

Activity – 02

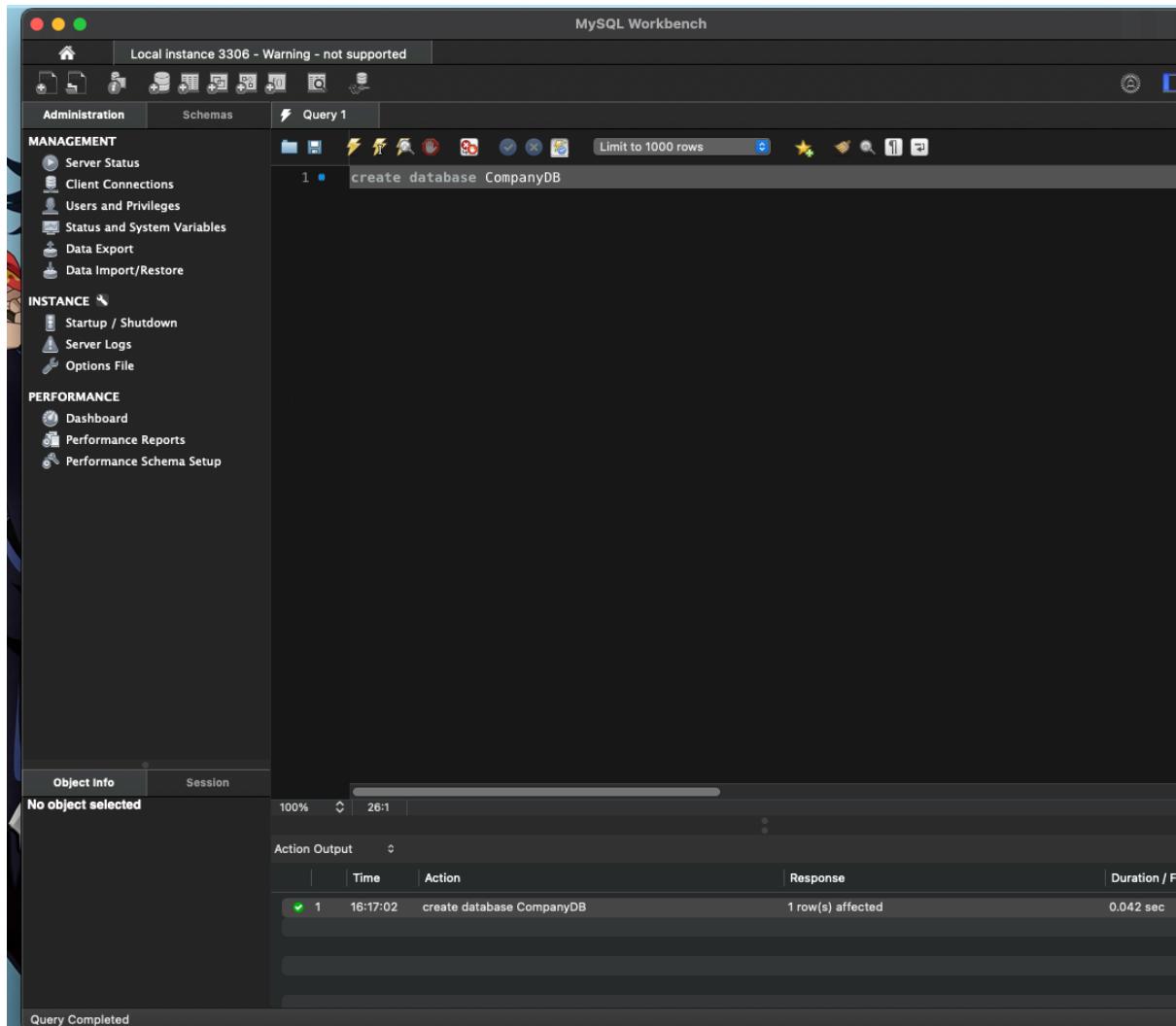
Ganesh Gundekarla

11700551

Question 1 : database creation :

