

JAVA SPRING BOOT – MySQL , MongoDB

ASSIGNMENT – 2

SWARNA SAI SREEKAR

20BCE7105

1. Create database and tables

The screenshot shows the MySQL Workbench interface. The 'SCHEMAS' panel on the left shows the 'assignment2' database selected. The 'SQL File 1' editor contains the following SQL code:

```
1 create database assignment2;
2 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(1,'Harika','CSE','Vijayawada','9845775674');
3 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(2,'Ravi','CSE','Kajapattinam','9450379832');
4 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(3,'saran','MECH','Hyderabad','7236780946');
5 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(4,'Heera','ECE','Banglore','8613964278');
6 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(5,'Heera','ECE','Banglore','8613964278');
7 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(6,'charan','BBA','Tamilnadu','7031278254');
8 select * from assignment2.university;
```

The 'Result Grid' shows the output of the 'select * from assignment2.university;' query:

unid	Student	Branch	Address	phoneno
1	Harika	CSE	Vijayawada	9845775674
2	Ravi	CSE	Kajapattinam	9450379832
3	saran	MECH	Hyderabad	7236780946
4	Heera	ECE	Banglore	8613964278
5	Heera	ECE	Banglore	8613964278
6	charan	BBA	Tamilnadu	7031278254

The 'Output' panel shows the execution of the insert statements, with messages indicating that 1 row(s) affected for each insert operation.

Update command

The screenshot shows the MySQL Workbench interface. The 'SQL File 2' editor contains the following SQL code:

```
1 create database assignment2;
2 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(1,'Harika','CSE','Vijayawada','9845775674');
3 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(2,'Ravi','CSE','Kajapattinam','9450379832');
4 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(3,'saran','MECH','Hyderabad','7236780946');
5 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(4,'Heera','ECE','Banglore','8613964278');
6 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(5,'Heera','ECE','Banglore','8613964278');
7 insert into assignment2.university(unid,Student,Branch,Address,phoneno) values(6,'charan','BBA','Tamilnadu','7031278254');
8 select * from assignment2.university;
9 update assignment2.university set Branch='ECE' where phoneno='7031278254';
```

The 'Result Grid' shows the output of the 'select * from assignment2.university;' query, which is identical to the one in the first screenshot.

The 'Output' panel shows the execution of the update statement, with messages indicating that 1 row(s) affected for each insert operation and 5 row(s) returned for the select statement. The final message shows an error:

```
18 13:42:42 [Update assignment2.university set Branch='ECE' where Student='charan'] Error-Code: 1175 You are using safe update mode and you tried to update a table without a WHERE clause
```

Delete command

The screenshot shows the SQL Studio interface with the 'university' table selected in the 'SCHEMAS' pane. The SQL editor contains the following commands:

```
1 * create database assignment2;
2 * insert into assignment2.university(sno,Student,Branch,Address,phoneno) values(1,'harika','CSE','vijayawada','9895756374');
3 * insert into assignment2.university(sno,Student,Branch,Address,phoneno) values(2,'har','CSE','Raipur','9496079112');
4 * insert into assignment2.university(sno,Student,Branch,Address,phoneno) values(3,'sarat','MECH','Hyderabad','7224783368');
5 * insert into assignment2.university(sno,Student,Branch,Address,phoneno) values(4,'siree','ECE','Bangalore','9852964170');
6 * insert into assignment2.university(sno,Student,Branch,Address,phoneno) values(5,'charan','BBA','Tandhadi','8852664278');
7 * select * from assignment2.university;
8 * update assignment2.university set Branch='BBA' where phoneno='7224783368';
9 * DELETE from assignment2.university where phoneno='9496079112';
```

The 'Result Grid' shows the data after the DELETE command:

sno	Student	Branch	Address	phoneno
1	harika	BBA	Hyderabad	7224783368
2	charan	BBA	Tandhadi	781278354
3	Harika	BCE	Bangalore	8852664278
4	Harika	CSE	Hyderabad	984775674
5	Harika	BBA	BBA	BBA

The 'Action Output' pane shows the execution results:

