# COMP20240 MySQL DB assignment (2023-2024): Departments Vacancies Database

**Purpose:** The purpose of the assignment is the implementation of an application that stores data in a MySQL database.

**STEP 1:** create a new database and name it with a name composed of your surname and UCD student number (example: smith123456). **Do not use either capital letters or any special character (e.g., apostrophes).**

**STEP 2:** your database will represent the following **scenario**: "Departments in a company advertise vacant positions which require specific skills (e.g., administrative, managerial, etc.). Candidates may be invited to interviews for the positions".

Your database must include the following information and **may include any other information that you consider necessary for representing the concepts and implementing the queries listed below**:

- **Candidate details**: candidate identifier, firstname, surname, address, telephone number, skills.
- **Department details**: department identifier, department name, address, telephone number.
- **Position details**: position identifier, department offering the position, type of position, skills required.
- **Interview details**: you must decide what information should be used to best represent this concept based on the constraints and information provided below.
- **Constraints:**
  - One department can request many interviews for a position.
  - One candidate can be invited to many interviews in relation to a position.
  - One department can hire many candidates in relation to a position.
  - Each candidate can have many skills.
  - Each position can require many skills.

**NOTE:** *You must create table(s) and relationships that will allow you to represent the fact that interviews occur on particular dates. Your database should also represent whether a candidate is offered a position.*

**STEP 3:** For every table, create a stored procedure that includes a parametric query that allows you to insert a new row in such a table.

**STEP 4: Implement the following queries (some of which are parametric) using stored procedures:**

1. Find the candidates with a given first name.
2. Find the surname of candidates with a given candidate identifier.
3. Find the departments with a given department name.
4. Find the candidates who have at least one skill required by a given position identifier.
5. Find the positions requiring a given skill.
6. Find the number of candidates that have been offered a position.
7. Find the number of positions that require administrative skills.
8. Find the identifier of candidates that were interviewed only on a given date.
9. Find the interviews that occurred on a given date.
10. Find the positions sorted according to the departments who are offering them.
11. Find the name and identifier of candidates that were interviewed more than once.

**STEP 5:** export your database onto a self-contained .sql file **which should have the same name as your database** (example: smith123456.sql).

**STEP 6**: prepare the **related documentation** as detailed in the next page (**deliverable 2).**

## Rules

- Each section and subsection of the deliverables must be completed **individually**.
- **All questions should be directed to the demonstrators during lab hours.**

**SUBMISSION INSTRUCTIONS:** The deliverables (.sql file + documentation pdf file as described below) must be submitted to Brightspace (submissions via email will not be accepted).

**Deliverables:**

1. The completed database, implemented using MySQL, exported and saved in a self-contained .sql file (**which should have the same name as your database as detailed in STEP 5**), MUST contain the following:

    - Tables used to correctly represent all concepts as described above (with appropriate primary keys, constraints, etc.) including additional attributes necessary to link the tables according to the required relationships (and any other assumptions you made).

    - Appropriate data types for all attributes and primary key(s).

    - Tables should be populated with **at least** 10 rows per table.

    - Correct insert procedures (STEP 3) and queries (STEP 4) as per information sheet implemented by means of stored procedures. Please name your insert procedures with the name of the corresponding table and the queries in the format **Qx_descriptive_name** where x is the query number (in STEP 4) and descriptive_name is the name you would give this query in the absence of query numbers.

    **REMEMBER TO CALL YOUR DATABASE WITH A NAME CONTAINING YOUR SURNAME AND STUDENT NUMBER (eg. smith123456) AND THE FINAL SELF-CONTAINED .SQL FILE ACCORDINGLY.**

2. **Supporting documentation** MUST be provided in one single PDF file including:
    - A short clear description of your database.

    - A discussion of any assumptions or additions you made.

    - A discussion of reaction policies used and why they were used.

    - The Entity-Relationship (ER) diagram of your database (generated as described in lab Exercise 5).

    - Please indicate clearly, in your documentation, the operating system you used (Mac, Windows, Ubuntu, etc).

    - **CALL YOUR .PDF FILE WITH THE SAME NAME AS YOUR DATABASE.**

## Submission deadline: Sunday, November 19th, 2023 at 11.59 p.m.