This is done by :

`create database CompanyDB;`



Question 02 : creation of table

Two tables are being created next , one is for Employees and one is for Salaries.

a. `create table Employees (`
`emp_id INT primary key,`
`emp_name VARCHAR(50),`
`department VARCHAR(30)`
`);`

MySQL Workbench

Local instance 3306 - Warning - not supported

Administration Schemas Query 1

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Object Info Session

No object selected

100% 1:11

Action Output

	Time	Action	Response	Duration / Fetch Time
✓ 1	16:17:02	create database CompanyDB	1 row(s) affected	0.042 sec
✗ 2	16:18:55	create table Employees (emp_id INT primary key, emp_name VARCHAR(50), department VARCHAR(30));	Error Code: 1046. No database selected Select the de...	0.0049 sec
✓ 3	16:19:39	USE CompanyDB	0 row(s) affected	0.0021 sec
✓ 4	16:19:49	create table Employees (emp_id INT primary key, emp_name VARCHAR(50), department VARCHAR(30));	0 row(s) affected	0.023 sec

Query Completed

```
1 • create database CompanyDB;
2
3 • USE CompanyDB;
4
5 • create table Employees (
6     emp_id INT primary key,
7     emp_name VARCHAR(50),
8     department VARCHAR(30)
9 );
10
11
12
```

Now we create another table which is named Salaries which has the attributes of id and monthly salary .

b. create table Salaries(

```
    emp_id INT,  
    monthly_salary DECIMAL(10,2),  
    Foreign key(emp_id) references Employees(emp_id)  
);
```

The screenshot shows the MySQL Workbench interface with the following details:

- Left Panel (Administration):**
 - MANAGEMENT:** Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore.
 - INSTANCE:** Startup / Shutdown, Server Logs, Options File.
 - PERFORMANCE:** Dashboard, Performance Reports, Performance Schema Setup.
- Center Panel (Query 1):**

```

1 create database CompanyDB;
2
3 USE CompanyDB;
4
5 create table Employees (
6     emp_id INT primary key,
7     emp_name VARCHAR(50),
8     department VARCHAR(30)
9 );
10
11 create table Salaries(
12     emp_id INT,
13     monthly_salary DECIMAL(10,2),
14     Foreign key(emp_id) references Employees(emp_id)
15 );
16

```
- Action Output:**

	Time	Action	Response	Duration / Fetch Time
✓ 1	16:17:02	create database CompanyDB	1 row(s) affected	0.042 sec
✗ 2	16:18:55	create table Employees (emp_id INT primary key, emp_na...	Error Code: 1046. No database selected Select the de...	0.0049 sec
✓ 3	16:19:39	USE CompanyDB	0 row(s) affected	0.0021 sec
✓ 4	16:19:49	create table Employees (emp_id INT primary key, emp_na...	0 row(s) affected	0.023 sec
✓ 5	16:22:10	create table Salaries(emp_id INT, monthly_salary DECIM...	0 row(s) affected	0.0092 sec
- Bottom Panel:** Object Info (No object selected), Session, Action Output (1:11), Query Completed.

Question 03

Insertion of data into create tables .

Firstly , we are going to insert the employee details into the Employees table and the salaries of the respected persons in the Salaries table.

a. Inserting data into Employees.

```

insert into Employees(emp_id,emp_name,department)values
(1,'John Doe','HR'),
(2,'Jane Smith','IT'),
(3,'Bob Johnson','Finance');

```

The screenshot shows the MySQL Workbench interface with the following details:

- Left Panel (Object Info):**
 - Table: employees2
 - Columns:

employee_id	int	AI PK
first_name	varchar(50)	
last_name	varchar(50)	
department	varchar(50)	
salary	decimal(10,2)	
- Center Panel (Query 1):**

```

14     Foreign key(emp_id) references Employees(emp_id)
15   );
16
17 insert into Employees(emp_id,emp_name,department)values
18   (1,'John Doe','HR'),
19   (2,'Jane Smith','IT'),
20   (3,'Bob Johnson','Finance');
21

```
- Action Output:**

	Time	Action	Response	Duration / Fetch Time
✓ 1	16:35:30	insert into Employees(emp_id,emp_name,department)values...	3 row(s) affected Records: 3 Duplicates: 0 Warnings:...	0.0012 sec
- Bottom Panel:** Object Info (Table: employees2), Session, Action Output (1:17), Query Completed.

b. Insertion done into salaries table .

```
insert into Salaries(emp_id,monthly_salary)values  
(1,5000.00),  
(2,6000.00),  
(3,5500.00);
```

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'Schemas' tree, with 'ganesh_bigdata_db' selected. The main pane contains the following SQL code:

```
15 );  
16  
17 • insert into Employees(emp_id,emp_name,department)values  
18     ('1','John Doe','HR'),  
19     ('2','Jane Smith','IT'),  
20     ('3','Bob Johnson','Finance');  
21  
22 • insert into Salaries(emp_id,monthly_salary)values  
23     (1,5000.00),  
24     (2,6000.00),  
25     (3,5500.00);  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37
```

The 'Object Info' tab at the bottom shows the 'employees' table definition:

Table: employees2	Session
Columns:	
employee_id	int AI PK
first_name	varchar(50)
last_name	varchar(50)

The 'Action Output' tab shows the execution log:

Action	Time	Response	Duration / Fetch Time
insert into Employees(emp_id,emp_name,department)values... 3 row(s) affected Records: 3 Duplicates: 0 Warnings:... 0.0012 sec	16:35:30		
insert into Salaries(emp_id,monthly_salary)values (1,5000.0... 3 row(s) affected Records: 3 Duplicates: 0 Warnings:... 0.0012 sec	16:36:47		

Question 04

a. Here , select * is used to basically retrieve everything from the table.

```
select * from Employees;
```

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'Schemas' tree, with 'ganesh_bigdata_db' selected. The main pane contains the following SQL code:

```
22 • insert into Salaries(emp_id,monthly_salary)values  
23     (1,5000.00),  
24     (2,6000.00),  
25     (3,5500.00);  
26  
27 • select * from Employees;
```

The 'Result Grid' pane shows the retrieved data:

emp_id	emp_name	department
1	John Doe	HR
2	Jane Smith	IT
3	Bob Johnson	Finance
HULL	HULL	HULL

The 'Object Info' tab at the bottom shows the 'employees' table definition:

Table: employees2	Session
Columns:	
employee_id	int AI PK
first_name	varchar(50)
last_name	varchar(50)

The 'Action Output' tab shows the execution log:

Action	Time	Response	Duration / Fetch Time
insert into Employees(emp_id,emp_name,department)values... 3 row(s) affected Records: 3 Duplicates: 0 Warnings:... 0.0012 sec	16:35:30		
insert into Salaries(emp_id,monthly_salary)values (1,5000.0... 3 row(s) affected Records: 3 Duplicates: 0 Warnings:... 0.0012 sec	16:36:47		
select * from Employees LIMIT 0, 1000	16:37:25	3 row(s) returned	0.00058 sec / 0.000...

