Consider the following database schema. Write the following queries in SQL.

**COURSES** = {CourseNo, Title, Credits}
**OFFERINGS** = {CourseNo, Section, Semester, Year, Prof-name, Schedule, RoomNo}
**PROFS** = {Prof-name, Department, Building, Office-No, Phone-No}
**STUDENTS** = {IDNo, Name, DOB, Department}
**REGISTRATION** = {IDNo, CourseNo, Section, Semester, Year}

**Q1**: List all the information regarding the course offerings for CSC 130.

**Q2**: List the titles of all courses that Professor Stilgar is teaching.

**Q3**: List all courses (Titles and number of credits) taught by Physics professors.

**Q4**: List all Mathematics professors who do not teach MAT 110 this semester.

**Q5**: List the names and ID numbers of students taking a course (at least one) with Professor Stilgar this semester.

**Q6**: List the names and ID numbers of students taking all courses Professor Stilgar is teaching this semester.

**Q7**: List all students (IDs and names) in the Math department, and for each student list the total number of credits he/she is taking in this semester. That is, the output is a relation with three columns, IDNo, Name, total Credits.

**Q8**: Create a view that lists all courses (course numbers) and their sections, and for each course section lists the total number of students registered in that course in this semester. That is, the view has three columns, CourseNo, Section, Enrollment.

**Question 2**

Give a brief description about *outer join* and explain, using examples, how (left, right, full) outer join is specified in SQL.

10% of the assignment mark is for typesetting. Figures and special symbols can be drawn/written by hand.