

# DBMS ASSIGNMENT-3

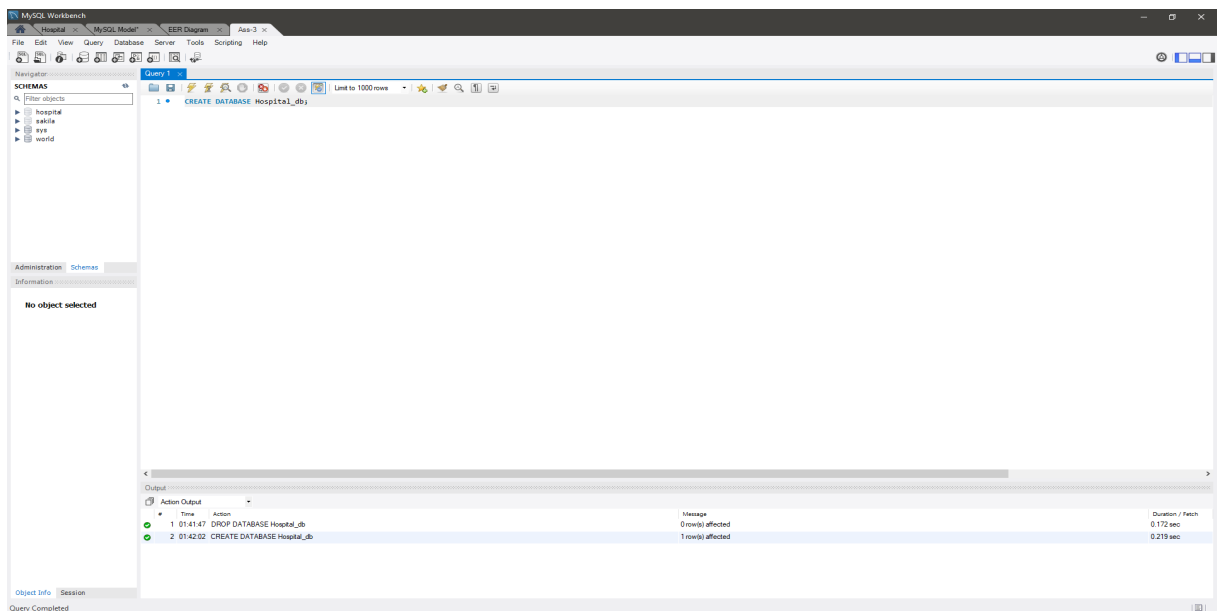
## HOSPITAL DATABASE

1. Show how to Create and Drop Database.

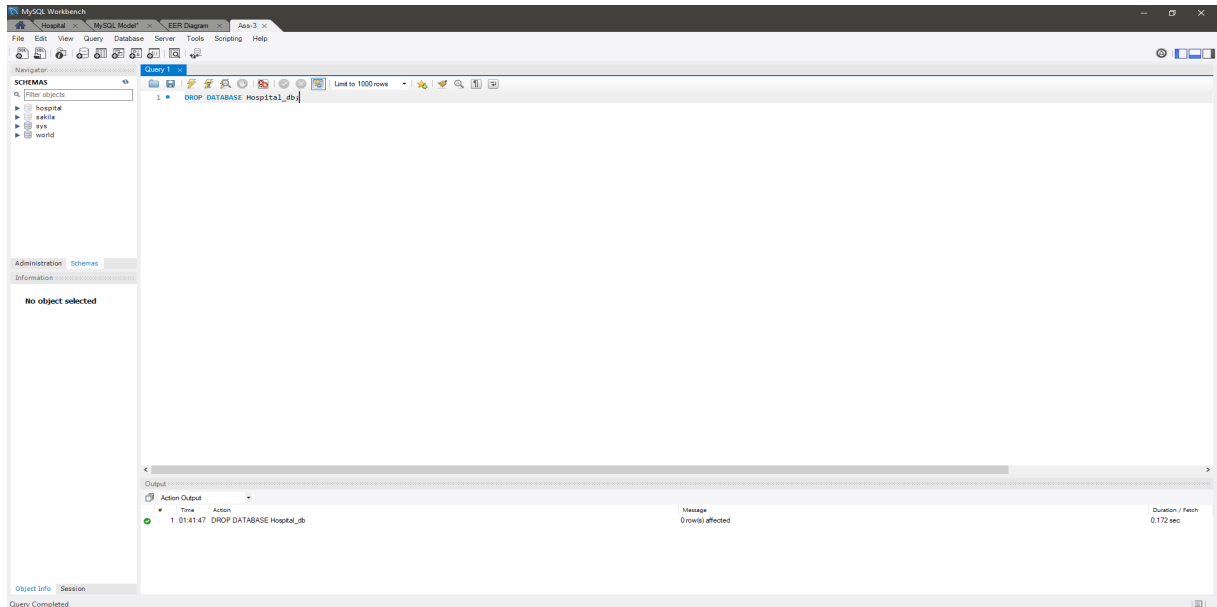
- Create Table

Query : create database hospital\_db;

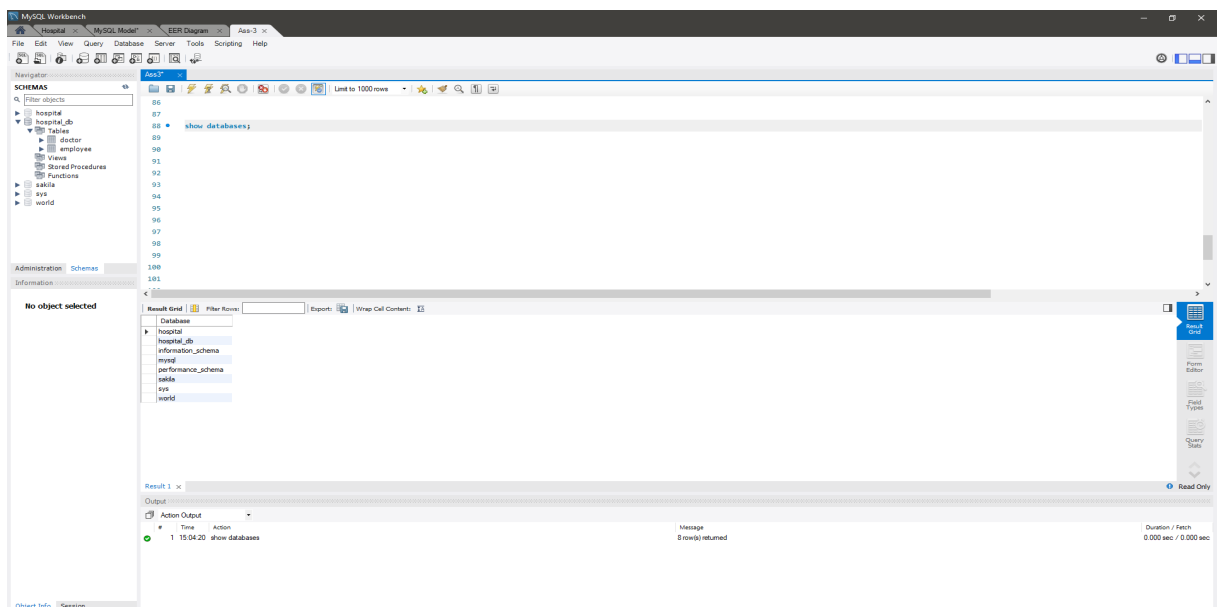
Output:



- Drop Database  
Query : drop database hospital\_db;  
Output:



2. Show all the Databases in the system.  
Query : show databases;  
Output:



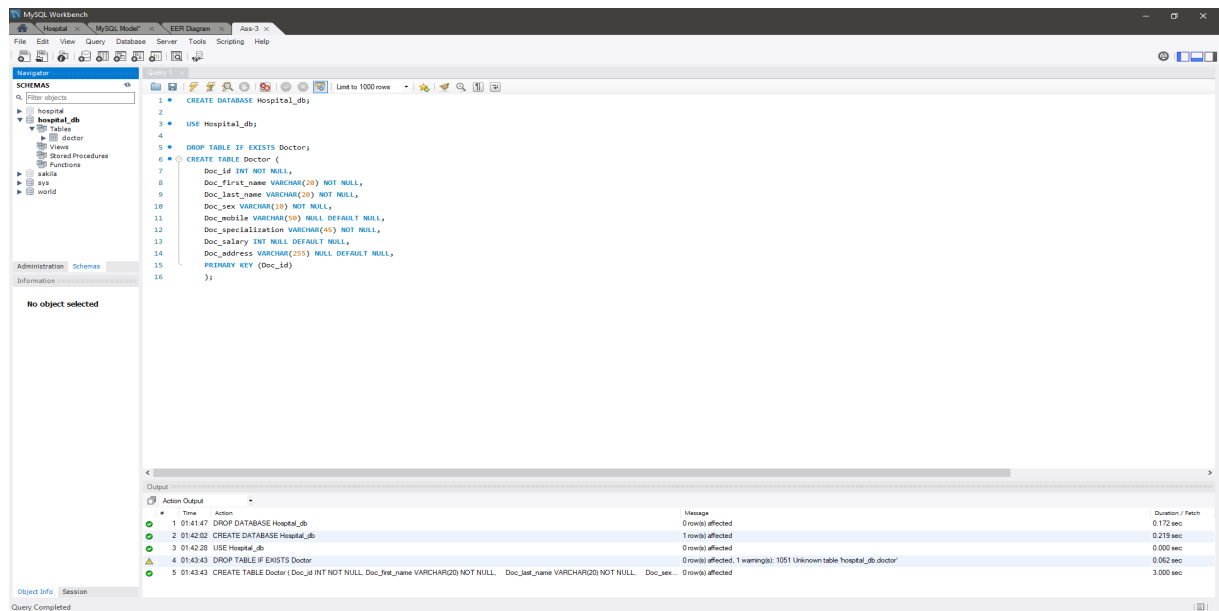
### 3. Create Table for your Database.

