

MODERN APPLICATION DEVELOPMENT

JAVA SPRING BOOT

WEEK-2 ASSIGNMENT

NAME : NISHANTH MEGAN K

REG NO : 20MIS0368

FILE LINK :

https://drive.google.com/drive/folders/1U4FiBzX0HIEfsasIOprROFWR7x-HbeEw?usp=share_link

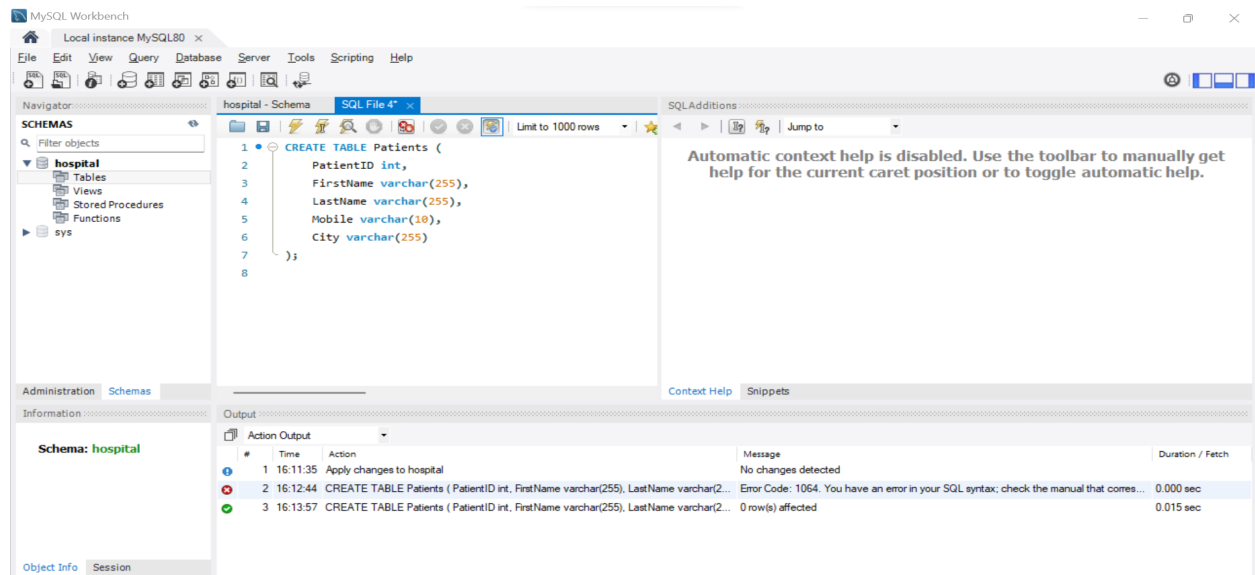
1) Create, Update, Delete commands in MySQL

Create table :

Code:

```
CREATE TABLE Patients (  
    PatientID int,  
    FirstName varchar(255),  
    LastName varchar(255),  
    Mobile varchar(10),  
    City varchar(255)  
);
```

Output:



Insert :

Code:

```
INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (1, 'Vicky', 'Jack', '8967543212', 'New York');
```

```
INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (2, 'Daniel', 'Smith', '7564343223', 'Los Angeles');
```

```
INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (3, 'John', 'Samuel', '6478655487', 'Chicago');
```

```
INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (4, 'Elizebeth', 'Victoria', '8954326754', 'San Francisco');
```

```
INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (5, 'Olivia', 'Brown', '6732548756', 'Seattle');
```

Output:

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator hospital - Schema SQL File 4* x

SCHEMAS

Filter objects

hospital

Tables

Views

Stored Procedures

Functions

sys

Administration Schemas

Information

Schema: hospital

Object Info Session

9 INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (1, 'Vikky', 'Jack', '8967543212', 'New York');
10
11
12 INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (2, 'Daniel', 'Smith', '7564343223', 'Los Angeles');
13
14
15 INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (3, 'John', 'Samuel', '6478655487', 'Chicago');
16
17
18 INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (4, 'Elizebeth', 'Victoria', '8954326754', 'San Francisco');
19
20
21 INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Context Help Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
8	16:37:35	INSERT INTO Persons (PersonID, FirstName, LastName, Mobile, City) VALUES (2, ...	Error Code: 1054. Unknown column 'Mobile' in field list	0.000 sec
9	16:42:08	INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City) VALUES (1, ...	1 row(s) affected	0.000 sec
10	16:42:08	INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City) VALUES (2, ...	1 row(s) affected	0.000 sec
11	16:42:08	INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City) VALUES (3, ...	1 row(s) affected	0.000 sec
12	16:42:08	INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City) VALUES (4, ...	1 row(s) affected	0.000 sec
13	16:42:08	INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City) VALUES (5, ...	1 row(s) affected	0.000 sec

