Student Admission — Phase 3: Data Modeling & Relationships

1. Introduction
This phase covers designing the **data model** for Student Admission in Salesforce. It defines standard and custom objects, fields, record types, page layouts, and relationships to support admission workflows, student applications, course enrollments, and reporting.

2. Objectives
- Configure **standard & custom objects** for students, applications, courses, and enrollments.
- Add **fields** to capture admission-related information.
- Create **record types** for UG/PG applications.
- Customize **page layouts & compact layouts** for user-friendly data entry.
- Use **Schema Builder** to visualize objects and relationships.
- Understand and apply **Lookup vs Master-Detail vs Hierarchical relationships**.
- Implement a **junction object** for Student ↔ Course (Enrollment).
- Explore **External Objects** for integrating external admission data.

3. Configuration Steps
Step 1 — Standard & Custom Objects
- **Standard Objects** used:
  - Contact → (can represent Students, but we created a custom object for clarity)
  - User → Admin, Officer, Student users
  - Reports & Dashboards → analytics
- **Custom Objects created**:
  - **Student__c** → represents student details.
  - **Application__c** → represents student applications for admission.
  - **Course__c** → represents academic courses offered.
  - **Enrollment__c** → junction object linking Student__c and Course__c.

Account	Account	Standard Object			
Activity	Activity	Standard Object			
Address	Address	Standard Object			
Agent Work	AgentWork	Standard Object			
Alternative Payment Method	AlternativePaymentMethod	Standard Object			
API Anomaly Event Store	ApiAnomalyEventStore	Standard Object			
Application	Application__c	Custom Object	9/18/2025	✓	▼
Appointment Category	AppointmentCategory	Standard Object			
Appointment Invitation	AppointmentInvitation	Standard Object			
Appointment Invitee	AppointmentInvitee	Standard Object			
Appointment Topic Time Slot	AppointmentTopicTimeSlot	Standard Object			
Approval Submission	ApprovalSubmission	Standard Object			

Email Message	EmailMessage	Standard Object			
Engagement Channel Type	EngagementChannelType	Standard Object			
Engagement Channel Work Type	EngagementChannelWorkType	Standard Object			
Enrollment	Enrollment_c__c	Custom Object	9/18/2025	✓	▼
Entitlement	Entitlement	Standard Object			
Entitlement Contact	EntitlementContact	Standard Object			

Store	WebStore	Standard Object			
Store Buyer Group	WebStoreBuyerGroup	Standard Object			
Store Catalog	WebStoreCatalog	Standard Object			
Student	Student_c__c	Custom Object	9/18/2025	✓	▼
Task	Task	Standard Object			
Time Slot	TimeSlot	Standard Object			
User	User	Standard Object			

Contract Line Item	ContractLineItem	Standard Object			
Coupon	Coupon	Standard Object			
Course	Course_c__c	Custom Object	9/18/2025	✓	▼
Credential Stuffing Event Store	CredentialStuffingEventStore	Standard Object			

Fig.Objects

Step 2 — Fields
Student__c Fields
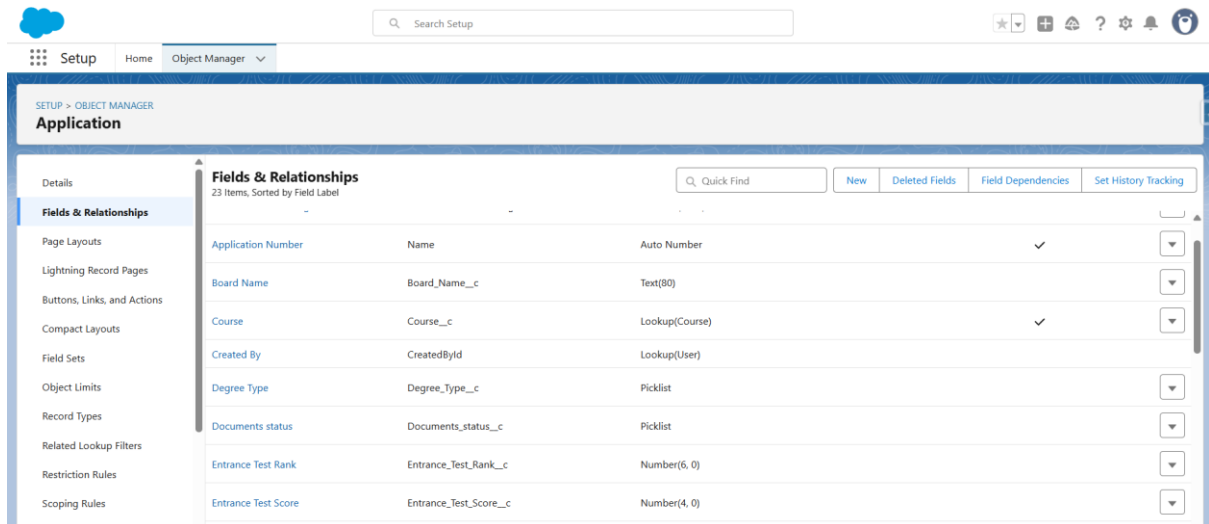- Student_ID__c (Auto Number: STU-{0000})
- Full_Name__c (Text)
- Email__c (Email)
- Phone__c (Phone)
- DOB__c (Date)
- Program__c (Picklist: UG, PG, Diploma)

