Assignment 03
EPPS 6354 Information Management

Oliver Jack Myers

UTD Spring 2025

1.i SQL to get Student ID’s

SQL for Students:

```
SELECT ID
FROM student;
```

Enter SQL commands here

```
1 SELECT ID FROM student;
```

ID
00128
12345
19991
23121
44553
45678
54321
55739
70557
76543
76653
98765
98988

1.ii SQL to get Instructors

SQL for Instructors:

```
SELECT name
FROM instructor;
```

Enter SQL commands here

```
1 SELECT name FROM instructor;
```

name
Srinivasan
Wu
Mozart
Einstein
El Said
Gold
Katz
Califieri
Singh
Crick
Brandt
Kim

1.iii SQL to get Departments

SQL for Departments:

```
SELECT dept_name
FROM department;
```

Enter SQL commands here

```
1 SELECT dept_name
2 FROM department;
```

dept_name
Biology
Comp. Sci.
Elec. Eng.
Finance
History
Music
Physics

2.i Find the ID and name of each student who has taken at least one Comp. Sci. course;
make sure there are no duplicate names in the result.

SQL:

```
SELECT DISTINCT student.ID,
student.name

FROM student, takes, course

WHERE student.ID = takes.ID

  AND takes.course_id =
course.course_id

  AND course.dept_name = 'Comp.
Sci.';
```

Enter SQL commands here

```
1 SELECT DISTINCT student.ID, student.name
2 FROM student, takes, course
3 WHERE student.ID = takes.ID
4   AND takes.course_id = course.course_id
5   AND course.dept_name = 'Comp. Sci.';
```

Execute | Save the db | Load an SQLite database file: Choose File no file selected

ID	name
00128	Zhang
12345	Shankar
45678	Levy
54321	Williams
76543	Brown
98765	Bourikas

2.ii Add grades to the list

SQL:

```
SELECT DISTINCT student.ID,
student.name, takes.grade

FROM student, takes, course

WHERE student.ID = takes.ID

  AND takes.course_id =
course.course_id

  AND course.dept_name =
'Comp. Sci.';
```

Enter SQL commands here

```
1 SELECT DISTINCT student.ID, student.name, takes.grade
2 FROM student, takes, course
3 WHERE student.ID = takes.ID
4   AND takes.course_id = course.course_id
5   AND course.dept_name = 'Comp. Sci.';
```

Execute | Save the db | Load an SQLite database file: Choose File no file selected

ID	name	grade
00128	Zhang	A
00128	Zhang	A-
12345	Shankar	C
12345	Shankar	A
45678	Levy	F
45678	Levy	B+
45678	Levy	B
54321	Williams	A-
54321	Williams	B+
76543	Brown	A
98765	Bourikas	C-
98765	Bourikas	B

2.iii Find the ID and name of each student who has not taken any course offered before 2017.

SQL:

```
SELECT DISTINCT s.ID, s.name
FROM student s
WHERE s.ID NOT IN (
  SELECT t.ID
  FROM takes t
  WHERE t.year < 2017
);
```

Enter SQL commands here

```
1 SELECT DISTINCT s.ID, s.name
2 FROM student s
3 WHERE s.ID NOT IN (
4   SELECT t.ID
5   FROM takes t
6   WHERE t.year < 2017
7 );
```

Execute | Save the db | Load an SQLite database file: Choose File no file selected

ID	name
00128	Zhang
12345	Shankar
19991	Brandt
23121	Chavez
44553	Peltier
45678	Levy
54321	Williams
55739	Sanchez
70557	Snow
76543	Brown
76653	Aoi
98765	Bourikas
98988	Tanaka

2.iv For each department, find the max salary of instructors in that department.

