books returned on time).
 - **Metrics** – One key number (e.g. total borrowings this month).
 - **Tables** – Raw report data in a grid.
-

Steps (Click by Click)

- **Go to the App Launcher → type “Dashboards” → click Dashboards.**
- **Click “New Dashboard”.**
- **Enter Dashboard Name** – e.g. Library Performance Dashboard or Teacher Engagement Dashboard.
- **Choose a Folder** (select the folder you created earlier for Reports/Dashboards).

- **Click “Create”.**
- **Click “+ Component”** (to add your first chart, metric or table).
- **Select a Report** you already built (for example: “Books Borrowed This Month” or “Top Students”).
- **Choose Visualization Type:**
 - Bar Chart
 - Pie Chart
 - Gauge
 - Metric
 - Table

Configure Data Display:

- Choose which fields, groupings, and filters you want for this chart or metric.

Click “Add”.

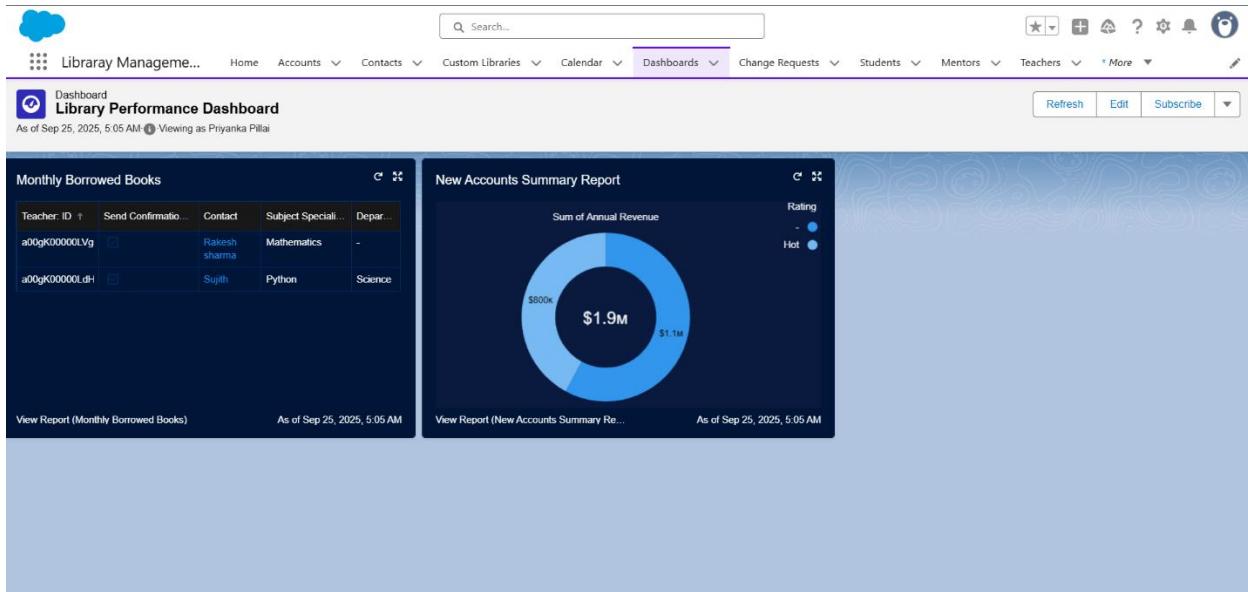
Repeat “+ Component” for more charts/metrics if needed.

Click “Save” → then “Done”.

Click “Refresh” anytime to pull in the latest data.

Result:

Your dashboard now shows live numbers and visuals from your Salesforce reports for Teachers, Students and Librarians.



Step 3: Field-Level Security

What it is:

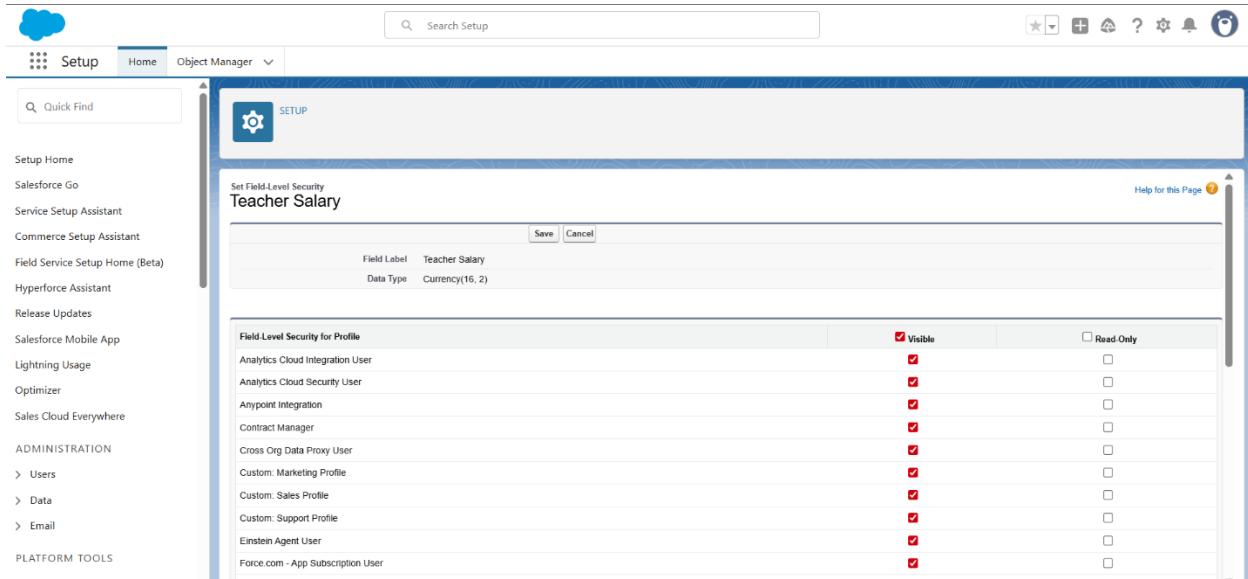
Field-Level Security controls who can see or edit individual fields on a record.

