Faculty of Computing, Engineering and Science Assessment Brief

Module Title: **Advanced Databases and Modelling**

Module Code: **IS3S662**

Module Leader: **Joseph Griffiths**

Assessment Type: **Set exercise / tasks - not-time constrained**

Assessment Title: **Milestone 2**

Weighting: **30%**

Word count: **N/A**

Submission Date: **13/12/2024 by 23:59**

Return Date: **21/01/2025**

## Assignment Details

Before you begin, please refer to the appendix for the setup of the database.

You must create the tables shown in the appendix.

1. Populate the tables by writing a procedure that inserts a new record into the database (the records shown in the appendix must be used). The procedure must:
   * Check for reasonable inputs.
   * Put a new record in the “course” table
   * Put a corresponding new record in the “modules” table.
2. Write a procedure that retrieves a count on the number of courses and modules.
3. Write a procedure getCourseDetails which accepts a Course ID and returns the Course Title, Course Leader, Description, Date Modified and number of credits. The main block should call the procedure with a course id and output the course’s details.
4. Write a PL/SQL block which utilises the getCourseDetails procedure and prints the data for each record.
5. Write a procedure that deletes a course and all corresponding modules from the database.
6. Write a trigger that reports how many modules are present after any insert/update/delete operation.

**Important:** For each question, you are expected to show your input and output. This is expected in the form of a screenshot. You are expected to detail what your code is demonstrating. Please ensure a script of your code is attached at the end of your submission.

## Appendix: Database Design and Setup

**Course Table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Course ID** | **Title** | **Description** | **Course Leader** | **Date Modified** | **Number of Credits** |
| C001 | Computing | Focuses on developing skills that employers demand for roles in the IT industry. | Richard Jones | 20-APR-2024 | 120 |
| C002 | Business | Focuses on management and looks at the different functions of business and how they inter-relate. | Thomas Page | 03-MAR-2021 | 120 |
| C003 | History | Focus on modern history, from about 1450 to the present day, including regional, national, European, American and global perspectives. | Adam Richards | 12-MAY-2023 | 120 |

**Module Table**

|  |  |
| --- | --- |
| **Course ID** | **Module ID** |
| C001 | IS1S464 |
| C001 | IS3S662 |
| C002 | BS1S737 |
| C002 | BS3S374 |
| C003 | HS3S773 |
| C003 | HS2S484 |

## The SQL to create the tables:

*CREATE TABLE course (*

*course\_id VARCHAR2(13) NOT NULL PRIMARY KEY,*

*title VARCHAR2(200),*

*description VARCHAR2(2000),*

*course\_leader VARCHAR2(200),*

*date\_modified DATE,*

*number\_of\_credits NUMBER*

*);*

*CREATE TABLE modules (*

*module\_id VARCHAR2(100) NOT NULL PRIMARY KEY,*

*course\_id VARCHAR2(13) REFERENCES course (course\_id)*

*);*

## Guidance on Format of Assessment

**Note:** Students are reminded not to include this assignment brief with the assignment submission. Please ensure the document you submit is in the format of a Word file, that your work is structured appropriately (i.e. Question 1 Input, Output, Explanation …), and the code you have used to answer the questions is pasted in at the end of your submission.

## Learning Outcomes Assessed

* To critically evaluate the requirements and subsequently design an appropriate solution for a problem of defined scope using advanced design/modelling techniques in non-trivial situations.
* To produce an appropriate solution for a problem of defined scope using advanced design/modelling techniques in non-trivial situations.