SQL:

```
SELECT dept_name,
MAX(salary) AS max_salary

FROM instructor

GROUP BY dept_name;
```

Enter SQL commands here

```
1 SELECT dept_name, MAX(salary) AS max_salary
2 FROM instructor
3 GROUP BY dept_name;
```

Execute | Save the db | Load an SQLite database file: Choose File no file selected

dept_name	max_salary
Biology	72000
Comp. Sci.	92000
Elec. Eng.	80000
Finance	90000
History	62000
Music	40000
Physics	95000

2.v Find the lowest, across all departments, of the per-department maximum salary computed by the preceding query.

```
SELECT dept_name, max_salary
FROM (
    SELECT dept_name, MAX(salary) AS max_sa
FROM instructor
GROUP BY dept_name
) AS dept_max
WHERE max_salary = (
SELECT MIN(max_salary)
FROM (
SELECT dept_name, MAX(salary) AS max_salary
FROM instructor
GROUP BY dept_name
) AS dept_max_inner
);
```

Enter SQL commands here

```
1  SELECT dept_name, max_salary
2  FROM (
3    SELECT dept_name, MAX(salary) AS max_salary
4    FROM instructor
5    GROUP BY dept_name
6  ) AS dept_max
7  WHERE max_salary = (
8    SELECT MIN(max_salary)
9    FROM (
10     SELECT dept_name, MAX(salary) AS max_salary
11     FROM instructor
12     GROUP BY dept_name
13   ) AS dept_max_inner
14 );
```

Execute | Save the db | Load an SQLite database file: Choose File no file selected

dept_name	max_salary
Music	40000

2.vi Add names to the list

```
SELECT dept_name  max_salary
FROM (
    SELECT dept_
FROM instructor
GROUP BY dept_nar
) AS dept_max
WHERE max_salary
SELECT MIN(max_s
FROM (
SELECT dept_name
FROM instructor
GROUP BY dept_nar
) AS dept_max_in
);
```

Enter SQL commands here

```
1  SELECT i.name, i.dept_name, i.salary
2  FROM instructor i
3  WHERE i.salary = (
4    SELECT MIN(max_salary)
5    FROM (
6      SELECT dept_name, MAX(salary) AS max_salary
7      FROM instructor
8      GROUP BY dept_name
9    ) AS dept_max
10 )
11 AND i.salary = (
12   SELECT MAX(salary)
13   FROM instructor
14   WHERE dept_name = i.dept_name
15 );
```

Execute | Save the db | Load an SQLite database file: Choose File no file selected

name	dept_name	salary
Mozart	Music	40000

4. Find instructor (with name and ID) who has never given an A grade in any course she or he has taught. (Instructors who have never taught a course trivially satisfy this condition.)

Enter SQL commands here

```
SELECT DISTINCT instructor.ID, instructor.name
FROM instructor
WHERE instructor.ID NOT IN (
  SELECT teaches.ID
  FROM teaches, takes
  WHERE teaches.course_id = takes.course_id
    AND teaches.sec_id = takes.sec_id
    AND teaches.semester = takes.semester
    AND teaches.year = takes.year
    AND takes.grade = 'A'
);

```

Execute | Save the db | Load an SQLite database file: Choose File no file selected

ID	name
12121	Wu
15151	Mozart
22222	Einstein
32343	El Said
33456	Gold
45565	Katz
58583	Califieri
76543	Singh
98345	Kim

5. Write SQL query to find the number of students in each section. The result columns should
appear in the order “courseid, secid, year, semester, num”. You do not need to output sections
with 0 students.

Enter SQL commands here

```
1 SELECT course_id, sec_id, year, semester, COUNT(ID) AS num
2 FROM takes
3 GROUP BY course_id, sec_id, year, semester
4 HAVING COUNT(ID) > 0;
```

Execute | Save the db | Load an SQLite database file: Choose File no file selected

course_id	sec_id	year	semester	num
BIO-101	1	2017	Summer	1
BIO-301	1	2018	Summer	1
CS-101	1	2017	Fall	6
CS-101	1	2018	Spring	1
CS-190	2	2017	Spring	2
CS-315	1	2018	Spring	2
CS-319	1	2018	Spring	1
CS-319	2	2018	Spring	1
CS-347	1	2017	Fall	2
EE-181	1	2017	Spring	1
FIN-201	1	2018	Spring	1
HIS-351	1	2018	Spring	1
MU-199	1	2018	Spring	1
PHY-101	1	2017	Fall	1