Advanced DML Queries

Today we will also work on the database about a university. The entity-relational model for this case study is depicted in Figure 1. All this work should be done in the MySQL database management system.

1 Database Schema

Remember that the structure of the University is as follows:

assessment (assessmentID, moduleID, weight, typeID, name, description)
assessmentType (typeID, name)
course (courseID, name, departmentID, years, credits)
department (departmentID, name, numCourses, city, address, postalCode)
mark (assessmentID, studentID, score, timestamp)
module (moduleID, name, course_id, credits, year)
student(studentID, fname, lname, dateOfBirth, phoneNo, address, city, postCode)
takesCourse(studentID, courseID)

2 Activities

1. List modules that belong to computing courses
2. Display the names of the departments that deliver some PhD course
3. How many students have higher scores than the average score of student 34? (Note there are 378)
4. How many students have higher scores than all the scores of student 34? (Note there are 37)
5. Display the names of the students that have higher marks than all the marks of student 34
6. Display the id of all departments that give more than one PhD course. (Note there is just one department that meets these criteria)
7. List the name of all courses in department 52 together with the number of credits that each course has. You must obtain the following output (the order of the rows may be different):

BSc in Neuroscience	360
MSc in Neuroscience	380
PhD in Neuroscience	20

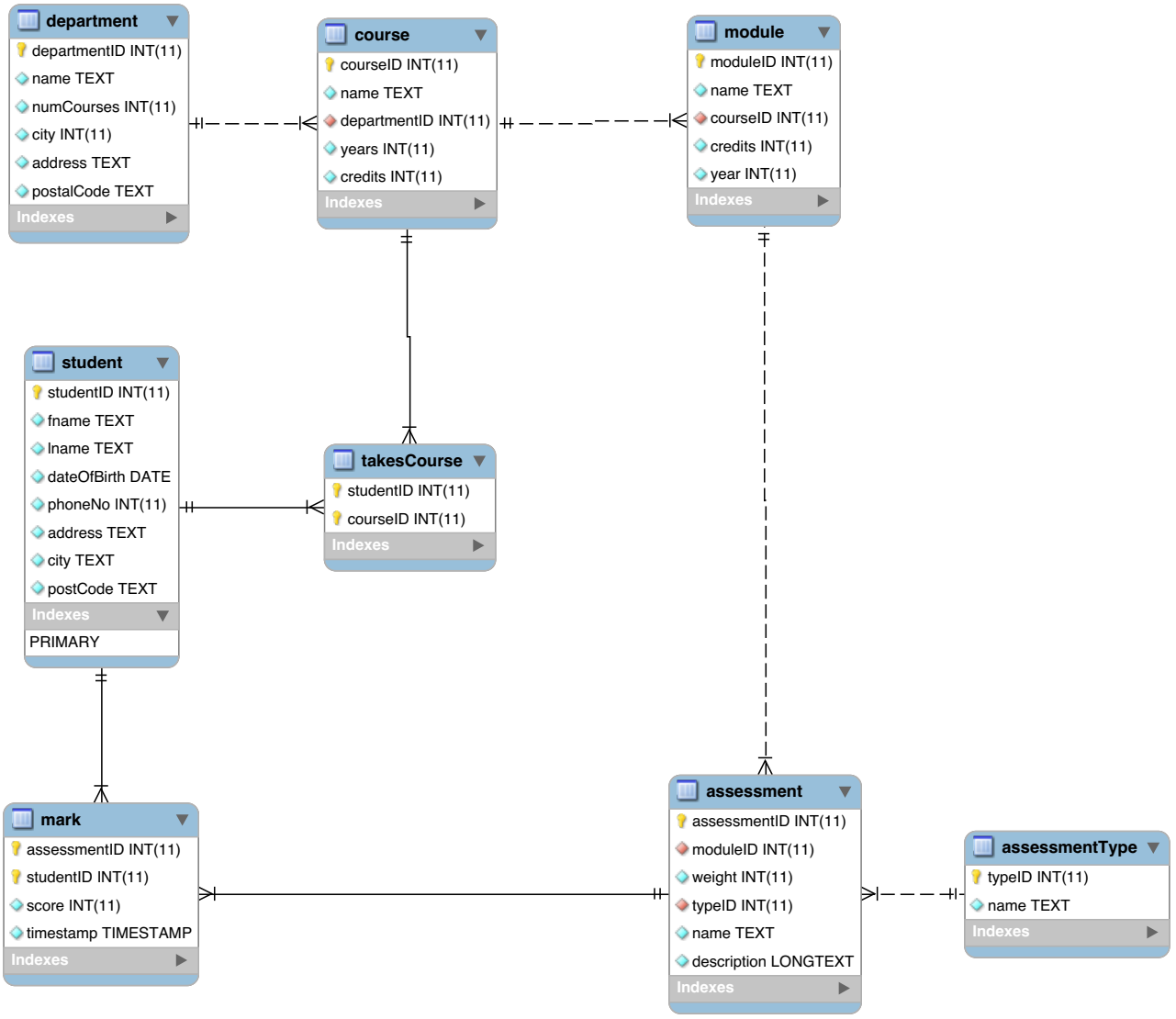
8. Display the names of the students that are not taking any course
9. List the name and description of all assessments of type 'Coursework'
10. What is the average mark for assessments of module 4638573? Solution: 34.667
11. List the description of all assessments and the maximum mark obtained by the students in this assignment. Note that all assignments should be displayed, even if they have not been marked.
12. What is the average mark for all the examinations? Solution 50.5958
13. List all the departments together with the names of the courses it offers
14. List the names of the students together with their scores

