LAB 1: CREATING DATABASE and Table

Database Name: Online posal

SQL for creating tables

```sql
CREATE TABLE CUSTOMERS

    CustomerID   int NOT NULL,
    CustomerName   varchar (50),
    ContactName    varchar (100),
    customerAge    int,
    postalcode     varchar (50),
    country  varchar (50),
    PRIMARY KEY (customerID)
);
```

```sql
CREATE TABLE Category

categoryID  int NOT NULL,
categoryName  varchar(100),
Description  varchar (50),
PRIMARY KEY (categoryID)
);
```

```sql
CREATE TABLE Employee
(
  employeeID  int NOT NULL,
  lastName        varchar (100),
  firstName       varchar(100),
  birthDate       date,
  notes     varchar (250),
  PRIMARY KEY (employeeID)
);
```

```sql
CREATE TABLE Shipper
(
  ShipperID    int   NOT NULL,
  ShipperName    varchar(100),
  Phone          varchar(100),
  PRIMARY KEY (shipperID)
);
```

```sql
CREATE TABLE Suppliers
(
  SupplierID int    NOT NULL,
  SupplierName  varchar(100),
  Contact Name  varchar(100),
  address  varchar (50),
  city  varchar (50),
  Postal code  varchar (50),
  Phone         varchar (50),
  PRIMARY KEY  (supplierID)
);
```

```sql
CREATE TABLE ORDER
(
  OrderID int NOT NULL,
  CustomerID int,
  employeeID int,
  ShipperID int,
  PRIMARY KEY (orderID)
  FOREIGN KEY (customerID) REFRENCES customer (customerID)
  ON DELETE CASCADE
  FOREIGN KEY (employeeID) REFRENCES employee (employeeID)
  ON DELETE CASCADE
  FOREIGN KEY (shipperID) REFRENCES shipper (shipperID)
  ON DELETE CASCADE
);
```

```sql
CREATE TABLE Products
(
    productID int NOT NULL,
    productName varchar (50),
    supplier ID int,
    categoryID int,
    unit    varchar (50),
    price   varchar (50),
    PRIMARY KEY (product ID),
    FOREIGN KEY (category ID) REFRENCES categoryid
                                        (categoryID)
    ON DELETE CASCADE
);
```

```sql
CREATE TABLE order Details
(
    orderDetailID int NOT Null
    orderID int,
    productID int,
    PRIMARY key (orderDetailID)
    FOREIGN KEY (orderID) REFRENCES order (orderID)
    ON DELETE CASCADE,
    FOREIGN key (productID) .REFRENCES product (productID)
    ON DELETE CASCADE
);
```

LAB-2

SQL for inserting values in TABLES.

FOR CUSTOMER

INSERT INTO customer
VALUES (1, "Alfreds Flutterkisle", 'Ober Str. 57', 'Frankfurt', 12209,
       'Germany');

VALUES (2, 'Antonio Moreno Taqueria', 'Mataderos 2317',
       'Memico O.F.', '05023, 'Menico');

INSERT INTO
VALUES (3, Around The.horn', 120 Hanover', 'London',
       'WA1 1DP', 'UK');

VALUES (4, 'Berglunds Snabbkop', 'Brausvasen 8', 'Lulea J',
       'S-958 22', 'Sweden');

VALUES (5. 'Blauer see Delikateson', 'Foster=57',
       'Maheim', '68306', 'Gemary');

FOR Category

INSERT INTO Category
VALUES (1, 'Beverages', 'soft drinks, coffees+, teas, beers')
INSERT INTO category
VALUES (2, 'Condiments', 'sweet & savory saures, spreads &
       seasonings');
VALUES (3, 'confections'); "Desserts, candles & sweet breads') -
VALUES (4, 'Dairy products', 'cheese');

FOR table employees

INSERT INTO employees
VALUES (1, 'Davolio', 'Nancy', '1968-12-08', 'Education in psycology!);
VALUES (2, 'Fuller', 'Andrew', '1952-02-15', 'Phd in international marketing) -
VAUES (3 'Laverling', 'Janet', '1963-08-30', 'Janet has BS degree in
       chemistry from Boston College');

