Siyi Guo 737008 Database System Assignment2

1. Count the number of students in each course at the University. Print the course name, as well as the number of students. (1 mark)

```
SELECT  Course.name, Count(Student.id) AS Number_of_Student
FROM  Student INNER JOIN Course
ON Student.course = Course.id
GROUP BY Course.name;
```

name	Number_of_Student
Bachelor of Design	9
Bachelor of Science	30
Master of Information Systems	28
Master of Information Technology	33

4 Rows

2. Is there any subject failed by more than one student? List the subject code as well as the number of failures. (1 mark)

```
SELECT CONCAT(area, yearlevel, code) AS Subject, COUNT(student) AS Number_of_Student
FROM StudentTakesSubject
WHERE result < 50 || result IS NULL
GROUP BY CONCAT(area, yearlevel, code)
HAVING COUNT(Student) > 1;
```

Subject	Number_of_Student_Failed
COMP20003	2
COMP90041	2
INFO20003	3

3 Rows

3. For the students who have completed at least one subject at undergraduate level, how many points does each student need to complete their degree? (2 marks)

```
SELECT Student, Course.creditpoints - SUM(if(result >= 50 , Subject.creditpoints, 0)) AS
Credit_to_Complete
FROM Course INNER JOIN Student INNER JOIN StudentTakesSubject STS INNER JOIN Subject
ON Course.id = Student.course AND Student.id = STS.student
AND STS.area = Subject.area AND STS.yearlevel = Subject.yearlevel AND STS.code = Subject.code
WHERE STS.yearlevel < 9
GROUP BY Student
```

Student	Credit_to_Complete
123006	237.5
123010	225.0
123011	262.5
123012	262.5
123018	225.0
123036	225.0
123037	300.0
123041	275.0
123055	225.0

9 Rows

4. List the student number, lastname, course and GPA of students who have completed more than 4 subjects at undergraduate level? (To calculate GPA you need to (1) multiply the student's result per subject by their credit points, (2) sum them up for all the subjects 3 the student has taken and (3) divide it by the sum of the credit points these subjects are worth) (2 marks)

```
SELECT id as student_number, lastname, course, SUM(result * creditpoints) / SUM(creditpoints) as GPA
FROM Student INNER JOIN StudentTakesSubject STS INNER JOIN Subject
ON id = student AND STS.area = Subject.area AND STS.code = Subject.code AND Subject.yearlevel =
STS.yearlevel
WHERE STS.yearlevel < 9
GROUP BY id
HAVING COUNT(STS.yearlevel) > 4;
```

student_number	lastname	course	GPA
123006	Belew	B-SCI	75.33333
123010	Bruton	B-SCI	64.50000
123018	Francia	B-SCI	77.33333
123036	Ketterman	B-SCI	75.50000
123055	Millner	B-SCI	73.50000

5 Rows

5. Which lecturer awarded the highest mark and what subject(s) was it (print the lecturer's full name, the mark and the entire subject code e.g. "INFO20003")? (2 marks)

```
SELECT CONCAT(firstname, " ", lastname) AS fullname, result, CONCAT(STS.Area, STS.yearlevel,
STS.code) AS subjec_code
FROM Lecturer INNER JOIN Subject INNER JOIN StudentTakesSubject STS
ON id = lecturer
AND Subject.area = STS.area AND Subject.code = STS.code AND Subject.yearlevel = STS.yearlevel
WHERE result = (SELECT MAX(result) FROM StudentTakesSubject);
```

fullname	result	subjec_code
Marv Jackson	94	COMP10002

1 Row

6. For each student who has completed COMP10001 print their name, result and their academic grade (H1,H2A etc). (2 marks)

```
SELECT CONCAT(firstname, ' ', lastname) AS name,  result,
(CASE
        WHEN result >= 80 THEN "H1"
        WHEN result >= 75 THEN "H2A"
        WHEN result >= 70 THEN "H2B"
        WHEN result >= 65 THEN "H3"
        WHEN result >= 50 THEN "P"
        WHEN result = 49 THEN "NH"
        WHEN result >= 0 THEN "N"
        ELSE "UNKNOWN"
END) AS AcademicGrade
FROM Student INNER JOIN StudentTakesSubject
ON Student.id = StudentTakesSubject.student
WHERE area = "COMP" && yearlevel = 1 && code = 0001;
```

name	result	AcademicGrade
Lon Belew	73	H2B
Wai Bruton	77	H2A
Roseline Francia	91	H1
Rudolf Ketterman	71	H2B
Shaunta Millner	74	H2B

5 Rows

7. Find the names of lecturers who teach at both undergraduate and postgraduate level. (3 marks)

```
SELECT id AS lecturer_id, CONCAT(firstname, ' ', lastname) AS lecturer_name
FROM Lecturer INNER JOIN Subject
ON id = lecturer
WHERE id IN (SELECT id FROM Lecturer INNER JOIN Subject ON id = lecturer WHERE yearlevel >= 9)
AND yearlevel <9
```

lecturer_id	lecturer_name
222222	Ada Lovelace
444444	Grace Hopper

2 Rows

8. List the lecturers who teach across all study areas. (3 marks)

```
SELECT Lecturer.id AS lecturer_id, CONCAT(firstname, ' ', lastname) AS lecturer_name
FROM Lecturer INNER JOIN Subject INNER JOIN StudyArea
ON Lecturer.id = Subject.lecturer AND StudyArea.id = area
GROUP BY Lecturer.id
HAVING COUNT(DISTINCT area) = (SELECT COUNT(DISTINCT name) FROM StudyArea)
```

lecturer_id	lecturer_name
444444	Grace Hopper

1 Row

9. Have any students from Gilberton suburb enrolled into Bachelor of Science course repeated a subject at undergraduate level? (3 marks)