- b. We can select a specific column column using SELECT and FROM and also we can provide a condition along with it using WHERE .

select monthly_salary from Salaries;

```

Administration Schemas
Schemas
> ganesh_bigdata_db
-> ganeshdb
-> Tables
  < employees
    > Columns
    > Indexes
    > Foreign Keys
    > Triggers
  > employees1
  > employees2
-> Views
-> Stored Procedures
-> Functions
-> sys
-> testing
-> testing1

Query 1
Limit to 1000 rows
22 • insert into Salaries(emp_id,monthly_salary)values
23      (1,5000.00),
24      (2,6000.00),
25      (3,5500.00);
26
27 • select * from Employees;
28
29 • select monthly_salary from Salaries;
30
31
32
33
34
35
36
100% 38:29
Result Grid Filter Rows: Search Export:
monthly_salary
5000.00
6000.00
5500.00
Salaries 3
Object Info Session
Table: employees2
Columns:
  employee_id int AI PK
  first_name varchar(50)
  last_name varchar(50)
  department varchar(50)
  salary decimal(10,2)
Action Output
Time Action Response Duration / Fetch Time
1 16:35:30 insert into Employees(emp_id,emp_name,department)values... 3 row(s) affected Records: 3 Duplicates: 0 Warnings:... 0.0012 sec
2 16:36:47 insert into Salaries(emp_id,monthly_salary)values (1,5000.00), (2,6000.00), (3,5500.00); 3 row(s) affected Records: 3 Duplicates: 0 Warnings:... 0.0012 sec
3 16:37:25 select * from Employees LIMIT 0, 1000 3 row(s) returned 0.00058 sec / 0.000...
4 16:38:12 select monthly_salary from Salaries LIMIT 0, 1000 3 row(s) returned 0.00039 sec / 0.000...

```

Question 05

Using joins .

- a. Here , combining of the data can be done using Employees and Salaries using the ‘inner join’. So we get the result in one table showing both tables .

select * from Employees inner join Salaries ON Employees.emp_id=Salaries.emp_id;

The screenshot shows the MySQL Workbench interface. On the left, the Schemas tree displays two databases: 'ganesh_bigda...' and 'ganeshdb'. Under 'ganeshdb', there are tables like 'employees', 'employees1', and 'employee...'. The central pane shows a SQL editor with the following code:

```

22 • insert into Salaries(emp_id,monthly_salary)values
23     (1,5000.00),
24     (2,6000.00),
25     (3,5500.00);
26
27 • select * from Employees;
28
29 • select monthly_salary from Salaries;
30
31 • select * from Employees inner join Salaries ON Employees.emp_id=Salaries.emp_id;
32
33
34
35
36

```

The Result Grid shows the data inserted into the Salaries table:

emp_id	emp_name	department	emp_id	monthly_salary
1	John Doe	HR	1	5000.00
2	Jane Smith	IT	2	6000.00
3	Bob Johnson	Finance	3	5500.00

The Action Output pane shows the execution history:

Time	Action	Response	Duration / Fetch Time
16:35:30	insert into Employees(emp_id,emp_name,department)values (1,John Doe,HR)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.0012 sec
16:36:47	insert into Salaries(emp_id,monthly_salary)values (1,5000.00), (2,6000.00), (3,5500.00)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.0012 sec
16:37:25	select * from Employees LIMIT 0, 1000	3 row(s) returned	0.00058 sec / 0.000...
16:38:12	select monthly_salary from Salaries LIMIT 0, 1000	3 row(s) returned	0.00039 sec / 0.000...
16:39:36	select * from Employees inner join Salaries ON Employees.emp_id=Salaries.emp_id;	3 row(s) returned	0.00065 sec / 0.000...

- b. All the employees with their salaries can be listed using the left join method.

```
select      *      from      Employees      left      join      Salaries      on
Employees.emp_id=Salaries.emp_id;
```

The screenshot shows the MySQL Workbench interface with the same schema setup. The central pane shows the following SQL code:

```

22 • insert into Salaries(emp_id,monthly_salary)values
23     (1,5000.00),
24     (2,6000.00),
25     (3,5500.00);
26
27 • select * from Employees;
28
29 • select monthly_salary from Salaries;
30
31 • select * from Employees left join Salaries on Employees.emp_id=Salaries.emp_id;
32
33 •
34
35
36