Date of Birth	Date_of_Birth__c	Date		▼
Email	Email__c	Email		▼
Last Modified By	LastModifiedById	Lookup(User)		
Name	Name__c	Text(50)		▼
Owner	OwnerId	Lookup(User,Group)	✓	
Phone no	Phone_no__c	Phone		▼
Program	Program__c	Picklist		▼
student	Name	Text(80)	✓	▼
Student ID	Student_ID__c	Auto Number		▼

Application__c Fields
- Application_Number__c (Auto Number: APP-{0000})
- Student__c (Lookup → Student__c)
- Course__c (Lookup → Course__c)
- Status__c (Picklist: Draft, Submitted, Under Review, Accepted, Rejected)
- Submission_Date__c (Date)
- Intake__c (Picklist: Jan 2026, Jul 2026, etc.)
- Documents_Status__c (Picklist: Pending, Uploaded)
- Fee_Payment_Status__c (Picklist: Not Paid, Partial, Paid)

Application Number	Name	Auto Number	✓	▼
Board Name	Board_Name__c	Text(80)		▼
Course	Course__c	Lookup(Course)	✓	▼
Created By	CreatedById	Lookup(User)		
Degree Type	Degree_Type__c	Picklist		▼
Documents status	Documents_status__c	Picklist		▼
Entrance Test Rank	Entrance_Test_Rank__c	Number(6, 0)		▼
Entrance Test Score	Entrance_Test_Score__c	Number(4, 0)		▼
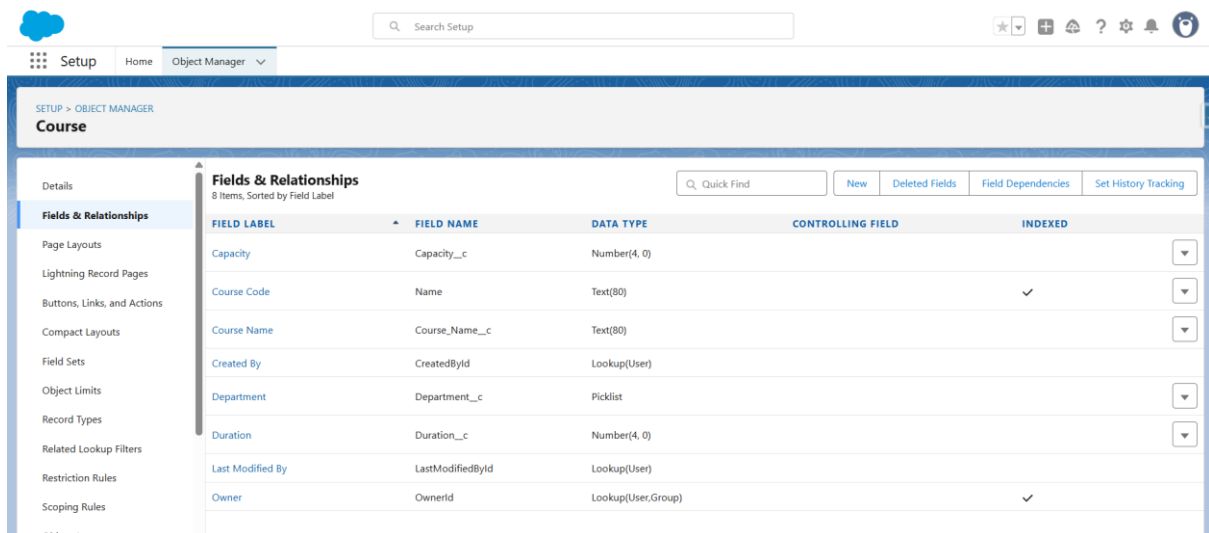
Fields & Relationships
23 Items, Sorted by Field Label

