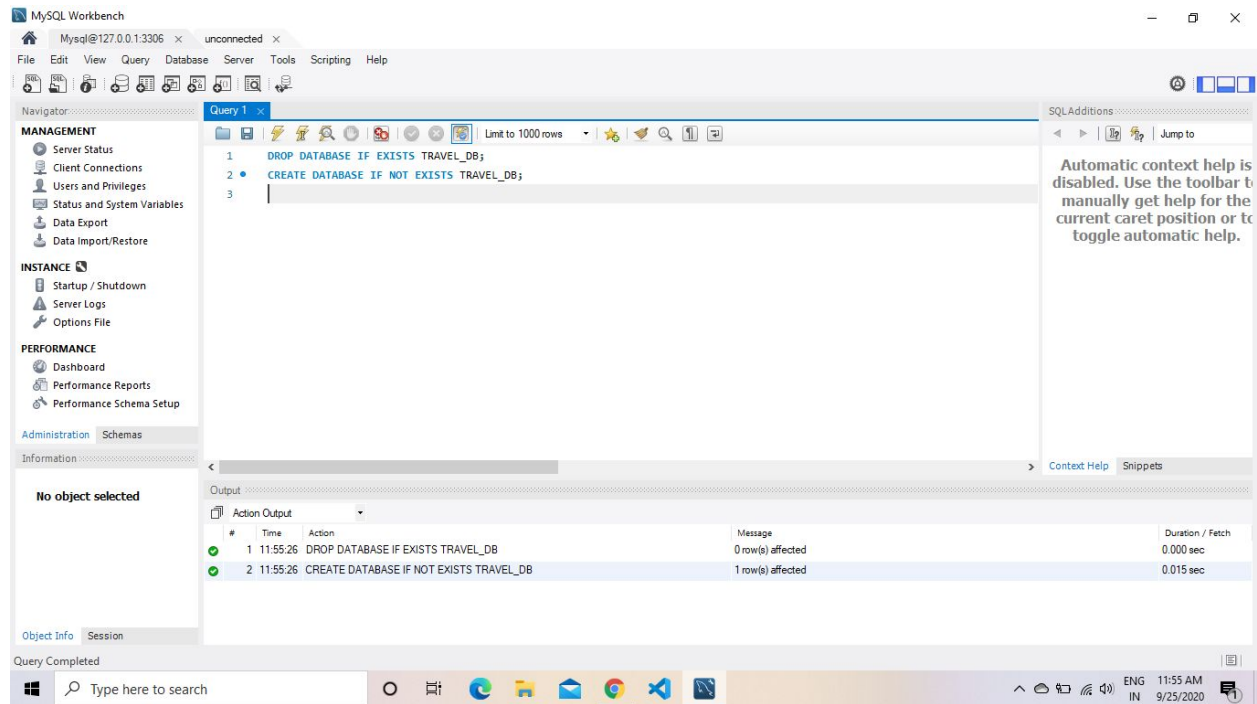


# ASSIGNMENT 3

## 1. SHOW HOW TO CREATE AND DROP DATABASE

QUERY -> **DROP DATABASE IF EXISTS Travel;**  
**CREATE DATABASE IF NOT EXISTS Travel;**



## 2.SHOW ALL THE DATABASES IN THE SYSTEM

QUERY-> **SHOW DATABASES;**

The screenshot shows the MySQL Workbench interface. The left sidebar contains the 'MANAGEMENT' and 'INSTANCE' sections. The central query editor displays the following SQL script:

```
1 DROP DATABASE IF EXISTS TRAVEL_DB;
2 CREATE DATABASE IF NOT EXISTS TRAVEL_DB;
3
4 SHOW databases;
5
```

