Karlov Alexandr

Laboratory work 3

1. Write the following queries in SQL, using the university schema:
    a. Find all courses worth more than 3 credits;
    b. Find all classrooms situated either in 'Watson' or 'Packard' buildings;
    c. Find all courses offered by the Computer Science department;
    d. Find all courses offered during fall;
    e. Find all students who have more than 45 credits but less than 90;
    f. Find all students whose names end with vowels;
    g. Find all courses which have course 'CS-101' as their prerequisite;

a) SELECT course_id
FROM course
WHERE credits > 3;

b) SELECT room_number
FROM classroom
WHERE building = 'Watson' or building = 'Packard';

c) SELECT course_id
FROM course
WHERE dept_name = 'Comp. Sci.';

d) SELECT course_id
FROM section
WHERE semester = 'Fall';

e) SELECT ID
FROM students
WHERE tot_cred > 45 and tot_cred < 90;

f) SELECT ID
FROM students
WHERE SIMILAR TO '%[a|o|i|y|e|u]';

g) SELECT course_id
FROM prereq
WHERE prereq = 'CS-101';

2. Write the following queries in SQL, using the university schema:
    a. For each department, find the average salary of instructors in that department and list them in ascending order. Assume that every department has at least one instructor;
    b. Find the building where the biggest number of courses takes place;
    c. Find the department with the lowest number of courses offered;
    d. Find the ID and name of each student who has taken more than 3 courses from the Computer Science department;
    e. Find all instructors who work either in Biology, Philosophy, or Music departments;
    f. Find all instructors who taught in the 2018 year but not in the 2017 year;

a) SELECT name, dept_name, avg(salary) AS av_salary
FROM instructor
GROUP BY dept_name
ORDER BY av_salary;

b) SELECT bmax.building, max(course_count)
FROM (SELECT count(course_id) AS course_count, building FROM section) as bmax;

c) SELECT deptmin.dept_name, min(course_count)
FROM (SELECT count(course_id) AS course_count, dept_name FROM course) as dmin;

d) SELECT student.ID, student.name
FROM student, takes
WHERE student.dept_name = 'Comp. Sci.' and (SELECT count(course_id) FROM takes)>3;

e) SELECT *
FROM instructor
WHERE dept_name = 'Biology' OR dept_name = 'Philosophy' OR dept_name = 'Music';

f)SELECT *
FROM instructor
where id IN
(SELECT id
FROM teaches
WHERE year = 2018 AND teaches.id NOT IN(SELECT id
FROM teaches
WHERE year = 2017);