S1, 2025

HIT 234 Assignment 1

# Submission Details:

For Internal and External students: submission is via Learnline. Instructions on how to submit the assignment is located under Assessments / Guidelines.

Please answer **ALL** questions.

# Part A - Single and multiple table queries

Value: 10%

For the first five questions you have to provide your answer in the form of a SQL statement, with description before as necessary. For example:

1. Example Question: Select all columns from the `Customer` table Example Answer:

-- This answer will select out all the columns from the Customer table SELECT \* from Customer;

# Questions 1 – 5:

1. List in ascending order the Last Names of all borrowers with Card Number less than 150.
2. What are the records of those loans between 3 June 2014 and 8 Mar 2018?
   * The output should include the name of the cardholder, and title of the book
   * List in ascending order by last name and first name
3. Find the total numbers of loans with last name beginning with D through E (including E); include last name beginning with Q as well as Z also.
   * The output of the above question should be a column containing the sum of all the loan based on the condition in the query, alternatively the output could be a table with columns that has the last name as per the condition in the query condition, also the table contains the list of loans.
4. Find all borrowers for a loan that have the Date-in before 15-March-15 and the Card number between 100 and 300 in Karama or Darwin.
   * The output for the above question is a list of all borrowers (first name and last name) based on the condition in the query.
5. Find the number of loans, which have been made from each branch?
   * Note all branches need to be included even those with no loans
   * The output of the above question is a table containing the list of loans only as per the condition in the query

# Question 6:

1. Create your own question based on the library database, and also provide a SQL Statement to answer your question.

# Part B – Table and Advanced Queries

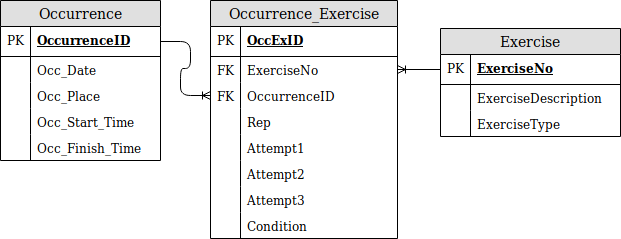
Value: 6%

The questions are broken into: Q1 (4 marks), Q2 (3 marks), Q3 and Q4 (3 marks). For all questions, provide the answer in the form of a SQL statement, with description before as necessary.

## Questions

|  |
| --- |
| **NOTES** |
| Questions 1 and 2 require you to answer in the form of a SQL statement to create a table and insert data into it. The question and answer is only about the statements to create and put data into the  `Occurrence\_Exercise` table, however it is recommended that you create the statements to create and insert data into all 3 tables, so that you can test that your statements are correct. Please only provide in your answer the SQL statement for the `Occurrence\_Exercise` table. |
| To understand the terminology:   * **A rep** is a number of times you do an attempted weight. * **A set** is an attempt and in this case, there are 3 attempts (attempt 1, attempt 2, attempt 3). * **An attempt contains the weight.**   So to understand the weight you would lift for a single exercise in the Exercise attempt table you would need to consider the rep multiplied by attempts by the weight. So for a single attempt:  e.g. for 9001  rep multiplied by attempts by the weight  8 reps x 3 attempts x 15kg = 360 total kg’s lifted |

1. The following Gym application database for weight session recording is as follows. Write and run SQL statements for creating the Occurrence\_Exercise table.
   * Include constraint values for the Condition Column: Heavy, Medium, Light



1. Write an SQL statement that will insert the data into **Occurrence\_Exercise** table only.
   * Definition of rep stands for repetition.
   * Each exercise attempt has the rep number as indicated in the occurrence table.

Occurrence Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| OccurrenceID | Occ\_Date | Occ\_Place | Occ\_Start\_Time | Occ\_Finish\_Time |
| 1 | 14/03/18 | University\_Gym | 16:00:00 | 16:30:00 |
| 2 | 15/03/18 | University\_Gym | 16:30:00 | 17:00:00 |
| 3 | 16/03/18 | University\_Gym | 16:10:00 | 17:20:00 |
| 4 | 17/03/18 | Home | 07:00:00 | 08:00:00 |
| 5 | 17/03/18 | University\_Gym | 17:00:00 | 18:00:00 |

Occurrence\_Exercise Table

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| OccExID | ExerciseNo | OccurrenceID | Rep | Attempt1 | Attempt2 | Attempt3 | Condition |
| 9001 | 001A | 1 | 8 | 15 | 15 | 15 | Light |
| 9002 | 002A | 1 | 8 | 70 | 75 | 70 | Medium |
| 9003 | 003A | 2 | 12 | 90 | 95 | 90 | Heavy |
| 9004 | 004A | 3 | 12 | 200 | 200 | 200 | Medium |
| 9005 | 006A | 3 | 8 | 110 | 110 | 115 | Heavy |
| 9006 | 005A | 4 | 10 | 20 | 25 |  | Medium |
| 9007 | 001A | 4 | 8 | 10 | 12 | 10 | Medium |
| 9008 | 002A | 5 | 8 | 85 | 90 | 115 | Heavy |

Exercise Table

|  |  |  |
| --- | --- | --- |
| ExerciseNo | ExerciseDescription | ExerciseType |
| 001A | Situps | Abs |
| 002A | Bench Press | Chest |
| 003A | Dead Lifts | Back |
| 004A | Leg Press | Legs |
| 005A | Seated Shoulder Press | Shoulders |
| 006A | Squats | Legs |

|  |  |  |
| --- | --- | --- |
| 007A | Lat pull downs | Lats |

Create the following queries:

1. Find the exercise which contains the maximum total weight lifted
   * Show: Exercise, Maximum weight
   * Ensure you show captions for each of the columns
   * Hint: you would need to consider the number of times the weight is lifted and weight

e.g. rep multiplied by attempt multiplied by the weight

1. Find the exercise that the user was not able to complete 3 attempts in, or the exercise that the user has not attempted.
   * Show appropriate columns
   * List in ascending order exercise type
   * Do not show repeated names
   * Ensure you show captions for each of the columns

# Part C – All Topics

Value: 14%

For the first four questions refer to the Library Database. For question 5 refer to CDU Customer Invoice Database.

The questions are broken into: Q1-5.

For all questions, provide the answer in the form of a SQL statement, with description before as necessary.

## Questions Using the Library Database

1. What are the 5 most popular books?
   * Show ASBN, Title.
   * HINT: To limit results use ROWNUM.
2. Which branches currently have no loans out?
   * Show all branch names.
   * List in ascending order by branch.
3. Find the **longest time** a book has been loaned out for each branch
   * Show only branch name and number of days.
   * List in ascending order by branch name.
   * Show all branches, even if there are no loaned out books. Therefore, a branch may have a null value for the longest loaded out book.
   * Format the number of days to one decimal place.
4. Find the borrower that has returned a book on the dated 16 December 2010 and 8 May 2012
   * Show two different methods.

## Question Using the CDU Customer Invoice Database

1. Devise a delete query, which will eliminate duplicates for customer columns `ID`,

`CustomerLastName`, `CustomerFirstName` and invoice columns `Invoice\_Date` and

`Invoice\_No`.

* + Ensure the query removes all duplicate data.