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
Scoping Rules

Quick Find | New | Deleted Fields | Field Dependencies | Set History Tracking

Course__c Fields
- Course_Code__c (Text)
- Course_Name__c (Text)
- Duration__c (Number, Months)
- Seats__c (Number)
- Department__c (Picklist: CS, Business, Arts, etc.)

Fields & Relationships
8 Items, Sorted by Field Label

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
Scoping Rules
Object Access

Quick Find | New | Deleted Fields | Field Dependencies | Set History Tracking

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED	
Capacity	Capacity__c	Number(4, 0)			▼
Course Code	Name	Text(80)		✓	▼
Course Name	Course_Name__c	Text(80)			▼
Created By	CreatedById	Lookup(User)			
Department	Department__c	Picklist			▼
Duration	Duration__c	Number(4, 0)			▼
Last Modified By	LastModifiedById	Lookup(User)			
Owner	OwnerId	Lookup(User,Group)		✓	

Enrollment__c Fields
- Enrollment_Number__c (Auto Number: ENR-{0000})
- Student__c (Master-Detail → Student__c)
- Course__c (Master-Detail → Course__c)
- Enrollment_Date__c (Date)
- Academic_Year__c (Picklist: 2025–26, 2026–27, etc.)
- Status__c (Picklist: Enrolled, Waitlisted, Cancelled)
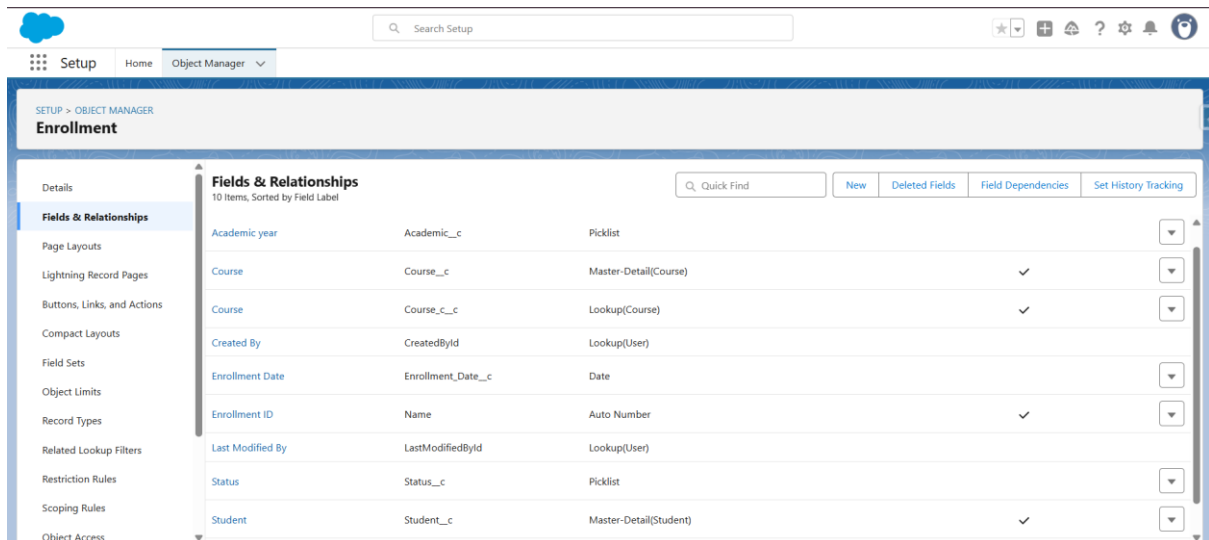
Fig. Fields