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator hospital - Schema SQL File 4* x

SCHEMAS

Filter objects

hospital

Tables

Views

Stored Procedures

Functions

sys

Administration Schemas

Information

Schema: hospital

Object Info Session

7
8
9 SELECT * FROM Patients;
10 INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (1, 'Vikky', 'Jack', '8967543212', 'New York');
11
12 INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City)
VALUES (2, 'Daniel', 'Smith', '7564343223', 'Los Angeles');
13

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Context Help Snippets

Result Grid

PatientID	FirstName	LastName	Mobile	City
1	Vikky	Jack	8967543212	New York
1	Vikky	Jack	8967543212	New York
2	Daniel	Smith	7564343223	Los Angeles
3	John	Samuel	6478655487	Chicago
4	Elizebeth	Victoria	8954326754	San Francisco
5	Olivia	Brown	6732548756	Seattle

Patients 2 x

Read Only

Context Help Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
9	16:42:08	INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City) VALUES (1, ...	1 row(s) affected	0.000 sec
10	16:42:08	INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City) VALUES (2, ...	1 row(s) affected	0.000 sec
11	16:42:08	INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City) VALUES (3, ...	1 row(s) affected	0.000 sec
12	16:42:08	INSERT INTO Patients (PatientID, FirstName, LastName, Mobile, City) VALUES (4, ...	1 row(s) affected	0.000 sec

Update:

Code:

UPDATE Patients

SET City = 'London'

WHERE PatientID = 4;

MySQL Workbench interface showing the 'hospital' schema. The SQL editor contains the following code:

```

3  FirstName varchar(255),
4  LastName varchar(255),
5  Mobile varchar(10),
6  City varchar(255)
7  );
8  • SELECT * FROM Patients;
9  • TRUNCATE Patients (PatientID, FirstName, LastName, Mobile, City)

```

The Result Grid displays the following data:

PatientID	FirstName	LastName	Mobile	City
1	Vikky	Jack	8967543212	New York
1	Vikky	Jack	8967543212	New York
2	Daniel	Smith	7564343223	Los Angeles
3	John	Samuel	6478655487	Chicago
4	Elzebeth	Victoria	8954326754	London
5	Olivia	Brown	6732548756	Seattle

The Output pane shows the execution of the UPDATE query:

#	Time	Action	Message	Duration / Fetch
21	18:38:33	UPDATE Patients SET City = 'London' WHERE PatientID = 4	Error Code: 1175. You are using safe update mode and you tried to update a table ...	0.000 sec
22	18:39:33	UPDATE Patients SET City = 'London' WHERE PatientID = 4	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
23	18:39:49	SELECT * FROM Patients LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Delete:

DELETE FROM Patients WHERE PatientID = 3

Output:

MySQL Workbench interface showing the 'hospital' schema. The SQL editor contains the following code:

```

3  FirstName varchar(255),
4  LastName varchar(255),
5  Mobile varchar(10),
6  City varchar(255)
7  );
8  • SELECT * FROM Patients;
9  • TRUNCATE Patients (PatientID, FirstName, LastName, Mobile, City)

```

The Result Grid displays the following data:

PatientID	FirstName	LastName	Mobile	City
1	Vikky	Jack	8967543212	New York
1	Vikky	Jack	8967543212	New York
2	Daniel	Smith	7564343223	Los Angeles
4	Elzebeth	Victoria	8954326754	London
5	Olivia	Brown	6732548756	Seattle

The Output pane shows the execution of the DELETE query:

#	Time	Action	Message	Duration / Fetch
23	18:39:49	SELECT * FROM Patients LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
24	18:40:40	DELETE FROM Patients WHERE PatientID = 3	1 row(s) affected	0.000 sec
25	18:41:03	SELECT * FROM Patients LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

2) Create table and joins in MySQL

Code:

Create table:

```
CREATE TABLE Employers (
  Id INT PRIMARY KEY,
  Name VARCHAR(50),
  Email VARCHAR(50)
);
```

```
SELECT * FROM Employers;
```

Output:

