

Database Systems Lab (CL2005)

Final Exam

Date: May 21st 2025

Instructor(s)

Muhammad Kamran, Durraiz Waseem, Ahmad
Jawad Mustasim, Hadiya Kashif, Hira Tayyab,
Zeeshan Nazar, Seemab Ayyub, Haiqa Saman,
Aqsa Khalid

Total Time (Hrs):

2

Total Marks:

45

Total Questions:

3

Roll No

Section

Student Signature

Do not write below this line

Submission Path

Windows + R: \\Exam\Lab Submissions\Muhammad Kamran\DB Lab Final\ [your section]

Important Note

- Submit a single .sql file named using your roll number e.g. 123-1234.sql.
- After submitting, verify your file is visible in the submission folder of your section, and check that a file size is shown next to it (to ensure it's not an empty 0 KB file)



Attempt all the questions.

CLO #: 3

Q1: Basic SQL

[8 marks]

- Write a SQL statement to print your roll number and section (don't use print keyword) (1 Mark)
- Write a SQL statement to display 3 random numbers (2 Marks)
- Write the answer of the following query as SQL comment where the table website is given (1 Marks)

Select string_agg(Title) from website

Title	Link
Google	https://www.google.com/
FAST NU FLEX	https://flexstudent.nu.edu.pk/
Slate NUCES	http://slate.nu.edu.pk/portal

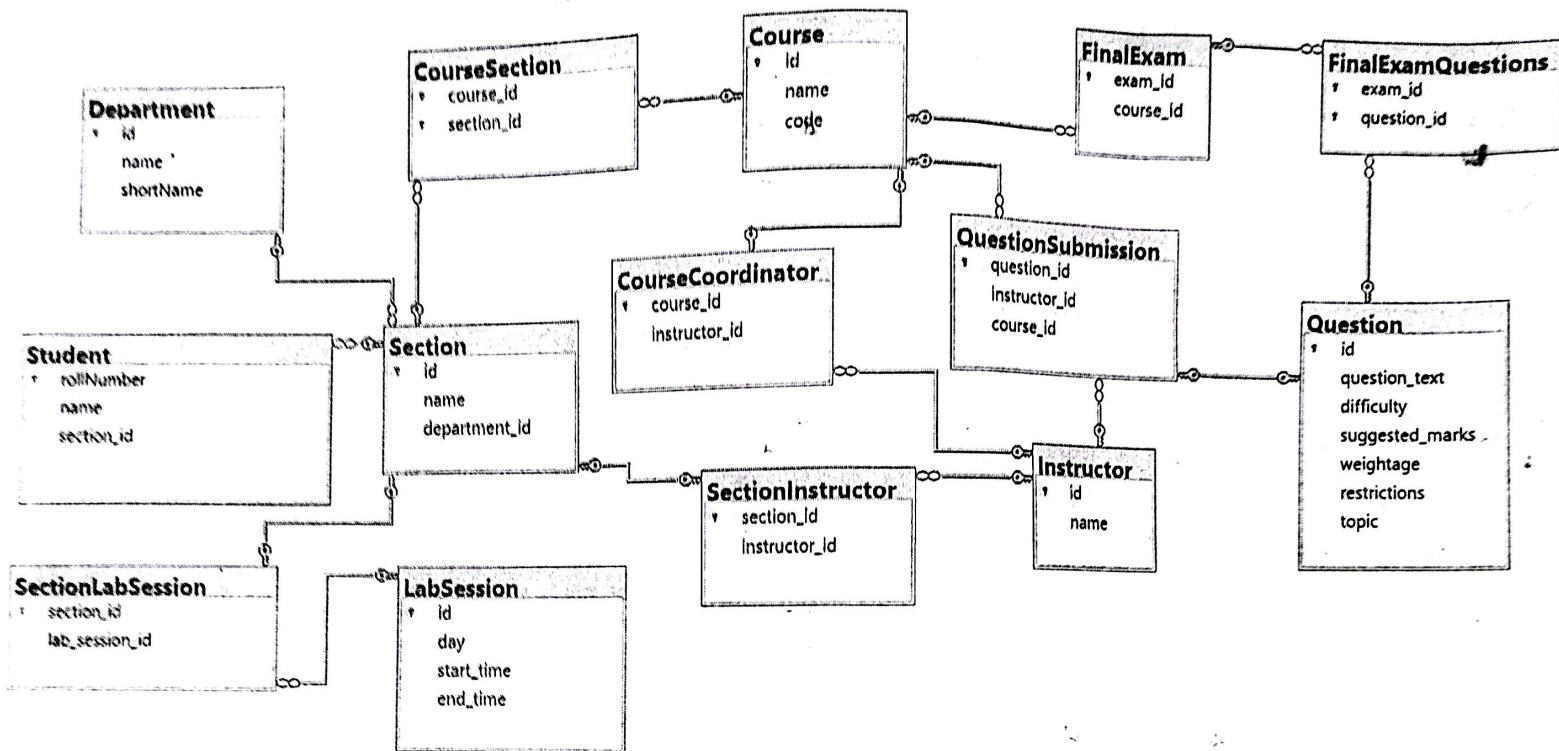
- Write a statement to print a message "Hello World" (1 Mark)
- Write a SQL Block to print numbers from 1 to n where n is last digit of your roll number (2 Marks)
- What is the purpose of scope_identity function in SQL Server? (write as comment) Example is given (1 Mark)

```
INSERT INTO ThemeSuggestion (student_id, theme_name, suggestion_mode)
VALUES (5, 'Dark Forests', 'email');
```

```
INSERT INTO SuggestionAudit (suggestion_id, action, timestamp)
VALUES (SCOPE_IDENTITY(), 'Inserted', GETDATE());
```