Step 3 — Record Type

- **Application__c Record Types**:
  - UG Application
  - PG Application
- Each record type has its own **page layout** (UG requires Entrance Exam Score, PG requires UG GPA).
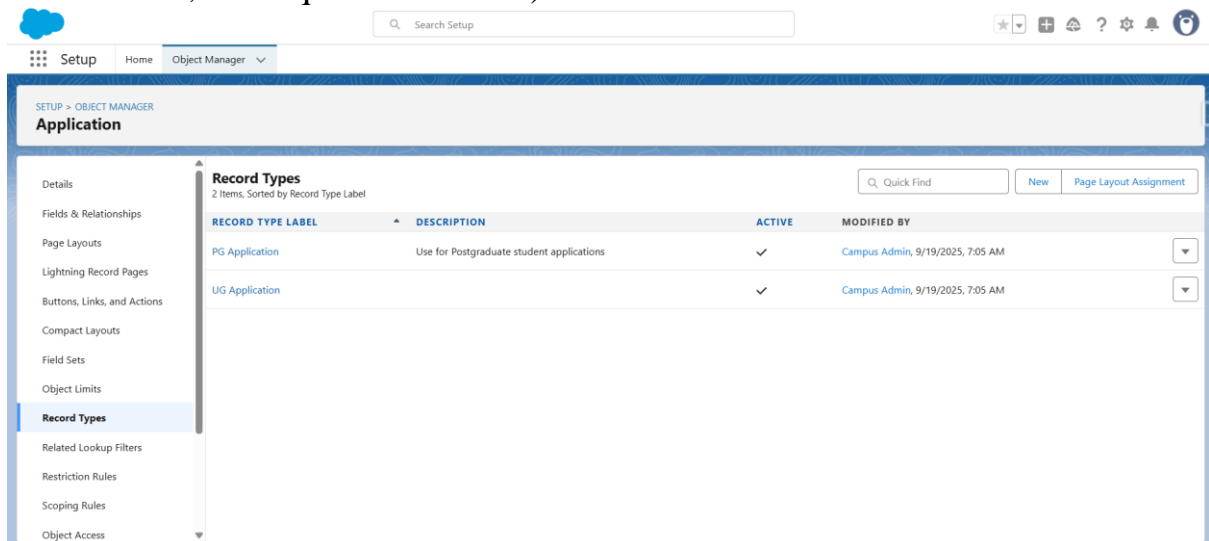
Fig.RecordTypes

Step 4 — Page Layouts

- **Student__c Layout** → personal info, related Applications & Enrollments.
- **Application__c Layouts** → UG layout shows entrance exam field, PG layout shows UG GPA field.
- **Course__c Layout** → seats, duration, department.
- **Enrollment__c Layout** → linked Student & Course, status, academic year.

Page Layouts
1 Items, Sorted by Page Layout Name

PAGE LAYOUT NAME	CREATED BY	MODIFIED BY	
Student Application Layout	Campus Admin, 9/18/2025, 5:57 AM	Campus Admin, 9/20/2025, 10:40 AM	

Page Layouts
3 Items, Sorted by Page Layout Name

PAGE LAYOUT NAME	CREATED BY	MODIFIED BY	
Application Layout	Campus Admin, 9/18/2025, 9:36 PM	Campus Admin, 9/20/2025, 10:10 AM	
PG Application Layout.	Campus Admin, 9/19/2025, 5:04 AM	Campus Admin, 9/19/2025, 7:29 AM	
UG Application Layout	Campus Admin, 9/19/2025, 4:50 AM	Campus Admin, 9/19/2025, 7:24 AM	

Page Layouts
1 Items, Sorted by Page Layout Name

PAGE LAYOUT NAME	CREATED BY	MODIFIED BY	
Enrollment Layout	Campus Admin, 9/18/2025, 6:47 AM	Campus Admin, 9/20/2025, 10:44 AM	

Page Layouts
1 Items, Sorted by Page Layout Name

PAGE LAYOUT NAME	CREATED BY	MODIFIED BY	
Course Layout	Campus Admin, 9/18/2025, 6:27 AM	Campus Admin, 9/20/2025, 10:34 AM	

