

DATA 504
Project 1: Writing SQL Queries in Oracle
(Due: 11:00pm on January 17, 2021)

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

Students(B#, first_name, last_name, status, email, bdate, deptname)
Courses(prog_code, course#, title)
Course_credit(course#, credits)
Classes(classid, prog_code, course#, sect#, year, semester, limit, class_size, room)
Enrollments(B#, classid, lgrade)

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 script file proj1_tables_script21.txt as proj1_tables_script21.sql in your bingsuns account.

To run the above script from your Oracle account, use

```
SQL> start proj1_tables_script21
```

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

2. Project Requirements

There are 12 statements (see Section 3 below) in this project and you are asked to write one or more SQL queries for each statement. Your query should take into consideration that the tuples currently in the database may change. In other words, your query must be correct for any valid state of every table. It is very important that you understand each query statement correctly. If you have any doubt about the correct interpretation of a statement, please ask the instructor for clarification by posting your questions in the Project 1 thread in the discussion forum in MyCourses. The query (or queries) for each statement is worth **5** points.

It is suggested that you first test each query individually and save each query in a different file (with extension .sql). After all queries have been tested to your satisfaction, you can run all the queries in a sequence and save the entire session in a spool file. Suppose you have saved your queries in files query1.sql, ..., query12.sql. Follow the steps below to generate the spool file after you have logged on Oracle:

```
SQL> set echo on
SQL> spool project1.txt
SQL> start query1
.....
SQL> start query12
SQL> spool off
```

To prepare your submission of Project 1, follow the steps below:

- (1) **Edit file project1.txt** by **adding your name** at the top of the file and **adding the following honesty statement**: “I have done this assignment completely on my own. I have not copied it, nor have I given my solution to anyone else. I understand that if I am involved in plagiarism or cheating I will have to sign an official form that I have cheated and that this form will be stored in my official university record. I also understand that I will receive a grade of 0 for the involved assignment and my grade will be reduced by one level (e.g., from A to A- or from B+ to B) for my first offense, and that I will receive a grade of “F” for the course for any additional offense of any kind.” No other changes to the file are permitted.
- (2) Submit project1.txt to the Project 1 submission folder in MyCourses by 11:00pm on January 17, 2021.

3. Query Statements

The following are the 12 query statements:

1. Find the B#, the first name and last name of every CS seniors. In the output, first name and last name of each student should be concatenated with a space in between under a new column header name.
2. Find the B#, first name, last name and birth date of each student who has taken at least one course. The header of the last column needs to be “birth date” (without the quotes and there is a space between birth and date). To not have header “birth date” truncated in the output, run SQL> column “birth date” format a10 before you run your query, here 10 is the new column/header width in character for “birth date”.)
3. Find the B#, first name, last name of each student who has taken at least one CS course and at least one math course. Write an SQL query that uses neither intersect nor distinct but the results have no duplicate.
4. (12 points) Find the B#, first name and last name of each student who has taken at least one course but has never received an A for any course he/she has taken. Write a query with an uncorrelated subquery and write another query with a correlated subquery.
5. Find the B#, first name and last name of each student who has taken at least one course and received an A for every class he/she has taken. Count only classes for which he/she received a non-null grade. In addition, do not use any aggregate (set) function.
6. Find the classid, prog_code and course# of each undergraduate class (i.e., course# < 500) that was offered in Spring 2019. For each such class, also list the number of seats available (computed by limit – class_size) under the header “seats_available”.
7. Find the B# and the total number of credits earned by each student.
8. Find the prog_code and course# of the course that has been attended by the largest number of students. If a course has been offered multiple times (i.e., has multiple classes), the numbers of students for all these offerings should be added for the course. Note that it is possible that multiple courses may have the same highest total attendance; in this case, all such courses should be output.
9. Find the B#, first name and last name of every student who has taken at least 2 classes. Also output the number of classes each such student has taken.
10. Find the classid, prog_code and course# of each class that has been taken by all juniors.
11. Find the B#, first name and last name of every student who has taken all CS classes offered in Spring 2019.

12. Find the prog_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 year and semester information for each qualified course.