```

The Result Grid shows the data from the Employees table and the joined Salaries table:

emp_id	emp_name	department	emp_id	monthly_salary
1	John Doe	HR	1	5000.00
2	Jane Smith	IT	2	6000.00
3	Bob Johnson	Finance	3	5500.00

The Action Output pane shows the execution history for the left join query:

Time	Action	Response	Duration / Fetch Time
16:35:30	insert into Employees(emp_id,emp_name,department)values (1,John Doe,HR)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.0012 sec
16:36:47	insert into Salaries(emp_id,monthly_salary)values (1,5000.00), (2,6000.00), (3,5500.00)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.0012 sec
16:37:25	select * from Employees LIMIT 0, 1000	3 row(s) returned	0.00058 sec / 0.000...
16:38:12	select monthly_salary from Salaries LIMIT 0, 1000	3 row(s) returned	0.00039 sec / 0.000...
16:39:36	select * from Employees left join Salaries on Employees.emp_id=Salaries.emp_id;	3 row(s) returned	0.00065 sec / 0.000...
16:41:50	select * from Employees left join Salaries on Employees.emp_id=Salaries.emp_id;	3 row(s) returned	0.0018 sec / 0.0001...

Question 06

a. WE can perform avg of monthly salaries using an inbuilt method in sql

Avg(column_name) provides us with the avg salary of the employees.

```
select avg(monthly_salary) as average_salary from Salaries;
```

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' tree displays 'ganesh_bigdata' and 'ganeshdb'. Under 'ganeshdb', there are 'Tables' (including 'employees', 'employees1', and 'employee...'), 'Views', 'Stored Proc...', 'Functions', and system tables like 'sys' and 'testing'. The central pane shows the SQL editor with the following code:

```
32
33 • select * from Employees left join Salaries on Employees.emp_id=Salaries.emp_id;
34
35 • select avg(monthly_salary) as average_salary from Salaries;
36
37
38
39
40
41
42
43
44
45
46
```

The results grid shows a single row with the value '5500.000000' under the column 'average_salary'. Below the results grid is the 'Action Output' section, which lists the following actions:

Action	Time	Response	Duration / Fetch Time
1	16:35:30	insert into Employees(emp_id,emp_name,department)values (1,'John Doe', 'Sales')	0.0012 sec
2	16:36:47	insert into Salaries(emp_id,monthly_salary)values (1,5000.00), (2,6000.00)	0.0012 sec
3	16:37:25	select * from Employees LIMIT 0, 1000	0.00058 sec / 0.000...
4	16:38:12	select monthly_salary from Salaries LIMIT 0, 1000	0.00039 sec / 0.000...
5	16:39:36	select * from Employees inner join Salaries ON Employees.emp_id=Salaries.e...	0.00065 sec / 0.000...
6	16:41:50	select * from Employees left join Salaries on Employees.emp_id=Salaries.e...	0.0018 sec / 0.00001...
7	16:43:03	select avg(monthly_salary) as average_salary from Salaries LIMIT 0, 1000	0.0021 sec / 0.00000...

b. Total number of employees.

, it provides you a number of how many employees each are in each department of the company with branch .

```
select department,count(emp_id) as total_employees from Employees
group by department;
```

```

32
33 •   select * from Employees left join Salaries on Employees.emp_id=Salaries.emp_id;
34
35 •   select avg(monthly_salary) as average_salary from Salaries;
36
37 •   select department,count(emp_id) as total_employees from Employees group by department;
38
39
40
41
42
43
44
45
46

