

SmartBridge
Modern Application Development (Java Spring Boot)
Week-2 Assignment

Student Name: Naman
Registration Number: 20BCT0202
Campus: VIT Vellore

MySQL Commands

-- Creating tables in MySQL

```
CREATE TABLE customers (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    cname VARCHAR(50),  
    email VARCHAR(50),  
    phno NUMERIC(10)  
);
```

```
CREATE TABLE orders (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    customer_id INT,  
    product_id INT,  
    quantity INT,  
    FOREIGN KEY (customer_id) REFERENCES customers(id)  
);
```

-- Create Row commands in MySQL

```
INSERT INTO customers (cname, email, phno)  
VALUES ('John Doe', 'johndoe@example.com', 9898098980),  
      ('Jane Smith', 'janesmith@example.com', 9112091120),  
      ('Bob Johnson', 'bobjohnson@example.com', 9001001009),  
      ('Sam Smith', 'samsmith@example.com', 9002090020);
```

```
INSERT INTO orders (customer_id, product_id, quantity)  
VALUES (1, 14627, 5),  
      (3, 13020, 10),  
      (3, 10023, 2),  
      (4, 10551, 20),  
      (1, 14002, 3),  
      (2, 10420, 11);
```

```
SELECT * FROM customers;  
SELECT * FROM orders;
```

-- Update, Delete Row commands in MySQL

```
UPDATE customers  
SET email = 'newemail@example.com', phno = 9010090100  
WHERE id = 1;
```

```
UPDATE orders  
SET quantity = 4  
WHERE id = 1;
```

```
DELETE FROM orders  
WHERE customer_id = 3;
```

```
SELECT * FROM customers;  
SELECT * FROM orders;
```

```
-- Joins in MySQL
SELECT orders.id AS order_id, customer_id, customers.cname, orders.product_id,
orders.quantity
FROM customers INNER JOIN orders ON customers.id = orders.customer_id;

SELECT orders.id AS order_id, customer_id, customers.cname, orders.product_id,
orders.quantity
FROM customers LEFT JOIN orders ON customers.id = orders.customer_id;

SELECT orders.id AS order_id, customer_id, customers.cname, orders.product_id,
orders.quantity
FROM customers RIGHT JOIN orders ON customers.id = orders.customer_id;

SELECT * FROM customers FULL JOIN orders;

-- Modifying tables in MySQL
ALTER TABLE customers
ADD address VARCHAR(80),
MODIFY cname VARCHAR(40),
DROP COLUMN phno;

-- Dropping tables in MySQL
DROP TABLE orders;
```

Output:

Output

Action Output

#	Time	Action	Message
✓ 1	21:34:07	SELECT orders.id AS order_id, customer_id, customers.cname, orders.product_id, orders.qu...	4 row(s) returned
✓ 2	21:34:07	SELECT orders.id AS order_id, customer_id, customers.cname, orders.product_id, orders.qu...	5 row(s) returned
✓ 3	21:34:07	SELECT orders.id AS order_id, customer_id, customers.cname, orders.product_id, orders.qu...	4 row(s) returned
✓ 4	21:34:07	SELECT * FROM customers FULL JOIN orders LIMIT 0, 1000	16 row(s) returned
✓ 5	21:35:23	DROP TABLE `smartbridge`.`orders`	0 row(s) affected
✓ 6	21:35:30	DROP TABLE `smartbridge`.`customers`	0 row(s) affected
✓ 7	21:35:33	CREATE TABLE customers (id INT AUTO_INCREMENT PRIMARY KEY, cname VAR...	0 row(s) affected
✓ 8	21:35:33	CREATE TABLE orders (id INT AUTO_INCREMENT PRIMARY KEY, customer_id INT...	0 row(s) affected
✓ 9	21:35:33	INSERT INTO customers (cname, email, phno) VALUES ('John Doe', 'johndoe@example.co...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0
✓ 10	21:35:33	INSERT INTO orders (customer_id, product_id, quantity) VALUES (1, 14627, 5), (3, 130...	6 row(s) affected Records: 6 Duplicates: 0 Warnings: 0
✓ 11	21:35:33	SELECT * FROM customers LIMIT 0, 1000	4 row(s) returned
✓ 12	21:35:33	SELECT * FROM orders LIMIT 0, 1000	6 row(s) returned
✓ 13	21:35:33	UPDATE customers SET email = 'newemail@example.com', phno = 9010090100 WHERE id...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
✓ 14	21:35:33	UPDATE orders SET quantity = 4 WHERE id = 1	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
✓ 15	21:35:33	DELETE FROM orders WHERE customer_id = 3	2 row(s) affected
✓ 16	21:35:33	SELECT * FROM customers LIMIT 0, 1000	4 row(s) returned
✓ 17	21:35:34	SELECT * FROM orders LIMIT 0, 1000	4 row(s) returned
✓ 18	21:35:34	SELECT orders.id AS order_id, customer_id, customers.cname, orders.product_id, orders.qu...	4 row(s) returned
✓ 19	21:35:34	SELECT orders.id AS order_id, customer_id, customers.cname, orders.product_id, orders.qu...	5 row(s) returned
✓ 20	21:35:34	SELECT orders.id AS order_id, customer_id, customers.cname, orders.product_id, orders.qu...	4 row(s) returned

