Database project report

Business Requirement:

1- Department Management:
   - Each department must have a unique identifier, name, location and budget.
   - Departments can offer multiple courses.

2- Course Management:
   - Each course must have a unique ID, title and credit hours.
   - A course can have prerequisites that link it to other courses.
   - Every course must be associated with a department.

3- Student Management:
   - Each student must have a unique ID, first name, last name, phone number ,sex ,department id and total credit.
   - Every student must register in one department.
   - Each student should receive grades for the sections the takes.

4- Instructor Management:
   - Each instructor must have a unique ID, first name, last name, phone number ,sex ,department id and salary.
   - Each instructor instructs only one department.
   - Instructors can teach multiple sections.

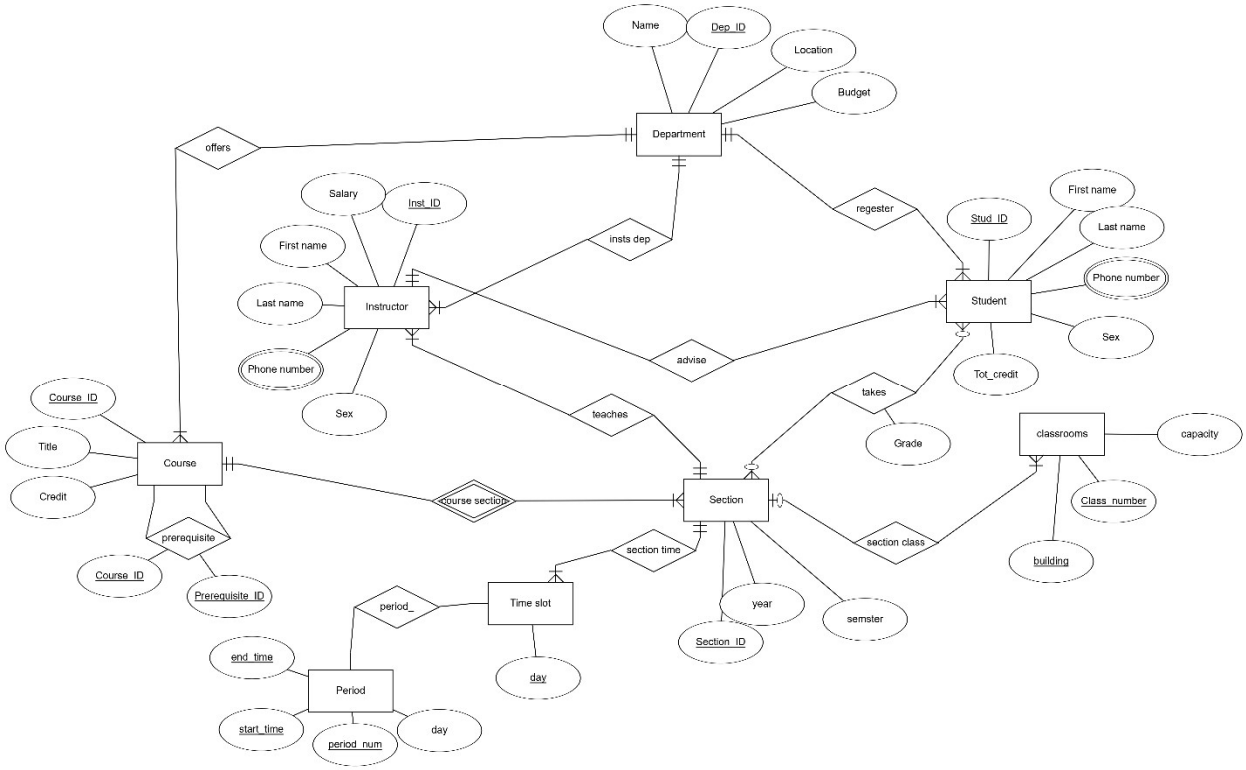
5- Section Management:
   - Each section must have a unique ID, semester and year.
   - A section must be associated with a specific schedule and classroom.

