SQL BASICS AND TASK

Task 2) Perform basic sql commands create,update,delete,insert and select.....

…….create table.....

```
CREATE TABLE STUD(
        Id INT PRIMARY KEY NOT NULL,
        Name VARCHAR(20),
        Place VARCHAR(50),
        Course VARCHAR(20)
        );
```

…..insert values .....

```
INSERT INTO STUD(Id,Name,Place,Course) VALUES (1, 'Arjun', 'Alappuzha', 'MCA'),
(2, 'Sneha', 'Kollam', 'BSc'),
(3, 'Neha', 'Kochi', 'B.Tech'),
(4, 'Meera', 'Palakkad', 'M.Tech'),
(5, 'Arun', 'Malappuram', 'B.Ed'),
(6, 'Rahul', 'Trivandrum', 'MBA'),
(7, 'Sandeep', 'Kannur', 'BBA'),
(8, 'Vishnu', 'Thrissur', 'MSc'),
(9,'Remesh','Kochi','BCA'),
(10,'shilpa','Trivandrum','MCA');
```

--------display table data ----------

```
SELECT * FROM STUD

 -- display distinct values....

SELECT DISTINCT Course  FROM STUD;

--display data in  sorted form.....

SELECT * FROM STUD ORDER BY Name ASC;

  -- display data followed by a where clause.....

SELECT * FROM STUD WHERE Place='Kollam' AND Name LIKE 'S%';

SELECT * FROM STUD WHERE Course='MCA' OR PLACE ='Trivandrum';

SELECT * FROM STUD WHERE NOT Place='Kollam';

SELECT TOP 3 * FROM  STUD;

 --   update existing record ....

UPDATE STUD SET Course='.NET' WHERE Id=1;

UPDATE STUD SET Name='Shilpa' WHERE ID=10;

 --     delete record ......

DELETE  FROM STUD WHERE Id=5;
```

Task 3) Display second highest salary.....

Task 4) Display number of employees in each department.....

```
CREATE TABLE Employee (
   EmpId INT PRIMARY KEY,
   EmpName VARCHAR(50),
   salary DECIMAL(10,2)
);

INSERT INTO Employee (EmpId, EmpName, salary) VALUES
(1, 'Arjun', 50000),
(2, 'Neha', 60000),
(3, 'Rahul', 70000),
(4, 'Sneha', 80000),
(5, 'Vishnu', 90000);

SELECT * FROM Employee;


SELECT MAX(salary) as Second_highest_Salary FROM Employee WHERE salary NOT IN (SELECT
MAX(SALARY) FROM Employee) ;


SELECT MIN(salary)  as Minimum_salary from  Employee ;

SELECT COUNT(EmpName) as Total_Emp from Employee;
```

```
SELECT * FROM Employee WHERE salary BETWEEN 70000 AND 100000;

ALTER TABLE Employee ADD Department VARCHAR(20);

UPDATE  Employee SET Department = 'Sales' WHERE EmpId =1 OR EmpId=2;

UPDATE  Employee SET Department = 'Marketing' WHERE EmpId =3;

SELECT Department,COUNT(Empid) as number_of_employees FROM Employee GROUP BY
Department;

UPDATE  Employee SET Department = 'Developing' WHERE EmpId =4;

UPDATE  Employee SET Department = 'Tesing' WHERE EmpId =5;
```

Task 5) perform Join operations....

```
CREATE TABLE customer (
    customer_id INT PRIMARY KEY,
    name VARCHAR(50),
    phone VARCHAR(15)
);

CREATE TABLE orders (
    order_id INT PRIMARY KEY,
    customer_id INT,
    order_name VARCHAR(50),
    order_price DECIMAL(10,2),
    FOREIGN KEY (customer_id) REFERENCES customer(customer_id)
);

INSERT INTO customer (customer_id, name, phone) VALUES
(1, 'Alice', '9876543210'),
(2, 'Bob', '9123456789'),
(3, 'charlie', '9988776655');


INSERT INTO orders (order_id, customer_id, order_name, order_price) VALUES
(101, 1, 'Laptop', 800.00),
(102, 2, 'Phone', 500.00),
(103, 1, 'Mouse', 20.00);
```

```
SELECT * FROM customer;


SELECT* FROM orders;



-- INNER JOIN.....


SELECT
customer.customer_id,customer.name,customer.phone,orders.order_id,orders.order_name,orders.o
rder_price from customer INNER JOIN orders ON customer.customer_id=orders.customer_id;



INSERT INTO customer (customer_id, name, phone)

VALUES (4, 'alex', '9847432219');   -- adding new record to customer table....


--  LEFT JOIN.....


SELECT
customer.customer_id,customer.name,customer.phone,orders.order_id,orders.order_name,orders.o
rder_price from customer LEFT JOIN orders ON customer.customer_id=orders.customer_id;


--  RIGHT JOIN.....


SELECT
customer.customer_id,customer.name,customer.phone,orders.order_id,orders.order_name,orders.o
rder_price from customer RIGHT JOIN orders ON customer.customer_id=orders.customer_id;


-- FULL JOIN.....


SELECT
customer.customer_id,customer.name,customer.phone,orders.order_id,orders.order_name,orders.o
rder_price from customer FULL JOIN orders  ON customer.customer_id=orders.customer_id;
```

Task 6) primary key and foreign key constraints.....

```
CREATE TABLE Persons (
    ID int NOT NULL PRIMARY KEY,   -- primary key : ID
    LastName varchar(255) NOT NULL,
    FirstName varchar(255),
    Age int
);

CREATE TABLE OrderDetails(

    OrderID int NOT NULL PRIMARY KEY,
    OrderNumber int NOT NULL,
    PersonID int FOREIGN KEY REFERENCES Persons(ID) --foreign key as personID from perosons ID (pk)
);
```