Customers Table

	id	cname	email	phno
▶	1	John Doe	johndoe@example.com	9898098980
	2	Jane Smith	janesmith@example.com	9112091120
	3	Bob Johnson	bobjohnson@example.com	9001001009
	4	Sam Smith	samsmith@example.com	9002090020
*	NULL	NULL	NULL	NULL

Orders Table

	id	customer_id	product_id	quantity
▶	1	1	14627	5
	2	3	13020	10
	3	3	10023	2
	4	4	10551	20
	5	1	14002	3
	6	2	10420	11
*	NULL	NULL	NULL	NULL

Customers Table After Update, Delete Row Command

	id	cname	email	phno
▶	1	John Doe	newemail@example.com	9010090100
	2	Jane Smith	janesmith@example.com	9112091120
	3	Bob Johnson	bobjohnson@example.com	9001001009
	4	Sam Smith	samsmith@example.com	9002090020
*	NULL	NULL	NULL	NULL

Orders Table After Update, Delete Row Command

	id	customer_id	product_id	quantity
▶	1	1	14627	4
	4	4	10551	20
	5	1	14002	3
	6	2	10420	11
*	NULL	NULL	NULL	NULL

Inner Join

	order_id	customer_id	cname	product_id	quantity
▶	1	1	John Doe	14627	4
	5	1	John Doe	14002	3
	6	2	Jane Smith	10420	11
	4	4	Sam Smith	10551	20

Left Join

	order_id	customer_id	cname	product_id	quantity
▶	1	1	John Doe	14627	4
	5	1	John Doe	14002	3
	6	2	Jane Smith	10420	11
	NULL	NULL	Bob Johnson	NULL	NULL
	4	4	Sam Smith	10551	20

Right Join

	order_id	customer_id	cname	product_id	quantity
▶	1	1	John Doe	14627	4
	4	4	Sam Smith	10551	20
	5	1	John Doe	14002	3
	6	2	Jane Smith	10420	11

Full Join

	id	cname	email	phno	id	customer_id	product_id	quantity
▶	4	Sam Smith	samsmith@example.com	9002090020	1	1	14627	4
	3	Bob Johnson	bobjohnson@example.com	9001001009	1	1	14627	4
	2	Jane Smith	janesmith@example.com	9112091120	1	1	14627	4
	1	John Doe	newemail@example.com	9010090100	1	1	14627	4
	4	Sam Smith	samsmith@example.com	9002090020	4	4	10551	20
	3	Bob Johnson	bobjohnson@example.com	9001001009	4	4	10551	20
	2	Jane Smith	janesmith@example.com	9112091120	4	4	10551	20
	1	John Doe	newemail@example.com	9010090100	4	4	10551	20
	4	Sam Smith	samsmith@example.com	9002090020	5	1	14002	3
	3	Bob Johnson	bobjohnson@example.com	9001001009	5	1	14002	3
	2	Jane Smith	janesmith@example.com	9112091120	5	1	14002	3
	1	John Doe	newemail@example.com	9010090100	5	1	14002	3
	4	Sam Smith	samsmith@example.com	9002090020	6	2	10420	11
	3	Bob Johnson	bobjohnson@example.com	9001001009	6	2	10420	11
	2	Jane Smith	janesmith@example.com	9112091120	6	2	10420	11
	1	John Doe	newemail@example.com	9010090100	6	2	10420	11

