Applied Databases

Higher Diploma in Science in Data Analytics

1   Description

This document describes the final project specification for the Applied Databases module.

2   Marks

This project is worth 60% of the marks for the module.

2.1   Marking Scheme

85% of the marks will be awarded for implementing the functionality described in this document.

15% of the marks will be awarded for innovation and extra functionality.

Please describe your innovation (if any) in a document entitled *innovation.doc* or *innovation.pdf* which should be stored in the *Innovation* folder of your project.

NOTE: You may be invited to an MS Teams meeting for a viva explanation of any or all parts of your submission.

2.1.1   Plagiarism

Plagiarism will be dealt with in accordance with the institute's Plagiarism policy.

3   Submission

Deadline for submissions is **Tuesday May 4th 2021 at 9:00am**.

- Firstly, download the file GXXXXXXXX.7z from the *MongoDB & MySQL* section of Moodle.
- Unzip it.
- Rename the unzipped folder from GXXXXXXXX to your Student Number e.g. G12345678
- The folder contains 4 sub-folders:
  - **Innovation**
    Write a Word/PDF document explaining any innovation/extra functionality you provided and place in this folder.
    If none – just leave folder empty.

  - **Mongo-Queries**
    This folder contains 4 files, corresponding to each MongoDB question.
    Write only the exact MongoDB command for each question into the appropriate file.

  - **MySQL-Queries**
    This folder contains 4 files, corresponding to each MySQL question.
    Write only the exact MySQL command for each question into the appropriate file.

  - **PythonApp**
    Write your Python App in this folder.

- When you are finished, compress the folder – which is now called your Student number (using Winzip or 7zip) and upload to Moodle before the deadline.

4   Functionality

4.1   MySQL

See *Questions.pdf* in the *MongoDB & MySQL* section of Moodle.

4.2   MongoDB

See *Questions.pdf* in the *MongoDB & MySQL* section of Moodle.

4.3 Python

The following python application should be based on the following databases:

- MySQL
  Download *moviesDB.sql* from Moodle and import into MySQL. The database is called **moviesDB**.
  When connecting to the MySQL database use the username *root* and *no* password (password="").

- Mongo
  Download *movieScriptsDB.json* from Moodle and import into MongoDB.
  The database <u>must</u> be called **movieScriptsDB**, and the collection <u>must</u> be called **movieScripts**.

4.3.1 Python Application

Write a python application that displays a main menu as follows:

```
Movies DB
---------

MENU
====
1 - View Films
2 - View Actors by Year of Birth & Gender
3 - View Studios
4 - Add New Country
5 - View Movie with Subtitles
6 - Add New MovieScript
x - Exit application
Choice:
```

Figure 1 Main Menu

The choices are as follows:

- **1 (View Films)**
  The user is shown the list of Film Names (in alphabetical order) and the Names of the actors (in alphabetical order) who starred in the film in groups of 5:

```
Choice: 1

Films
-----
Armageddon    :  Bruce Willis
Armageddon    :  Billy Bob Thornton
Armageddon    :  Ben Affleck
Armageddon    :  Liv Tyler
Armageddon    :  Steve Buscemi
-- Quit (q) --
```

Figure 2 First group of films & actors

If the user presses any key except *q* the next 5 Film Names and Names of actors who starred in the film are shown:

```
Choice: 1

Films
-----
Armageddon    :  Bruce Willis
Armageddon    :  Billy Bob Thornton
Armageddon    :  Ben Affleck
Armageddon    :  Liv Tyler
Armageddon    :  Steve Buscemi
-- Quit (q) --
Armageddon    :  Owen Wilson
Armageddon    :  Michael Clarke Duncan
Armageddon    :  Peter Stormare
Around the World in 80 Days  :  Jackie Chan
Around the World in 80 Days  :  Steve Coogan
-- Quit (q) --
```

Figure 3 Next group of films & actors

7

And so on until the user presses *q*:

Figure 4 All films have been retrieved from the database

Whenever the user presses *q* he/she is returned to the Main Menu.

- **2 (View Actors by Year of Birth & Gender)**

  The user is asked to enter a year:

  Figure 5 Enter Year of Birth

  When a year is entered, the user is then asked to enter a Gender:

  Figure 6 Enter Gender

  When a Gender is entered, the name, birth month and Gender of all actors born the given year, with the given Gender, are displayed and the user is returned to the Main Menu.