```

departme...	total_employee...
HR	1
IT	1
Finance	1

Result 7

Action Output		Duration / Fetch Time	
	Action	Time	Response
✓ 1	insert into Employees(emp_id,emp_name,department)values (1,John Do...)	16:35:30	3 row(s) affected Records: 3 Duplicates: 0 Warnings... 0.0012 sec
✓ 2	insert into Salaries(emp_id,monthly_salary)values (1,5000.00), (2,6000...	16:36:47	3 row(s) affected Records: 3 Duplicates: 0 Warnings... 0.0012 sec
✓ 3	select * from Employees LIMIT 0, 1000	16:37:25	3 row(s) returned 0.00058 sec / 0.000...
✓ 4	select monthly_salary from Salaries LIMIT 0, 1000	16:38:12	3 row(s) returned 0.00039 sec / 0.000...
✓ 5	select * from Employees inner join Salaries ON Employees.emp_id=Salarie...	16:39:36	3 row(s) returned 0.00065 sec / 0.000...
✓ 6	select * from Employees left join Salaries on Employees.emp_id=Salaries.e...	16:41:50	3 row(s) returned 0.0018 sec / 0.00001...
✓ 7	select avg(monthly_salary) as average_salary from Salaries LIMIT 0, 1000	16:43:03	1 row(s) returned 0.0021 sec / 0.00000...
✓ 8	select department,count(emp_id) as total_employees from Employees gro...	16:47:05	3 row(s) returned 0.0019 sec / 0.00001...

Question 07

- a. Add a unique value

The term unique prevents duplicates to the emp_id in the table "Employees", so if you are beginning to inset a new row with an emp_id which is already in existance in the table , it provides us error , useful for data integrity .

`alter table Employees add constraint unq_val_emp_id unique(emp_id);`

The screenshot shows the MySQL Workbench interface. The left pane displays the 'SCHEMAS' tree with databases like 'ganesh_bigda...', 'ganeshdb', and 'sys'. The right pane shows a query editor with the following SQL code:

```

35 • select avg(monthly_salary) as average_salary from Salaries;
36
37 • select department,count(emp_id) as total_employees from Employees group by department;
38
39 • alter table Employees add constraint unq_val_emp_id unique(emp_id);
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59

```

Below the query editor is an 'Object Info' panel for the 'employees2' table, showing columns: employee_id (AI PK), first_name, last_name, department, and salary.

Action	Time	Action	Response	Duration / Fetch Time
✓ 1	16:35:30	insert into Employees(emp_id,emp_name,department)values	(1,John Doe,IT)	3 row(s) affected Records: 3 Duplicates: 0 Warnings... 0.0012 sec
✓ 2	16:36:47	insert into Salaries(emp_id,monthly_salary)values	(1,5000.00), (2,6000.00)	3 row(s) affected Records: 3 Duplicates: 0 Warnings... 0.0012 sec
✓ 3	16:37:25	select * from Employees LIMIT 0, 1000		3 row(s) returned 0.00058 sec / 0.000...
✓ 4	16:38:12	select monthly_salary from Salaries LIMIT 0, 1000		3 row(s) returned 0.00039 sec / 0.000...
✓ 5	16:39:36	select * from Employees inner join Salaries ON Employees.emp_id=Salaries.emp_id		3 row(s) returned 0.00065 sec / 0.000...
✓ 6	16:41:50	select * from Employees left join Salaries on Employees.emp_id=Salaries.emp_id		3 row(s) returned 0.0018 sec / 0.00001...
✓ 7	16:43:03	select avg(monthly_salary) as average_salary from Salaries LIMIT 0, 1000		1 row(s) returned 0.0021 sec / 0.00000...
✓ 8	16:47:05	select department,count(emp_id) as total_employees from Employees gro...		3 row(s) returned 0.0019 sec / 0.00001...
✓ 9	16:48:38	alter table Employees add constraint unq_val_emp_id unique(emp_id);		0 row(s) affected Records: 0 Duplicates: 0 Warnings... 0.026 sec

Query Completed

Question – 08

- So , the employee table employee can be changed his department from anything to 'IT' can be done using 'UPDATE ' and followed by a condition using 'WHERE ' .

`update Employees set Department='IT' where emp_id=1;`

Here , we update employees in department where employee id is '1' to IT .