#	Time	Action	Message
21	13:44:47	Update assignment2.university set Branch='BBA' where phoneno='781278354'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
22	13:44:51	select * from assignment2.university LIMIT 0, 1000	5 row(s) returned
23	13:45:30	Update assignment2.university set Branch='BBA' where phoneno='7224783368'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
24	13:45:35	select * from assignment2.university LIMIT 0, 1000	5 row(s) returned
25	13:51:45	DELETE from assignment2.university where phoneno='9496079112'	1 row(s) affected
26	13:51:51	select * from assignment2.university LIMIT 0, 1000	4 row(s) returned

2. Creating Tables

The screenshot shows the SQL Studio interface with the 'department' table selected in the 'SCHEMAS' pane. The SQL editor contains the following commands:

```
5 * insert into assignment2.university(sno,Student,Branch,Address,phoneno) values(5,'Harika','ECE','Bangalore','8852664278');
6 * insert into assignment2.university(sno,Student,Branch,Address,phoneno) values(6,'charan','BBA','Tandhadi','781278354');
7 * select * from assignment2.university;
8 * update assignment2.university set Branch='BBA' where phoneno='7224783368';
9 * DELETE from assignment2.university where phoneno='9496079112';
10 * create TABLE assignment2.FACULTY(SNO INT,DEPARTMENTID INT,FACULTYID INT,FACULTYNAME VARCHAR(75) , PRIMARY KEY(SNO),
11 FOREIGN KEY(DEPARTMENTID) REFERENCES DEPARTMENT(DEPARTMENTID));
12 * insert into assignment2.DEPARTMENT VALUES (1,1,'CSE');
13 * insert into assignment2.DEPARTMENT VALUES (2,2,'ECE');
14 * insert into assignment2.DEPARTMENT VALUES (3,3,'MECH');
15 * insert into assignment2.DEPARTMENT VALUES (4,4,'MATH');
16 * insert into assignment2.DEPARTMENT VALUES (5,5,'ENG');
17 * select * from assignment2.department order by sno;
```

The 'Result Grid' shows the data after the commands:

sno	departmentid	departmentname
1	1	CSE
2	2	BCE
3	3	MECH
4	4	MATH
5	5	ENG

The 'Action Output' pane shows the execution results:

#	Time	Action	Message
37	15:21:37	select * from assignment2.department LIMIT 0, 1000	5 row(s) returned
38	15:22:57	select * from assignment2.department order by sno LIMIT 0, 1000	5 row(s) returned
39	15:23:59	create TABLE assignment2.STUDENT(SNO INT,DEPARTMENTID INT,STUDENTID INT,STUDENTNAME VARCHAR(75) , PRIMARY KEY(SNO), FOREIGN KEY(DEPARTMENTID) REFERENCES DEPARTMENT(DEPARTMENTID));	Error Code: 1064 You have an error in your SQL syntax; check the manual that correspond to this MySQL server version for the right syntax to use near 'create TABLE assignment2.STUDENT(SNO INT,DEPARTMENTID INT,STUDENTNAME VA'
40	15:24:56	create TABLE assignment2.DHLOSNO INT,DEPARTMENTID INT,STUDENTID INT,STUDENTNAME VARCHAR(75) , PRIMARY KEY(SNO), FOREIGN KEY(DEPARTMENTID) REFERENCES DEPARTMENT(DEPARTMENTID));	Error Code: 1050 Table 'student' already exists
41	15:25:24	create TABLE assignment2.DHLOSNO INT,DEPARTMENTID INT,STUDENTID INT,STUDENTNAME VARCHAR(75) , PRIMARY KEY(SNO), FOREIGN KEY(DEPARTMENTID) REFERENCES DEPARTMENT(DEPARTMENTID));	0 row(s) affected
42	15:26:32	select * from assignment2.department order by sno LIMIT 0, 1000	5 row(s) returned