6- Classroom Management:
   - Each classroom must have a unique class number, capacity and a building.
   - Classrooms can be assigned to sections.

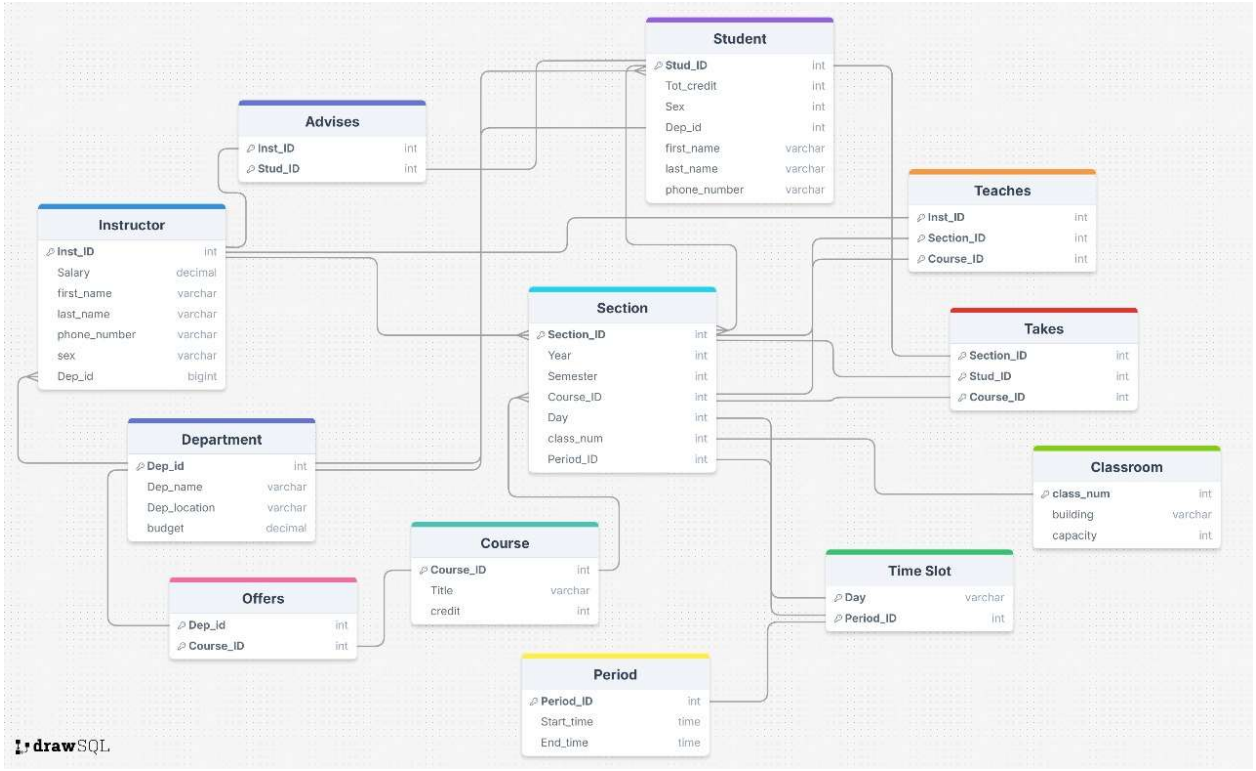
7- Time Slot Management:
   - Each time slot must include a day and period number.
   - Each section must be assigned a specific time slot.
   -

8- Scheduling and Registration:
   - Students can register for available sections based on their schedules.
   - Sections must have assigned instructors and classrooms.

ERD:



Schema:



SQL explanation:

1- Database creation:
   - “create database University;”
     Creates a new database named University.
   - “use Final_university;”
     Specifies the database context to execute subsequent queries.