Query:

```
CREATE TABLE Doctor (  
  Doc_id INT NOT NULL,  
  Doc_first_name VARCHAR(20) NOT NULL,  
  Doc_last_name VARCHAR(20) NOT NULL,  
  Doc_sex VARCHAR(10) NOT NULL,  
  Doc_mobile VARCHAR(50) NULL DEFAULT NULL,  
  Doc_specialization VARCHAR(45) NOT NULL,  
  Doc_salary INT NULL DEFAULT NULL,  
  Doc_address VARCHAR(255) NULL DEFAULT NULL,  
  PRIMARY KEY (Doc_id)  
);
```

```
CREATE TABLE employee (  
  E_id INT PRIMARY KEY,  
  E_first_name VARCHAR(20) NOT NULL,  
  E_last_name VARCHAR(20) NOT NULL,  
  E_sex VARCHAR(10) NOT NULL,  
  E_mobile VARCHAR(50) NULL DEFAULT NULL,  
  E_salary INT NULL DEFAULT NULL,  
  E_address VARCHAR(255) NULL DEFAULT NULL  
);
```

Output:

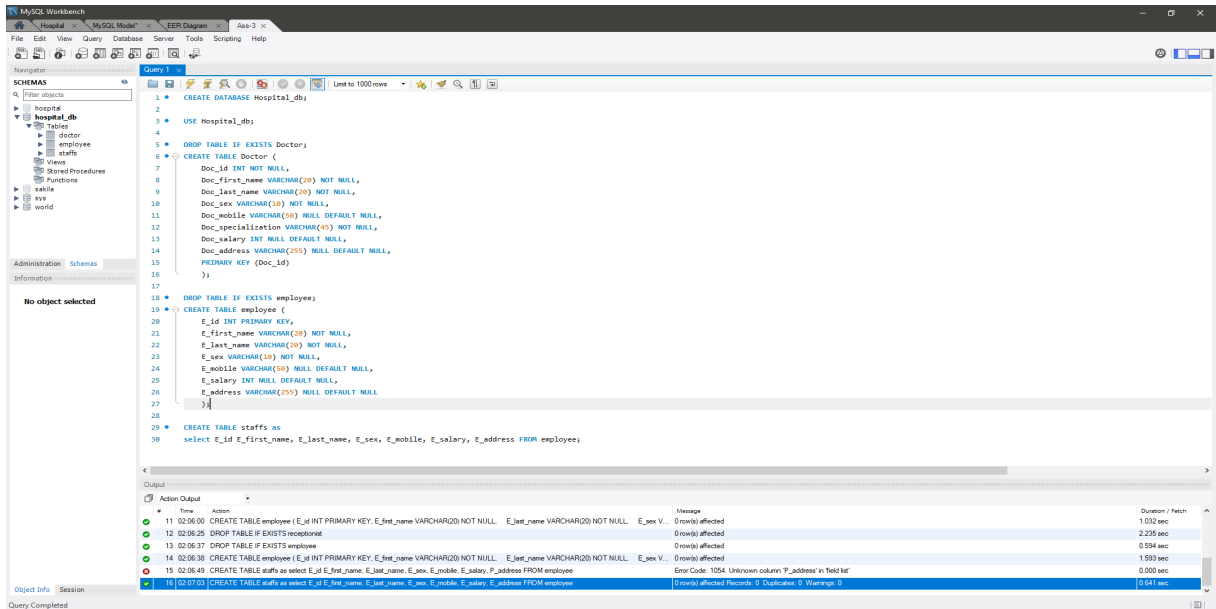


#### 4. Show how select can be used for Creating table

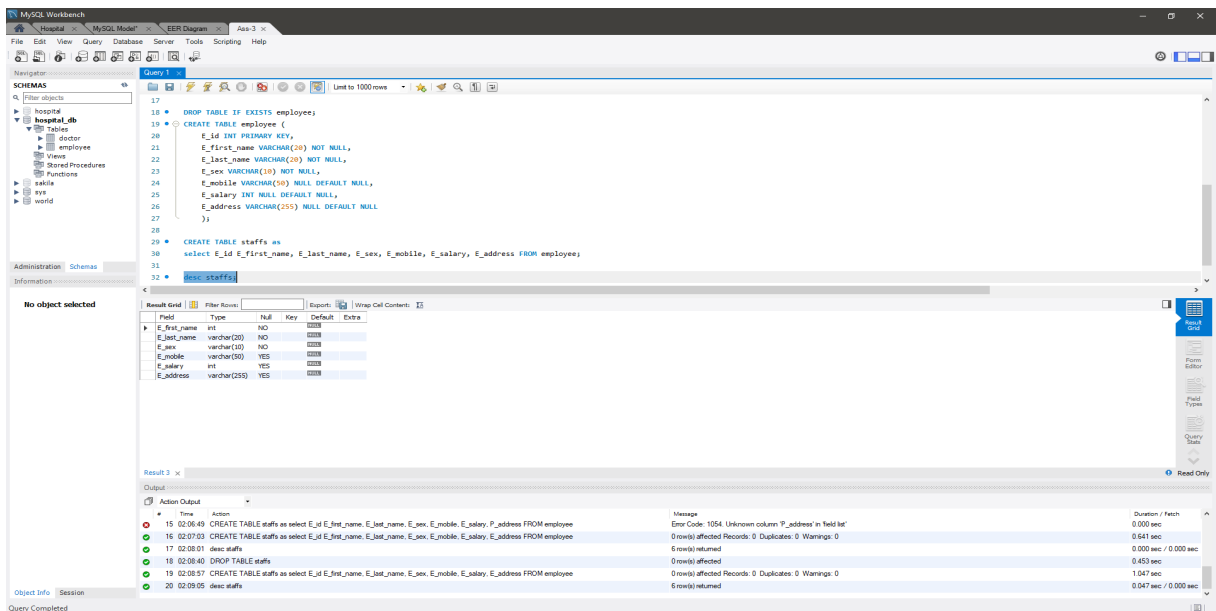
Query: CREATE TABLE staffs as

select E\_id E\_first\_name, E\_last\_name, E\_sex, E\_mobile, E\_salary, E\_address FROM employee;

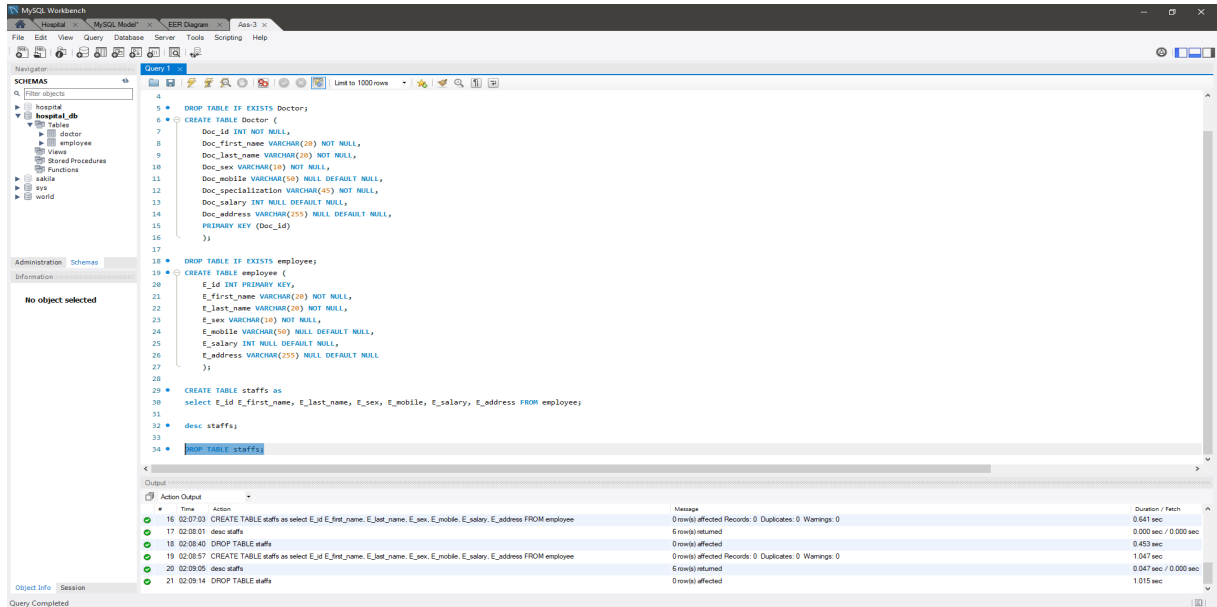
Output:



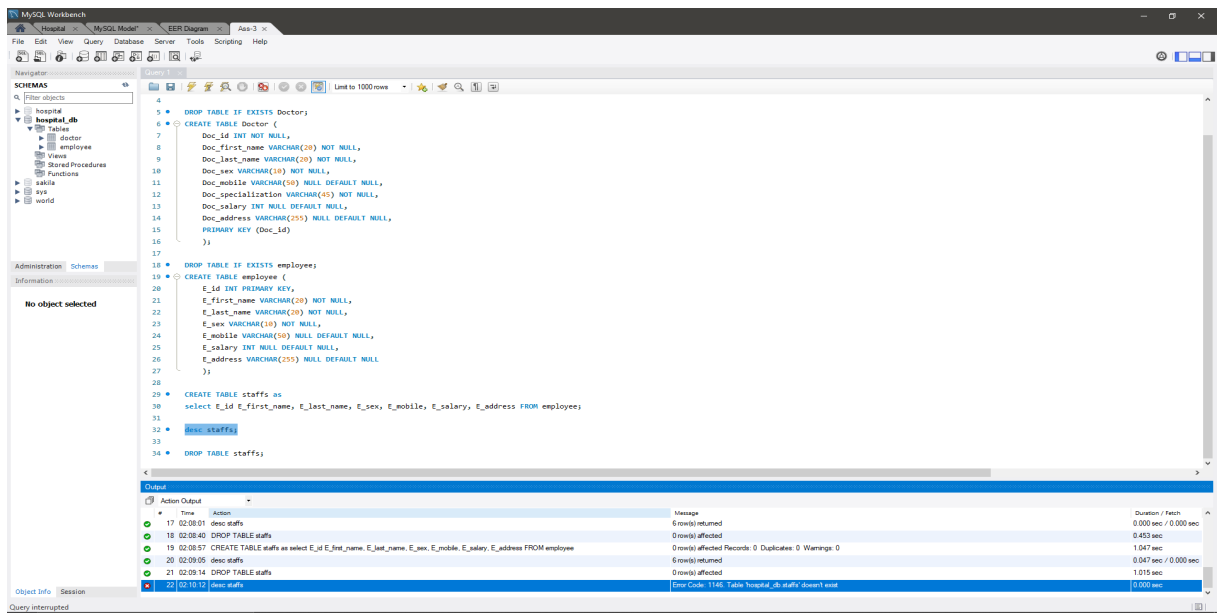
Verification using describe query:



5. Drop table.  
Query : DROP TABLE staffs;  
Output:

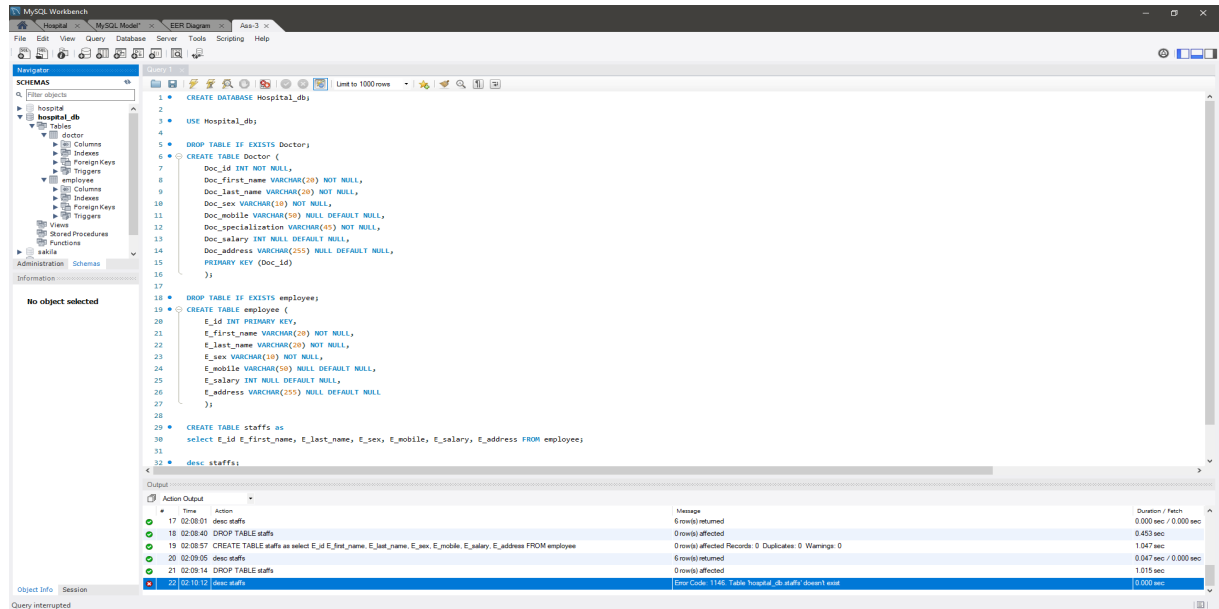


Verification whether the table is dropped or not by accessing the table using describe query :