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'Schemas' tree with 'hospital' selected. The main editor shows the SQL script:

```
29
30 CREATE TABLE Employers (
31   Id INT PRIMARY KEY,
32   Name VARCHAR(50),
33   Email VARCHAR(50)
34 );
35
36 SELECT * FROM Employers;
37
```

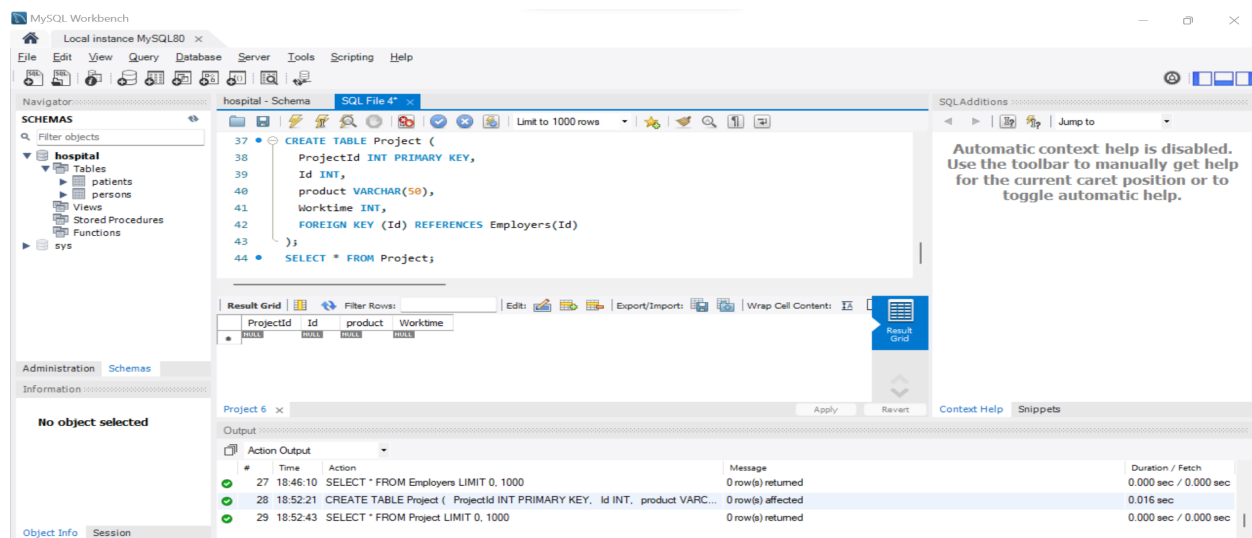
The 'Result Grid' is empty. The 'Output' pane at the bottom shows the execution results:

#	Time	Action	Message	Duration / Fetch
25	18:41:03	SELECT * FROM Patients LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
26	18:45:43	CREATE TABLE Employers (Id INT PRIMARY KEY, Name VARCHAR(50), Em...	0 row(s) affected	0.015 sec
27	18:46:10	SELECT * FROM Employers LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

Create Table Project :

```
CREATE TABLE Project (
  ProjectId INT PRIMARY KEY,
  Id INT,
  product VARCHAR(50),
  Worktime INT,
  FOREIGN KEY (Id) REFERENCES Employers(Id)
);
SELECT * FROM Project;
```

Output :



Inner Join :

Code:

```
SELECT Employers.Name, Project.product
FROM Employers
INNER JOIN Project ON Employers.Id = Project.Id;
```

Output:

MySQL Workbench interface showing a query execution error. The query is as follows:

```

41 Worktime INT,
42 FOREIGN KEY (Id) REFERENCES Employers(Id)
43 );
44 • SELECT * FROM Project;
45
46 • SELECT Employers.Name, Project.product
47 FROM Employers
48 INNER JOIN Project ON Employers.Id = Project.Id;

```

The error message is: "Error Code: 1052. Column 'Id' in on clause is ambiguous". The output shows 0 rows returned.

