Assessment 2. Assignment 2, COIT20247 T1 2025

Due Date: Friday of Week 10 (11:55 pm AEST, May 23rd, 2025)
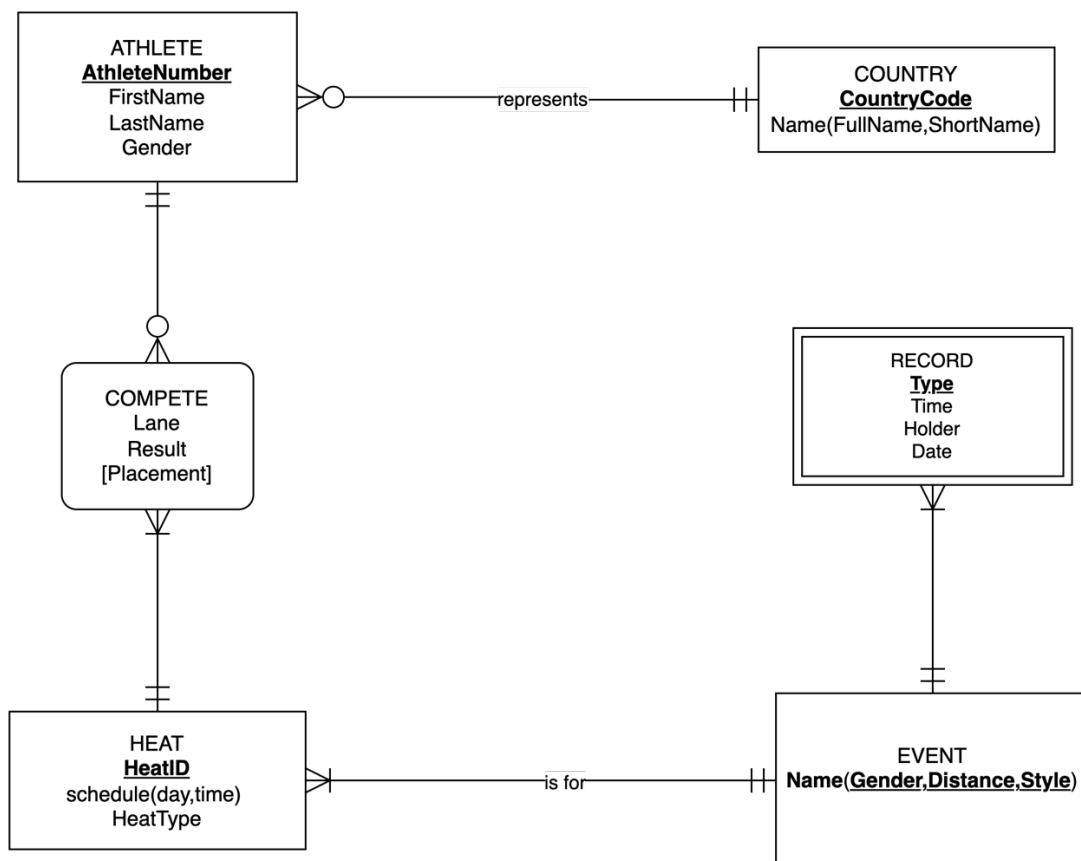
Weighting: 35%

OBJECTIVES

This assessment item relates to the unit learning outcomes numbers 1, 2 and 3 as stated in the unit profile.

ASSIGNMENT SPECIFICATION

You are required to create an SQL database from the following Olympic Swimming Event ERD based on the Assignment 1 specifications.

For the RECORD entity, the record Type is either "Olympic" or "World".

ASSESSMENT TASKS

1 ER MAPPING AND NORMALIZATION

1. Map the ER diagram, from the sample solution, into a **set of relations** in at least Third Normal Form (**3NF**). You must ensure that your relations meet 3NF. There is no need to show your working. You must specify the primary key, and foreign key (if any) for each relation.