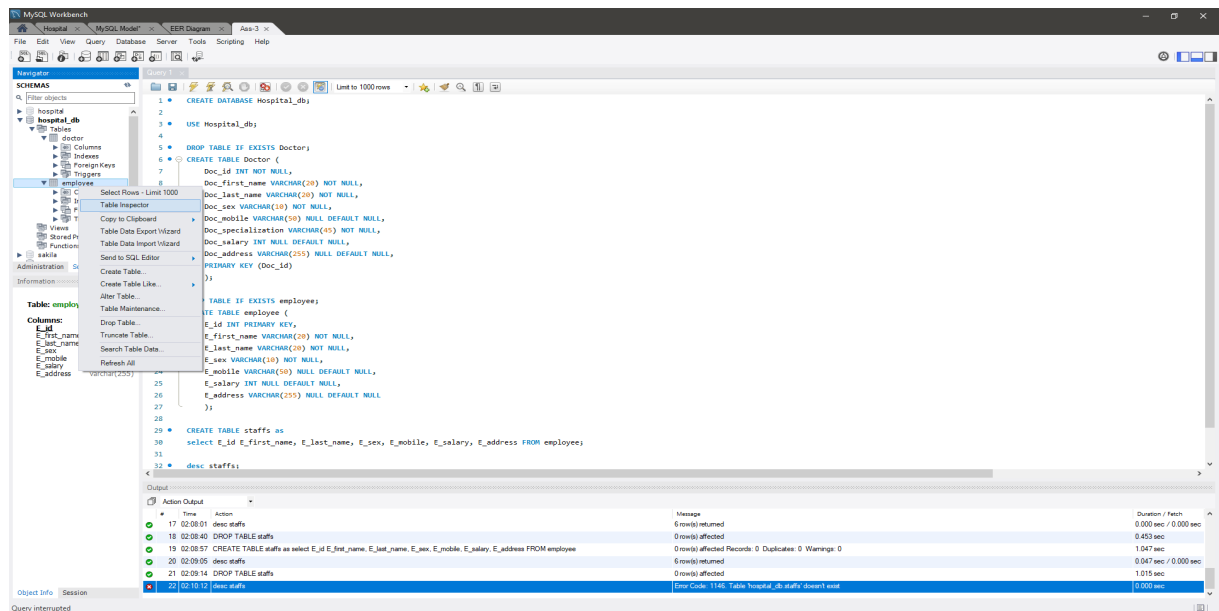


6. Show how to check the schema of the tables.

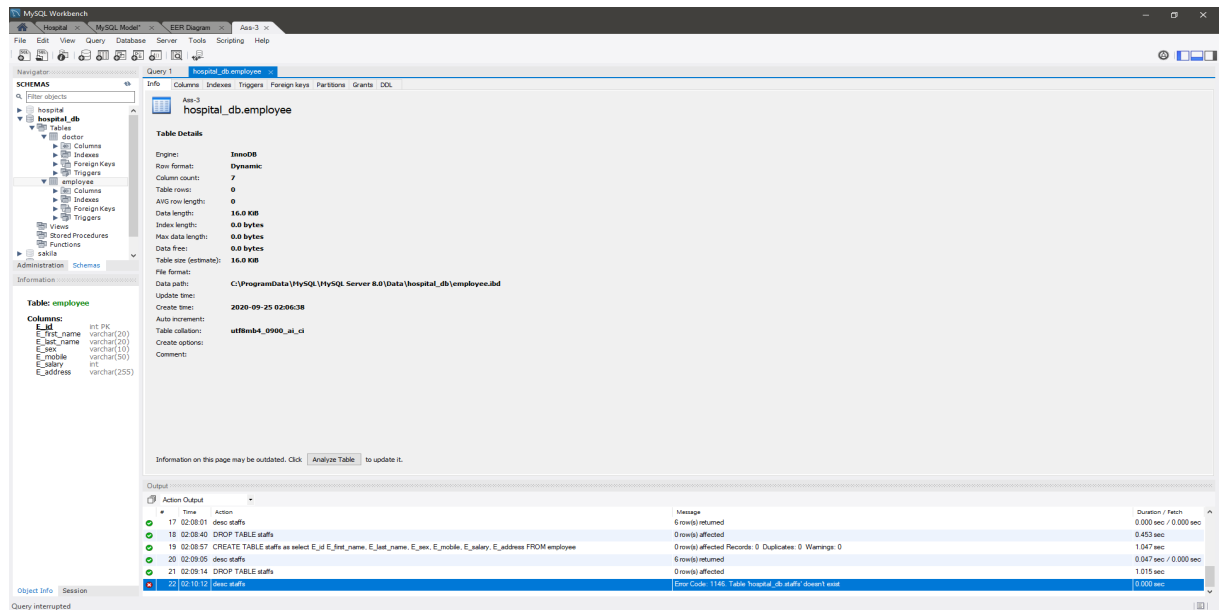
- Hover over the table name in the left side schemas panel of Workbench



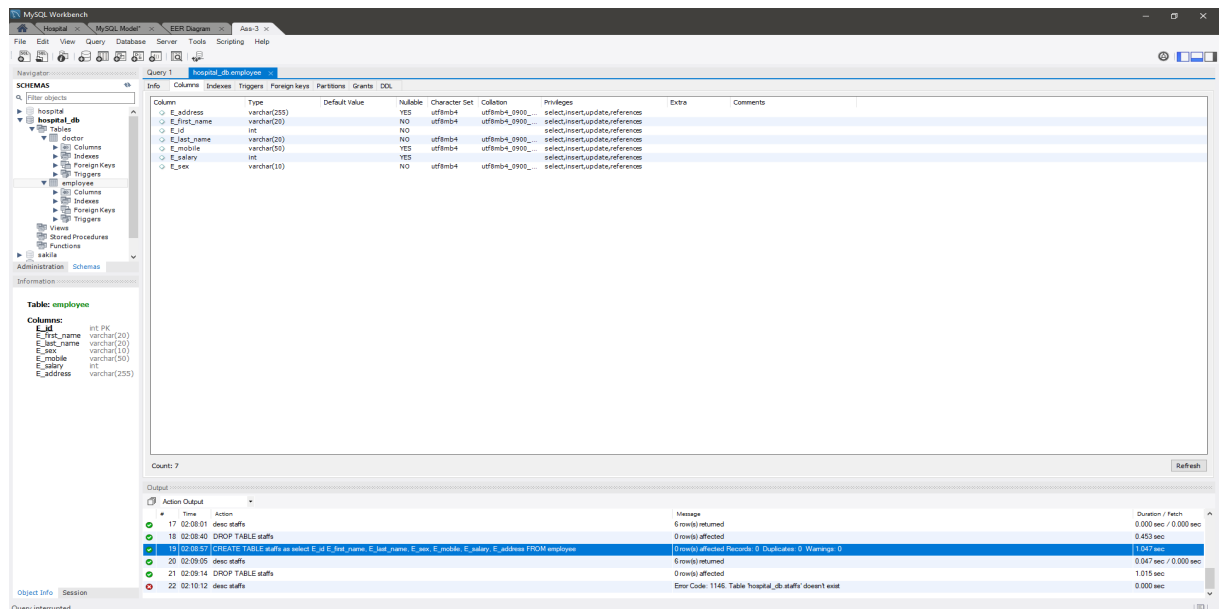
Right click on the table required and click on Table Inspector.



A new tab opens by the name hospital\_db.employee



Now you can click on whatever you want to get information about the table. Like columns, indexes etc.  
Columns:



## Indexes:

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'hospital\_db' database. The 'Indexes in Table' tab is selected for the 'employee' table. The 'Columns in table' section lists the columns and their properties. The 'Index Details' section shows the primary index on 'E\_id'. The 'Output' window at the bottom shows the execution of a query to create a table 'staffs' from the 'employee' table.

Column	Type	Nullable	Indexes
E_id	int	NO	PRIMARY
E_first_name	varchar(20)	NO	
E_last_name	varchar(20)	NO	
E_sex	varchar(10)	NO	
E_mobile	varchar(20)	YES	
E_salary	int	YES	
E_address	varchar(255)	YES	

Time	Action	Message	Duration / Fetch
17 02:08:01	desc staffs	6 rows(s) returned	0.000 sec / 0.000 sec
18 02:08:40	DROP TABLE staffs	6 rows(s) affected	0.453 sec
19 02:08:57	CREATE TABLE staffs as select E_id E_first_name, E_last_name, E_sex, E_mobile, E_salary, E_address FROM employee	6 rows(s) affected Records: 0 Duplicates: 0 Warnings: 0	1.047 sec
20 02:09:05	desc staffs	6 rows(s) returned	0.047 sec / 0.000 sec
21 02:09:14	DROP TABLE staffs	6 rows(s) affected	1.015 sec
22 02:10:12	desc staffs	Error Code: 1146: Table 'hospital_db.staffs' doesn't exist	0.000 sec

You can also get the schema of the database using schema inspector like this :

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left shows the 'hospital\_db' database. The 'Schema Inspector' is open for the 'hospital\_db' database. The 'CREATE TABLE' tab is selected, showing the SQL code for creating the 'employee' table. The 'Output' window at the bottom shows the execution of a query to create a table 'staffs' from the 'employee' table.

```

USE Hospital_db;

DROP TABLE IF EXISTS Doctor;
CREATE TABLE Doctor (
  Doc_id INT NOT NULL,
  Doc_first_name VARCHAR(20) NOT NULL,
  Doc_last_name VARCHAR(20) NOT NULL,
  Doc_sex VARCHAR(10) NOT NULL,
  Doc_mobile VARCHAR(20) NULL DEFAULT NULL,
  Doc_specialization VARCHAR(10) NOT NULL,
  Doc_salary INT NULL DEFAULT NULL,
  Doc_address VARCHAR(255) NULL DEFAULT NULL,
  PRIMARY KEY (Doc_id)
);

DROP TABLE IF EXISTS employee;
CREATE TABLE employee (
  E_id INT PRIMARY KEY,
  E_first_name VARCHAR(20) NOT NULL,
  E_last_name VARCHAR(20) NOT NULL,
  E_sex VARCHAR(10) NOT NULL,
  E_mobile VARCHAR(20) NULL DEFAULT NULL,
  E_salary INT NULL DEFAULT NULL,
  E_address VARCHAR(255) NULL DEFAULT NULL
);

CREATE TABLE staffs as
select E_id E_first_name, E_last_name, E_sex, E_mobile, E_salary, E_address FROM employee;

desc staffs;
  
```

Time	Action	Message	Duration / Fetch
17 02:08:01	desc staffs	6 rows(s) returned	0.000 sec / 0.000 sec
18 02:08:40	DROP TABLE staffs	6 rows(s) affected	0.453 sec
19 02:08:57	CREATE TABLE staffs as select E_id E_first_name, E_last_name, E_sex, E_mobile, E_salary, E_address FROM employee	6 rows(s) affected Records: 0 Duplicates: 0 Warnings: 0	1.047 sec
20 02:09:05	desc staffs	6 rows(s) returned	0.047 sec / 0.000 sec
21 02:09:14	DROP TABLE staffs	6 rows(s) affected	1.015 sec
22 02:10:12	desc staffs	Error Code: 1146: Table 'hospital_db.staffs' doesn't exist	0.000 sec



MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Query 1 hospital\_db.employee hospital\_db

Schemas

Filter objects

hospital\_db

Tables

Columns

Indexes

ForeignKeys

Triggers

Views

Stored Procedures

Functions

Information

Schema: hospital\_db

Name	Engine	Version	Row Format	Rows	Avg Row Length	Data Length	Max Data Length	Index Length	Data Free	Auto Incr...	Create Time	Update Time	Check Time	Collation	Checksum
doctor	InnoDB	10	Dynamic	0	0	16.0 KB	0.0 bytes	0.0 bytes	0.0 bytes	0	2020-09-25 11:43:43			utf8mb4_990...	
employee	InnoDB	10	Dynamic	0	0	16.0 KB	0.0 bytes	0.0 bytes	0.0 bytes	0	2020-09-25 12:16:38			utf8mb4_990...	

