Database Systems
Project 1: Writing SQL Queries in Oracle
Max Total: 15 points

The following normalized tables from the Student Registration System (some tables have been simplified) will be used in this project:

Students(sid, firstname, lastname, status, gpa, email)
Courses(dept_code, course#, title)
Course_credit(course#, credits)
Classes(classid, dept_code, course#, sect#, year, semester, limit, class_size)
Enrollments(sid, classid, lgrade)
Grades(lgrade, ngrade)

As a general clarification, we assume that no student takes the same course (including different sections of the same course) more than once. If you have questions about these tables, please contact the instructor for clarification.

1. **Preparation**

Save the sql file proj1_tables.sql in your harveyv account using UShare (Read Oracle guidelines available on Brigtspace).

Run it in your Oracle account:

SQL> start proj1_tables

Then check whether all tables are created correctly in your Oracle account.

2. Query Statements

There are 15 statements in this project, each of which is 1 point. You need to write one or more SQL query for each statement.

Your queries must consider that the tuples currently in the database may change. In other words, your queries must be correct not only for the provided data but also for any valid data. For this project, each query needs returns distinct rows only (i.e., remove duplicate rows in your result). Furthermore, you are disallowed to use unnecessary "select distinct" in your queries. Unremoved duplicates or unnecessary use of "select distinct" will result in points deduction. You are encouraged to use methods other than "select distinct" to remove/avoid duplicates.

The 15 query statements follow:

1. Find the first name of each student who has taken at least one CS course and at least one math course.

2. Find the dept_code and course# of each course that was not offered in Spring 2024.

3. Find the firstname of each student who has never received a C for any course he/she has taken. If you write a nested query, make sure the subquery is uncorrelated.

4. Find the lastname of each student who has received an A for every course he/she has taken. Count only classes for which he/she received a non-null grade. GPA information is not permitted to be used in this query. If multiple students have the same last name, return the first one only, while removing the others with the same last name.

5. Find the dept_code and course# of each course that has been offered the smallest number of times (each record in the classes table corresponds to a course offering). Note that it is possible that more than one course may satisfy this query condition; in this case, all such courses need be retrieved.

6. Find the sid and firstname of every student who has taken more than 2 classes.

7. Find every class (all attributes are needed) that is offered by the CS department in the Spring semester of 2024 and has less than 3 students enrolled. For this query, you are not allowed to use the size information from the classes table.

8. Find the sid and first name of every student who has taken all 200-level Math courses. Here we are referring to courses, not classes. (A course, e.g., CS 240, can be offered in different semesters where the different implementations of the course are different classes.)

9. Find the title of each course that has been taken by student B003 but not by student B005.

10. Find the first name of every student who has taken at least one course that has been taken by student B002. Note that here we are talking about taking the same course, not just the same class.

11. Find the dept_code and course# of each course that has two or more classes in the same semester of the same year. The query should also show the semester and year information for each qualified course.

12. Find the sid and lastname of each student who has received at least one highest grade in one of the classes he/she has taken. Suppose all possible grades are (A, B, C, D, F). Note that the highest grade given to students in a class is not necessarily A, for example, when all students in the class did poorly.

13. List the dept_code, course# and title of each course that has been taken by student B007. For each such course, also list the grade the student received. If the grade is null for a course, output "To be assigned" as the grade information for the course.

14. Find the dept_code, course# and title of each course whose title contains "data" and that has been taken by all students whose GPA is higher than 3.3. Note that even though a qualified course is required to be taken by all students whose GPA is higher than 3.3, it may also be taken by some students whose GPA is not higher than 3.3.

15. Compute the GPA for each student from the student's number grades (ngrade) for all the courses he/she has taken (ignore the GPA values already in the students table). The GPA of a student is computed by dividing the sum of his/her number grades by the number of classes he/she has taken and received a non-null number grade. If a student has not received any non-null grade yet, the student's GPA will be null. For each student, the sid, last name, and the computed GPA (name column head as cgpa) of the student need be displayed. Display the results in ascending non-null GPA values.

3. How to create a spool file for submission

You need to generate a spool file and submit it through Gradescope as follows.

1) After writing your queries, save them in separate plain text files with sql extension. Specifically, save them as 1.sql, 2.sql, …, 15.sql.

2) Follow the steps below to generate the spool file, called project1.txt, in Oracle:

```
SQL> set echo on
SQL> spool project1.txt
SQL> start 1
SQL> start 2
.......
SQL> start 15
SQL> spool off
```

3) Write your name(s) at the top of project1.txt.
4) Convert the project1.txt file to PDF
5) Submit it through Gradescope.