

DBMS – ASSIGNMENT 3 (LAB TEST)

BSDS, Second Year, 2025–2026

Date: April 20, 2026

Total: 5 marks

SUBMISSION INSTRUCTIONS

1. Submit all the solutions in a single file.
2. Naming convention for your submission file (assuming BSD-BG/CC/DD-24xx is your roll number): BSD-BG/CC/DD-24xx-assign3 (.rtf, .docx, .doc, .pdf, .tex, etc.).
3. To submit a solution file (say BSD-BG/CC/DD-24xx-assign3.pdf), ensure that it is not password protected and mail to <assignisik@gmail.com> with the subject line as follows: DBMS BSDS 2024-28 BSD-BG/CC/DD-24xx Assignment 3.

NOTE: All the solutions of this Lab Test must only include SQL queries. They should be written in a generalized way (as applicable). The time and space efficient queries will receive better marks.

Q1. Suppose the students in a classroom were asked what animal they wish to be in the future birth, if at all possible. Consider the following schema representing the data received from these students. Note that the attribute *futurebirth* is a multi-valued attribute.

- WISH = $\langle \underline{studentid} : integer, studentname : string, futurebirth : string \rangle$

The primary key corresponding to the aforementioned relation is underlined. Write the following queries in SQL.

- (i) Find the names of the users who does not wish to be human again.
- (ii) Find the names of the students who mentioned multiple animal names.
- (iii) Find all the duplicate list of animal names (the wish list is exactly the same for more than one student) mentioned.

Q2. Consider the following schema that summarizes the performance of the students in a placement session of an institute.

- HIRING = $\langle \underline{studentid} : integer, company : string, salay : real, joining : date \rangle$

The primary key corresponding to the aforementioned relation is underlined. Write the following queries in SQL.

- (i) Find the names of all the companies who have hired multiple students.
- (ii) Find the name of the company offering the second highest salary.
- (iii) Find the name of the company offering the earliest joining.

Q3. Consider the following schema denoting the people who have accounts in the banks SBI and UBI till date. Note that an account holder can have multiple accounts in a bank but only one account of a single type, namely ‘savings’ or ‘current’.

- SBI = $\langle \underline{\text{accountid}} : \text{integer}, \text{name} : \text{string}, \underline{\text{type}} : \text{string}, \text{balance} : \text{real} \rangle$
- UBI = $\langle \underline{\text{accountid}} : \text{integer}, \text{name} : \text{string}, \underline{\text{type}} : \text{string}, \text{balance} : \text{real} \rangle$

The primary keys corresponding to the aforementioned relations are underlined. Write the following queries in SQL.

- Find the names of all the people who have account in both SBI and UBI.
- Find the names of all the people who have only one account in SBI and multiple accounts in UBI.
- Find the names of all the people who have more money deposited in SBI than UBI.

Q4. Consider the following schema that represents the estimated GDP (in billions) of various countries around the world in 2026.

- GDP = $\langle \underline{\text{countryid}} : \text{integer}, \text{countryname} : \text{string}, \text{GDP} : \text{integer} \rangle$

The primary key corresponding to the aforementioned relation is underlined. Write the following queries in SQL.

- Find all the countries whose name include the word “islands”.
- Find the mean GDP of all the countries around the world.
- Find the names of all the countries who have a GDP more than ‘India’.

Q5. Consider the following schema representing the Professors who worked for MIT and Stanford. Assume that all the names are unique across both the relations.

- MIT = $\langle \underline{\text{mitid}} : \text{integer}, \text{name} : \text{string}, \text{joining} : \text{date}, \text{currentsalary} : \text{real} \rangle$
- Stanford = $\langle \underline{\text{stanfordid}} : \text{integer}, \text{name} : \text{string}, \text{joining} : \text{date}, \text{currentsalary} : \text{real} \rangle$

The primary key corresponding to the aforementioned relation is underlined. Write the following queries in SQL.

- Find the names of all the Professors who have worked in MIT but not in Stanford.
- Find the names of all the Professors who have worked in both MIT and Stanford, but earned a better salary while working in MIT than in Stanford.
- Find the names of all the Professors who have worked in both MIT and Stanford, but have joined in MIT earlier than Stanford.

Q6. Create a trigger to set control after the update operations on a table. In case an update operation happens on it, it should replace the new value with the old value to get the table back into original form.

- Q7. Create a view from a table, which has an auto-incremented unique identifier ID, that will take all the rows that appear in the middle of it. Note that there can be multiple rows in the middle if the number of rows in the original table is even.