Inner Join

The screenshot shows a SQL query window with the following SQL code:

```
23 * Insert into assignment2.FACULTY VALUES (4,20,200,'sheela');
24 * Insert into assignment2.FACULTY VALUES (5,2,400,'harish');
25 * select * from assignment2.FACULTY ORDER BY SNO;
26
27 * SELECT * FROM assignment2.department JOIN assignment2.FACULTY ON DEPARTMENT.DEPARTMENTID=FACTY.DEPARTMENTID;
```

The 'Result Grid' displays the following data:

sno	departmentid	departmentname	SNO	DEPARTMENTID	FACULTYID	FACULTYNAME
1	1	CSE	1	1	203	Rajesh
1	1	CSE	2	1	300	chandu
3	2	MECH	5	2	400	harish
2	3	ECE	3	3	203	Rajesh
4	20	MATH	4	20	280	sheela

The 'Action Output' window shows the following messages:

#	Time	Action	Message
50	15:33:00	Insert into assignment2.CHILD VALUES (1,1,7915,Prasanth)	1 row(s) affected
51	15:33:03	Insert into assignment2.CHILD VALUES (2,2,6790,Ranu)	1 row(s) affected
52	15:33:42	select * from assignment2.FACULTY ORDER BY SNO LIMIT 0, 1000	Error Code: 1048, Unknown database 'assignment2'
53	15:34:22	select * from assignment2.FACULTY ORDER BY SNO LIMIT 0, 1000	5 row(s) returned
54	15:36:55	SELECT * FROM assignment2.department JOIN assignment2.FACULTY ON DEPARTMENT.DEPARTMENTID=FACTY.DEPARTMENTID;	5 row(s) returned

Left Join

The screenshot shows a SQL query window with the following SQL code:

```
23 * Insert into assignment2.FACULTY VALUES (4,20,200,'sheela');
24 * Insert into assignment2.FACULTY VALUES (5,2,400,'harish');
25 * select * from assignment2.FACULTY ORDER BY SNO;
26
27 * SELECT * FROM assignment2.department LEFT JOIN assignment2.FACULTY ON DEPARTMENT.DEPARTMENTID=FACTY.DEPARTMENTID;
```

The 'Result Grid' displays the following data:

sno	departmentid	departmentname	SNO	DEPARTMENTID	FACULTYID	FACULTYNAME
1	1	CSE	1	1	203	Rajesh
1	1	CSE	2	1	300	chandu
3	2	MECH	5	2	400	harish
2	3	ECE	3	3	203	Rajesh
4	20	MATH	4	20	280	sheela
5	23	ENG	0000	0000	0000	0000

The 'Action Output' window shows the following messages:

#	Time	Action	Message
51	15:33:03	Insert into assignment2.CHILD VALUES (2,2,6790,Ranu)	1 row(s) affected
52	15:33:42	select * from assignment2.FACULTY ORDER BY SNO LIMIT 0, 1000	Error Code: 1048, Unknown database 'assignment2'
53	15:34:22	select * from assignment2.FACULTY ORDER BY SNO LIMIT 0, 1000	5 row(s) returned
54	15:36:55	SELECT * FROM assignment2.department LEFT JOIN assignment2.FACULTY ON DEPARTMENT.DEPARTMENTID=FACTY.DEPARTMENTID;	5 row(s) returned

Right Join

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'assignment2' database is selected, and the 'Tables' folder is expanded. The 'assignment2' database is highlighted. The main pane displays a SQL query window with the following code:

```

23 * Insert into assignment2.FACULTY VALUES (1,20,200,'sheela');
24 * Insert into assignment2.FACULTY VALUES (1,2,400,'Nareesh');
25 * select * from assignment2.FACULTY ORDER BY SNO)
26
27 * SELECT * FROM assignment2.department RIGHT JOIN assignment2.FACULTY ON DEPARTMENT.DEPARTMENTID= FACULTY.DEPARTMENTID

```

The 'Result Grid' shows the output of the query:

sno	departmentid	departmentName	SNO	DEPARTMENTID	FACULTYID	FACULTYNAME
1	1	CSE	1	1	203	Rajesh
1	1	CSE	2	1	300	chandu
2	3	ECE	3	3	203	Rakesh
4	20	MATH	4	20	280	sheela
3	2	MECH	5	2	450	Nareesh

The 'Output' pane at the bottom shows 'No object selected'.

