Dear Students,

In marking your submissions we used an auto-marker. The auto-marker is never complete and cannot be designed to check all code variations. However, it can save a lot of time in making sure that a submission is correct and aligns with the assignment requirements. We understand that in many cases the auto-marker would fail even if a small error exists in the code. For these reasons we ONLY used the auto-marker as a hint, to help in the evaluation of your assignment. In many cases we had to go on to your code and look, even fix minor errors to make sure your submission passes the auto-marker correctly. In all cases your final mark is equal or higher than the one indicated by the auto-marker, BUT never lower.

We designed the auto-marker so that failure to a specific test would automatically create a reported feedback to your report. This is the case for all students that their code was not throwing Exceptions. So in cases where you didn't get all points, there is a small report of the test it failed. That would be enough to explain why points were cut. In cases that there were exceptions thrown early in the auto-marker, your assignment was assigned zero points due to major programming/compilations issues.

If your assignment falls in this category and you think/know that there is a minor issue that was causing the auto-marker to fail, then you should submit a remarking request where you suggest specific steps to take or changes to make in order to overpass the compilation/exception problems. We will then fix the problem and/or look into your java code and make a judgment of its correctness and completeness, looking more carefully parts of the code that the auto-marker would check.

PART I - Interactive SQL

Regarding the Interactive SQL part of your A2, the auto-marker was applied on each implemented query. Please find below an overview of the tests run by the auto-marker and points gained/lost for each of them. We assumed that a method is correct (i.e., deserves full marks) and for each test failed we were deducting some portion denoted as (-1, -3, etc.) below. Note that these are not mutual exclusive tests, so they are not meant to add to the full marks. A set of if/else conditions were applied to check most of the cases.

=============================================

SQL Auto-marker Scores

=============================================

# Query 1-10

(-1) Minor Errors (not captured by the auto-marker: formatting, wrong attribute names, etc.)

(-3) The actual answer differs significantly from the expected answer OR the query is structurally correct but minor issues exist, so a subset/superset of correct answers is returned.

(-5) No answer, random output or zero effort

=============================================

PART II - Embedded SQL

Regarding the JDBC part of your A2, the auto-marker was applied on each implemented java method. Please find below an overview of the tests run by the auto-marker and points gained/lost for each of the cases tested. There are 10 methods in total, each weighting 5 points, for a total of 50 marks. The points near each single test represent the weight of the particular test.

For example, for testing the listCountryLanguages() method we had 3 separate tests, all with its weight (+1, +3, +1). Your score in each method is the sum of these weights. For example, if you pass the two first tests your score is 4/5 in listCountryLanguages().

=============================================

JDBC Auto-marker Scores

=============================================

# connectDB

(+5) Connected to the database

# insertCountry

(+2) Inserts valid country

(+1) Did not insert an invalid country

(+1) Inserts could be verified

(+1) Passed above all test cases

# getCountriesNextToOceanCount

(+3) Returned appropriate values for existing country.

(+2) Returned appropriate values for non-existing players.

# getOceanInfo

(+3) Returned appropriate records for existing country

(+2) Returned empty String for non-existent players

# deleteNeighbour

(+2) Deletes a valid country

(+2) Did not delete a non-existing country

(+1) Deletion could be verified

# listCountryLanguages

(+1) The number of returned tuples is the expected one.

(+3) Listed correct the players of the specific football club.

(+1) Did not list players of a football club that does not exist.

# chgHDI

(+3) Update the hdi table successfully with valid value

(+2) Did not change hdi table with invalid value

# updateHeight

(+2) If the return value is true.

(+3) Updates the value successfully.

# updateDB

(+2) The number of returned tuples is the expected one.

(+3) All values in tuples are correct.

# disconnectDB

(+5) disconnected from the database

=============================================