#	Time	Action	Message	Duration / Fetch
29	18:52:43	SELECT * FROM Project LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
30	18:58:00	SELECT Employers.Name, Project.product FROM Employers INNER JOIN Project ...	Error Code: 1052. Column 'Id' in on clause is ambiguous	0.000 sec
31	18:58:55	SELECT Employers.Name, Project.product FROM Employers INNER JOIN Project ...	0 row(s) returned	0.000 sec / 0.000 sec

Left Join:

SELECT Employers.Name, Project.product
 FROM Employers
 left JOIN Project ON Employers.Id = Project.Id;

MySQL Workbench interface showing the same query executed with a LEFT JOIN. The query is as follows:

```

45
46 • SELECT Employers.Name, Project.product
47 FROM Employers
48 INNER JOIN Project ON Employers.Id = Project.Id;
49
50 • SELECT Employers.Name, Project.product
51 FROM Employers
52 left JOIN Project ON Employers.Id = Project.Id;

```

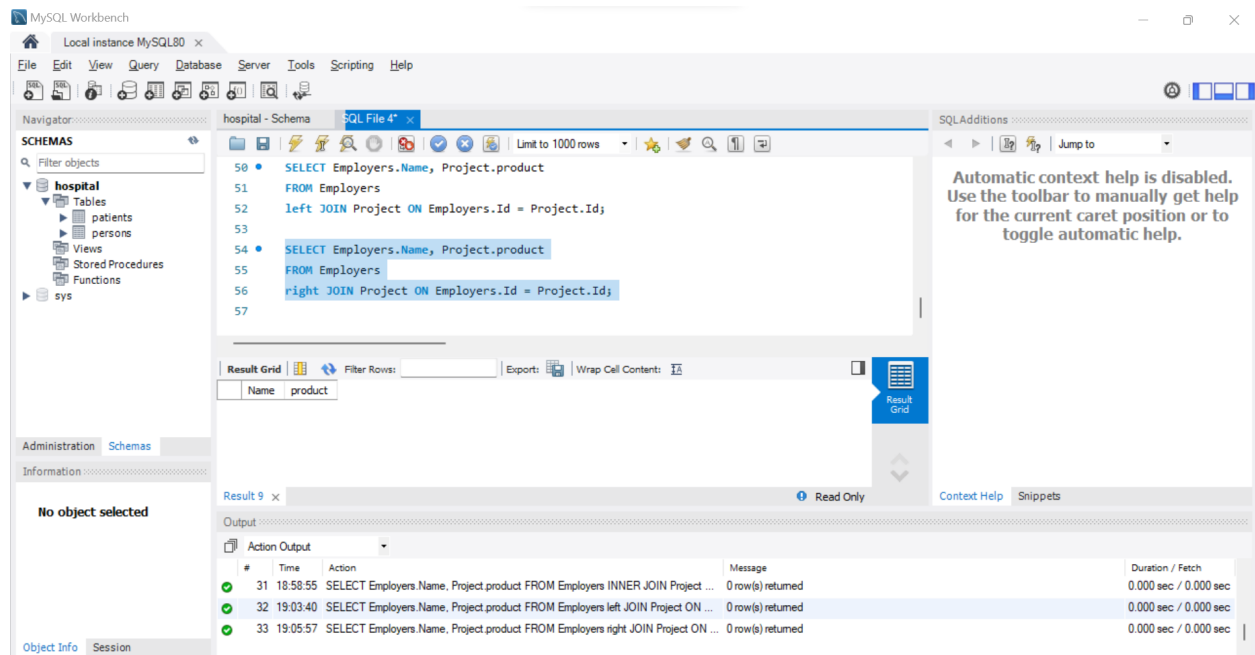
The error message is: "Error Code: 1052. Column 'Id' in on clause is ambiguous". The output shows 0 rows returned.

#	Time	Action	Message	Duration / Fetch
30	18:58:00	SELECT Employers.Name, Project.product FROM Employers INNER JOIN Project ...	Error Code: 1052. Column 'Id' in on clause is ambiguous	0.000 sec
31	18:58:55	SELECT Employers.Name, Project.product FROM Employers INNER JOIN Project ...	0 row(s) returned	0.000 sec / 0.000 sec
32	19:03:40	SELECT Employers.Name, Project.product FROM Employers left JOIN Project ON ...	0 row(s) returned	0.000 sec / 0.000 sec

Right Join :

```
SELECT Employers.Name, Project.product
FROM Employers
right JOIN Project ON Employers.Id = Project.Id;
```

Output :