Fig.PageLayouts

Step 5 — Compact Layouts

- **Application__c Compact Layout** → shows Application Number, Student, Status, Intake, Submission Date.
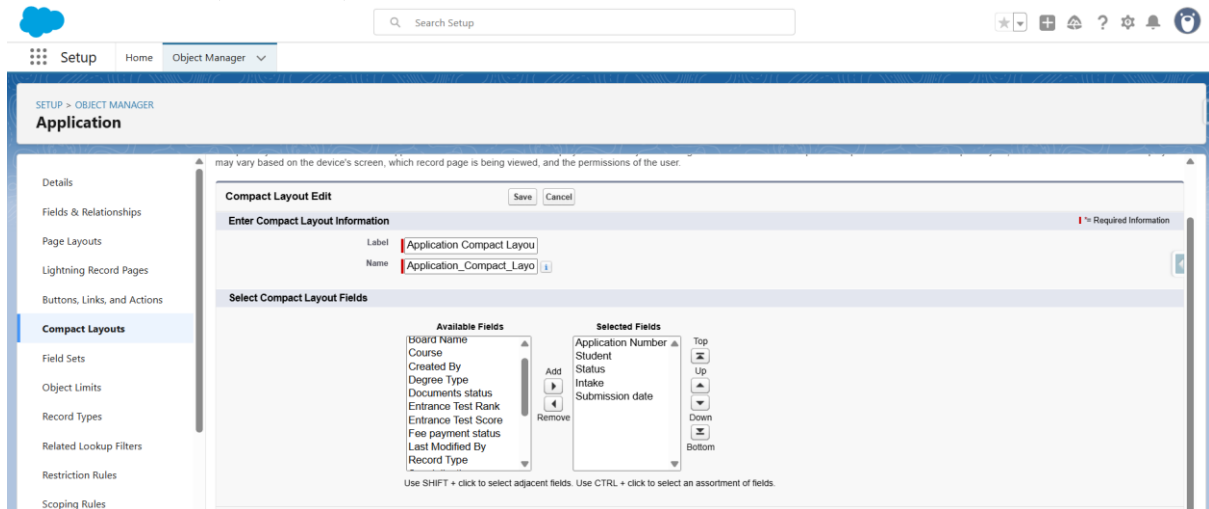