NAME: M. VAISHNAVI DEVI
REGISTRATION NO: 20BCE7454

ASSIGNMENT - 2

1) **Create, update, delete commands in my sql?**

Code:

Update command:

```
-- create a table
CREATE TABLE students (
 id INTEGER PRIMARY KEY,
 name VARCHAR(30) NOT NULL,
 gender CHAR(1) NOT NULL
);
-- insert some values
INSERT INTO students VALUES (1, 'Ram', 'M');
INSERT INTO students VALUES (2, 'Sita', 'F');
INSERT INTO students VALUES (3, 'Janaki', 'F');
-- fetch some values
UPDATE students
SET name = 'Arjun'
WHERE id = 1;
SELECT * FROM students;
```

OUTPUT:

Before updating

id	name	gender
1	Ram	M
2	Sita	F
3	Janaki	F

OUTPUT:

After updating

id	name	gender
1	Arjun	M
2	Sita	F
3	Janaki	F

Delete commands:

```
-- create a table
CREATE TABLE students (
 id INTEGER PRIMARY KEY,
 name VARCHAR(30) NOT NULL,
 gender CHAR(1) NOT NULL
);
-- insert some values
INSERT INTO students VALUES (1, 'Ram', 'M');
INSERT INTO students VALUES (2, 'Sita', 'F');
INSERT INTO students VALUES (3, 'Janaki', 'F');
-- fetch some values
DELETE FROM students
WHERE id = 2;
SELECT * FROM students;
```

OUTPUT:

Before deleting

id	name	gender
1	Arjun	M
2	Sita	F
3	Janaki	F

After deleting

id	name	gender
1	Ram	M
3	Janaki	F

2) Create a table and perform joins in mySql

Inserting data:

Code:

```
CREATE TABLE customers (
 id INT PRIMARY KEY,
 name VARCHAR(50),
 email VARCHAR(50)
);
CREATE TABLE orders (
 id INT PRIMARY KEY,
 order_date DATE,
 customer_id INT,
 FOREIGN KEY (customer_id) REFERENCES customers(id)
);
INSERT INTO customers (id, name, email)
VALUES (1, 'Arjun', 'arjun@example.com');
INSERT INTO customers (id, name, email)
VALUES (2, 'Krishna', 'krishna@example.com');
INSERT INTO customers (id, name, email)
VALUES (3, 'Ankitha', 'ankitha@example.com');
INSERT INTO customers (id, name, email)
VALUES (4, 'Nithya', 'nithya@example.com');
INSERT INTO customers (id, name, email)
VALUES (5, 'Karna', 'karna@example.com');
INSERT INTO orders (id, order_date, customer_id)
VALUES (1001, '2023-05-03', 1);
INSERT INTO orders (id, order_date, customer_id)
VALUES (1002, '2023-05-04', 1);
INSERT INTO orders (id, order_date, customer_id)
VALUES (1003, '2023-05-05', 2);
INSERT INTO orders (id, order_date, customer_id)
VALUES (1004, '2023-05-06', 3);
INSERT INTO orders (id, order_date, customer_id)
VALUES (1005, '2023-05-07', 4);
INSERT INTO orders (id, order_date, customer_id)
VALUES (1006, '2023-05-08', 4);
INSERT INTO orders (id, order_date, customer_id)
VALUES (1007, '2023-05-09', 5);
select * from customers;
select * from orders;
```

OUTPUT:

id	name	email
1	Arjun	arjun@example.com
2	Krishna	krishna@example.com
3	Ankitha	ankitha@example.com
4	Nithya	nithya@example.com
5	Karna	karna@example.com

id	order_date	customer_id
1001	2023-05-03	1
1002	2023-05-04	1
1003	2023-05-05	2
1004	2023-05-06	3
1005	2023-05-07	4
1006	2023-05-08	4
1007	2023-05-09	5

Performing joins:

Code:

```
SELECT customers.name, orders.order_date
FROM customers
INNER JOIN orders ON customers.id = orders.customer_id;
```

OUTPUT:

name	order_date
Arjun	2023-05-03
Arjun	2023-05-04
Krishna	2023-05-05
Ankitha	2023-05-06
Nithya	2023-05-07
Nithya	2023-05-08
Karna	2023-05-09

3) Create, update, delete commands in mongodb?

Update command:

Code:

```
db.students.insertMany([
```

```
 { id: 1, name: 'Ryan', gender: 'M' },
 { id: 2, name: 'Joanna', gender: 'F' }
]);
db.students.find({ gender: 'F' });
db.students.updateOne(
 { id: 1 },
 { $set: { name: "Arun", gender: "M" } }
);
```

OUTPUT:

```
 mycompiler_mongodb> ... ... ... {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("647374acd80605b9d08a774e"),
    '1': ObjectId("647374acd80605b9d08a774f")
  }
}
mycompiler_mongodb> [
  {
    _id: ObjectId("647374acd80605b9d08a774f"),
    id: 2,
    name: 'Joanna',
    gender: 'F'
  }
]
mycompiler_mongodb> ... ... ... {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
mycompiler_mongodb>
```

After updating

```
mycompiler_mongodb> [
  {
    _id: ObjectId("647376ab3ea4348f8a1ba3b6"),
    id: 1,
    name: 'Arun',
    gender: 'M'
  },
  {
    _id: ObjectId("647376ab3ea4348f8a1ba3b7"),
    id: 2,
    name: 'Joanna',
    gender: 'F'
  }
]
mycompiler_mongodb>
```

Deleting commands:

Code:

db.students.deleteOne({ id: 2 });
db.students.find()

After deleting:

```
mycompiler_mongodb> { acknowledged: true, deletedCount: 1 }
mycompiler_mongodb> [
  {
    _id: ObjectId("64737649afe11c2ce55017b9"),
    id: 1,
    name: 'Arun',
    gender: 'M'
  }
]
mycompiler_mongodb>
```