Full Join

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'assignment2' database is selected, and the 'Tables' folder is expanded. The 'assignment2' database is highlighted. The main pane displays a SQL query window with the following code:

```

25 * select * from assignment2.FACULTY ORDER BY SNO)
26
27 * SELECT * FROM assignment2.department FULL JOIN assignment2.FACULTY

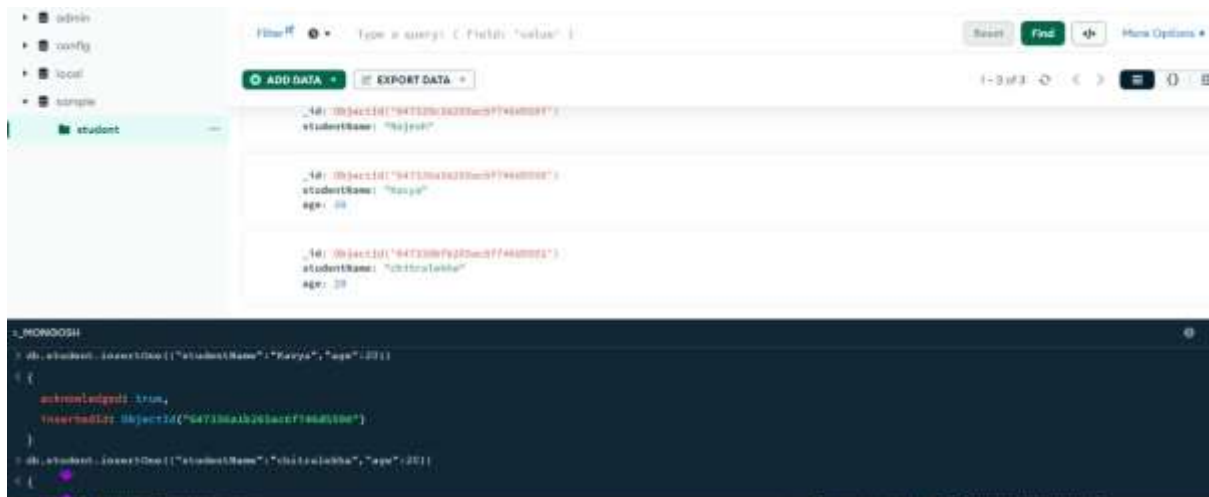
```

The 'Result Grid' shows the output of the query:

sno	departmentid	departmentName	SNO	DEPARTMENTID	FACULTYID	FACULTYNAME
5	23	ENG	1	1	203	Rajesh
4	20	MATH	1	1	203	Rajesh
2	3	ECE	1	1	203	Rajesh
3	2	MECH	1	1	203	Rajesh
1	1	CSE	1	1	203	Rajesh
5	23	ENG	2	1	300	chandu
4	20	MATH	2	1	300	chandu
2	3	ECE	2	1	300	chandu
3	2	MECH	2	1	300	chandu
1	1	CSE	2	1	300	chandu
5	23	ENG	3	3	203	Rakesh
4	20	MATH	3	3	203	Rakesh
2	3	ECE	3	3	203	Rakesh
3	2	MECH	3	3	203	Rakesh
1	1	CSE	3	3	203	Rakesh
5	23	ENG	4	20	280	sheela
4	20	MATH	4	20	280	sheela
2	3	ECE	4	20	280	sheela
3	2	MECH	4	20	280	sheela
1	1	CSE	4	20	280	sheela
5	23	ENG	5	2	450	Nareesh
4	20	MATH	5	2	450	Nareesh

The 'Output' pane at the bottom shows 'Action Output'.

