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

SELECT Course.name AS CourseName, COUNT(Student.id) AS NumberOfStudents
FROM Student INNER JOIN Course on Course.id = Student.course
GROUP BY CourseName;

CourseName	NumberOfStudents
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

SELECT CONCAT(area, yearlevel, code) AS SubjectCode, COUNT(student) AS NumberOfFailures
FROM Subject NATURAL JOIN StudentTakesSubject
WHERE result < 50
GROUP BY SubjectCode
HAVING NumberOfFailures > 1;

SubjectCode	NumberOfFailures
INFO20003	2

1 Row

3. For the students who have completed at least one subject at <mark>undergraduate level</mark>, how many points does each student need to complete their degree?

SELECT student AS Student, (300 - SUM(creditpoints))AS PointsNeeded
FROM StudentTakesSubject NATURAL JOIN Subject
WHERE result >= 50
GROUP BY student;

Student	PointsNeeded
123006	237.5
123008	225.0
123009	262.5
123010	225.0
123011	262.5
123012	262.5
123018	225.0
123036	225.0
123039	275.0
123041	275.0

11 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 the student has taken and (3) divide it by the sum of the credit points these subjects are worth*)

SELECT id AS StudentNumber, lastName, Course,
SUM(result*creditpoints)/SUM(creditpoints) AS WAM
FROM Student INNER JOIN StudentTakesSubject ON id = student NATURAL JOIN Subject
WHERE result IS NOT NULL
GROUP BY id
HAVING COUNT(CASE WHEN yearlevel < 4 THEN 1 ELSE NULL END)> 4;

StudentNumber	lastName	Course	WAM
123006	Belew	B-SCI	75.33333
123010	Bruton	B-SCI	73.71429
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_WS(" ", firstname, lastname) AS lecturerName, result AS mark,
CONCAT(area, yearlevel, code) AS SubjectCode
FROM Lecturer INNER JOIN Subject ON id = lecturer NATURAL JOIN StudentTakesSubject
WHERE result IN
(SELECT MAX(result)
FROM StudentTakesSubject);

lecturerName	mark	SubjectCode
Mary Jackson	94	COMP10002

1 Row

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

SELECT CONCAT_WS(" ", firstname, lastname) AS studentName, result,
(CASE
WHEN result >= 80 then "H1"
WHEN result BETWEEN 75 and 79 then "H2A"
WHEN result BETWEEN 70 and 74 then "H2B"
WHEN result BETWEEN 65 and 69 then "H3"
WHEN result BETWEEN 50 and 64 then "P"
WHEN result <= 49 then "N" END) AS academicGrade
FROM Student INNER JOIN StudentTakesSubject ON id = student
WHERE CONCAT(area, yearlevel, code) = "COMP10001" ;

studentName	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 CONCAT_WS(" ", firstname, lastname) AS lecturerName
FROM Lecturer INNER JOIN Subject ON id = lecturer
where yearlevel >= 4 and Lecturer.id IN
(SELECT lecturer
FROM Subject
WHERE yearlevel < 4)
GROUP BY Lecturer.id;

lecturerName
Ada Lovelace
Grace Hopper

2 Rows

8. List the lecturers who teach across all study areas.

SELECT CONCAT_WS(" ", firstname, lastname) AS lecturerName
FROM Lecturer INNER JOIN Subject ON id = lecturer
where area in ("COMP", "ISYS", "INFO")
GROUP BY id
HAVING COUNT(DISTINCT area) = 3;

lecturerName
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 CONCAT_WS(" ", firstname, lastname) AS StudentName
FROM Student NATURAL JOIN Suburb INNER JOIN StudentTakesSubject ON id = student
WHERE name = "Gilberton" AND course = "B-SCI"
GROUP BY StudentName
HAVING COUNT(CONCAT(area, yearlevel, code)) > 1;

StudentName
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 StudentEvaluation
(
`lecturer` mediumint(8) unsigned NOT NULL,
`student` mediumint(8) unsigned NOT NULL,
`area` char(4) NOT NULL,
 `yearlevel` tinyint(3) unsigned NOT NULL,
 `code` char(4) NOT NULL,
 `rating` int (2) unsigned NOT NULL,
 `comments` varchar(250),
 PRIMARY KEY (`lecturer`,`student`,`area`,`yearlevel`,`code`),
 FOREIGN KEY (`student`) REFERENCES `Student` (`id`) ON DELETE NO ACTION ON UPDATE
NO ACTION,
 FOREIGN KEY (`area`, `yearlevel`, `code`) REFERENCES `Subject` (`area`, `yearlevel`, `code`)
ON DELETE NO ACTION ON UPDATE NO ACTION,
 FOREIGN KEY (`lecturer`) REFERENCES `Lecturer` (`id`) ON DELETE NO ACTION ON UPDATE
NO ACTION
 )ENGINE=InnoDB DEFAULT CHARSET=utf8;
```