## Assignment 2: Aggregates and Nested Subqueries

### Statement : Write the SQL queries using aggregates, grouping, ordering statements and nested subqueries for given statements on given schema.

### Intermediate SQL: Aggregates and grouping and ordering

The following questions are all based on the university schema

* Find the number of instructors who have never taught any course. If the result of your query is empty, add the appropriate data (and include corresponding insert statements) to ensure the result is not empty.NOTE: IN THE SUBMISSION FILE, PASTE DATA INSERTED BELOW THIS STATEMENT AS A REMARK.

1. select count(\*) from instructor where id not in (select id from teaches);

+----------+

| count(\*) |

+----------+

| 3 |

+----------+

* Find the total capacity of every building in the university.

1. select building, sum(capacity) as total\_capacity from classroom group by building;

+----------+----------------+

| building | total\_capacity |

+----------+----------------+

| Packard | 500 |

| Painter | 10 |

| Taylor | 70 |

| Watson | 80 |

+----------+----------------+

* Find all departments that have at least one instructor, and list the names of the departments along with the number of instructors;   order the result in descending order of number of instructors.

1. select dept\_name, count(id) as no\_of\_instructors from instructor group by dept\_name order by no\_of\_instructors desc;

+------------+-------------------+

| dept\_name | no\_of\_instructors |

+------------+-------------------+

| Comp. Sci. | 3 |

| Finance | 2 |

| History | 2 |

| Physics | 2 |

| Biology | 1 |

| Elec. Eng. | 1 |

| Music | 1 |

+------------+-------------------+

* For each student, compute the total credits they have successfully completed, i.e. total credits of courses they have taken, for which they have a non-null grade other than 'F'. Do NOT use the tot\_creids attribute of student.

1. select s.name, sum(c.credits) as total\_credits from takes t join course c on t.course\_id = c.course\_id join student s on t.id = s.id where t.grade is not null and t.grade != 'F' group by s.name;

+----------+---------------+

| name | total\_credits |

+----------+---------------+

| Zhang | 7 |

| Shankar | 14 |

| Brandt | 3 |

| Chavez | 3 |

| Peltier | 4 |

| Levy | 7 |

| Williams | 8 |

| Sanchez | 3 |

| Brown | 7 |

| Aoi | 3 |

| Bourikas | 7 |

| Tanaka | 4 |

+----------+---------------+

**B. Nested Subqueries** : Write the following queries for university schema.

* Find the id and title of all courses which do not require any prerequisites.

1. select course\_id, title from course where course\_id not in (select course\_id from prereq);

+-----------+----------------------------+

| course\_id | title |

+-----------+----------------------------+

| BIO-101 | Intro. to Biology |

| CS-101 | Intro. to Computer Science |

| FIN-201 | Investment Banking |

| HIS-351 | World History |

| MU-199 | Music Video Production |

| PHY-101 | Physical Principles |

+-----------+----------------------------+

* Find the names of students who have not taken any biology dept. courses.

1. select name from student where id in (select id from takes where course\_id != 'BIO%');

+----------+

| name |

+----------+

| Zhang |

| Shankar |

| Brandt |

| Chavez |

| Peltier |

| Levy |

| Williams |

| Sanchez |

| Brown |

| Aoi |

| Bourikas |

| Tanaka |

+----------+