3. MongoDB Create and Insert



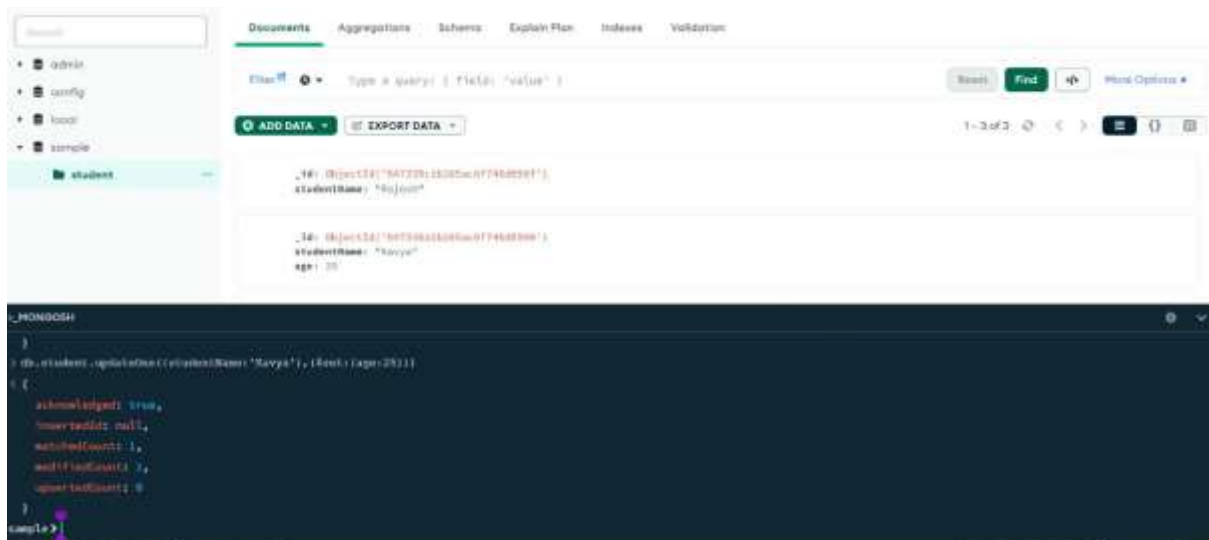
The screenshot shows the MongoDB Compass interface. On the left, the 'student' collection is selected. The main panel displays three documents in the 'student' collection:

- Document 1: `{ "_id": ObjectId("647336a3256c9ff94b00ff7"), "studentName": "Rajesh" }`
- Document 2: `{ "_id": ObjectId("647336a3256c9ff94b00ff8"), "studentName": "Kavya", "age": 20 }`
- Document 3: `{ "_id": ObjectId("647336a3256c9ff94b00ff9"), "studentName": "Chitralekha", "age": 20 }`

Below the documents, the MongoDB shell shows the commands used to insert these documents:

```
> use student; insertOne({ "studentName": "Kavya", "age": 20 })
{
  acknowledged: true,
  insertedId: ObjectId("647336a3256c9ff94b00ff8")
}
> use student; insertOne({ "studentName": "Chitralekha", "age": 20 })
{
  acknowledged: true,
  insertedId: ObjectId("647336a3256c9ff94b00ff9")
}
```

Update Command



The screenshot shows the MongoDB Compass interface. On the left, the 'student' collection is selected. The main panel displays two documents in the 'student' collection:

- Document 1: `{ "_id": ObjectId("647336a3256c9ff94b00ff7"), "studentName": "Rajesh" }`
- Document 2: `{ "_id": ObjectId("647336a3256c9ff94b00ff8"), "studentName": "Kavya", "age": 20 }`

Below the documents, the MongoDB shell shows the command used to update the document:

```
> use student; updateOne({ studentName: "Kavya" }, { $set: { age: 25 } })
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Delete Command



The screenshot shows the MongoDB Compass interface. On the left, the 'student' collection is selected. The main panel displays two documents in the 'student' collection:

- Document 1: `{ "_id": ObjectId("647336a3256c9ff94b00ff8"), "studentName": "Kavya", "age": 20 }`
- Document 2: `{ "_id": ObjectId("647336a3256c9ff94b00ff9"), "studentName": "Chitralekha", "age": 20 }`

Below the documents, the MongoDB shell shows the command used to delete the document:

```
> use student; deleteOne({ studentName: "Rajesh" })
{
  acknowledged: true,
  deletedCount: 1
}
```