**Lab Exercise 9**

This week in the lecture we have looked at transforming data into information, this often requires combining aggregation, sub-queries and functions. In this lab exercise you are required to work in small groups of 2 or 3 and generate code to extract the following information.

*Note: The queries require the use of the makeuniv database as in previous labs, by this point in the course you should be familiar with the database structure and the data held within it.*

1. The university needs to know which subjects have a high number of students failing, generate a query which will show management the subjects which have less than 10

students enrolled on them

SELECT subjectid, COUNT(studentid)

FROM Enrolled

GROUP BY subjectid

HAVING COUNT(studentid) < 10;

SELECT subjectid,COUNT(mark)

FROM (SELECT subjectid, COUNT(studentid)

FROM Enrolled

GROUP BY subjectid

HAVING COUNT(studentid) < 10)

INNER JOIN Marks USING (subjectid)

WHERE mark < 40

GROUP BY subjectid;

1. Adapt the query should display the results in a format that is more readable, displaying the subject name not the ID and concatenating the output with English to aid the reader.

SELECT sname || ‘ has ‘ || COUNT(studentid) || ‘ students’

FROM Enrolled INNER JOIN Subject USING (subjectid)

GROUP BY subjectid

HAVING COUNT(studentid) < 10;

SELECT sname || ‘ has ‘ || COUNT(studentid) || ‘ students’

FROM (SELECT subjectid, COUNT(studentid)

FROM Enrolled

GROUP BY subjectid

HAVING COUNT(studentid) < 10)

INNER JOIN Marks USING (subjectid) INNER JOIN Subject USING (subjectid)

WHERE mark < 40

GROUP BY subjectid;

1. Which program/major has the largest average (display only that program/major)

SELECT major, AVG(mark)

FROM Student INNER JOIN Marks USING (studentid)

GROUP BY major

HAVING AVG(mark) =

(SELECT MAX(avg\_mark)

FROM

(SELECT AVG(mark) AS avg\_mark

FROM Student INNER JOIN Marks USING (studentid)

GROUP BY major)

);

Alternative:

SELECT major, avg\_mark

FROM

(SELECT major, AVG(mark) AS avg\_mark

FROM Student INNER JOIN Marks USING (studentid)

GROUP BY major)

WHERE avg\_mark =

(SELECT MAX(avg)

FROM

(SELECT AVG(mark) AS avg

FROM Student INNER JOIN Marks USING (studentid)

GROUP BY major)

);

// selecting maximum val of avg\_mark after the table is created in the FROM statement

1. The timetable is held in the class table, management wish to know how many students are enrolled on each class (you may assume that students attend all classes for each subject that is timetabled). Write code that will display the day, time and subject in addition to the number of students allocated to that class slot. The output should be formatted as detailed below:

[subject name] running on a [day] at [time] has [number] students enrolled

SELECT sname || ‘ running on a ’ || day || ‘ at ’ || classtime || ‘ has ‘ || COUNT(studentid) || ‘ students enrolled’

FROM Class INNER JOIN Subject USING (subjectid) INNER JOIN Enrolled USING (subjectid)

GROUP BY sname;

You have approx 20 min to complete the above questions, the solutions will then be discussed as a group.