3) Create , update, delete commands in mongo

Create:

```
db.Patients.insertOne({patientid:"1",name:"Micheal",age:"20",City:"London"})
```

```
db.Patients.insertOne({patientid:"3",name:"John",age:"21",City:"Paris"})
```

```
db.Patients.insertOne({patientid:"2",name:"Maxwell",age:"19",City:"Bangalore"})
```

```
db.Patients.insertOne({patientid:"4",name:"Smith",age:"23",City:"San Francisco"})
```

```
db.Patients.insertOne({patientid:"5",name:"Virat",age:"22",Adress:"Delhi"})
```



```

MongoDB Compass - localhost:27017/Hospital.Patient
Connect Edit View Collection Help

localhost:27017 Documents Hospital.Patient

My Queries

> MONGODB

> db.Patients.insertOne({patientid:"1",name:"Micheal",age:"20",City:"London"})

db.Patients.insertOne({patientid:"3",name:"John",age:"21",City:"Paris"})

db.Patients.insertOne({patientid:"2",name:"Maxwell",age:"19",City:"Bangalore"})

db.Patients.insertOne({patientid:"4",name:"Smith",age:"23",City:"San Francisco"})

db.Patients.insertOne({patientid:"5",name:"Virat",age:"22",Adress:"Delhi"})

< {
  acknowledged: true,
  insertedId: ObjectId("64735df450f18fa8b1a10bf6")
}
test>

```

Update :

```
db.Patients.updateOne({_id:ObjectId('64735df450f18fa8b1a10bf6')}, {$set:{age:"24"}})
```

```
db.Patients.findOne({age:"24"})
```

```

MongoDB Compass - localhost:27017/Hospital.Patient
Connect Edit View Collection Help

localhost:27017 Documents Hospital.Patient

My Queries

> MONGODB

db.Patients.insertOne({patientid:"4",name:"Smith",age:"23",City:"San Francisco"})

db.Patients.insertOne({patientid:"5",name:"Virat",age:"22",Adress:"Delhi"})

< {
  acknowledged: true,
  insertedId: ObjectId("64735df450f18fa8b1a10bf6")
}
> db.Patients.updateOne({_id:ObjectId('64735df450f18fa8b1a10bf6')}, {$set:{age:"24"}})

db.Patients.findOne({age:"24"})

< {
  _id: ObjectId("64735df450f18fa8b1a10bf6"),
  patientid: '5',
  name: 'Virat',
  age: '24',
  Adress: 'Delhi'
}
test>

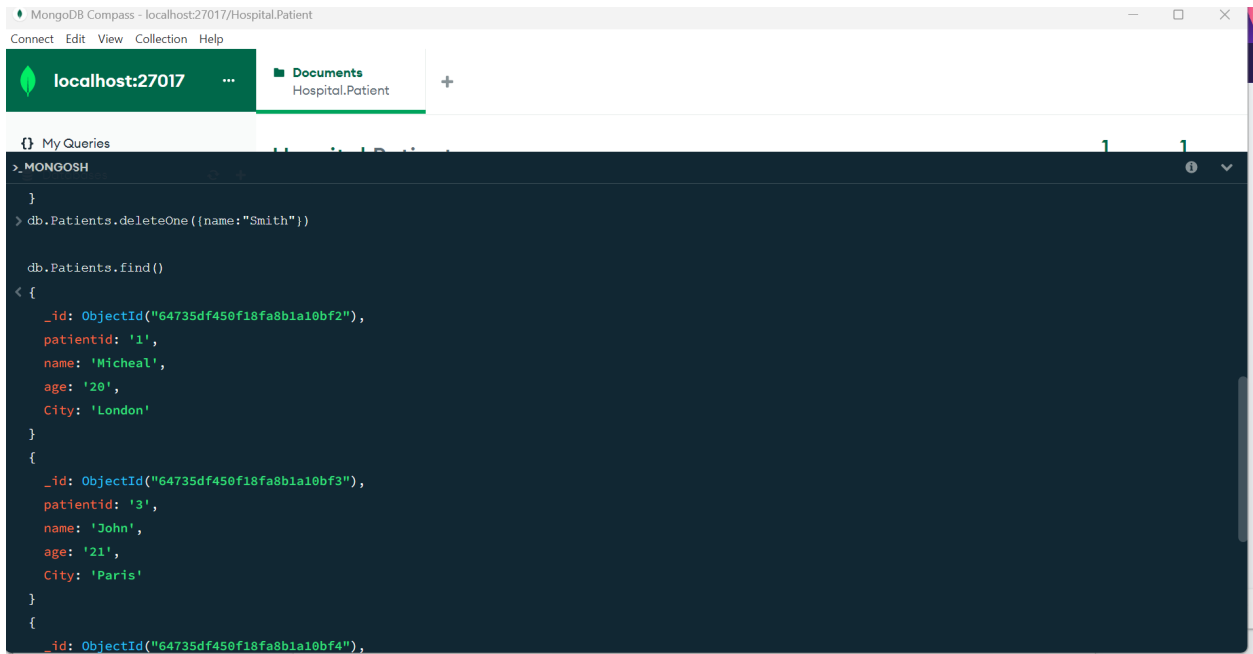
```