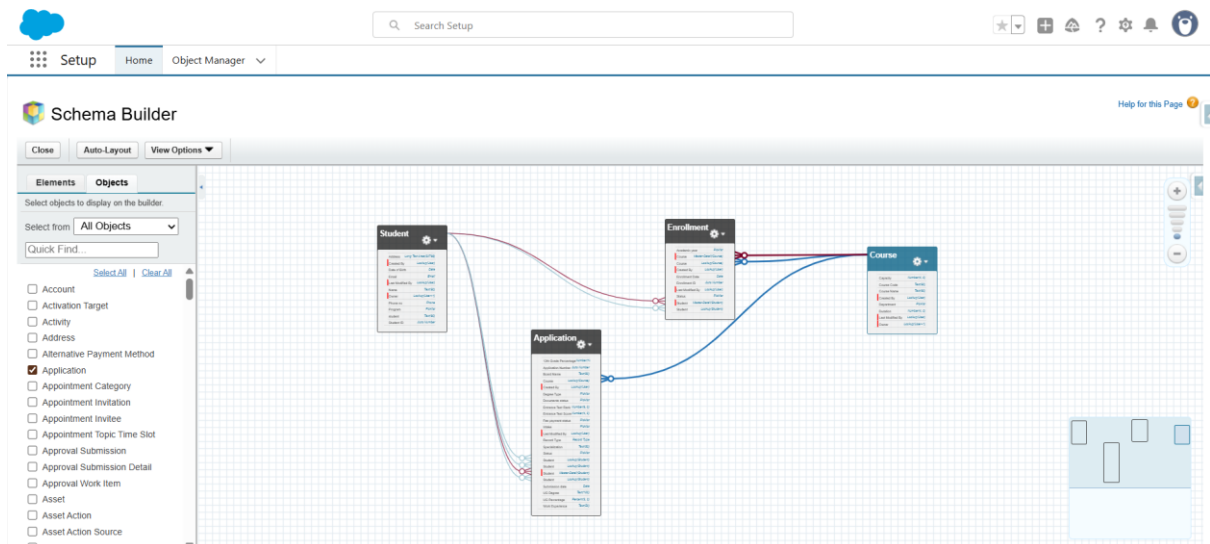- **Enrollment__c Compact Layout** → shows Enrollment Number, Student, Course, Status.
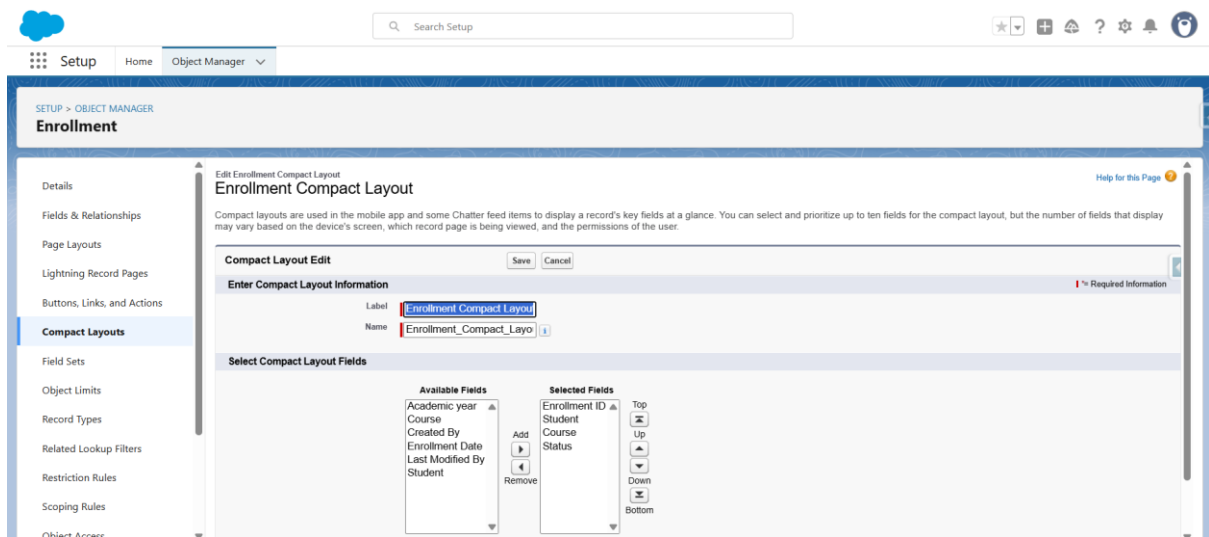
Fig.CompactLayouts

