**Assignment 1 – University Course Registration System**

**Name:** Gurjit Singh  
**Student ID:** 300217308  
**Course:** CIS Assignment 1

**1. Scenario**

My student ID ends with **08**, which maps to the **University Course Registration System**.

**2. ER Model**

Main entities: **Students, Instructors, Departments, Courses, Classrooms, Enrollments (weak), Prerequisites (recursive)**.

* Students enroll in Courses (Enrollment stores grade, semester).
* Instructors teach Courses.
* Departments offer Courses.
* Courses scheduled in Classrooms.
* Courses may have prerequisites.

📎 Deliverable: er-diagram.pdf

**3. Data Dictionary**

Each table includes attributes, data types, PK/FK constraints, and metadata.  
📎 Deliverables: tables.md, constraints.md

**4. Normalization**

* All tables in **1NF** (atomic attributes).
* All tables in **2NF** (no partial dependencies).
* Transitive dependency (course → dept\_id → dept\_name) resolved.

📎 Deliverables: functional-dependencies.md, 1nf-analysis.md, 2nf-analysis.md

**5. PostgreSQL Implementation**

* schema.sql → CREATE TABLE + comments.
* sample\_data.sql → ≥5 rows per table.
* test\_queries.sql → 5 queries (JOIN, aggregate, subquery, etc.).
* indexes.sql → indexes for foreign keys.

|  |  |
| --- | --- |
| Notation: | Strong Entity  Weak Entity  Relationship  Primary Key |
| 1, N, M | Cardinality |

**University Course Registration System ER Diagram**

Department

Instructor

Student

Course

Enrollment

Prerequisite

ID

ID

ID

ID

ID

ID

Belongs

To

Emplo

ys

Offers

Teaches

Enrolls

In

Has

Prereq

1

1

1

N

1

N

1

N

N

1

M

N

N

1

# Core Components

* **6 Entities**: Student, Course, Instructor, Department, Enrollment, Prerequisite
* **2 Weak Entities**: Enrollment, Prerequisite (double borders)
* **6 Relationships**: Clear connections with proper cardinalities
* **Primary Keys**: Underlined attributes for each entity

# Key Relationships with Corrected Cardinalities

* **Belongs To**: Department (1) - Student(N)
* **Employs**: Department (1) - Instructor(N)
* **Offers**: Department (1) - Course(N)
* **Teaches**: Instructor (1) - Course(N)
* **Enrolls In**: Student (1) - Enrollment(N) and Course (1) - Enrollment(N)
* **Has Prerequisite**: Course(M) - Course(N) - Recursive relationship