2. Select any two of your relations from the previous step (1), and perform the following for each of those two relations:
   - List all the **functional dependencies that** exist in the relation.
   - Demonstrate that the relation meets **Third Normal Form (3NF).**

For your help, an example for a Student relation's 3NF justification has been provided below:

STUDENT (StudentID, Name, Address, DateOfBirth)

i.   The primary key is StudentID, which identifies a student. There will be one student name per Student ID because only one name is recorded for each student. Even though a student may have multiple addresses (e.g. a home address and a work address), the case study states that there is a need to record the home address only, and, hence, there will be only one address for each student. There will be only one date of birth for each student. Thus, there are no repeating groups. Hence, this relation is in 1NF.

ii. The primary key is StudentID. This is the only candidate key since Name is not guaranteed to be unique (two different students may have the same name). Address is also not guaranteed to be unique. This means that Student ID functionally determines every other attribute in the table. There is a single-valued simple candidate key (StudentId), and therefore, no partial dependencies are possible. Hence, the relation is in 2NF.

iii. Name cannot be used to functionally determine any other attribute in the table since two different students may have the same name; likewise for address and date of birth. Therefore, there are no transitive dependencies in the table. Therefore, it meets the requirements of first, second and third normal form.

2 USE MYSQL WORKBENCH TO CREATE THE LOGICAL DATABASE MODEL

Consider the E-R model on the Olympic Swimming Event for the use case of Assignment 1, Use *MySQL Workbench 8.0 CE* to create a logical database model with the relations as in Part 1(E-R mapping and normalization), including specifying the primary keys, foreign keys, creating relationships as appropriate, and creating the E-R diagram. Save the file as `olympics.mwb`

3 RELATIONAL DATABASE IMPLEMENTATION USING MYSQL SERVER

Use MySQL Server to perform the physical implementation of the designed database from the above logical model and export the database file as `olympics.sql`. In implementing the ER model provided, you must complete the following tasks:
   - Create all the relations in a **MySQL database**. Consider each attribute in every table and make appropriate choices regarding **data types & sizes, null allowed or not**. Your choices should be appropriate for each attribute and should support data integrity.

- Indexing can speed up the search for particular tables or queries. Review the guideline given in the textbook regarding the setup of the **index**. You should at least ensure the last name in the `Athlete` table is indexed, and the HeatID attribute in the `Compete` table is indexed.
- Populate the database with sample data of your own. You must include sufficient sample data to test your queries. *Please note that the expected result of query questions should depend on the actual sample data populated.*

You can also implement your database using MySQL Workbench or via Command Line Client. (Note: Week 6 lecture and tutorial provide the instructions on how to export a MySQL database file)

DATA INTEGRITY

You are required to implement a constraint of data integrity to enforce the requirement that at most 8 swimmers compete in a heat. Save the constraint in a file named `MyIntegrityConstraint.sql`

TRIGGER AND STORED PROCEDURE

In databases, a trigger is a set of instructions that automatically executes in response to a specific event on a particular table. Triggers can be used to enforce business rules and maintain data integrity. A stored procedure is a piece of prepared SQL code that you can save, execute, and reuse at any time. You are required to write a trigger and a stored procedure as specified below:

Trigger. Write a trigger that updates the table Record when, in a heat, a swimmer breaks an Olympic or World record. In case of a World record, the trigger must update the corresponding Olympic and World records. Value of the new holder should be the full name.

Stored procedure. Write a stored procedure that, when executed, produces the list of swimmers that participate in a heat. The list should be ordered by lane number, and for each athlete, it shows the name, the nationality (CountryCode) of the swimmer.

Save the SQL code for the trigger and stored procedure in a file named `MyTriggerAndProcedure.sql`

Note: To test this, under the current database schema selected on MySQL Workbench, run the following command first:

SET SQL_SAFE_UPDATES = 0;

SQL QUERIES

Create queries to answer the following information requests. You should type the query codes manually using SQL view/editor under MySQL Server. The marking process of the queries for information requests may also consider the effectiveness of your SQL statements that have been used in the queries. It is important that your SQL statement must reflect the correct business logic & SQL syntax. (**Use enough sample data so that the output is produced by each and every query)** Under MySQL Server, save all query question SQL statements as a file named `QuerySolutions.sql` (each question starts with a line comment such as - - Q1 solution).

Q1 Query 1: List the women's 100m Freestyle finalists who did not qualify for the women's 50m final? You are required to provide two solutions one using NOT IN and the other using a Left Join). Show the Full name and country of each athlete.

Q2 Query 2: Show the Australian swimmers who qualify for a final. For each swimmer, show the name, the event and the result (time and placement) of each final.

