National University of Computer and Emerging Sciences



Lab Manual 02

Database Systems Lab

| Course Instructor | s**ir Naveed** |
| --- | --- |
| Lab Instructor (s) | Samia Akhter  Sir Basam |
| Date | SEP 9, 2023 |

Department of Computer Science

FAST-NU, Lahore, Pakistan

## 

## 

## 

## Objectives

After performing this lab, students shall be able to:

(DML)

* Crud operation
* Like operator
* Arithmetic Operators
* Set operations
* Built in functions

# Lab Exercise

**Question 1:** You are tasked with performing CRUD operations on a database with the following schema for a fictional library management system: **(12 marks)**

**Schema:**

* **Table:** Books
  + Columns:
    1. BookID (Primary Key)
    2. Title (Title of the book)
    3. Author (Author of the book)
    4. PublicationYear (Year the book was published)
    5. ISBN (International Standard Book Number)

INSERT INTO Books (BookID, Title, Author, PublicationYear, ISBN)

VALUES

(1, 'To Kill a Mockingbird', 'Harper Lee', 1960, '9780061120084'),

(2, '1984', 'George Orwell', 1949, '9780451524935'),

(3, 'The Great Gatsby', 'F. Scott Fitzgerald', 1925, '9780743273565'),

(4, 'Pride and Prejudice', 'Jane Austen', 1813, '9780141439518'),

**Your task is to write SQL queries for the following CRUD operations:**

**Create:**

1. Insert a new book into the Books table with the following details:
   * Title: "The Catcher in the Rye"
   * Author: "J.D. Salinger"
   * Publication Year: 1951
   * ISBN: 9780316769174

**Read:** 2. Retrieve the details (Title, Author, Publication Year, ISBN) of all books in the database.

**Update:** 3. Update the book with the ISBN "9780316769174" to change its title to "Catcher in the Rye."

**Delete:** 4. Delete the book with the ISBN "9780316769174" from the database.

Please write the SQL queries for each of the above CRUD operations and provide the results (if applicable) after performing each operation.

**Question 2:** In your database, there is a table called Products that stores information about various products. The schema for the Products table is as follows: **(12 marks)**

**Schema:**

* **Table:** Products
  + Columns:
    1. ProductID (Primary Key)
    2. ProductName (Name of the product)
    3. Category (Category to which the product belongs)
    4. Price (Price of the product)

Below are some sample values in the Products table:

| **ProductID** | **ProductName** | **Category** | **Price** |
| --- | --- | --- | --- |
| 1 | Laptop | Electronics | 800.00 |
| 2 | Desk Chair | Furniture | 120.00 |
| 3 | Smartphone | Electronics | 500.00 |
| 4 | Office Desk | Furniture | 250.00 |
| 5 | Wireless Mouse | Electronics | 20.00 |
| 6 | Bookshelf | Furniture | 150.00 |
| 7 | Headphones | Electronics | 100.00 |

Your task is to write SQL queries using the LIKE operator to retrieve specific information about products based on the provided sample values:

**Query 1:**Retrieve the names of all products that contain the word "Chair" anywhere in their name.

**Query 2:** Retrieve the names and prices of all products in the "Electronics" category.

**Query 3:** Retrieve the names and prices of all products with a price less than $200.

**Query 4:** Retrieve the names and categories of products that contain the word "Desk" anywhere in their name.

Please write SQL queries for each of the above queries using the LIKE operator and provide the results based on the provided sample values.

**Question 3: (20 marks)**

**Table Name: orders**

**Columns: order\_id (INT), customer\_id (INT), order\_date (DATE), total\_amount (DECIMAL)**

| **Order\_id** | **Customer\_id** | **Order\_date** | **Total\_amount** |
| --- | --- | --- | --- |
| 1 | 101 | 2023-09-01 | 150.50 |
| 2 | 102 | 2023-09-02 | 75.20 |
| 3 | 103 | 2023-09-03 | 200.00 |
| 4 | 104 | 2023-09-04 | 80.75 |
| 5 | 105 | 2023-09-05 | 300.30 |

1. Calculate the total revenue generated from all orders in the orders table.
2. Find the average order amount in the orders table.
3. Update the total\_amount column to increase all order amounts by 10%.
4. Select orders where the total amount is greater than $500.
5. Calculate the sum of total amounts for orders placed in 2023

Table Name: **employees**

Columns: employee\_id (INT), first\_name (VARCHAR), last\_name (VARCHAR), department\_id (INT)

| **Employee\_id** | **First\_name** | **Last\_name** | **Department\_id** |
| --- | --- | --- | --- |
| 1 | John | Smith | 101 |
| 2 | Sarah | Johnson | 102 |
| 3 | Michael | Williams | 103 |
| 4 | Emily | Davis | 104 |
| 5 | David | Lee | 105 |

Table Name: **departments**

columns: department\_id (INT), department\_name (VARCHAR)

| **Department\_id** | **Department\_name** |
| --- | --- |
| 101 | Sales |
| 102 | Marketing |
| 103 | Finance |
| 104 | Human Resources |
| 105 | Research and Development |

1. Perform a union operation between the employees and departments tables to get a combined list of employees and department names.
2. Find the employees who are not assigned to any department using the set difference operation.
3. Determine the common department(s) between two sets of employees using the intersection operation.
4. List all departments and the number of employees in each department using a join and grouping.
5. Find the departments that have no employees assigned to them.

**Question 4: (20 marks)**

Table Name: **products**

Columns: product\_id (INT), product\_name (VARCHAR), description (TEXT), price (DECIMAL), manufacture\_date (DATE)

| **Product\_id** | **Product\_name** | **Description** | **Price** | **Manufacture\_date** |
| --- | --- | --- | --- | --- |
| 1 | Laptop | High-performance laptop | 900 | 2023-05-15 |
| 2 | Smartphone | Latest smartphone model | 699 | 2023-06-20 |
| 3 | Headphones | Noise-canceling headphones | 149 | 2023-07-10 |
| 4 | Tablet | 10-inch touchscreen tablet | 299 | 2023-06-05 |
| 5 | Camera | DSLR camera with zoom lens | 799 | 2023-08-12 |

1. Retrieve the first 3 characters of each product name in the products table using the SUBSTRING function.
2. Calculate the age of each product in days (from the manufacture\_date column to the current date) using the **DATEDIFF** function and **GETDATE**.
3. Create a new column called discounted\_price in the products table, which is 10% less than the original price, and then display the updated table.
4. Retrieve a list of products sorted by their prices in descending order using the **ORDER** **BY** clause.

.

.