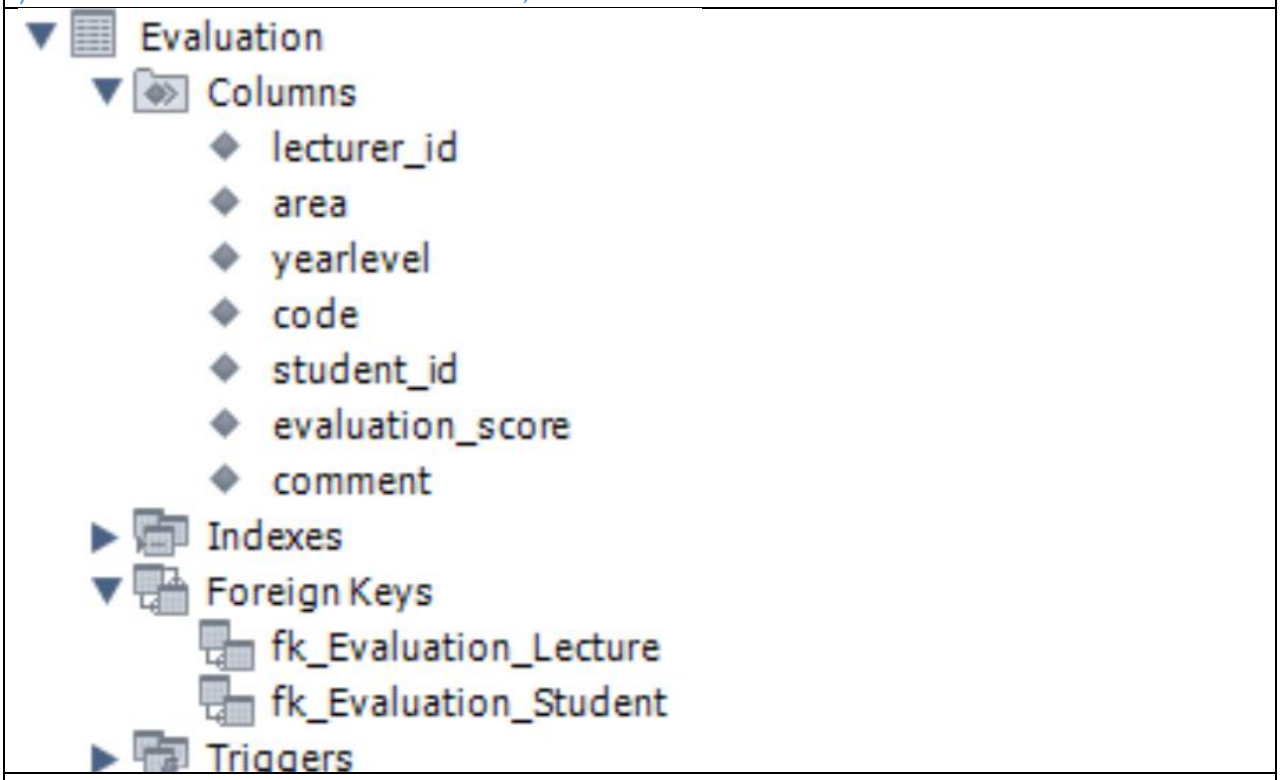
```
SELECT Student, CONCAT(firstname, ' ', lastname) AS Student_name
FROM StudentTakesSubject STS INNER JOIN Student INNER JOIN Suburb
ON STS.student = Student.id AND Student.postcode = Suburb.postcode
WHERE Student.course = "B-SCI" AND Suburb.name = "Gilberton" AND yearlevel < 9
GROUP BY Student
HAVING COUNT(DISTINCT CONCAT(area, yearlevel, STS.code)) < COUNT(CONCAT(area, yearlevel, STS.code))
```

Student	Student_name
123037	Fidelia Khang

1 Row

10. The Dean has asked you to design a table that will record the student evaluations for each lecturer for each subject he has taught in each academic semester. You are to write the DDL to create the table including all suitable attributes and write the references to the Foreign Keys. (1 mark)

```
CREATE TABLE `Evaluation` (
  `lecturer_id` mediumint(8) unsigned NOT NULL,
  `area` char(4) NOT NULL,
  `yearlevel` tinyint(3) unsigned NOT NULL,
  `code` char(4) NOT NULL,
  `student_id` mediumint(8) unsigned NOT NULL,
  `evaluation_score` tinyint(3) NOT NULL,
  `comment` text,
  PRIMARY KEY(`lecturer_id`),
  KEY `fk_Evaluation_Student` (`student_id`),
  KEY `fk_Evaluation_Subject1_idx` (`area`,`yearlevel`,`code`),
  CONSTRAINT `fk_Evaluation_Lecture` FOREIGN KEY (`lecturer_id`) REFERENCES `Lecturer` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `fk_Evaluation_Student` FOREIGN KEY (`student_id`) REFERENCES `Student` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,
  CONSTRAINT `fk_Evaluation_Subject1` FOREIGN KEY (`area`, `yearlevel`, `code`) REFERENCES `Subject` (`area`, `yearlevel`, `code`) ON DELETE NO ACTION ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```



0 Rows