15. List the names of all assessments that have never been marked
16. List the names of the students together with their scores and the weights of each assessment
17. List the names of the students together with their pondered scores (score*weight/100)
18. List the students together with their mark for the modules in which they have been assessed
19. List the ID, the name and the number of students on each course
20. Calculate the number of students in the most taken course
21. List the name and the number of students of the most taken course
22. Calculate the final marks for student 164 in all modules on course 36. You must obtain the following output:

name	total marks
Module in BSc in Neuroscience 18	5.4000
Module in BSc in Neuroscience 17	35.6400
Module in BSc in Neuroscience 16	25.7500
Module in BSc in Neuroscience 15	35.5000
Module in BSc in Neuroscience 14	43.5000
Module in BSc in Neuroscience 13	17.5000
Module in BSc in Neuroscience 12	27.6000
Module in BSc in Neuroscience 11	26.0700
Module in BSc in Neuroscience 10	35.5000
Module in BSc in Neuroscience 9	49.0000
Module in BSc in Neuroscience 8	30.8000
Module in BSc in Neuroscience 7	29.2500
Module in BSc in Neuroscience 6	58.2500
Module in BSc in Neuroscience 5	29.8000
Module in BSc in Neuroscience 3	29.5000
Module in BSc in Neuroscience 2	45.0000
Module in BSc in Neuroscience 1	19.1400

23. There is an spelling error on the city of some students it has been defined as 'Lodon' but it should be 'London'. Use a SQL query to correct this mistake
24. The city of all departments have not been defined. But you can know the city from the address (that is the address indicates the name of the city where the campus is). Use SQL queries to correct this mistake.
25. A query using the HAVING clause has an equivalent formulation without a HAVING clause. For example, reformulate the following query without using a HAVING clause:

```
SELECT studentID,count(assessmentID)
FROM mark
GROUP BY studentID
HAVING count(assessmentID)=5;
```


department
departmentID INT(11)
name TEXT
numCourses INT(11)
city INT(11)
address TEXT
postalCode TEXT
Indexes
course
courseID INT(11)
name TEXT
departmentID INT(11)
years INT(11)
credits INT(11)
Indexes
module
moduleID INT(11)
name TEXT
courseID INT(11)
credits INT(11)
year INT(11)
Indexes
student
studentID INT(11)
fname TEXT
lname TEXT
dateOfBirth DATE
phoneNo INT(11)
address TEXT
city TEXT
postCode TEXT
Indexes
PRIMARY
takesCourse
studentID INT(11)
courseID INT(11)
Indexes
mark
assessmentID INT(11)
studentID INT(11)
score INT(11)
timestamp TIMESTAMP
Indexes
assessment
assessmentID INT(11)
moduleID INT(11)
weight INT(11)
typeID INT(11)
name TEXT
description LONGTEXT
Indexes
assessmentType
typeID INT(11)
name TEXT
Indexes