Count: 2 Maintenance

Output

Action Output

#	Time	Action	Message	Duration / Fetch
17	02:08:01	desc staffs	6 row(s) returned	0.000 sec / 0.000 sec
18	02:08:40	DROP TABLE staffs	0 row(s) affected	0.403 sec
19	02:08:57	CREATE TABLE staffs as select E_id,E_first_name,E_last_name,E_sec_E_mobile,E_salary,E_address FROM employee	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	1.947 sec
20	02:09:05	desc staffs	6 row(s) returned	0.047 sec / 0.000 sec
21	02:09:14	DROP TABLE staffs	0 row(s) affected	1.015 sec
22	02:10:12	desc staffs	Error Code: 1146. Table 'hospital_db.staffs' doesn't exist	0.000 sec

Query interrupted

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Query 1 hospital\_db.employee hospital\_db

Schemas

Filter objects

hospital\_db

Tables

Columns

Indexes

ForeignKeys

Triggers

Views

Stored Procedures

Functions

Information

Schema: hospital\_db

Table	Column	Type	Default Value	Nullable	Character Set	Collation	Privileges	Extra	Comments
doctor	Doc_address	varchar(255)		YES	utf8mb4	utf8mb4_990...	select,insert,update,references		
doctor	Doc_first_name	varchar(20)		NO	utf8mb4	utf8mb4_990...	select,insert,update,references		
doctor	Doc_id	int		NO	utf8mb4	utf8mb4_990...	select,insert,update,references		
doctor	Doc_last_name	varchar(20)		NO	utf8mb4	utf8mb4_990...	select,insert,update,references		
doctor	Doc_mobile	varchar(50)		YES	utf8mb4	utf8mb4_990...	select,insert,update,references		
doctor	Doc_salary	int		YES	utf8mb4	utf8mb4_990...	select,insert,update,references		
doctor	Doc_specialization	varchar(45)		NO	utf8mb4	utf8mb4_990...	select,insert,update,references		
employee	E_address	varchar(255)		YES	utf8mb4	utf8mb4_990...	select,insert,update,references		
employee	E_first_name	varchar(20)		NO	utf8mb4	utf8mb4_990...	select,insert,update,references		
employee	E_id	int		NO	utf8mb4	utf8mb4_990...	select,insert,update,references		
employee	E_last_name	varchar(20)		NO	utf8mb4	utf8mb4_990...	select,insert,update,references		
employee	E_mobile	varchar(50)		YES	utf8mb4	utf8mb4_990...	select,insert,update,references		
employee	E_salary	int		YES	utf8mb4	utf8mb4_990...	select,insert,update,references		
employee	E_sex	varchar(10)		NO	utf8mb4	utf8mb4_990...	select,insert,update,references		

Count: 15 Refresh

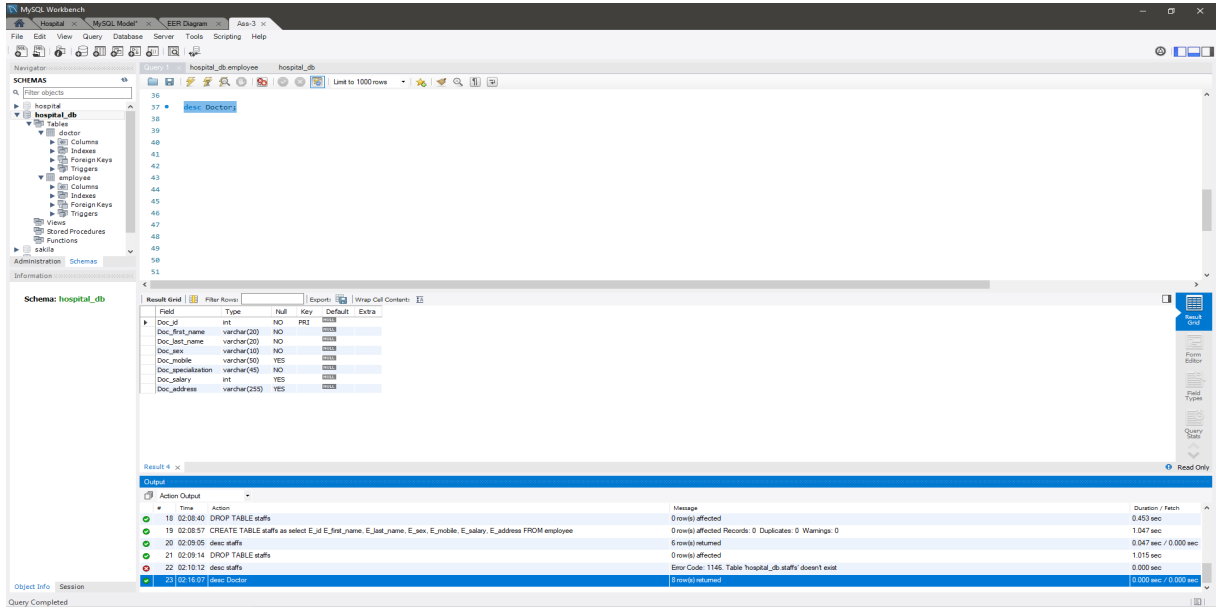
Output

Action Output

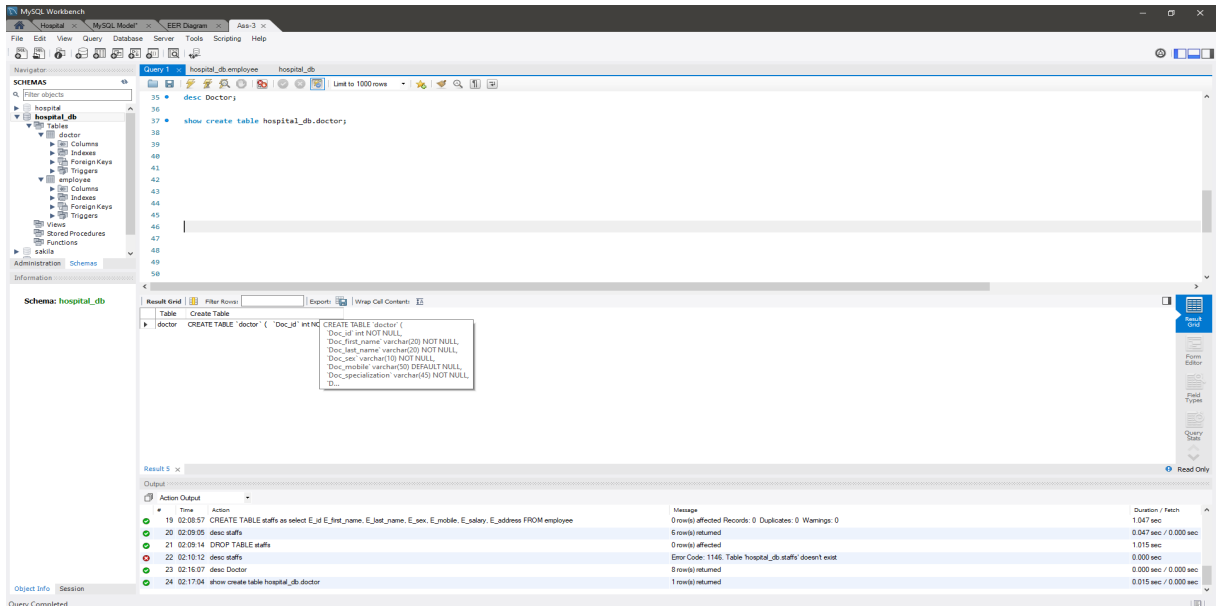
#	Time	Action	Message	Duration / Fetch
17	02:08:01	desc staffs	6 row(s) returned	0.000 sec / 0.000 sec
18	02:08:40	DROP TABLE staffs	0 row(s) affected	0.403 sec
19	02:08:57	CREATE TABLE staffs as select E_id,E_first_name,E_last_name,E_sec_E_mobile,E_salary,E_address FROM employee	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	1.947 sec
20	02:09:05	desc staffs	6 row(s) returned	0.047 sec / 0.000 sec
21	02:09:14	DROP TABLE staffs	0 row(s) affected	1.015 sec
22	02:10:12	desc staffs	Error Code: 1146. Table 'hospital_db.staffs' doesn't exist	0.000 sec