SQL for shippers

```
INSERT INTO shippers VALUES
(1', 'Speedy Enpress', '(503) 555-9831');
(2, 'United Package', '(503) 555-3199');
(3, 'Federal Shipping', (503) 555-9931');
```

SQL for table supplies

```
INSERT INTO suppliers VALUES
( 1, 'Enotidliquid', 'Charlotte Copper', '49 Gilbert St.', 'Londona',
  'EC1 4SD', 'UK', '((71)555-2222');
( 2, 'New Orleans Cajun Delights', 'Shelley Burke', 'P.O.Box 78934',
  'New Orleans', '70117', 'USA', '(100) 555-48223');
( 3, 'Grandma kellys Homestead', 'Regina Murphy', '707 Oxford Rd.',
  'Ann Arbor', '48104', 'USA', '(313) 555-5735');
(4, 'Tokyo Traders', 'Yoshi Nagase', '9-8 Sekimai Musashino-Shi',
  'Tokyo', '100', 'Japan'. '(03) 3555-5011'),
( 5, 'Cooperative de Quesos Las Cabras', 'Antonio del Valle
     Savedra', 'Calle dRosal 4', 'Oviedo'. '33007', 'Spain',
  '(981 598 7654') ;
```

SQL for products

```
INSERT INTO products VALUES
(1, 'Chais', 1, 1, '10 bones x 20 bags', 18);
(2, 'Chang', 1, 1, '24-12 oz bottles', 19);
(3, 'Aniseed Syrup', 1, 2, '12-550ml bottles', 10),
( 4, 'Chef Antos Cajus Seasoning', 2, 2, '48-6 oz jars', 22),
(5, 'Chef Antos Gumbo Mix', 2, 2, '36 bones', 21.35),
(6, 'Grandmas Bysenberry Spread', 3, 2, '22-8 oz jars', 25),
(7, 'Uncle Bobs Organic Dried Pears', 3, 1, '12-8 oz jars', 25)
```

SQl for orders

Inserd INTo Orders VALUES
(10249, 2, 1, '1996-07-05',1),
(10250, 4, 3, '1996-07-08', 2);
(10251, 1, 2, '1996-07-08',1),
(10252, 3, 1, '1996-07-09',2);
(10253, 5, 3, '1996-07-10',2);

SCSL for order Details

INSERT INTO order Details VALUES
(1, 10248, 1, 12);
(2, 10248, 2, 10),
(3, 10248, 3, 5),
(4, 10249, 4, 9),
(5, 10249, 5, 10),
(6, 10250, 6, 10);

Lab-3 – SELECTION & UPDATION & DELETION

SELECTION

```
SELECT *
FROM customers
Where country = "Germany"
```

```
SELECT customerID, customerName, PostalCode
FROM customers
Where country = "Germany"
```

```
SELECT *
FROM customers
Where customerName LIKE "a%.";
```

```
SELECT *
FROM employees
Where firstName LIKE "a%.";
```

UPDATION

```
UPDATE orderdetails
SET Quantity = Quantity * 1.1
```

```
UPDATE products
SET price = price * 5;
```

```
SELECT productID, ProductName, price * 1.20
FROM products
```

DELETION

```
DELETE FROM orderdetails
WHERE Quantity < 9;
```

BETWEEN & NOT BETWEEN

```
SELECT *
FROM Products
WHERE productID BETWEEN 3 AND 6;
```

```
SELECT *
FROM products
WHERE product IDNOT
BETWEEN 3 AND 6;
```

Lab-4

```
SELECT *
FROM suppliers
WHERE city IN ("Londona", "Tokyo")
```

```
SELET '
FROM suppliers
WHERE city = "Londona" OR city = "Tokyo"
```

```
SELECT AVG (price) AS Average Price
FROM products

SELET SUM( price) AS Sum Price
FROM products.
```

```
SELECT Min (price) As MinPrice
FROM products
```

```
SELET COUNT(price) AS Counted price
FROM products;
```

Practice

```
SELECT Address, MIN(SALARY) AS Min Salary, MAX (salary)
            AS MAX Salary
FROM tbl-emp
GROUP BY Address;
```

```
SELET Address, MAX (Salary) AS Man Salary
FROM tbl-emp
GROUP BY Address
HAVING MaxSalary >10000;
```