Exam DB

You are given following DB for Exam preparation process of a University where all instructors of a lab course submit questions and a coordinator (out of those instructors) finalizes the exam by selecting questions from the instructors.



CLO #: 2

Q2: (DDL) Write DDL statements for the scenario described in the exam context. [7 Marks]

- [1 Marks] Write a SQL statement to create a table named Topic with the following structure:
 - id: an auto-incremented integer that serves as the primary key
 - name: a VARCHAR(200) column
- [1 Mark] Write a SQL statement to insert distinct values from the topic column of the Question table into the name column of the newly created Topic table.
- [1 Marks] Write a SQL statement to add a column named topic_id to the Question table. This column should be defined as a foreign key referencing the id column in the Topic table.
- [1 Mark] Write a SQL statement to change the name of topic_id to corresponding_topic_id.
- [2 Marks] Write a SQL statement to update the topic_id column in the Question table by matching the existing topic column value with the corresponding name in the Topic table.
- [1 Mark] Write a SQL statement to remove the original topic column from the Question table.

CLO #: 3

Q3: Please write the queries for the following

[30 marks]

- a) [5 Marks] Write a SQL query to retrieve all distinct lab session time slots scheduled for the **Database Lab** course. For each time slot, display the day and time range (start_time to end_time), along with a comma-separated list of section names scheduled in that slot and a comma-separated list of instructors assigned to those sections.

day time_slot section_names instructors

- b) [5 Marks] Create a **scalar-valued function** named fn_InstructorLabWorkload that accepts an Instructor_ID as input and returns the **total number of hours per week** the instructor spends in lab sessions.
After defining the function, **add a computed column** to the Instructor table that uses this function to dynamically display each instructor's weekly lab workload.
- c) [5 Marks] Write a **trigger** on the SectionInstructor table to prevent the assignment of **multiple instructors to the same section**.
Ensure the trigger uses **transactions and rollback** to maintain consistency, and raises a meaningful error if the rule is violated.
- d) [12 Marks] Create a trigger on the QuestionSubmission table to handle automatic final exam generation per course.
- The trigger should execute after each insert, but operate only for the course(s) involved in the inserted rows.
 - The trigger must check if the total number of submitted questions for a course exceeds 3; only then should an exam be created.
 - It should not create or modify the exam if the current system date is later than June 05, 2025.
 - For each applicable course, insert a new record into the FinalExam table if no exam already exists for that course.
 - Select questions randomly from the QuestionSubmission table for the relevant course.
 - Ensure that selected questions are unique and that no question appears more than once in the same exam.
 - The total weightage of selected questions for the exam must not exceed 40; stop selecting when this threshold is reached or all of the questions have been used.
 - If an exam already exists for the course, do not re-insert or modify it.
 - Wrap all logic within a transaction to ensure atomicity; if any error occurs, the operation must be rolled back completely.
 - Use TRY...CATCH for error handling and raise a meaningful error message if constraints are violated or execution fails.
 - All inserts into FinalExam and FinalExamQuestions must maintain referential integrity through foreign key constraints.
- e) [3 Marks] Write a SQL query to retrieve the list of **questions selected for the final exam**, along with their **full details** (e.g., difficulty, topic, marks, weightage) and the **total weightage per exam**.

Just take it?