Step 6 — Schema Builder

- Used **Schema Builder** to visualize objects & relationships:
  - **Student__c → Application__c** (1:M)
  - **Student__c ↔ Course__c** (M:M via Enrollment__c junction)
  - **Course__c → Enrollment__c** (1:M)

Fig.SchemaBuilder

Step 7 — Lookup vs Master-Detail vs Hierarchical Relationships

- **Lookup Relationship:** Student__c → Application__c (loose relationship, application can exist independently).

- **Master-Detail Relationship:** Enrollment__c → Student__c and Course__c (deletes enrollments if parent is deleted).
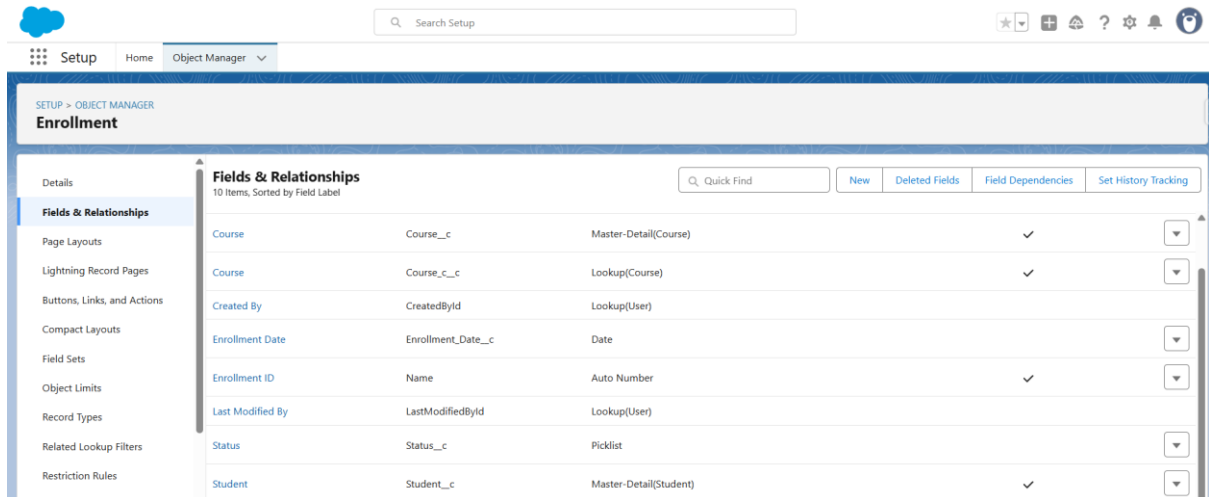
- **Hierarchical Relationship:** Used only on **User object** (e.g., reporting structure, not directly used here).
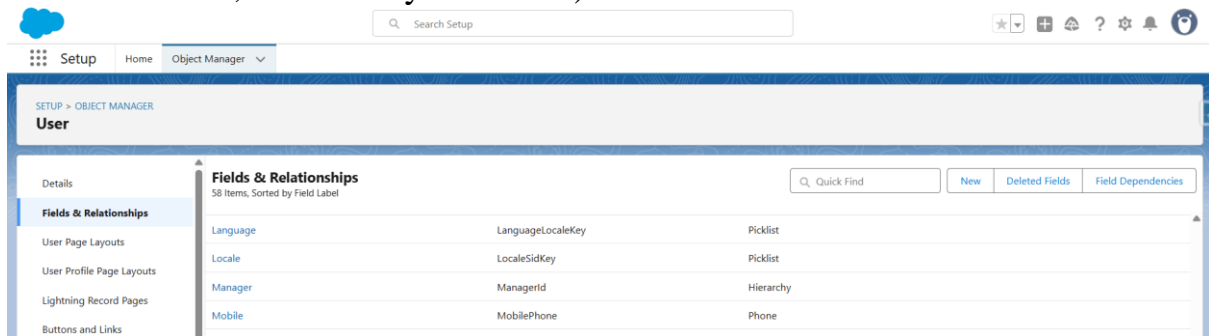
Fig.7 Relationships