MongoDB Commands

Creating a Database, Collections and Documents in MongoDB

```
> use SmartBridge
< already on db SmartBridge
> db.createCollection("Customers");
< { ok: 1 }
> db.customers.insertMany([
  { cname: 'John Doe', email: 'johndoe@example.com', phno: 9898098980 },
  { cname: 'Jane Smith', email: 'janesmith@example.com', phno: 9112091120 },
  { cname: 'Bob Johnson', email: 'bobjohnson@example.com', phno: 9001001009 },
  { cname: 'Sam Smith', email: 'samsmith@example.com', phno: 9002090020 }
]);
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("647390dbd54f06d4e79947ee"),
    '1': ObjectId("647390dbd54f06d4e79947ef"),
    '2': ObjectId("647390dbd54f06d4e79947f0"),
    '3': ObjectId("647390dbd54f06d4e79947f1")
  }
}
> db.createCollection("Orders");
< { ok: 1 }
> db.orders.insertMany([
  { customer_id: 1, product_id: 14627, quantity: 5 },
  { customer_id: 3, product_id: 13020, quantity: 10 },
  { customer_id: 3, product_id: 10023, quantity: 2 },
  { customer_id: 4, product_id: 10551, quantity: 20 },
  { customer_id: 1, product_id: 14002, quantity: 3 },
  { customer_id: 2, product_id: 10420, quantity: 11 }
]);
< {
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("64739121d54f06d4e79947f2"),
    '1': ObjectId("64739121d54f06d4e79947f3"),
    '2': ObjectId("64739121d54f06d4e79947f4"),
    '3': ObjectId("64739121d54f06d4e79947f5"),
    '4': ObjectId("64739121d54f06d4e79947f6"),
    '5': ObjectId("64739121d54f06d4e79947f7")
  }
}
```

SmartBridge.customers

Documents

Aggregations

Schema

Explain Plan

Indexes

Validation

Filter

Type a query: { field: 'value' }

ADD DATA

EXPORT DATA

_id: ObjectId('647390dbd54f06d4e79947ee')

cname: "John Doe"

email: "johndoe@example.com"

phno: 9898098980

_id: ObjectId('647390dbd54f06d4e79947ef')

cname: "Jane Smith"

email: "janesmith@example.com"

phno: 9112091120

_id: ObjectId('647390dbd54f06d4e79947f0')

cname: "Bob Johnson"

email: "bobjohnson@example.com"

phno: 9001001009

SmartBridge.orders

Documents

Aggregations

Schema

Explain Plan

Indexes

Validation

Filter

Type a query: { field: 'value' }

ADD DATA

EXPORT DATA

_id: ObjectId('64739121d54f06d4e79947f2')

customer_id: 1

product_id: 14627

quantity: 5

_id: ObjectId('64739121d54f06d4e79947f3')

customer_id: 3

product_id: 13020

quantity: 10

_id: ObjectId('64739121d54f06d4e79947f4')

customer_id: 3

product_id: 10023

quantity: 2

Updating a Document in MongoDB

```
> db.customers.updateMany(
  { id: 1 },
  {
    $set: {
      email: 'newemail@example.com',
      phno: 9010090100
    }
  }
);
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
SmartBridge>
```

Deleting Documents in MongoDB

```
> db.orders.deleteMany({ customer_id: 3 });
< {
  acknowledged: true,
  deletedCount: 2
}
SmartBridge>
```

Dropping a Collection in MongoDB

```
> db.orders.drop();
< true
SmartBridge>
```