

CO2102 Assignment 1: University Database Design and SQL Query

This project contributes 50% towards your CO2102 marks. The project contains 5 tasks. All tasks are compulsory and must be answered. The use of AI is strictly prohibited for this assessment.

Instructions

- Please find the deadline information on the CO2102 Blackboard Homepage.
- Resubmission before the deadline is allowed. However, only the last submitted file will be marked.
- Late submission will cause a penalty on the overall mark as per the University's regulations.
- This project weighs 50% and is calculated out of 100. Read the marking scheme section at the end for more details.
- Do not use your handwriting to draw the ERD but use software to achieve this.
- All drawn ERD must be copied as a .jpg or.png file and pasted into a word document. All normalization should also be included in the word document.
- After you complete and execute all the tasks using Workbench. You should save your file.
- Copy and paste the content of the saved file (your solution) onto a Microsoft Word file and save it. Do not submit a screenshot (image) of your SQL file. If you do this, 20% will be deducted from the overall mark.
- You should submit it via Turnitin on Blackboard. The submission is located under Assessment and Feedback - "CO2102 Project Submission".
- Your submission should only include a single file, titled: YourGroupNo.docx (e.g. Group199.docx).
- Once you are happy to proceed with the submission, click the submit button. Please note that if you do not follow the submission guidelines your work might not be marked.
- Be aware of PLAGIARISM. This assignment is a group (max of 3) piece of work. Any similarities reported by the plagiarism detection software between your work and any external/online or internal resources, the plagiarised parts will automatically be flagged and reported, and marks will be deducted in accordance with the University's guidelines.
- You must write your solution. If you reuse any artefact from any of the worksheets or labs, your work is treated as plagiarism and receives 0 marks.

Case Scenario:

The information system database at the university serves as a foundational repository that meticulously catalogues the essential components of the institution's scholarly framework. This database is a testament to the intricate architecture of data that supports the operational and educational activities of the university. The system comprises entities, each with distinct attributes and interconnected relationships that facilitate the management of academic processes.

Departments are the administrative units of the university, characterised by a department name, a designated building, and an allocated budget. These departments are pivotal in defining the academic landscape, offering a range of courses and serving as the professional affiliation for instructors.

Instructors are integral to the academic mission, identified by a unique ID and a name, with their remuneration detailed by salary. The relationship between instructors and departments is one of professional association, where each instructor is affiliated with a department. An interesting nuance is the advisory relationship, where an instructor may also serve as an advisor to another instructor, suggesting a mentorship dynamic within the faculty.

Students are central to the academic community, each with a unique ID, name, and a record of total credits earned, reflecting their academic progress. Students are also linked to an advisor, typically an instructor, who provides guidance and mentorship throughout their academic tenure.

Courses are the educational offerings, each identified by a course ID and title, and quantified by the credits they award. Courses may also have prerequisites, which are foundational courses that must be completed prior to enrolling in the course, thus establishing a sequential academic progression.

Sections represent the instances of a course offered within a specific semester and academic year. Each section is associated with a course and scheduled within a time slot, which defines the specific time and day of the week allocated for the course's delivery.

Time slots delineate the academic timetable, specifying the start and end times, and the day of the week when classes are conducted. These slots are critical in coordinating the scheduling of courses and sections.

Classrooms are the physical spaces where learning occurs, identified by a building and room number, with a defined capacity that dictates the number of students that can be accommodated.

The relationships within this database are meticulously defined. Instructors are linked to departments, where they teach courses and advise students. Students are associated with departments and instructors, establishing a triadic academic relationship. Courses are connected to departments, indicating the department's role in curricular offerings. The advisory relationship between students and instructors is a significant aspect of the academic experience, with instructors often serving as mentors and guides.

Sections are the conduit that connects courses to time slots and classrooms, creating a structured

academic schedule. The database also captures the prerequisite relationship between courses, reflecting the sequential nature of academic progression.

This interconnected system is dynamic, evolving with the academic calendar, the hiring of new instructors, the enrolment of students, and the introduction of new courses. It is designed to be robust and adaptable, ensuring that all aspects of university operations are efficiently managed and can be accessed and updated as needed.

Based on the above description, please complete the following tasks.