It protects sensitive data like Teacher Contact, Student DOB, Library Fines, Book Costs, etc., so only the right roles can view or update them.

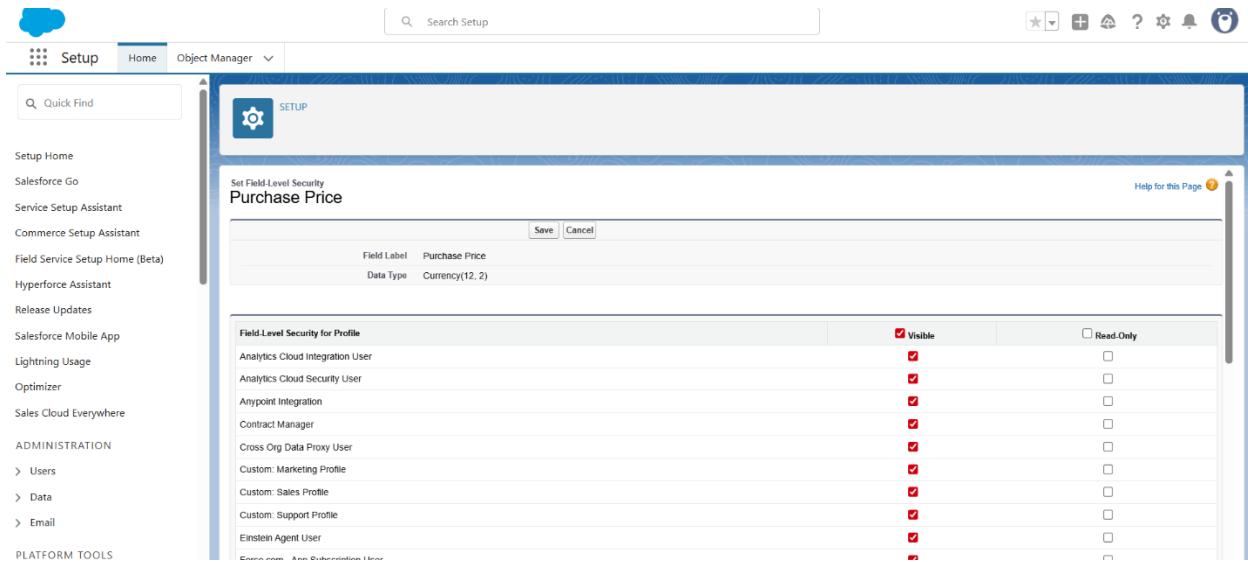
Steps (Click by Click)

- Go to Setup → Object Manager.
- Select your Object (example: Student or Teacher or Book).
- Click “Fields & Relationships”.
- Click the field you want to protect (example: Student DOB).
- Click “Set Field-Level Security”.
- For each Profile:
 - Visible → can see the field.
 - Read-Only → can see but cannot edit.
 - Hidden → cannot see at all.
- (Example: For Student DOB → Visible to Teacher & Librarian profiles, Hidden for Student profile.)
- Click “Save”.

- Repeat steps 4–7 for other sensitive fields like Teacher Salary, Book Purchase Price, Library Fine Amount, etc.
- Result:
 - Only authorized roles (Teacher, Librarian, Admin) can view or edit those sensitive fields. Students or other profiles see nothing or read-only according to your settings.
- Setup → Object Manager → Teacher → Fields & Relationships → New.
- Choose Field Type = Currency → Next.
- Field details:
 - Field Label: Teacher Salary
 - Field Name: Teacher_Salary__c
 - Length / Decimal Places: choose e.g. 16,2 (or your org standard)
 - Help Text: Annual salary in local currency. Visible to Admin only.
 - Description: Stores salary for payroll/audit
 - Click Next.
 - Set Field-Level Security (during creation)
 - On the Set Field-Level Security screen (shown while creating), pick visibility per profile:
 - visible for all
 - Click Next.
 - Add to Page Layouts
 - Choose page layouts where the field appears (usually Admin layouts only) → Save.



- Setup → Object Manager → Student → Fields & Relationships → New.
- Field Type = Currency → Next.
- Field details:
- Field Label: Purchase Price
- Field Name: Purchase_Price__c
- Length/Decimals: e.g., 12,2
- Help Text: Cost price of the book — visible to Librarians & Admins.
- Click Next.
- Set Field-Level Security (during creation)
- Choose visibility: all
- Next → Add to relevant Page Layout(s) → Save



- Setup → Object Manager → Student → Fields & Relationships → New.
- Field Type = Currency → Next.
- Field details:
- Field Label: Fine Amount
- Field Name: Fine_Amount__c
- Help Text: Fine charged for overdue returns — visible to Librarians only.
- Click Next.
- Set Field-Level Security
- Visibility choices: for all
- Next → Add to Page Layout(s) → Save.