  Figure 7 Female actors born in 1959

  If an invalid year is entered, the user is prompted again until a valid year is entered (a valid year is any integer).

  If an invalid Gender is entered, the user is prompted again until a valid gender (Male/Female) is entered.

  If no Gender is entered, the name, birth month and Gender of all actors born the given year, of either Gender, are displayed and the user is returned to the Main Menu.

  Figure 8 Male & Female actors born in 1971

- **3 (View Studios)**

  The user is shown the id and name of all studios, in ascending studio id order.

```
Choice: 3

Studios
--------
1    :    Universal Pictures
2    :    Paramount Pictures
3    :    Walt Disney Pictures
4    :    20th Century Fox
5    :    Columbia Pictures
6    :    Warner Bros. Pictures
7    :    Touchstone Pictures
8    :    Dreamworks
9    :    ImageMovers
10   :    Disney Pixar
11   :    MGM
12   :    Chris Lee Productions
13   :    Lucasfilm
14   :    New Line Cinema
15   :    Carolco Pictures
16   :    Miramax Films
17   :    Jerry Bruckheimer Films
18   :    Gaumont
19   :    Revolution Studios
20   :    Imagine Entertainment
21   :    Morgan Creek Productions
22   :    United Artists
23   :    Dimension Films
24   :    Braveworld Productions
25   :    Icon Productions
26   :    Carolina Bank
27   :    Tig Productions
28   :    Dune Entertainment
29   :    Beijing New Picture Film Co.
30   :    Big Talk Productions
31   :    Malpaso Company
32   :    British Film Council
33   :    Channel Four Films
34   :    Constantin Film
35   :    Bavaria Film
36   :    Bayerischer Rundfunk
37   :    RKO Radio Pictures
38   :    Toho Company
39   :    Kathbur Pictures
40   :    Avalon Studios
```

Figure 9 Studios

- **4 (Add New Country)**

  The user is asked to enter a Country ID:

  Figure 10 Enter Year of Birth

  When a Country ID is entered, the user is then asked to enter a Country Name:

  Figure 11 Enter Country Name

  When a Country Name is entered the country details are added to the database and the user is returned to the main menu:

  Figure 12 Country added to database

  If an invalid Country ID is entered, the user is prompted again until a valid Country ID is entered (a valid Country ID is any integer).

  If no Country Name is entered, the user is prompted again until Country Name is entered.

  If the ID and/or Country Name entered already exist in the database, the user should be informed of this.

  Figure 13 Country ID already exists

  Figure 14 Country Name already exists

- **5 (View Movies with Subtitles)**

  The user is asked to enter a Subtitle Language. If none has been entered, the user is again prompted to enter it.

  When a subtitle language has been entered, all movieScripts (in the **movieScripts** collection in the **movieScriptsDB** database in MongoDB) are found and the corresponding FilmName and the first 30 characters of the FilmSynopsis are shown.

  Figure 15 FilmName and partial FilmSynopsis of films with Spanish subtitles

- **6 (Add New MovieScript)**

The user is asked to enter a Film ID. If none has been entered or an invalid Film ID has been entered, the user is again prompted to enter it. (A valid Film ID is any integer).

The user is then prompted to enter a number of keywords.
When *-1* has been entered the user is then asked to enter a number of Subtitles Languages, when *-1* has been entered the new MovieScript is added to the **movieScripts** collection in the **movieScriptsDB** database in MongoDB.

```
Choice: 6

Add New Movie Script
--------------------
ID :
ID : asdfd
ID : 6
Keyword (-1 to end): ape
Keyword (-1 to end): new york
Keyword (-1 to end): -1
Subtitles Language (-1 to end): English
Subtitles Language (-1 to end): Spanish
Subtitles Language (-1 to end): -1

MovieScript: 6  added to database
```
Figure 16 Add New Movie Script

```
{ "_id" : 6, "keywords" : [ "ape", "new york" ], "subtitles" : [ "English", "Spanish" ] }
```
Figure 17 Add New Movie Script document added to movieScripts collection in the movieScriptsDB database

If a Film ID is entered that already exists in the movieScripts collection in the movieScriptsDB, the following error is returned:

```
Choice: 6

Add New Movie Script
--------------------
ID : 1
Keyword (-1 to end): -1
Subtitles Language (-1 to end): English
Subtitles Language (-1 to end): -1

*** ERROR ***: Movie Script with id:  1   already exists
```
Figure 18 Add New Movie Script – MovieScript with ID 1 already exists in movieScriptsDB.