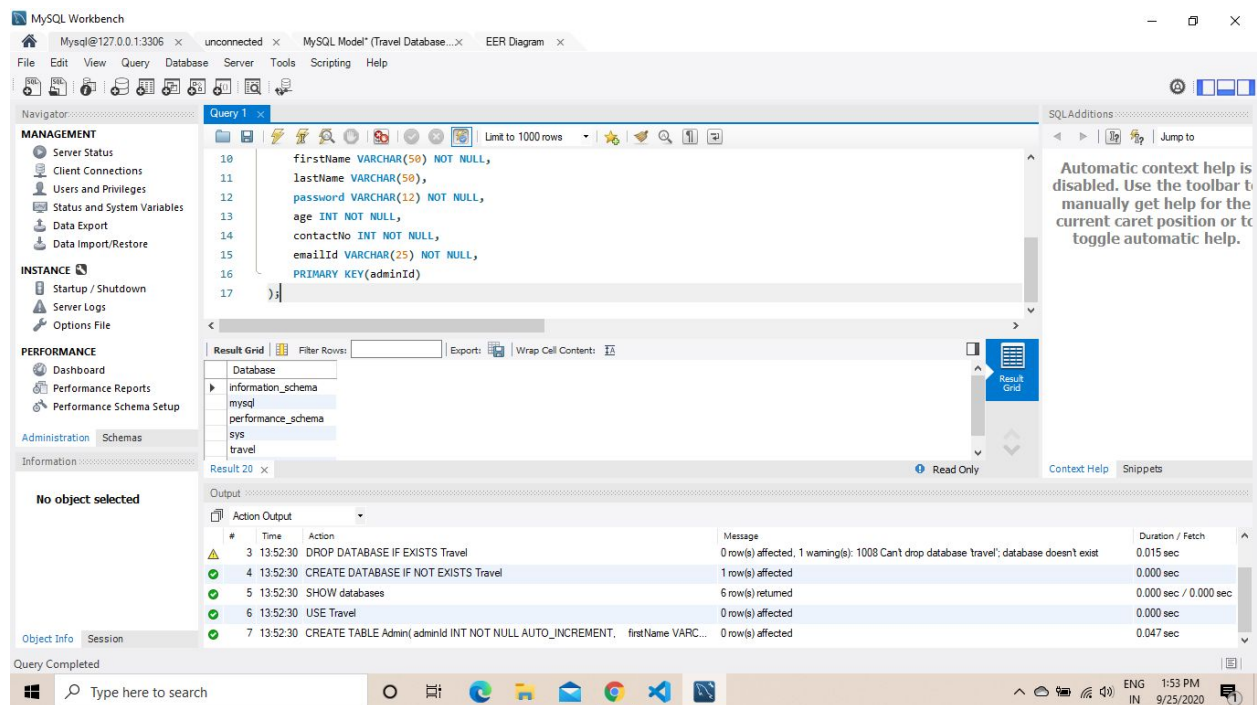
The 'Result Grid' is visible, showing the output of the queries. The 'Output' tab is selected, displaying the following results:

#	Time	Action	Message	Duration / Fetch
1	11:54:25	DROP DATABASE IF EXISTS TRAVEL_DB	5 row(s) affected	0.125 sec
2	11:54:25	CREATE DATABASE IF NOT EXISTS TRAVEL_DB	1 row(s) affected	0.000 sec
3	11:54:25	SHOW databases	5 row(s) returned	0.000 sec / 0.000 sec

The 'Result Grid' also shows a list of databases: information\_schema, mysql, performance\_schema, sys, and travel\_db. The 'Output' tab is selected, showing the results of the queries. The 'Query Completed' status is visible at the bottom.

### 3.CREATE TABLE FOR YOUR DATABASE

**QUERY-> CREATE TABLE Admin(  
adminId INT NOT NULL AUTO\_INCREMENT,  
firstName VARCHAR(50) NOT NULL,  
lastName VARCHAR(50),  
password VARCHAR(12) NOT NULL,  
age INT NOT NULL,  
contactNo INT NOT NULL,  
emailId VARCHAR(25) NOT NULL,  
PRIMARY KEY(adminId)  
);**



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

**QUERY-> CREATE TABLE User2 AS  
SELECT UserId, firstName  
FROM User;**

**DESC User2;  
DROP TABLE User2;**

The screenshot displays the MySQL Workbench interface. The 'Query' tab is active, showing the following SQL script:

```
30 );
31
32 -- Create table using select
33 CREATE TABLE User2 AS
34 SELECT UserId, firstName
35 FROM User;
36
37 DESC User2;
```

