

Group-7
DBMS ASSIGNMENT-3

1. SHOW HOW TO CREATE AND DROP DATABASE.

RESULT AFTER CREATING:

```
6      -- creates database
7  •    CREATE DATABASE IF NOT EXISTS Companydb1;
8  •    SHOW DATABASES;
```


< | Result Grid | Filter Rows: | Export: | Wrap C

	Database
▶	companydb
	companydb1
	information_schema
	mysql
	performance_schema
	sakila
	stud
	sys
	world

RESULT AFTER DROPPING:

```
6 -- deletes database
7 • DROP DATABASE companydb1;
8 • SHOW DATABASES;
```

<


Result Grid |  Filter Rows: | Ex

	Database
▶	companydb
	information_schema
	mysql
	performance_schema
	sakila
	stud
	sys
	world

2. SHOW ALL THE DATABASES ARE IN THE SYSTEM.

```
54 • SHOW DATABASES;
```

<

Result Grid |  Filter Rows:

	Database
▶	companydb
	companydb1
	information_schema
	mysql
	performance_schema
	sakila
	stud
	sys
	world

3. CREATE TABLE FOR YOUR DATABASE.


```
3 • CREATE TABLE Employee(  
4     Employee_id SMALLINT NOT NULL PRIMARY KEY,  
5     `First Name` VARCHAR(45),  
6     `Last Name` VARCHAR(45),  
7     DoB DATE,  
8     Salary INT  
9 );  
10 • DESCRIBE Employee;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	Field	Type	Null	Key	Default	Extra
▶	Employee_id	smallint	NO	PRI	NULL	
	First Name	varchar(45)	YES		NULL	
	Last Name	varchar(45)	YES		NULL	
	DoB	date	YES		NULL	
	Salary	int	YES		NULL	

4. SHOW HOW SELECT CAN BE USED FOR CREATING TABLE.

```
10 • CREATE TABLE Manager AS( SELECT * FROM Employee);
11 • DESCRIBE Manager;
12
```

Result Grid					
Filter Rows:		Export:  Wrap Cell Content			
Field	Type	Null	Key	Default	Extra
Employee_id	smallint	NO		NULL	
First Name	varchar(45)	YES		NULL	
Last Name	varchar(45)	YES		NULL	
DoB	date	YES		NULL	
Salary	int	YES		NULL	

5. DROP TABLE.

```
11 • DROP TABLE Manager;
12 • DESCRIBE Manager;
```

34 18:05:53 DESCRIBE Manager Error Code: 1146. Table 'companydb1.manager' doesn't exist

6. SHOW HOW TO CHECK THE SCHEMA OF THE TABLES.

18 • DESCRIBE Employee;

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	Field	Type	Null	Key	Default	Extra
▶	Employee_id	smallint	NO	PRI	NULL	
	First Name	varchar(45)	YES		NULL	
	Last Name	varchar(45)	YES		NULL	
	DoB	date	YES		NULL	
	Salary	int	YES		NULL	
	Branch_id	smallint	YES	MUL	NULL	

7. SHOW ALL THE TABLES FROM THE DATABASE.

22 • SHOW TABLES;

<




Result Grid | Filter Rows: |

	Tables_in_companydb1
▶	branch
	employee

8.INSERT 5-10 ROWS IN EACH OF THE TABLES IN YOUR DATABASE.

```
23 • INSERT INTO Branch VALUES
24     (01, 'Scranton', 'Pennsylvania'),
25     (02, 'Corporate', 'New York'),
26     (03, 'Nashua', 'New Hampshire'),
27     (04, 'Stamford', 'Connecticut'),
28     (05, 'Utica', 'New York');
29
30 • SELECT * FROM Branch;
```

<






Result Grid   Filter Rows: Edit: 

	Branch_id	Branch Name	Location
▶	1	Scranton	Pennsylvania
	2	Corporate	New York
	3	Nashua	New Hampshire
	4	Stamford	Connecticut
	5	Utica	New York
*	NULL	NULL	NULL

```

29 • INSERT INTO Employee VALUES
30     (101, 'Michael', 'Scott', '1970-03-15', 150500, 01),
31     (102, 'Jan', 'Levinson', '1971-12-10', 250000, 02),
32     (103, 'Andy', 'Bernard', '1975-09-20', 90000, 04),
33     (104, 'Karen', 'Filipelli', '1980-08-19', 95000, 05),
34     (105, 'Jim', 'Halpert', '1978-10-01', 90000, 01);
35 • SELECT * FROM Employee;

```



Result Grid						
Filter Rows: <input type="text"/>						
Edit:   						
Export/Import:  						
	Employee_id	First Name	Last Name	DoB	Salary	Branch_id
▶	101	Michael	Scott	1970-03-15	150500	1
	102	Jan	Levinson	1971-12-10	250000	2
	103	Andy	Bernard	1975-09-20	90000	4
	104	Karen	Filipelli	1980-08-19	95000	5
	105	Jim	Halpert	1978-10-01	90000	1
✱	NULL	NULL	NULL	NULL	NULL	NULL

9. SHOW USAGE OF SIMPLE SELECT STATEMENT.

```

36 • SELECT `First Name`, Salary FROM Employee;

```

Result Grid		
Filter Rows: <input type="text"/>		
Export:  		
	First Name	Salary
▶	Michael	150500
	Jan	250000
	Andy	90000
	Karen	95000
	Jim	90000

10. SELECT STATEMENT USING RELATIONAL AND LOGICAL OPERATORS.

```
37 • SELECT * FROM Employee WHERE Employee.Salary<100000;
```

Employee_id	First Name	Last Name	DoB	Salary	Branch_id
103	Andy	Bernard	1975-09-20	90000	4
104	Karen	Filipelli	1980-08-19	95000	5
105	Jim	Halpert	1978-10-01	90000	1
NULL	NULL	NULL	NULL	NULL	NULL

```
38 • SELECT * FROM Branch WHERE Location='New York' OR Location='Pennsylvania';
```

Branch_id	Branch Name	Location
1	Scranton	Pennsylvania
2	Corporate	New York
5	Utica	New York
NULL	NULL	NULL

11. ONE SIMPLE SUBQUERY USING SELECT.

```
39 -- Gives names of employees working in New York
40 • SELECT `First Name`
41 FROM Employee
42 WHERE Branch_id IN (SELECT Branch_id FROM Branch WHERE Location='New York');
```

First Name
Jan
Karen