The screenshot shows the MySQL Workbench interface. The left pane displays the 'SCHEMAS' tree with databases like 'ganesh_bigda...', 'ganeshdb', and 'sys'. The right pane shows a query editor with the following SQL code:

```

36
37 • alter table Employees add constraint unq_val_emp_id unique(emp_id);
38
39 • update Employees set Department='IT' where emp_id=1;
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

```

Below the query editor is an 'Object Info' panel for the 'employees2' table, showing columns: employee_id (AI PK), first_name, last_name, department, and salary.

Action	Time	Action	Response	Duration / Fetch Time
✓ 1	16:35:30	insert into Employees(emp_id,emp_name,department)values	(1,John Doe,IT)	3 row(s) affected Records: 3 Duplicates: 0 Warnings... 0.0012 sec
✓ 2	16:36:47	insert into Salaries(emp_id,monthly_salary)values	(1,5000.00), (2,6000.00)	3 row(s) affected Records: 3 Duplicates: 0 Warnings... 0.0012 sec
✓ 3	16:37:25	select * from Employees LIMIT 0, 1000		3 row(s) returned 0.00058 sec / 0.000...
✓ 4	16:38:12	select monthly_salary from Salaries LIMIT 0, 1000		3 row(s) returned 0.00039 sec / 0.000...
✓ 5	16:39:36	select * from Employees inner join Salaries ON Employees.emp_id=Salaries.emp_id		3 row(s) returned 0.00065 sec / 0.000...
✓ 6	16:41:50	select * from Employees left join Salaries on Employees.emp_id=Salaries.emp_id		3 row(s) returned 0.0018 sec / 0.00001...
✓ 7	16:43:03	select avg(monthly_salary) as average_salary from Salaries LIMIT 0, 1000		1 row(s) returned 0.0021 sec / 0.00000...
✓ 8	16:47:05	select department,count(emp_id) as total_employees from Employees...		3 row(s) returned 0.0019 sec / 0.00001...
✓ 9	16:48:38	alter table Employees add constraint unq_val_emp_id unique(emp_id);		0 row(s) affected Records: 0 Duplicates: 0 Warnings... 0.026 sec
✓ 10	16:49:41	update Employees set Department='IT' where emp_id=1;		1 row(s) affected Rows matched: 1 Changed: 1 Warnings... 0.0018 sec

Query Completed

b. Same monthly salary is done using the 'update' command.

```
update Salaries set monthly_salary=6500 where emp_id=2;
```

The screenshot shows the MySQL Workbench interface. The left sidebar displays the database schema with tables like 'employees', 'Salaries', and 'Employees1'. The main pane shows a query editor with the following SQL code:

```
37 select department,count(emp_id) as total_employees from Employees group by department;
38
39 alter table Employees add constraint unq_val_emp_id unique(emp_id);
40
41 update Employees set Department='IT' where emp_id=1;
42
43 update Salaries set monthly_salary=6500 where emp_id=2;
```

Below the query editor is a table titled 'Object Info' showing details for the 'employees2' table, which includes columns: employee_id (int AI PK), first_name (varchar(50)), last_name (varchar(50)), department (varchar(50)), and salary (decimal(10,2)).

Object Info	Session
Table: employees2	
Columns:	
employee_id	int AI PK
first_name	varchar(50)
last_name	varchar(50)
department	varchar(50)
salary	decimal(10,2)

The 'Action Output' section shows the execution log with 11 entries, detailing each step of the query execution. The final entry shows the update of the 'Salaries' table.