Query interrupted

- Using describe query :  
Query : desc employee;  
Output :



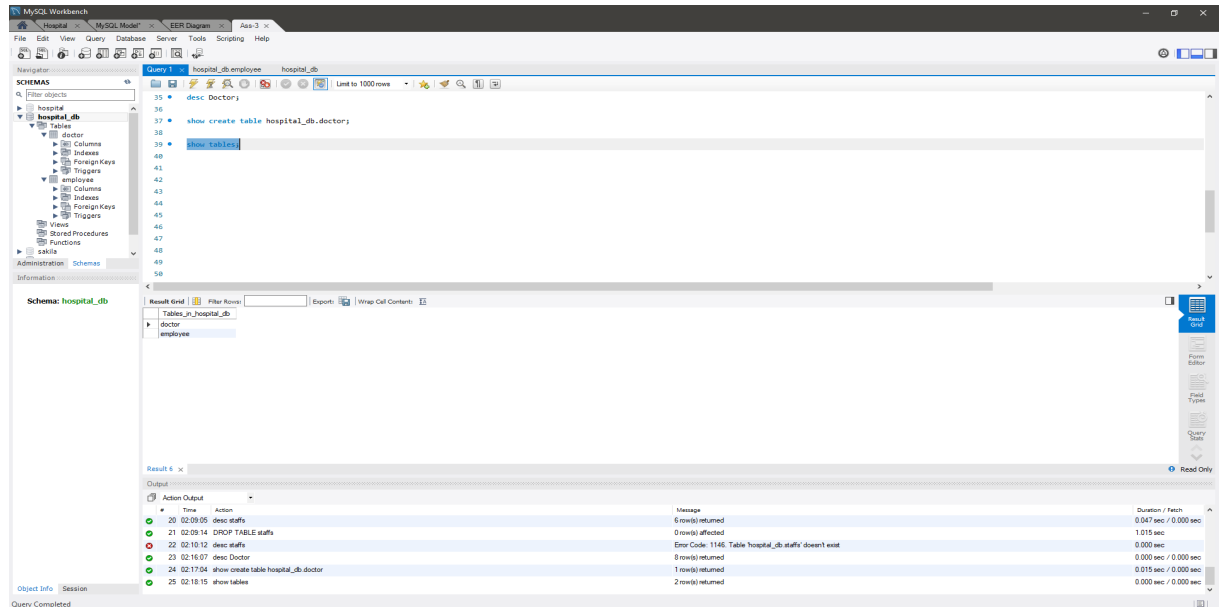
- Using show query :  
Query : show create table hospital\_db.employee;  
Output:



7. Show all the tables from the database.

Query : show tables;

Output:

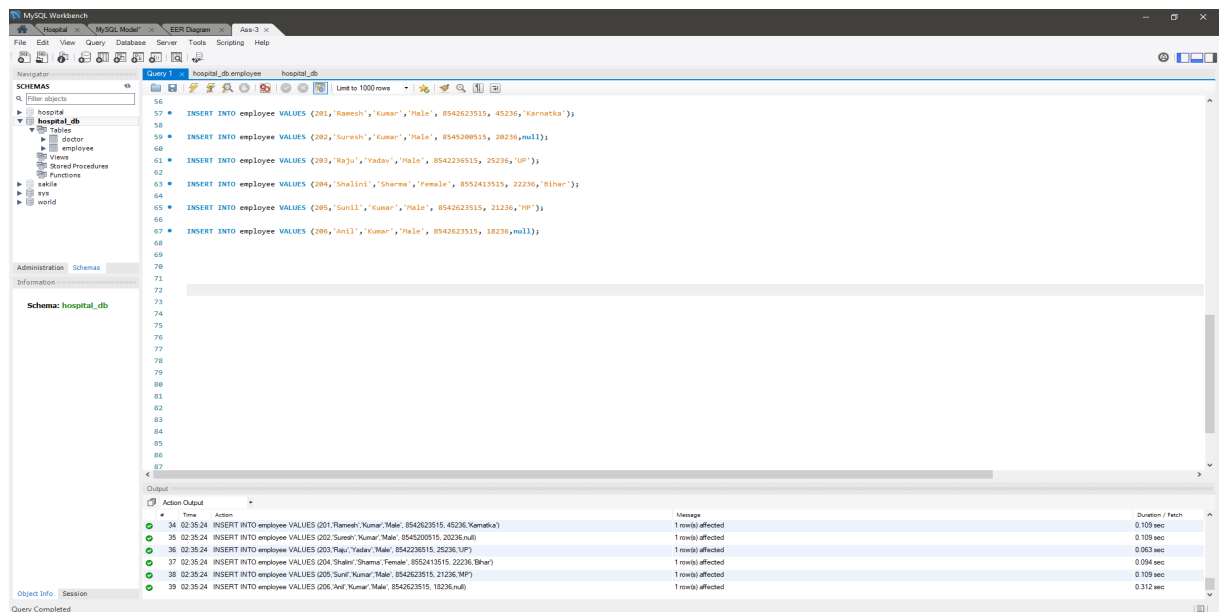


8. Insert 5 to 10 rows in each of the tables of your Database.

Inserting data into employee table :

Query : INSERT INTO employee VALUES (201,'Ramesh','Kumar','Male', 8542623515, 45236,'Karnataka');

Output:



Verifying whether data has been inserted properly or not by fetching the data using select query :

The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying a series of INSERT statements into the 'employee' table. The 'Result Grid' shows the output of the last query, which is a SELECT statement. The grid contains 10 rows of data, including columns for ID, first name, last name, sex, mobile, salary, and address.

ID	E_first_name	E_last_name	E_sex	E_mobile	E_salary	E_address
201	Ramesh	Kumar	Male	8542623515	45236	Karnataka
202	Suresh	Kumar	Male	8542620515	20236	UP
203	Raju	Yadav	Male	8542236515	25236	UP
204	Shah	Sharma	Female	8552413515	22236	Bihar
205	Sunil	Kumar	Male	8542623515	21236	MP
206	Ash	Kumar	Male	8542623515	18236	UP

The 'Action Output' pane at the bottom shows the execution of the queries, with messages indicating that 1 row was affected for each INSERT statement.

Query : INSERT INTO Doctor VALUES (101, 'Ajit','Sharma','Male', 7788498482, 'Cardiologist', 542135,null);  
Output:

The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying a series of INSERT statements into the 'doctor' table. The 'Result Grid' shows the output of the last query, which is a SELECT statement. The grid contains 10 rows of data, including columns for ID, first name, last name, sex, mobile, salary, and address.

ID	D_first_name	D_last_name	D_sex	D_mobile	D_salary	D_address
101	Ajit	Sharma	Male	7788498482	542135	Delhi
102	Aman	Sharma	Male	7788802482	542235	Delhi
103	Tanviya	Hishra	Female	9924549848	508135	UP
104	Karan	Sharma	Male	9856324569	402835	Chennai
105	Sanjay	Anand	Male	9952898482	602135	Chennai
106	Nehani	Kashri	Female	9966325420	602135	Mumbai
107	Ramesh	Ash	Male	7985248482	280135	UP

The 'Action Output' pane at the bottom shows the execution of the queries, with messages indicating that 1 row was affected for each INSERT statement.

## 9. Show usage of Simple Select Statement:

Query1 : select \* from Doctor;

Output:

**Query 1**

```
SELECT * FROM Doctor;
```

**Schema: hospital\_db**

Doc_id	Doc_first_name	Doc_last_name	Doc_sex	Doc_mobile	Doc_specialization	Doc_salary	Doc_address
101	Ajit	Sharma	Male	7786048462	Cardiologist	141235	Delhi
102	Akhan	Sharma	Male	7786024642	Cardiologist	141235	Delhi
103	Tanya	Mahra	Female	9924548468	Audiologist	200135	UP
104	Karan	Sharma	Male	9952015069	Audiologist	402035	Chennai
105	Sanjay	Anand	Male	9952084842	Cardiologist	602135	Chennai
106	Mohan	Keshri	Female	9963254040	Cardiologist	602135	Mumbai
107	Ramesh	Athre	Male	7985248462	Audiologist	200135	Delhi

**Action Output**

Time	Action	Message	Duration / Fetch
28 02:27:06	INSERT INTO Doctor VALUES (103, 'Tanya','Mahra','Female', '9924548468', 'Audiologist', '200135','UP')	1 row(s) affected	0.196 sec
29 02:27:06	INSERT INTO Doctor VALUES (104, 'Karan','Sharma','Male', '9952015069', 'Audiologist', '402035','Chennai')	1 row(s) affected	0.172 sec
30 02:27:06	INSERT INTO Doctor VALUES (105, 'Sanjay','Anand','Male', '9952084842', 'Cardiologist', '602135','Chennai')	1 row(s) affected	0.109 sec
31 02:27:06	INSERT INTO Doctor VALUES (106, 'Mohan','Keshri','Female', '9963254040', 'Cardiologist', '602135','Mumbai')	1 row(s) affected	0.094 sec
32 02:27:06	INSERT INTO Doctor VALUES (107, 'Ramesh','Athre','Male', '7985248462', 'Audiologist', '200135','Delhi')	1 row(s) affected	0.078 sec
33 02:28:08	SELECT * FROM Doctor LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec

Query 2 : SELECT E.salary from employee;

Output:

**Query 2**

```
SELECT E.salary from employee;
```

**Schema: hospital\_db**

E_salary
141235
200236
252236
222236
212236
182236

**Action Output**

Time	Action	Message	Duration / Fetch
36 02:35:24	INSERT INTO employee VALUES (203,'Raj','Yadav','Male', '854223515', '25236','UP')	1 row(s) affected	0.063 sec
37 02:35:24	INSERT INTO employee VALUES (204,'Sham','Sharma','Female', '8552413515', '22236','UP')	1 row(s) affected	0.084 sec
38 02:35:24	INSERT INTO employee VALUES (205,'Suri','Kumar','Male', '8542623515', '21236','MP')	1 row(s) affected	0.109 sec
39 02:35:24	INSERT INTO employee VALUES (206,'Anil','Kumar','Male', '8542623515', '18236','MH')	1 row(s) affected	0.312 sec
40 02:35:51	SELECT * FROM employee LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
41 02:36:56	SELECT E_salary from employee LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

## 10. Select Statement using Relational and Logical operators.

- Using Relational Operators :

(a) “>” operator:

Query : SELECT \* from Doctor WHERE Doc\_salary > 400000;

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left displays the 'hospital\_db' database with tables 'doctor', 'employee', 'patient', 'visit', 'xray', and 'world'. The 'Query' editor contains the following SQL script:

```

61. INSERT INTO employee VALUES (203,'Raju','Yadav','Male', 8542236515, 25236,'LP');
62.
63. INSERT INTO employee VALUES (204,'Shalini','Sharma','Female', 8552413515, 22236,'Bihar');
64.
65. INSERT INTO employee VALUES (205,'Sunil','Kumar','Male', 8542623515, 21236,'HP');
66.
67. INSERT INTO employee VALUES (206,'Anil','Kumar','Male', 8542623515, 18236,null);
68.
69. SELECT * from employee;
70.
71. SELECT E_salary from employee;
72.
73. SELECT * from Doctor WHERE Doc_salary > 400000;
74.
75.
76.

```

The 'Result Grid' shows the output of the last query, displaying columns: Doc\_id, Doc\_first\_name, Doc\_last\_name, Doc\_sex, Doc\_mobile, Doc\_specialization, Doc\_salary, and Doc\_address. The results are as follows:

Doc_id	Doc_first_name	Doc_last_name	Doc_sex	Doc_mobile	Doc_specialization	Doc_salary	Doc_address
201	Ami	Sharma	Male	7788888842	Cardiologist	342135	0000
202	Anam	Sha	Male	7798002462	Cardiologist	342235	Delhi
203	Tanya	Mahra	Female	9924540848	Audiologist	500135	LP
204	Karan	Sharma	Male	8850252569	Audiologist	400235	0000
205	Seraj	Anand	Male	995209482	Cardiologist	602135	Chennai
206	Mahesh	Kapri	Female	9966225420	Cardiologist	602135	Hydrabad

The 'Output' pane shows the execution log with the following messages:

```

39 02:35:24 INSERT INTO employee VALUES (206,'Anil','Kumar','Male', 8542623515, 18236,null)
40 02:35:51 SELECT *from employee LIMIT 0, 1000
41 02:35:56 SELECT E_salary from employee LIMIT 0, 1000
42 02:38:17 SELECT *from doctor WHERE Doc_salary > 400000 LIMIT 0, 1000
43 02:38:44 SELECT *from doctor WHERE Doc_salary > 400000
44 02:38:58 SELECT *from Doctor WHERE Doc_salary > 400000 LIMIT 0, 1000

```

(b) “<” operator:

Query : SELECT \* from employee WHERE E\_id < 204;

The screenshot shows the MySQL Workbench interface. The 'Schemas' pane on the left displays the 'hospital\_db' database with tables 'doctor', 'employee', 'patient', 'visit', 'xray', and 'world'. The 'Query' editor contains the following SQL script:

```

61. INSERT INTO employee VALUES (203,'Raju','Yadav','Male', 8542236515, 25236,'LP');
62.
63. INSERT INTO employee VALUES (204,'Shalini','Sharma','Female', 8552413515, 22236,'Bihar');
64.
65. INSERT INTO employee VALUES (205,'Sunil','Kumar','Male', 8542623515, 21236,'HP');
66.
67. INSERT INTO employee VALUES (206,'Anil','Kumar','Male', 8542623515, 18236,null);
68.
69. SELECT * from employee;
70.
71. SELECT E_salary from employee;
72.
73. SELECT * from Doctor WHERE Doc_salary > 400000;
74.
75. SELECT * from employee WHERE E_id < 204;
76.

```

The 'Result Grid' shows the output of the last query, displaying columns: E\_id, E\_first\_name, E\_last\_name, E\_sex, E\_mobile, E\_salary, and E\_address. The results are as follows:

E_id	E_first_name	E_last_name	E_sex	E_mobile	E_salary	E_address
201	Ramash	Kumar	Male	8542623515	40236	Karnataka
202	Suresh	Kumar	Male	8542020515	20236	0000
203	Raju	Yadav	Male	8542236515	25236	LP

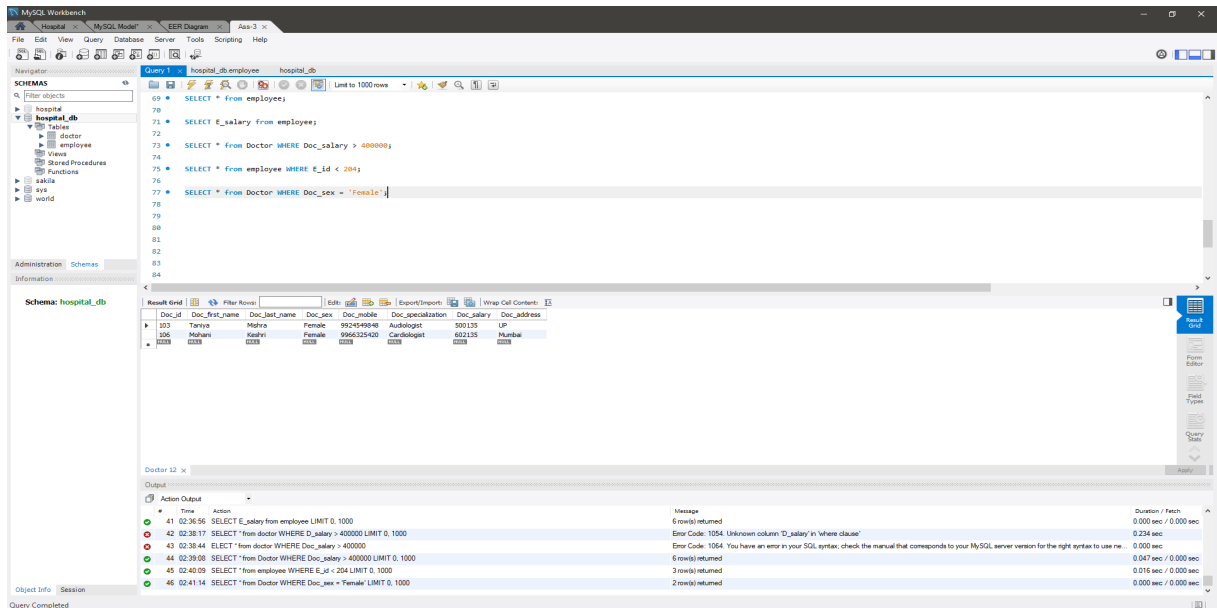
The 'Output' pane shows the execution log with the following messages:

```

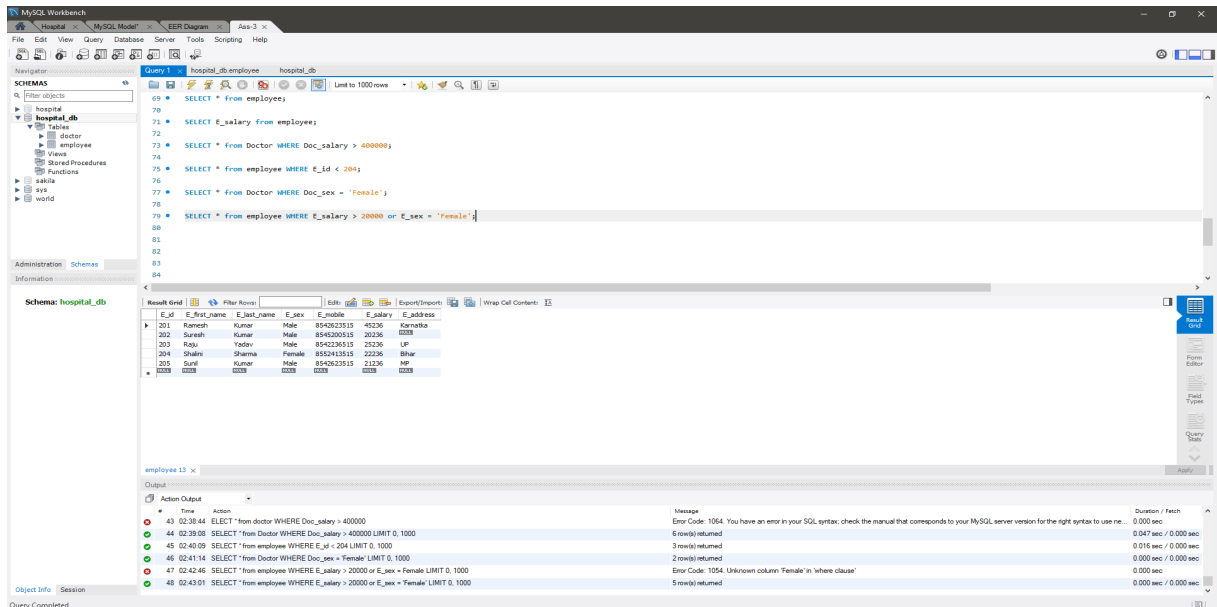
40 02:35:51 SELECT *from employee LIMIT 0, 1000
41 02:35:56 SELECT E_salary from employee LIMIT 0, 1000
42 02:38:17 SELECT *from doctor WHERE Doc_salary > 400000 LIMIT 0, 1000
43 02:38:44 SELECT *from doctor WHERE Doc_salary > 400000
44 02:39:08 SELECT *from Doctor WHERE Doc_salary > 400000 LIMIT 0, 1000
45 02:40:09 SELECT *from employee WHERE E_id < 204 LIMIT 0, 1000