Step 8 — Junction Objects
- **Enrollment__c** serves as a junction object to model **many-to-many relationship**:
    - A Student can enroll in many Courses.
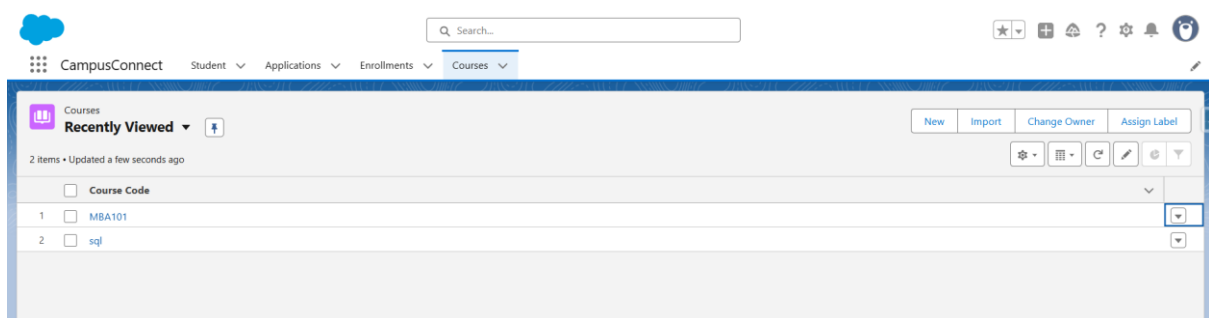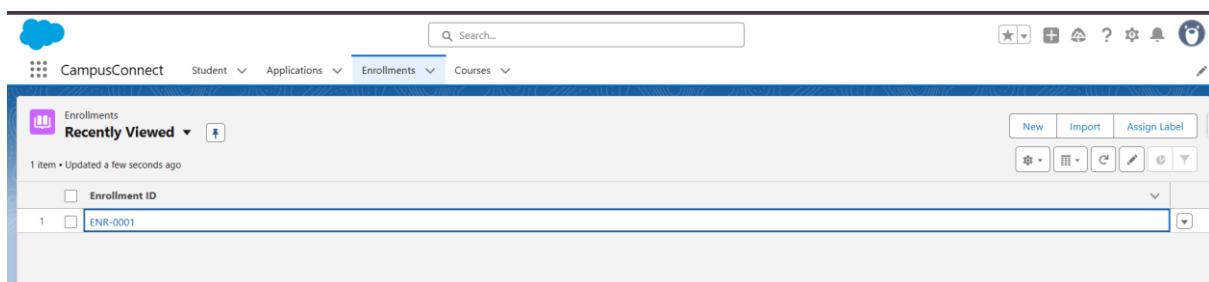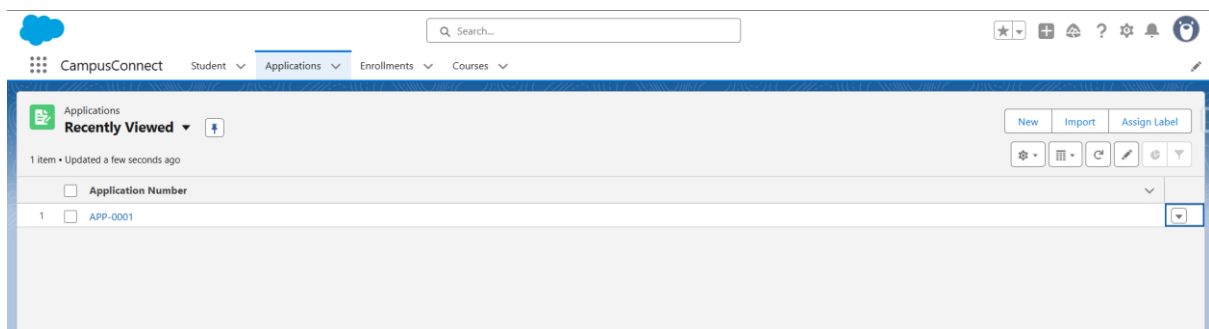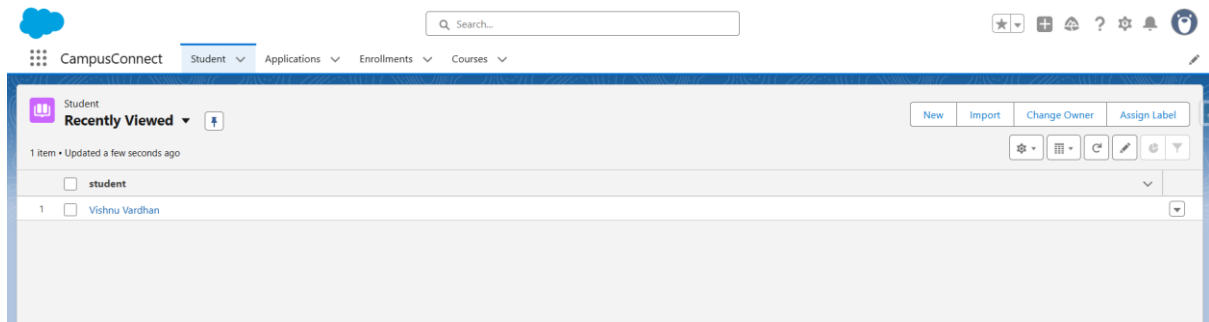   - A Course can have many Students.

Fig.JunctionObject

Step 9— Testing
- Created Student record (Vishnu Vardhan, UG).
- Created Application (MBA Program, Status = Submitted).
- Linked Application to Course (MBA).

- Application accepted → Enrollment created for Student ↔ Course.
- Verified compact layouts show key info.
- Verified Schema Builder displays all relationships correctly.

Fig.Testing