

Lab Assignment I

21/01/2025

Mark: 20

-
- Write the query, execute and paste screenshots, and make a pdf. Paste the text of the query too. You need to show a screenshot to demonstrate the correctness of your query. If you have made some changes that are not visible in the direct output, display the whole table again.
 - Submit one file .pdf file containing all answers and the file name should be **<roll no>_<name>_assignment1.pdf**
 - Write the questions before the answers.
 - Write a small justification for your solution.
 - Complete the exercise before 4.45 PM and submit the pdf in Moodle.
-

Find the Schema

- **Movie (movieID, name, genre, releaseYear)**
- **Actor (actorID, movieID, name, age)**

(movieID and actorID are unique in the above tables.)

-
1. Create a database named **MoviesDB**. Create the above tables in **MoviesDB** with suitable data types, and include the appropriate primary key, foreign key, and domain constraints. **3**
 2. Insert the following entries into the tables: **1**

Actor: (3, 1, "Tom Hanks", 64)

Movie: (1, "Forrest Gump", "Drama", 1994)

3. Insert the following entries into the tables using bulk insert: **2**

Movie:

- (2, "The Matrix", "Sci-Fi", 1999)
- (3, "The Godfather", "Crime", 1972)
- (4, "Inception", "Sci-Fi", 2010)
- (5, "The Shawshank Redemption", "Drama", 1994)

Actor:

- (2, 2, "Keanu Reeves", 59)
- (4, 3, "Marlon Brando", 80)
- (5, 4, "Leonardo DiCaprio", 46)
- (6, 5, "Morgan Freeman", 87)

4. Modify the **Movie** table to include an additional column rating (with a default value of 5) to store the rating of the movie. Update the ratings for all records as follows: **2**

(4, 3, 4, 5, 5) respectively.

5. What will happen if you try to insert the following into the **Actor** table? Why? **2**

(3, 1, "John Doe", 35)

6. Bulk insert all the values from a provided CSV file (**actor_insert.csv**) into the Actor table. **1**

(**Hint:** Download the CSV file and use the appropriate command for bulk insertion)

7. Display all contents of the **Movie** table. **1**
8. Display only the names and ages from the **Actor** table. **1**
9. Display the names of the actors who are older than 50 years old. **1**
10. List the names of the movies that were released after 2000. **1**

11. Increment the age of all “The Matrix” actors by 1 year. Display the entire **Actor** table after the operation. 2
12. Dump the database into a **.sql** file and include that in your submission. 1
13. Delete the **Actor** table. List existing relations in the database. 1
14. Delete the **MoviesDB** database. Then, list the existing databases. 1

Practice Questions

1. Restore the database
2. Create a new table Director (directorID, name, age). The directorID should be the primary key.
3. Insert the following information into Director
 - (1, “Christopher Nolan”, 53)
 - (2, “Francis Ford Coppola”, 85)
4. Modify the Movie table to include a new column directorID as a foreign key referencing the Director table.
5. Create a new table Review with columns reviewID (integer), movieID (integer), review(varchar), rating (integer), and reviewDate (date). The reviewID should be the primary key, and movieID should be a foreign key referencing the Movie table.
6. Create a CSV file with review data. Do a bulk insert to the Review table.