2- Table Creation:
   a- “department” Table
      -Represents university departments.
      -Fields:
         1) “dep_id” (Primary key, auto-incrementing ID)
         2) “dep_name” (Department name)
         3) “dep_location” (Location)
         4) “Budget”
   b- “instructor” Table
      -Fields:
         1) “inst_id” (Primary key, auto-incrementing ID)
         2) “first_name”, “last_name”
         3) “phone_number” (multivalued attribute)
         4) “sex”
         5) “salary”
         6) “dep_id” (Foreign key referencing “department”)
   c- “student” Table
      -Fields:
         1) “stud_id” (Primary key, auto-incrementing ID)
         2) “first_name”, “last_name”
         3) “phone_number” (multivalued attribute)
         4) “sex”
         5) “tot_credit”
         6) “dep_id” (Foreign key referencing “department”)

d- “course” Table
-Fields:
1) “course_id” (Primary key)
2) “title”, “credit”

e- “classroom” Table
-Fields:
1)“class_num” (Primary key, auto-incrementing ID)
2) “building”, “capacity

f- “period” Table
-Fields:
1) “period_id” (Primary key, auto-incrementing ID)
2) “start_time”, “end_time”

g- “[time slot]” Table
-Fields
1) “day”
2) “period_id” (Foreign key referencing “period”)
3) Composite Primary Key: “day”, “period”

h- “section” Table
-Fields:
1) “section_id” (Primary key, auto-incrementing ID)
2) “year”, “semester”
3) Relationships with other tables:
a- “course_ID” (Foreign key referencing “course”)
b- “day”, “period_id” (Foreign key referencing “[time slot]”)
c- “class_num” (Foreign key referencing “classroom”)
d- “inst_id” (Foreign key referencing “instructor”)

i- Junction Tables (To manage many-to-many relationships)
1) “[department offers courses]” → Links departments and courses.
2) “[instructor advises students]” → Links instructors and students.
3) “[instructors teach sections]”→ Links instructors and sections.
4) “[students take sections]”→ Links students, sections and courses.

3- Initial Data Insertion:

- "period" Table
  Inserts 6 time periods with start and end times (e.g., '8:30', '10:30').
- "[time slot]" Table:
  Inserts all combinations of days ('saturday ' to 'thursday') and time periods (1-6).

Some SQL statements to retrieve data:

1) Retrieve all departments
   "select * from departments;"
2) List all courses offered by each department
   "Select d.dep_nam, c.title
   From [department offers courses] doc
   Join department d on doc.dep_id = d.dep_id
   Join course c on doc.course_id = c.course_id ;"
3) List instructors in each department
   "Select i.first_name, i.last_name, d.dep_name
   From instructor i
   Join department d on i.dep_id = d.dep_id;"
4) Find all students are advised by a specific instructor
   "Select s.first_name, s.last_name
   From [instructor advises students] ias
   Join student s on ias.stud_id = s.stud_id
   Where ias.inst_id = 1; "replace 1 with the desired instructor id""
5) Retrieve class schedule for a specific day
   "Select ts.day,p.start_time,p.end_time,s.section_id,c.title,
   cl.building,cl.capacity
   From section s
   Join [time slot] ts on s.day = ts.day and s.period_id = ts.period_id
   Join period p on ts.period_id = p.period_id
   Join course c on s.course_ID = c.course_ID
   Join classroom cl on s.class_num = cl.class_num
   Where ts.day = 'monday'; "replace monday with the desired day""

6) List students enrolled in a specific course

“Select s.first_name, s.last_name
From [student takes sections] sts
Join student s on sts.stud_id = s.stud_id
Join cource c on sts.course_id = c.course_id
Where c.title = ‘Database’; “replace Database with the desired course title””

Team members:

عبدالرحمن ابراهيم دسوقي امام	23011100
سيف الدين يسري محمد محمود	23010152
عمر إبراهيم توفيق محمد	23011369
يحيى فوزى عبد العال عبد الموجود	23011617
أنس حسام محمد رفعت	23011229
عبدالرحمن هشام السيد محمد	23011331
زياد أحمد محمد مرسي	23011268
محمد وفيق محمد إبراهيم	23011500