```

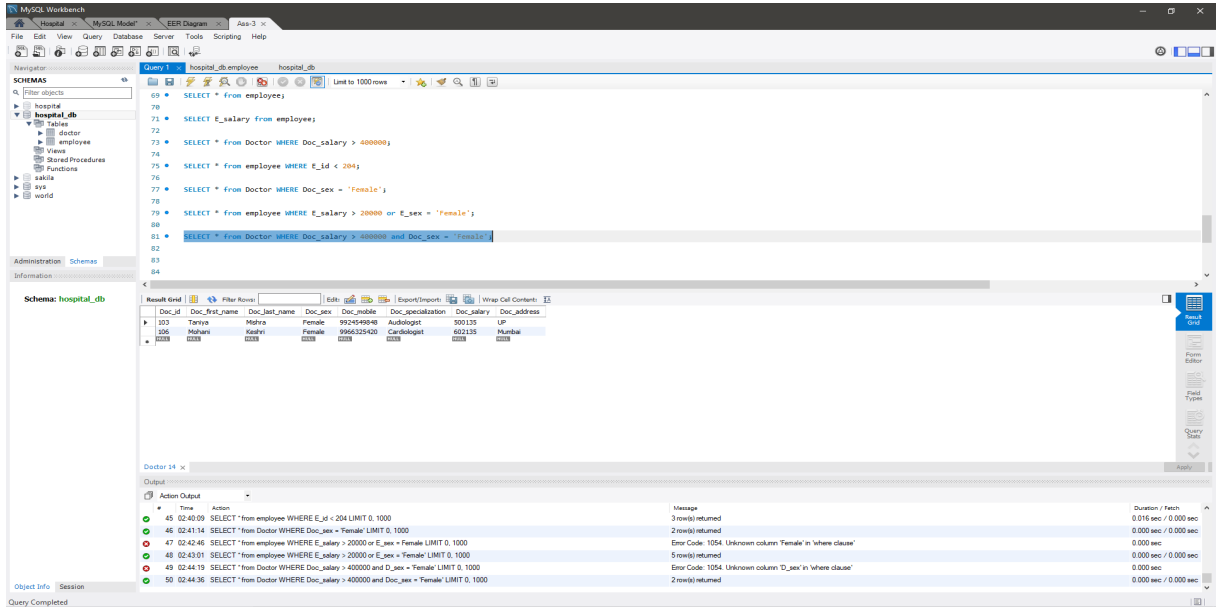
- (c) “=” operator:  
Query : SELECT \* from Doctor WHERE Doc\_sex = 'Female';



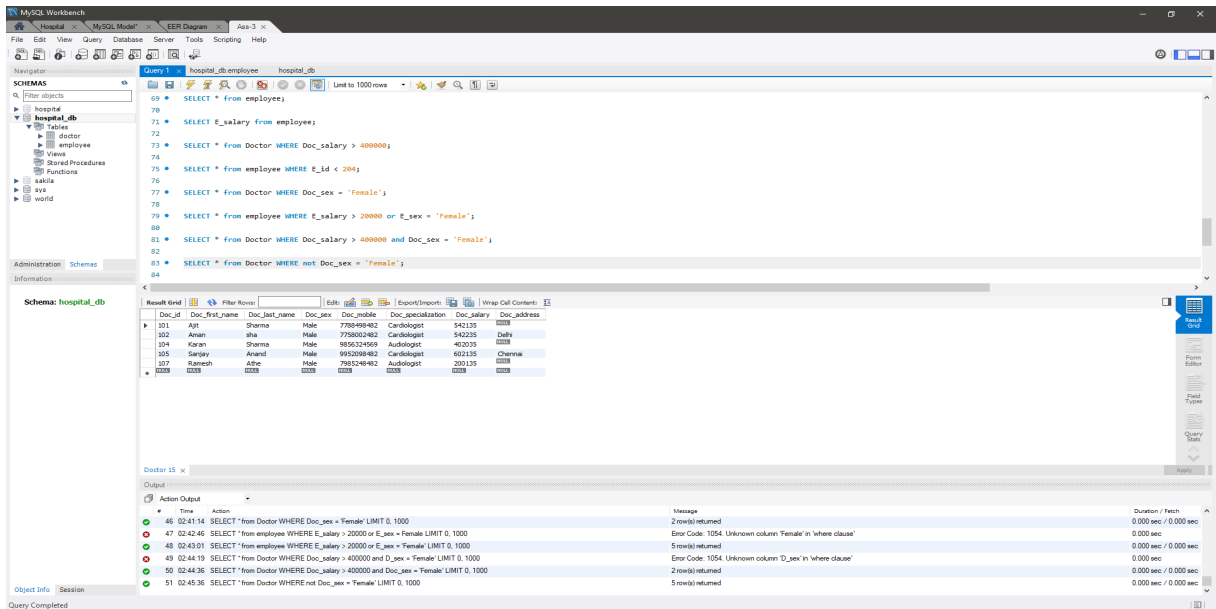
- (d) “OR” operator:  
Query : SELECT \* from employee WHERE E\_salary > 20000 or E\_sex = 'Female';



- (e) “AND” operator:  
Query : SELECT \* from Doctor WHERE Doc\_salary > 400000 and Doc\_sex = 'Female';



- (f) “NOT” operator:  
Query : SELECT \* from Doctor WHERE not Doc\_sex = 'Female';





11. One simple Subquery using select.

Query : select Doc.first\_name, Doc.salary from Doctor where Doc.salary > (select Doc.salary from Doctor where Doc\_id = 103);

The screenshot shows the MySQL Workbench interface with a query editor and a results pane. The query editor contains the following SQL code:

```
76
77 SELECT * from Doctor WHERE Doc_sex = 'Female';
78
79 SELECT * from employee WHERE E_salary > 20000 or E_sex = 'Female';
80
81 SELECT * from Doctor WHERE Doc_salary > 400000 and Doc_sex = 'Female';
82
83 SELECT * from Doctor WHERE not Doc_sex = 'Female';
84
85 select Doc.first_name, Doc.salary from Doctor where Doc.salary > (select Doc.salary from Doctor where Doc_id = 103);
86
87
88
89
90
91
```

The results pane shows the output of the query, which is a table with two columns: Doc\_first\_name and Doc\_salary. The results are as follows:

Doc_first_name	Doc_salary
Ajit	542135
Ansh	542135
Sangay	602135
Mohani	602135

The output pane shows the execution of the query, including the time taken and the number of rows returned. The query was executed successfully, returning 4 rows.