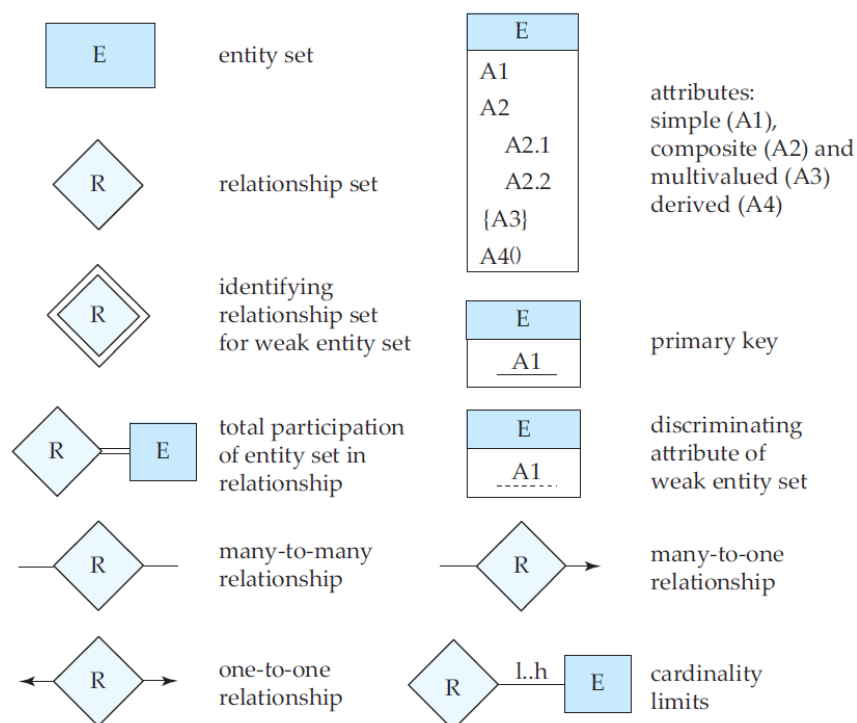
Task 1: (6 marks)

Identify all functional dependencies based on the *university* database description provided above, and represent each dependency in the following format:

EntityName(attribute1(PK), attribute2, ..., attributeN)

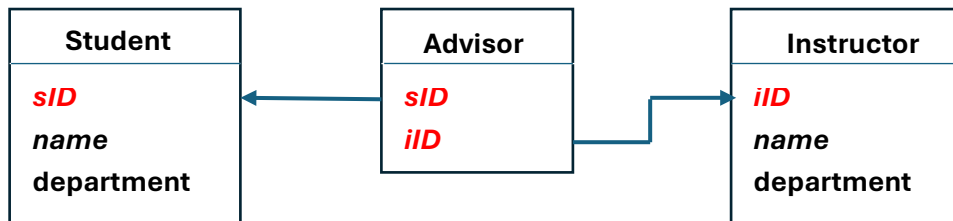
Task 2: (20 marks)

Design a logical *University* database model based on the previously identified functional dependencies by representing the schema in an entity-relationship (ER) diagram. Please use the following entity and relationship notations in your ER diagram:



Task 3: (30 marks)

- 1) Based on the ER diagram designed in Task 2, provide the specific database structure in a diagram by representing each table using ER entity notation (see Task 2) and linking the tables based on their associated attributes in the following format: (22 marks)



- 2) Additionally, analyse how the detailed design of the database schema from Task 2 achieves 1NF, 2NF, and 3NF by providing an example for each normal form based on the designed database structure. (8 marks)

Task 4: (24 marks)

List all necessary SQL statements used to create the *University* database structure designed in Task 3 (SQL statements to create the database and all tables identified in the design scheme).

Task 5: (20 marks)

Given the above database, write the specific SQL query statements based on the following requirements:

- 1) For all instructors in the university who have taught some courses, find their names and the course ID of all courses they taught.
- 2) List the names of instructors along with the titles of courses that they teach. (Write SQL query statement using NATURAL JOIN key words)
- 3) Find the names of instructors with salary amounts between \$90,000 and \$100,000.
- 4) To find the set of all courses taught either in Fall 2009 or in Spring 2010, or both.
- 5) Find the number of instructors in each department who teach a course in the Spring 2010 semester.
- 6) For each course section offered in 2009, find the average total credits (*tot_cred*) of all students enrolled in the section, if the section had at least 2 students.
- 7) Find all the courses taught in the Fall 2009 semester but not in the Spring 2010 semester. (Use subquery and NOT IN key words.)
- 8) Find the departments that have the highest average salary. (Use the ALL key words.)
- 9) Consider an update where we set the *tot_cred* attribute of each *student* tuple to the sum of the credits of courses successfully completed by the student.
- 10) Display a list of all students in the Comp. Sci. department, along with the course sections, if any, that they have taken in Spring 2009; all course sections from Spring 2009 must be displayed, even if no student from the Comp. Sci. department has taken the course section.

Task 6 (Individual Contribution):

In less than 60 words each (each group member can write up to 60 words, i.e., 180 words in total for a group with 3 members), can you explain how you have contributed to this group's work. Make sure to include the percentage of your contributions.

E.g. Kehinde contributed 40% of the project. He provided solutions to some of the questions.

Xiao did 60% of the project.

ABC did not contribute (0%) to the project.

Marking Scheme: *How is this Project Marked?*

- Each section of the work will be assessed in accordance with the marks assigned to it in the marking scheme. This is outlined below in accordance with each section/task.
- In order to ensure consistency and fairness, individual feedback will be provided on the system against each given mark.
- Your MySQL solution should run successfully on MySQL Workbench. Failed to do so, 30% of the overall mark will be deducted, and then your solution will be marked accordingly.
- All ERD must be drawn using draw.io or any other graphical tool. ERD drawn with pen and paper will result in 20% deduction from the overall mark.
- This assignment weighs 50% and is calculated out of 100.
- All group members are expected to contribute equally to this assignment (i.e. 33.3%). Refer to task 6. Students with 0% participation will be awarded 0. Students with less than 20% contribution may be awarded a maximum mark up to half of the group mark.