Delete :

```
db.Patients.deleteOne({name:"Smith"})
```

```
db.Patients.find()
```

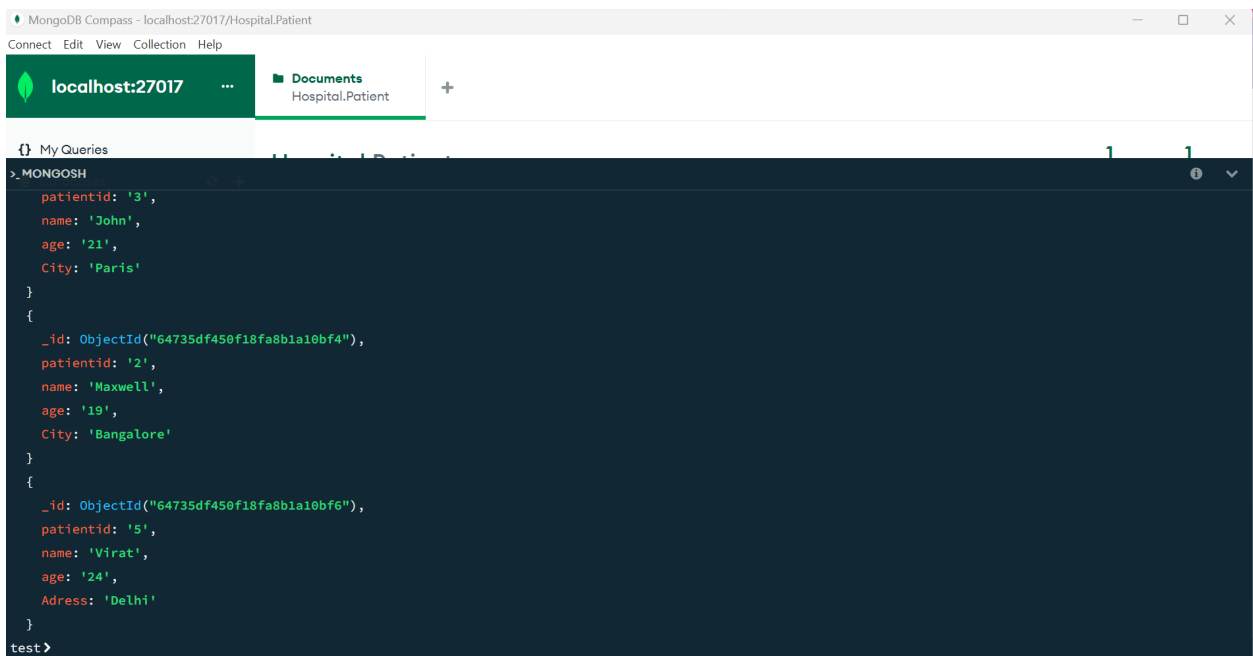
Output :



The screenshot shows the MongoDB Compass interface. The top bar indicates the connection to 'localhost:27017' and the 'Hospital.Patient' collection. The 'My Queries' tab is active, displaying a JavaScript console with the following commands and output:

```
> _MONGOSH
>
> db.Patients.deleteOne({name:"Smith"})

db.Patients.find()
< {
  _id: ObjectId("64735df450f18fa8b1a10bf2"),
  patientid: '1',
  name: 'Micheal',
  age: '20',
  City: 'London'
}
{
  _id: ObjectId("64735df450f18fa8b1a10bf3"),
  patientid: '3',
  name: 'John',
  age: '21',
  City: 'Paris'
}
{
  _id: ObjectId("64735df450f18fa8b1a10bf4"),
```



The screenshot continues from the previous one, showing the output of the 'find()' query. The remaining patients in the collection are:

```

  patientid: '3',
  name: 'John',
  age: '21',
  City: 'Paris'
}
{
  _id: ObjectId("64735df450f18fa8b1a10bf4"),
  patientid: '2',
  name: 'Maxwell',
  age: '19',
  City: 'Bangalore'
}
{
  _id: ObjectId("64735df450f18fa8b1a10bf6"),
  patientid: '5',
  name: 'Virat',
  age: '24',
  Adress: 'Delhi'
}
test>
```