If a Film ID is entered that does not exist in the moviesDB database in MySQL, the following error is printed, and the MovieScript is not added to the **movieScripts** collection in the **movieScriptsDB** database in MongoDB:

```
Choice: 6

Add New Movie Script
--------------------
ID : 999
Keyword (-1 to end): cars
Keyword (-1 to end): -1
Subtitles Language (-1 to end): -1

*** ERROR ***: Film with id:  999  does not exist in moviesDB
```
Figure 19 Add New Movie Script – Film with ID 999 does not exist in moviesDB.

- **x (Exit Application)**
  The program terminates.

- **Anything Else**
  The menu is shown again.

NOTES

- For menu option 3 (View Studios), the information should be read from the database **only once.**
  E.g. If the user chooses 3 (View Studios) the studios are read from the database and stored in the program.

  If the user chooses option 3 again, the information is **not** read from the database again. Instead, the information read the first time option 3 was chosen is used.

4.4   Testing

The MySQL and MongoDB sections are a pass/fail basis. Either you get all the marks or none.

You can test your answers as follows:

- Download the file **ExpectedQueryResults.7z** from the *MongoDB & MySQL* section of Moodle and unzip it.
- It contains two folders:
  - MySQL
    This has 4 files which each of which has the correct output for the corresponding MySQL question.

  - MongoDB
    This has 4 files which each of which has the correct output for the corresponding MongoDB question.

4.4.1 How to test your MySQL queries

- Write your MySQL query in the MySQL console.
- When you think its correct copy the query to appropriate file in the MySQL-Queries folder of your answer folder.
- Run the following command from the Windows command line:

```
mysql.exe -u root -proot moviesDB < MySQLQA.txt > MySQL-myAnsA.txt
```

mysql.exe is the location of mysql.exe e.g. "\Program Files\MySQL\MySQL Server 8.0\bin\mysql.exe"

-u root is the username, in this case root

-proot is the password, in this case root (no space between p and the password)

< The less than symbol means that the contents of the next file mentioned will be used as input to the mysql.exe command.

MySQLQA.txt is the location of the file with your MySQL query for this question e.g. "C:\Users\GHarrisson\Documents\GXXXXXXXX\MySQL-Queries\MySQLQA.txt"

> The greater than symbol means that the output from the mysql.exe command should be written to the file mentioned next

MySQL-myAnsA.txt is the location of the file your query result will be written to e.g. "C:\Users\GHarrisson\Documents\MyAnswers\MySQL-myAnsA.txt"

Figure 20 Creating MySQL result file

- Compare your answer with the correct answer:

```
fc MySQL-ANSA.txt MySQLQA-Ans.txt
```

fc the file tool compare program in windows.

MySQL-myAnsA.txt is the location of the file containing your query result.

MySQLQA-Ans.txt is the location of the correct answer for this query.

Figure 21 Checking your MySQL result with actual result

- If the result of the fc command is not **FC: no differences encountered** no marks will be awarded for the question.

4.4.2 How to test your MongoDB queries

- Write your MongoDB query in the MongoDB shell.
- When you think its correct copy the query to appropriate file in the Mongo-Queries folder of your answer folder.
- Run the MongoDB command from the Windows command line as follows:

`mongo.exe employeesDB < MongoDBQA.txt > Mongo-Q421ANS.txt`

`mongo.exe` is the location of mongo.exe e.g.

"\Program Files\MongoDB\Server\4.2\bin\mongo.exe"

`movieScriptsDB` is the name of the Mongo database

< The less than symbol means that the contents of the next file mentioned will be used as input to the mongo.exe command.