Action	Time	Response	Duration / Fetch Time
insert into Salaries(emp_id),monthly_salary values (1,6500);	16:38:17	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.00012 sec
select * from Employees LIMIT 0, 1000	16:37:25	3 row(s) returned	0.00058 sec / 0.000...
select monthly_salary from Salaries LIMIT 0, 1000	16:38:12	3 row(s) returned	0.00039 sec / 0.000...
select * from Employees inner join Salaries ON Employees.emp_id=Salaries.emp_id;	16:39:36	3 row(s) returned	0.00065 sec / 0.000...
select * from Employees left join Salaries on Employees.emp_id=Salaries.emp_id;	16:41:50	3 row(s) returned	0.0018 sec / 0.0001...
select avg(monthly_salary) as average_salary from Salaries LIMIT 0, 1000	16:43:03	1 row(s) returned	0.0021 sec / 0.0000...
select department,count(emp_id) as total_employees from Employees;	16:47:05	3 row(s) returned	0.0019 sec / 0.0001...
alter table Employees add constraint unq_val_emp_id unique(emp_id)	16:48:38	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.026 sec
update Employees set Department='IT' where emp_id=1	16:49:41	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.0018 sec
update Salaries set monthly_salary=6500 where emp_id=2	16:50:40	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.0022 sec

Query Completed

Question – 09

a. Delete of a row can be done using delete command .

```
delete from Employees where emp_id=2;
```

we are deleting from employees table , a row with emp_id=2 is being deleted now.

We cannot process it

Reason : Since we had deleted a row existing in the Employee table where the id=2 , because of the foreign key set to emp_id , we cannot be able to delete it directly .

This can be done using removing first from salaries then removing it from employees.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the database schema with tables like 'employees' and 'employees1'. The main area contains a query editor with the following SQL code:

```

37 • select department,count(emp_id) as total_employees from Employees group by department;
38
39 • alter table Employees add constraint unq_val_emp_id unique(emp_id);
40
41 • update Employees set Department='IT' where emp_id=1;
42
43 • update Salaries set monthly_salary=6500 where emp_id=2;
44
45 • delete from Employees where emp_id=2;
46
47
48
49
50
51
52

```

The 'Action Output' pane shows the execution log:

	Time	Action	Response	Dura...
✓ 1	16:35:30	insert in...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.001
✓ 2	16:36:47	insert in...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.001
✓ 3	16:37:25	select *...	3 row(s) returned	0.000
✓ 4	16:38:12	select m...	3 row(s) returned	0.000
✓ 5	16:39:36	select *...	3 row(s) returned	0.000
✓ 6	16:41:50	select *...	3 row(s) returned	0.001
✓ 7	16:43:03	select a...	1 row(s) returned	0.002
✓ 8	16:47:05	select d...	3 row(s) returned	0.001
✓ 9	16:48:38	alter tab...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.026
✓ 10	16:49:41	update...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.001
✓ 11	16:50:40	update...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.002
✗ 12	16:51:19	delete fr...	Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails ('companydb'.'salaries', CONSTRAINT 'salaries_i...'.	0.002

The status bar at the bottom indicates 'Query interrupted'.

Question 10.

Differences between DELETE , DROP , TRUNCATE

a. DELETE.

1. Here , the delete in sql is used in order to remove rows one or two from table by a set of conditions followed after delete .
2. This is used to manipulate the table.
3. This is considered to be slow than the truncate command if you are opting on large tables , the reason lies in logs in individual rows are being deleted at a time .

Example . Delete from GaneshDB where expenses='one dollar';

This deletes from the database ganeshDB , and the expenses with one dollar , all the rows are being deleted .

b. Drop

1. This is used to remove an entire table or a database .
2. This thing cannot be rewinded back.

3. This removes tables or database entirely with all its index,permissions.

4. Syntax : DROP table table_name;

Ex: drop table ganeshDB ;

This removes the table ganeshDB from the database

c. Truncate

1. The truncate command is used to preserve the table in the database but to remove all the rows in the existing table .

2. It's a data definition language one .

3. This thing cannot be taken back.

4. Faster compared to delete .

5. Ex : truncate table ganeshDB;

Unlike drop the table will be still be present but all row are deleted here .