Q3 Query 3: Show the medal tally for the swimming events. The list should show the name (Short Name) of the countries, and for each country that won at least one medal, you have to show the number of gold, silver, bronze medals, and the total number of medals. (A country is awarded a gold medal is one of its swimmers placed 1st in a final, a silver medal if a swimmer placed 2nd and a bronze medal if a swimmer placed 3rd). List the countries in order based on the number of (gold) medals they won. If two countries have the same number of gold medals, use the silver medals to order them, and if they have the same number of gold and silver medals, use the bronze medals to order them.

Q3 Query 4: Show the swimmer(s) who competed the most times. Provide the full name of the swimmer(s), country and the number of heats in which the swimmer competed.

Q4 Query 5: List the details of the swimmers who defeated another swimmer from the same country (there must be at least two swimmers from the same country in a heat). Show the names of the swimmers, the event, the heat, the times and the placement of the two swimmers. Show the results in the following format.

Event /Heat	Country Code	Full Name Swimmer one	Time Swimmer one	Placement Swimmer one	Full Name Swimmer two	Time Swimmer two	Placement Swimmer two

Q5 Query 5: Show the results of the men's 100m Medley semifinal one. Include the full name, country, placement and time for each swimmer. Order the list by placement.

Q6 Query 6. Who are the swimmers who broke the most world records in finals? (assume there are some swimmers when you populate the database).

HOW AND WHAT TO SUBMIT

You must make an electronic submission for this assignment using the appropriate assignment submission link in the unit web site.

Place the following files in a zip file named Ass2.zip and submit via Moodle Assignment 2 link.

1. MySQL Workbench logical database model file – `olympics.mwb`
2. MySQL database file – `olympics.sql` containing:
   a. Create table SQL statements
   b. Tables with sample data with insert SQL statements
   c. Create indexes
   d. The required stored procedure
3. The file of answering six information request questions – `QuerySolutions.sql`

4. Data integrity implementation file – `MyIntegrityConstraint.sql`
5. Trigger and stored procedure file – `MyTriggerAndProcedure.sql`
6. A Microsoft Word document containing:
   a. A set of relations mapped from E-R diagram (3NF);
   b. Two relations with functional dependencies and demonstration of normalization;
   c. Six SQL questions solutions (i.e. SQL statements for these 6 questions);
   d. the SQL code for the trigger; and
   e. the stored procedure SQL code.

Your document should contain appropriate identification information (your student number, name, unit number, etc.) and should contain the relevant headings for each section.

IMPORTANT NOTES

- You must work individually on this assignment.

- The model solution to this assignment will be released approximately TWO (2) weeks after the due date.

- Assignments that are submitted after the release of the model solution will receive zero marks.

- You must state in your assignment (at the end under the heading Acknowledgements is a good place) all materials and sources you used. This may be as simple as 'This assignment was completed using only the COIT20247 unit materials' to mention that you have used only the items provided by the unit COIT20247. The use of Generative AI must be documented in the assignment.

- Before submitting your assignment, review the marking criteria that are available in the assignment specification. Ensure that all the criteria are addressed in your submission.

- After submitting your assignment, please verify that your submission was successful, i.e. download your submitted files to ensure that they are correct.

- Penalty related to late submission and plagiarism is handled as per the university policies.

MARKING CRITERIA

Item	Maximum Marks
MySQL Workbench logical database model (olympics.mwb)	**3**
Relational database implementation	**19**
Queries	**13**
Total	**35**

Relational database implementation
- Set of relations correctly mapped from ERD (4 marks)
- Functional dependencies correctly listed for two of the relations (2 marks)
- Normalization correctly listed for two of the relations (2 marks)
- MySQL database implementation (olympics.sql) Create table and insert sample data (5 marks)
- Referential integrity (1 mark)
- Data types & sizes selected appropriately (1 mark)
- Indexes Indexes are appropriate (1 mark)
- Trigger and stored procedure (2 marks)
- Data integrity (1 marks)

Queries
- Query 1 (3 marks)
- Query 2 (2 marks)
- Query 3 (2 marks)
- Query 4 (2 marks)
- Query 5 (2 marks)
- Query 6 (2 marks)

Penalty related to plagiarism and copying is handled as per the university policies.

Penalty for late submissions: 5% of the total marks available per calendar day.