`MongoDBQA.txt` is the location of the file with your Mongo query for this question e.g. "C:\Users\GHarrisson\Downloads\GXXXXXXXX\Mongo-Queries\MongoDBQA.txt"

> The greater than symbol means that the output from the mongo.exe command should be written to the file mentioned next
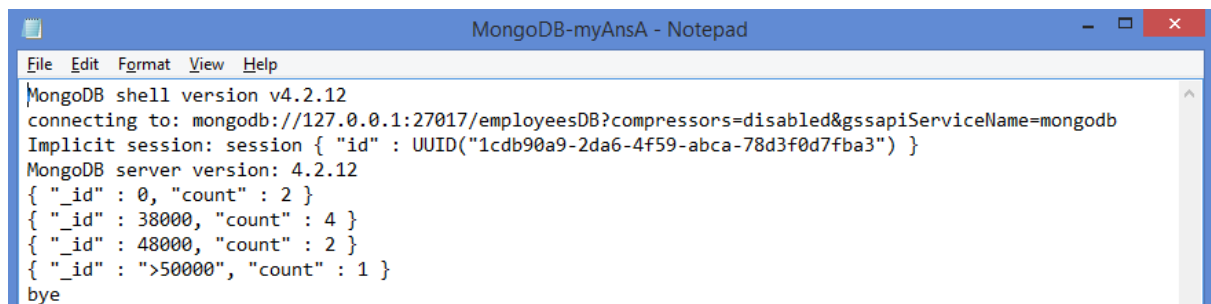
`MongoDB-myAnsA.txt` is the location of the file your query result will be written to e.g. "C:\Users\GHarrisson\Downloads\MyAnswers\MongoDB-myAnsA.txt"

```
C:\Users\Gerard>"\Program Files\MongoDB\Server\4.2\bin\mongo.exe" movieScriptsDB < C:\Users\Gerard\Downloads\GXXXXXXXX\Mongo-Queries\MongoDB
QA.txt > C:\Users\Gerard\Downloads\MyAnswers\MongoDB-myAnsA.txt
```
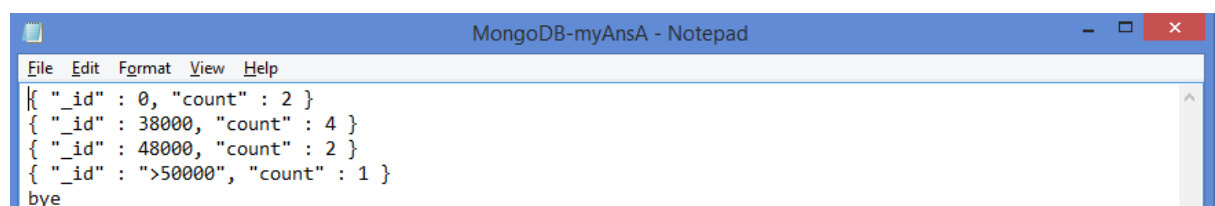
Figure 22 Creating Mongo Result file

- The first 4 lines of your query result file (**`MongoDB-myAnsA.txt`**) will have to be deleted so the fc command will only compare the actual output.

```
MongoDB shell version v4.2.12
connecting to: mongodb://127.0.0.1:27017/employeesDB?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("1cdb90a9-2da6-4f59-abca-78d3f0d7fba3") }
MongoDB server version: 4.2.12
{ "_id" : 0, "count" : 2 }
{ "_id" : 38000, "count" : 4 }
{ "_id" : 48000, "count" : 2 }
{ "_id" : ">50000", "count" : 1 }
bye
```

Figure 23 Mongo Query file before removal of first 4 lines

```
{ "_id" : 0, "count" : 2 }
{ "_id" : 38000, "count" : 4 }
{ "_id" : 48000, "count" : 2 }
{ "_id" : ">50000", "count" : 1 }
bye
```

Figure 24 Mongo Query file after removal of first 4 lines

- Compare your answer with the correct answer:

  `fc MongoDB-ANSA.txt MongoDBQB-Ans.txt`

  fc the file tool compare program in windows.

  MongoDB-myAnsA.txt is the location of the file <u>your</u> query result.

  MongoDBQA-Ans.txt is the location of the <u>correct</u> answer for this query.

```
C:\Users\Gerard>fc C:\Users\Gerard\Downloads\MyAnswers\MongoDB-myAnsA.txt C:\Users\Gerard\Downloads\ExpectedQueryResults\MongoDB\MongoDBQA-A
ns.txt
Comparing files C:\USERS\GERARD\DOWNLOADS\MYANSWERS\MongoDB-myAnsA.txt and C:\USERS\GERARD\DOWNLOADS\EXPECTEDQUERYRESULTS\MONGODB\MONGODBQA-
ANS.TXT
FC: no differences encountered
```

Figure 25 Checking your Mongo result with actual result

- If the result of the fc command is not **FC: no differences encountered** no marks will be awarded for the question.