

## 1<sup>st</sup> Internal Example Paper

23 marks

### Using DDL, DML and TCL Query

Create database with name as internal\_exam1

User internal\_exam1 and save all query in **internal\_exam1.sql**

You are tasked with managing an employee database for a company. You need to:

**10 marks**

1. **Create a table** to store the following employee details:

- EmployeeID (Primary Key)
- FirstName
- LastName
- Age
- Department
- Salary

1 mark

2. **Insert the following records** into the table:

EmployeeID	FirstName	LastName	Age	Department	Salary
1	John	Doe	28	HR	50000.00
2	Jane	Smith	34	IT	60000.00
3	Alice	Johnson	25	Finance	45000.00
4	Bob	Brown	40	Marketing	70000.00
5	Charlie	Davis	38	IT	75000.00

1 mark

3. **Retrieve** all employee records who have a salary greater than 55,000.

1 mark

4. **Update the salary** of the employee with EmployeeID 1 to 55,000.

1 mark

5. **Increase the salary** by 10% for all employees in the IT department.  
1 mark
6. **Delete** the record of the employee with EmployeeID 5.  
1 mark
7. **Retrieve** all employees in the HR department and display their  
FirstName, LastName, and Salary  
1 mark
8. **Add a column** to the table to store the hire date of the employee.  
1 mark
9. **Add a column** named JoinDate (DATE) to the table to store the date the  
employee joined the company.  
1 mark
10. **Retrieve and display employees sorted by their department.**  
1 mark

You are managing the inventory for a bookstore. Complete the following tasks:  
**7 marks**

1. **Create a table** Books with these columns:
  - a. BookID (Primary Key, INT)
  - b. Title (VARCHAR)
  - c. Author (VARCHAR)
  - d. Publisher (VARCHAR)
  - e. Price (DECIMAL)

1 mark
2. **Insert the following records** into the Books table:

BookID	Title	Author	Publisher	Price
1	The Great Gatsby	F. Scott	Scribner	15.00
2	1984	George Orwell	Harvill Secker	20.00
3	Moby Dick	Herman Melville	HarperCollins	18.00
4	To Kill a Mockingbird	Harper Lee	J.B. Lippincott	12.00
5	The Catcher in the Rye	J.D. Salinger	Little, Brown	14.00

1 mark

3. **Update the price** of the book with BookID = 1 to 17.00. 1 mark
4. **Retrieve** all books priced above 15.00 and display their Title, Author, and Price. 1 mark
5. **Delete the record** of the book with BookID = 4. 1 mark
6. **Alter the table** by adding a column Genre (VARCHAR) to store the genre of the book. 1 mark
7. **Retrieve** books published by "HarperCollins" and **sort** them by Title in ascending order. 1 mark

## Order Management System 6 marks

You are managing an **e-commerce database**. When a customer places an order, it involves updating the Inventory and creating an entry in the Orders table. Both operations need to be completed successfully, or neither should be applied in case of failure.

1. **Create the following tables:**
  - Orders (OrderID, CustomerName, OrderDate, TotalAmount)
  - Inventory (ProductID, ProductName, StockQuantity, Price)
2. **Insert the following records** into Inventory:

ProductID	ProductName	StockQuantity	Price
201	Laptop	50	700.00
202	Smartphone	100	400.00
203	Headphones	200	50.00

**Scenario:** A customer places an order for 2 laptops and 1 smartphone.

- **Insert the order details** into the Orders table.
- **Update the stock quantity** in the Inventory table by deducting the ordered quantity (2 laptops, 1 smartphone).
- **If both operations succeed, commit the transaction.**
- **If any operation fails (e.g., if insufficient stock is available), rollback the entire transaction.**

- **Questions:**

- How can you ensure that both the order creation and stock update are handled atomically (both must succeed or both must fail)?
- What would happen if the stock for the laptop is insufficient? How would the transaction be handled?