**Note:** See the next page for marking criteria.

## Marking Criteria/Rubric

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Fail (0-39) | Pass (40-59) | Merit (60-69) | Distinction (70+) | Mark |
| Question 1: Input the data (10%) | Data is mostly incorrect or missing.  No attempt to use a procedure for data insertion. | Data is inputted but contains significant errors (i.e. wrong values, incomplete entries).  Data is inserted directly without using a procedure. | Data is mostly accurate with only minor errors (i.e. few incorrect or missing values).  Data is inserted using a procedure for both tables, but the procedure has some minor errors. | All the data is accurately and completely inputted.  Data is inserted for both tables using a well-structured procedure.  Code is well-commented. |  |
| Question 2: Create a Count (10%) | Count is incorrect or not attempted.  The function may fail to execute or return an incorrect count. | Count function works but contains significant errors (i.e. incorrect total).  The output is incorrect due to errors in the query. | Count function is mostly correct with minor errors.  The function accurately counts for one table (i.e. the number of courses but not the number of modules). | Count function is fully accurate and works correctly for all relevant tables (i.e. performs a count for both the courses and the modules).  Code is well-commented. |  |
| Question 3: Procedure for course details (10%) | Procedure is either not created or is incorrect.  The procedure does not execute or produce meaningful output. | Procedure is created but contains significant errors (i.e. incorrect SQL).  The procedure executes but does not return an accurate output of course details. | Procedure is mostly correct with some minor errors (i.e. slight inaccuracies in the data returned).  The procedure returns most relevant course details. | Procedure is correct and returns all relevant course details.  Includes validation for inputs and handles exceptions.  Procedure is well optimised; code is well-commented. |  |
| Question 4: Utilising a procedure to print all records (20%) | Procedure is not created or is incorrect.  Does not print any records or does not execute properly. | Procedure is created but with significant errors (i.e. missing or incorrect data fields/missing record set).  The procedure prints records but the data is inaccurate. | Procedure works with minor errors (i.e. slight formatting errors).  Prints all records required for the tables, some non-critical data is missing. | Procedure accurately prints all records from the required tables without error.  Output is well formatted, clear and code is well-commented. Procedure is efficient. |  |
| Question 5: Creating a procedure to delete records (10%) | Procedure is incorrect, missing or fails to execute.  The procedure may cause unintended side effects such as deleting incorrect records. | Procedure is created but contains significant errors (i.e. partial deletions).  Records are deleted, but not all intended records are removed. | Procedure mostly works but with some minor errors (i.e. deletes course record, but not all copies).  The procedure successfully deletes records as intended in most instances, with some minor inaccuracies. | Procedure accurately and completely deletes all intended records without some errors.  Procedure is optimised and is well-commented. |  |
| Question 6: Creating a trigger (20%) | Trigger is incorrect or not created.  Trigger fails to execute, or it does not produce the expected outcome (e.g. no statement is printed). | Trigger is created but contains significant errors (i.e. only partially functional).  Trigger works for one operation (i.e. insert) but fails for the others (i.e. update and delete). | Trigger mostly works with some minor errors (i.e. some operations may not trigger correctly).  Trigger may have some slight inaccuracies. | Trigger functions correctly and efficiently for all intended operations (insert, update and delete).  Code is well-structured and is well-commented.  Trigger is optimised for performance and reliability. |  |
| Report (20%) | Report is incomplete, poorly written or lacks clarity.  Output is not shown, and explanations of the PL/SQL statements are missing or unclear. | Report covers basic elements but lacks detail and clarity.  Output is shown, but the explanations are brief and do not fully justify the PL/SQL statements. | Report is clear and mostly complete, with some minor errors.  The output of the PL/SQL statements are shown with screenshots, and the script is included at the end of the report.  The report is structured, but some parts may lack clarity. | Report is comprehensive, well-structured and error free.  Includes clear screenshots of both the input and the output, with detailed explanations.  Code is attached at the end and each question is clearly headed.  The report shows a deep understanding of the tasks and presents the work professionally. |  |

**Note**: All grades are provisional until they are ratified by the exam board.

## Submission Details

Please submit your report in the 'Assessment Submission' tab on the modules Blackboard page. Your submission needs to be in word format with your student ID on the title page.

## What happens next?

Your marked assessment should be available 20 working days after submission. However, please be advised that this may be subject to change in the event of Bank Holidays, University Closure or staff sickness. If there is something about the feedback you have been given that you are unclear about, please see your module tutor.

## Feedback Method

Your feedback will be available on Blackboard via the ‘My Grades’ tab or at the location where you submitted the report.

## Late Submission

The University’s [Regulations for Taught Courses (pdf)](https://registry.southwales.ac.uk/documents/2009/Regulations_for_Taught_Courses.pdf) (ref. 75-77) specify that students can submit coursework within the late submission period (five working days) but will be subject to penalty (grades capped at 40%).

Where the five working day late submission policy applies, disabled students with an [Individual Support Plan (ISP)](https://universityofsouthwales.sharepoint.com/sites/Student_Services/SitePages/en-gb/individual-support-plans.aspx)in place are permitted to submit within the late submission period (five working days), without penalty.

## Retrieval in the Event of Failure

This module runs resits within the summer to retrieve any failures.

## Extenuating Circumstances

[https://advice.southwales.ac.uk/a2z/extenuating-circumstances](https://advice.southwales.ac.uk/a2z/extenuating-circumstances/)

## Referencing, Plagiarism and Good Academic Practice

[https://advice.southwales.ac.uk/a2z/referencing-plagiarism-and-good-academic-practice](https://advice.southwales.ac.uk/a2z/referencing-plagiarism-and-good-academic-practice/)

## Learning Support Resources

[https://studyskills.southwales.ac.uk](https://studyskills.southwales.ac.uk/)

## Your Assessment Queries

Please use the seminar sessions for general assessment related queries and individual support will be available within your weekly two-hour practical classes / workshops.