Assessment Task Information

|  |  |
| --- | --- |
| Key details: | |
| **Assessment title:** | Practical Assignment: Coursework |
| **Module Name:** | Database Design and Implementation |
| **Module Code:** | IY466 |
| **Tutor’s Name:** | Dr Joseph Bolarinwa |
| **Assessment will be set on:** | 6/3/24 |
| **Assessment is due on:** | 7/4/24 |
| **Assessment weighting:** | 50% |

|  |
| --- |
| **Assessment Instructions** |
| What do you need to do for this assessment?  -You are required to attempt to cover all sections in the report. The aim of the report is to show your in-depth knowledge of database design and implementation and your ability to apply your knowledge by writing a report showing the careful steps undertaken to create a functional database.  -Before you start writing your report, you need to study all the material provided for students, e.g. the main textbook, lecture notes and the material uploaded via VLE.  - You have been tasked with designing and implementing a database structure to store information about a list of **movies** scraped from IMDb.  The CSV file is located with this coursework brief, titled: **movies.csv.** Also, it can be found at <https://www.kaggle.com/datasets/danielgrijalvas/movies>.  Inspect the csv file and carefully plan how you are to split this unnormalised data into a relational database which can be efficiently queried using SQL statements.  -Normalise the database to 1NF and then 2NF, continue to 3NF if you see fit.  Include your steps and workings in your report, explaining your process.  (You do not have to draw out tables, just describe your steps and write down your entities and attributes once it is fully normalised.)  -Draw your normalised database in an entity-relation diagram.  You may use any diagram drawing software of your choice or alternatively, you could draw the diagram by hand and scan your work.  -Implement the database in the **MYSQL** using python programming.  In your report, you must show screenshots of your finished tables as well as the **SQL+Python** code you used to create the database.  -Import the dataset into your database application by using **Python** code.  You must figure out a solution to splitting up the dataset ready to import to your relational tables.  Show screenshots of your imported data in your report, including the number of records successfully imported into your tables.  -Write and execute 5 SQL statements to find the following:  1. The average, minimum and maximum budget of drama movies.  2. The minimum budget of each genre.  3. The average score of each rating (ignore not rated movies).  4. The average runtime of all movies between the years 2014 and 2016.  5. The number of movies released before 2000.  Include your SQL statements and screenshots of their results in your report.  Good SQL syntax and practices are required for full marks here.  Write a reflective report of ~500 words detailing where you think the coursework went well and where you think you could improve if you were to do it again.  Analyse the techniques you took over the course of this project. How would you change your approach if you were to undertake a similar task again?  - The assignment specification and marking criteria will be released online via VLE. You can download the assignment specification. After finishing the report, you must upload the report via the VLE module page through two links. The first link is called “Submit via Turnitin,” and the second link is called “the Assignment”. Also, you are required to send a copy of the report to your tutor.   * It is the student’s responsibility to submit a low similarity report (i.e. not more than 25%). * Make sure that the submitted document is in PDF format. Otherwise, it will not be accepted. |
| **Structure:**   1. Title page with student name and student ID 2. Introduction and context 3. Normalisation process 4. Entity-relationship diagram 5. SQL database Implementation using Python 6. Data importing using Python 7. SQL statements to show correct implementation 8. Reflective Segment. 9. References   The total marks of the report are 100. |
| **Theory and/or task resources required for the assessment:**  **The** report should **cover** topics regarding relational databases, design and all SQL programming learned up to this date**.** |
| **Referencing style:**  You are encouraged to use Harvard referencing style. you can refer to the following websites  <https://www.uwe.ac.uk/study/study-support/study-skills>  <https://www.uwe.ac.uk/study/study-support/study-skills/referencing/introduction-to-referencing> |
| **Expected word count:**  ~1500. |
| **Learning Outcomes Assessed:**  Explain the principles of relational databases, raltional algebra and SQL and be able to demonstrate a proficiency in writing Structured Query Language  Describe the stpes required to design a dta base, within the systems development life cycle and be able to differentiate among conceptual, logical and physical database design  Explain the different approaches to data and object modelling by producing models for some simple problems  Create relatively complex SQL database and use websites to interface with these databases |
| **Submission Requirements:**  You must submit your report by using your computer. Then, you must submit the assessment electronically via the VLE module page through the Assignment activity that called “Submit via Turnitin”.   * you must submit your report online via the VLE before the deadline * You must send a copy of the report via email to your tutor. * The document must be in PDF format.   Assessment violating any of the requirements above will incur penalties or will not be accepted. |
| **How will this assessment be marked?**  **The assignment will be marked using the following criteria:**  **Introduction and context (5%):** how successfully you managed to explain an overview of your tasks  **Normalisation of your database (15%):** how accurately you were able to break the data into a relational model. Make sure you justify your decisions and take care of your process.  **Drawing an entity-relation diagram (10%)**: how consistent was your diagram with your normalised database? Did you include everything needed from an E-R diagram, and would a database designer be able to understand the structure of your database without seeing a finished product?  **Implementation of your database (20%)**: Were your tables created properly, taking into account constraints and correct data types, with appropriate names for your database and tables? Make sure that you explain the Python code that creates the table.  **Importing of data (10%)**: Data is imported in the correct tables and is split up in accordance with your relational database. Make sure that you explain the Python code that reads data from csv file and puts it into the database table.  **SQL Statements (15%):** Each statement has been attempted with correct outputs and clear evidence of such.  **Reflective report (10%)**: how well you were able to reflect on your overall implementation, understanding where you may improve if you were to undertake this coursework again.  **References (5%)**: Using proper and related references and following the Harvard referencing style  **Language and structure (10%)**: Your report is well structured, with clear language and justification of each of your steps.  The total marks are 100, and it weighs 50% of the module. |
| **How will you get feedback?**  You can get the result after it is moderated. |