The 'Result Grid' is visible, showing the structure of the 'User2' table:

Field	Type	Null	Key	Default	Extra
UserId	int	NO		0	
firstName	varchar(50)	NO			

The 'Output' pane at the bottom shows the execution log for the queries:

#	Time	Action	Message	Duration / Fetch
18	13:55:39	USE Travel	0 row(s) affected	0.000 sec
19	13:55:39	CREATE TABLE Admin( adminId INT NOT NULL AUTO_INCREMENT, firstName VARCHAR(50) NOT NULL, adminName VARCHAR(50) NOT NULL, adminEmail VARCHAR(50) NOT NULL, PRIMARY KEY (adminId))	0 row(s) affected	0.047 sec
20	13:55:39	CREATE TABLE User( userId INT NOT NULL AUTO_INCREMENT, adminId INT NOT NULL, firstName VARCHAR(50) NOT NULL, PRIMARY KEY (userId), FOREIGN KEY (adminId) REFERENCES Admin(adminId))	0 row(s) affected	0.062 sec
21	13:55:39	CREATE TABLE User2 AS SELECT UserId, firstName FROM User	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
22	13:55:39	DESC User2	2 row(s) returned	0.000 sec / 0.000 sec

The 'Query Completed' status is shown at the bottom left. The system tray at the bottom right indicates the date and time as 9/25/2020, 1:55 PM.

## 5.DROP TABLE

QUERY-> DROP TABLE TABLE\_NAME;

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
31
32 -- Create table using select
33 CREATE TABLE User2 AS
34 SELECT UserId, firstName
35 FROM User;
36
37 DESC User2;
38 DROP TABLE User2;
39
```

The output window shows the execution results:

#	Time	Action	Message	Duration / Fetch
27	13:57:20	CREATE TABLE Admin( adminId INT NOT NULL AUTO_INCREMENT, firstName VARC...	0 row(s) affected	0.031 sec
28	13:57:20	CREATE TABLE User (userId INT NOT NULL AUTO_INCREMENT, adminId INT NOT N...	0 row(s) affected	0.047 sec
29	13:57:20	CREATE TABLE User2 AS SELECT UserId, firstName FROM User	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.047 sec
30	13:57:20	DESC User2	2 row(s) returned	0.015 sec / 0.000 sec
31	13:57:20	DROP TABLE User2	0 row(s) affected	0.016 sec

The bottom status bar indicates "Query Completed".

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

**QUERY-> SELECT \* FROM INFORMATION\_SCHEMA.TABLES;**

The screenshot displays the MySQL Workbench interface. The 'Query' tab is active, showing a SQL query that defines a table structure and then executes a SELECT statement. The query is as follows:

```
104 bookingAgencyId INT NOT NULL,  
105 placeId INT NOT NULL,  
106 PRIMARY KEY(bookingAgencyId, placeId),  
107 FOREIGN KEY(bookingAgencyId) REFERENCES BookingAgency(bookingAgencyId),  
108 FOREIGN KEY(placeId) REFERENCES Places(placeId)  
109 );  
110  
111 SELECT * FROM INFORMATION_SCHEMA.TABLES  
112
```

The 'Result Grid' shows the output of the SELECT statement, displaying a list of tables in the database. The table has columns: TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, TABLE\_TYPE, ENGINE, VERSION, ROW\_FORMAT, TABLE\_ROWS, AVG\_ROW\_LENGTH, and DATA\_LENGTH. The data is as follows:

TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	TABLE_TYPE	ENGINE	VERSION	ROW_FORMAT	TABLE_ROWS	AVG_ROW_LENGTH	DATA_LENGTH
def	mysql	columns_priv	BASE TABLE	InnoDB	10	Dynamic	0	0	16384
def	mysql	component	BASE TABLE	InnoDB	10	Dynamic	0	0	16384
def	mysql	db	BASE TABLE	InnoDB	10	Dynamic	2	8192	16384
def	mysql	default_roles	BASE TABLE	InnoDB	10	Dynamic	0	0	16384

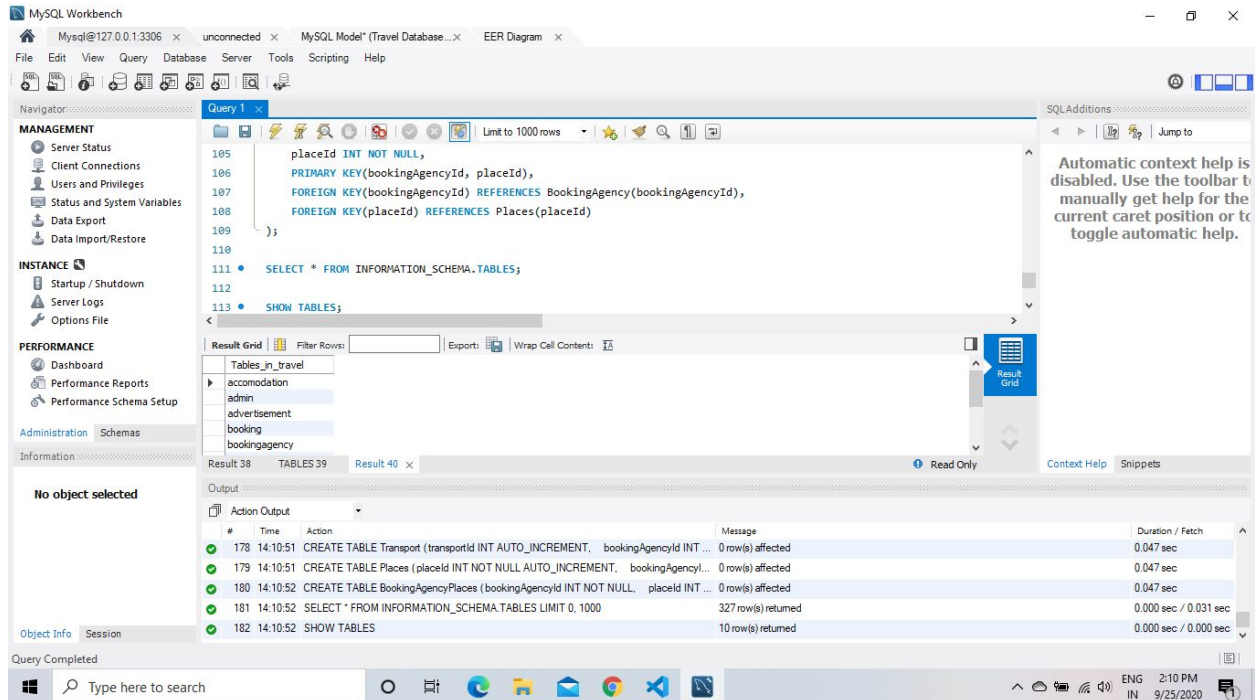
The 'Output' tab shows the execution log, indicating that the query was successful and returned 327 rows. The log entry is:

```
105 14:04:48 SELECT * FROM INFORMATION_SCHEMA.TABLES LIMIT 0, 1000  
327 row(s) returned
```

The 'Object Info' tab shows that no object was selected.

## 7.SHOW ALL THE TABLES FROM THE DATABASE

QUERY-> **SHOW TABLES;**

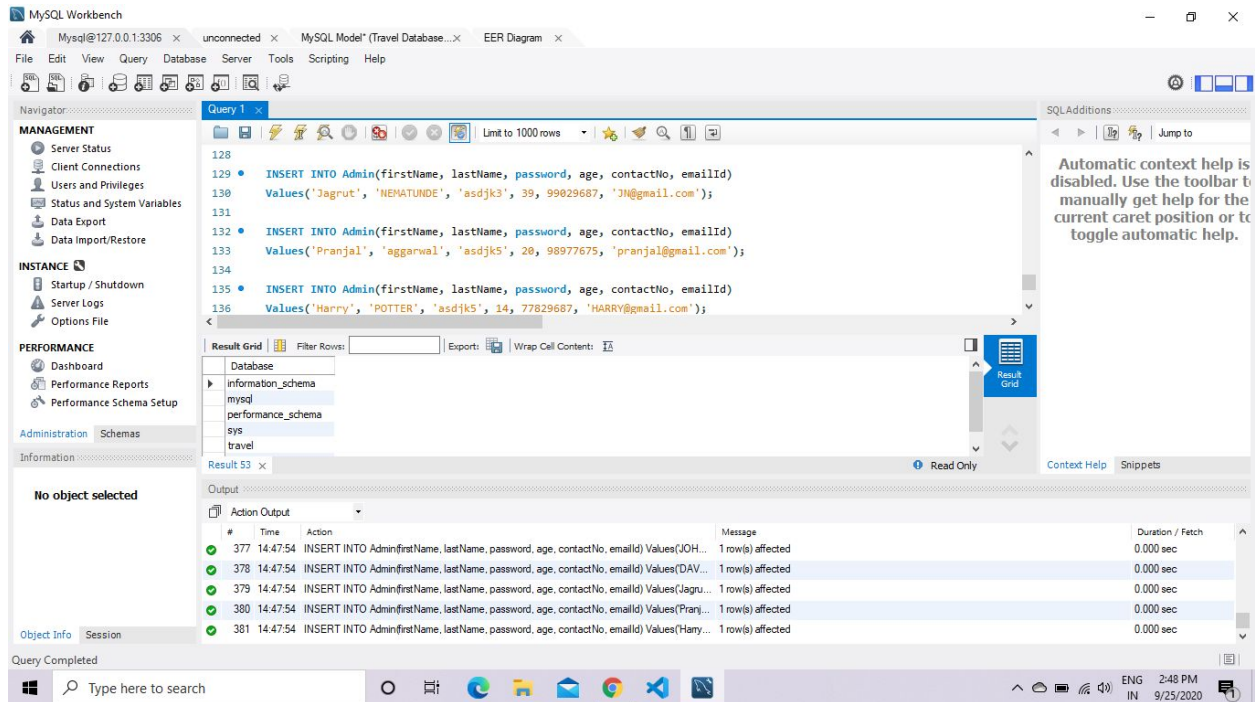


## 8.INSERT 5-10 ROWS IN EACH TABLE OF YOUR DATABASE

QUERY-> **INSERT INTO Admin(firstName, lastName, password, age, contactNo, emailId)**

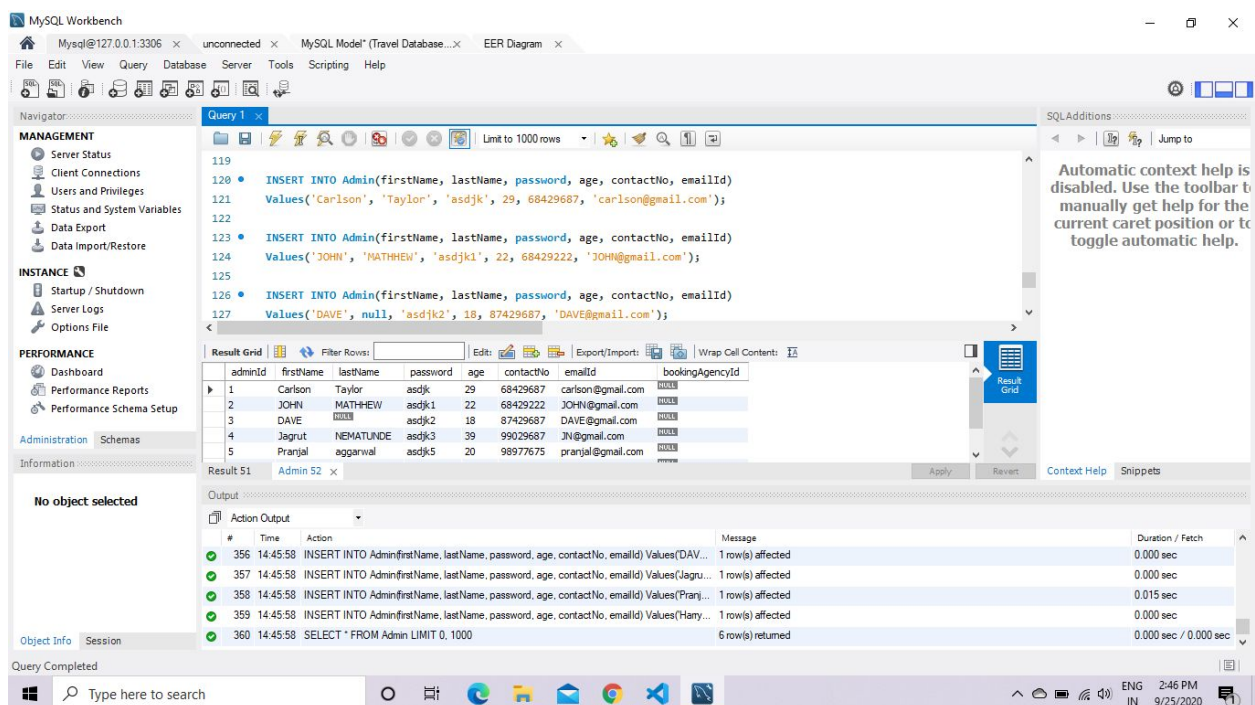
**Values('Carlson', 'Taylor', 'asdj', 29, 68429687, 'xyz@gmail.com');**





## 9.SHOW USAGE OF SIMPLE SELECT STATEMENT

QUERY-> **SELECT \* FROM Admin;**





## 10.SELECT STATEMENT USING RELATIONAL AND LOGICAL OPERATORS

**QUERY-> SELECT transportType, charges, name, adminId FROM Transport as t, BookingAgency as b WHERE t.charges >= 2500 or b.bookingAgencyId = 1;**

The screenshot displays the MySQL Workbench interface. The SQL editor contains the following query:

```
-- SELECT * FROM Transport;

-- A select query using logical and relational operators
SELECT transportType, charges, name, adminId FROM Transport as t, BookingAgency as b WHERE t.charges >= 2500 or b.bookingAgencyId = 1;
```

The query results are shown in the Result Grid, displaying 21 rows. The columns are transportType, charges, name, and adminId. The results are as follows:

transportType	charges	name	adminId
Flight	5000	READY GO	3
Flight	10000	READY GO	3
Train	2500	READY GO	3
Bus	500	READY GO	3
Flight	10000	READY GO	3

The Output pane shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
652	15:32:31	INSERT INTO Transport VALUES (2, 1, 'Flight', 10000)	1 row(s) affected	0.000 sec
653	15:32:31	INSERT INTO Transport VALUES (3, 1, 'Train', 2500)	1 row(s) affected	0.000 sec
654	15:32:31	INSERT INTO Transport VALUES (4, 1, 'Bus', 500)	1 row(s) affected	0.000 sec
655	15:32:31	INSERT INTO Transport VALUES (5, 1, 'Flight', 10000)	1 row(s) affected	0.000 sec
656	15:32:31	SELECT transportType, charges, name, adminId FROM Transport as t, BookingAgency as b ...	21 row(s) returned	0.000 sec / 0.000 sec

## 11. ONE SIMPLE SUBQUERY USING SELECT

**QUERY-> SELECT firstName, lastName, transportType FROM User as u, Transport as t WHERE transportType IN (SELECT transportType FROM Transport where transportType = 'Flight');**

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL code:

```
-- A sub-query using select
SELECT firstName, lastName, transportType FROM User as u, Transport as t WHERE transportType IN (SELECT transportType FROM Transport where transportType = 'Flight');
```

The Results window displays the following data:

firstName	lastName	transportType
HARRY	Styles	Flight
HARRY	Styles	Flight
HARRY	Styles	Flight
Ronald	Styles	Flight
Ronald	Styles	Flight
Ronald	Styles	Flight
Katy	Mathews	Flight

The Output window shows the following actions:

#	Time	Action	Message	Duration / Fetch
712	15:33:46	INSERT INTO Transport VALUES (3, 1, 'Train', 2500)	1 row(s) affected	0.000 sec
713	15:33:46	INSERT INTO Transport VALUES (4, 1, 'Bus', 500)	1 row(s) affected	0.000 sec