GYM MANAGEMENT SYSTEM

GROUP MEMBERS:	
ID:	**NAME:**
18-36798-1	MD. RIFAT
17-35185-2	MD. KHAIRUL ALAM MAZUMDER
17-35500-3	MD. NAFIS ISHRAK
15-30616-3	MD. FAZLA RABBI

COURSE: INTRODUCTION TO DATABASE

SECTION: F

COURSE TEACHER: JUENA AHMED NOSHIN

CONTENTS:

- INTRODUCTION
- SCENARIO DESCRIPTION
- ER DIAGRAM
- NORMALIZATION
- TABLE CREATION
- DATA INSERTION
- QUERY WRITING
- CONCLUSION

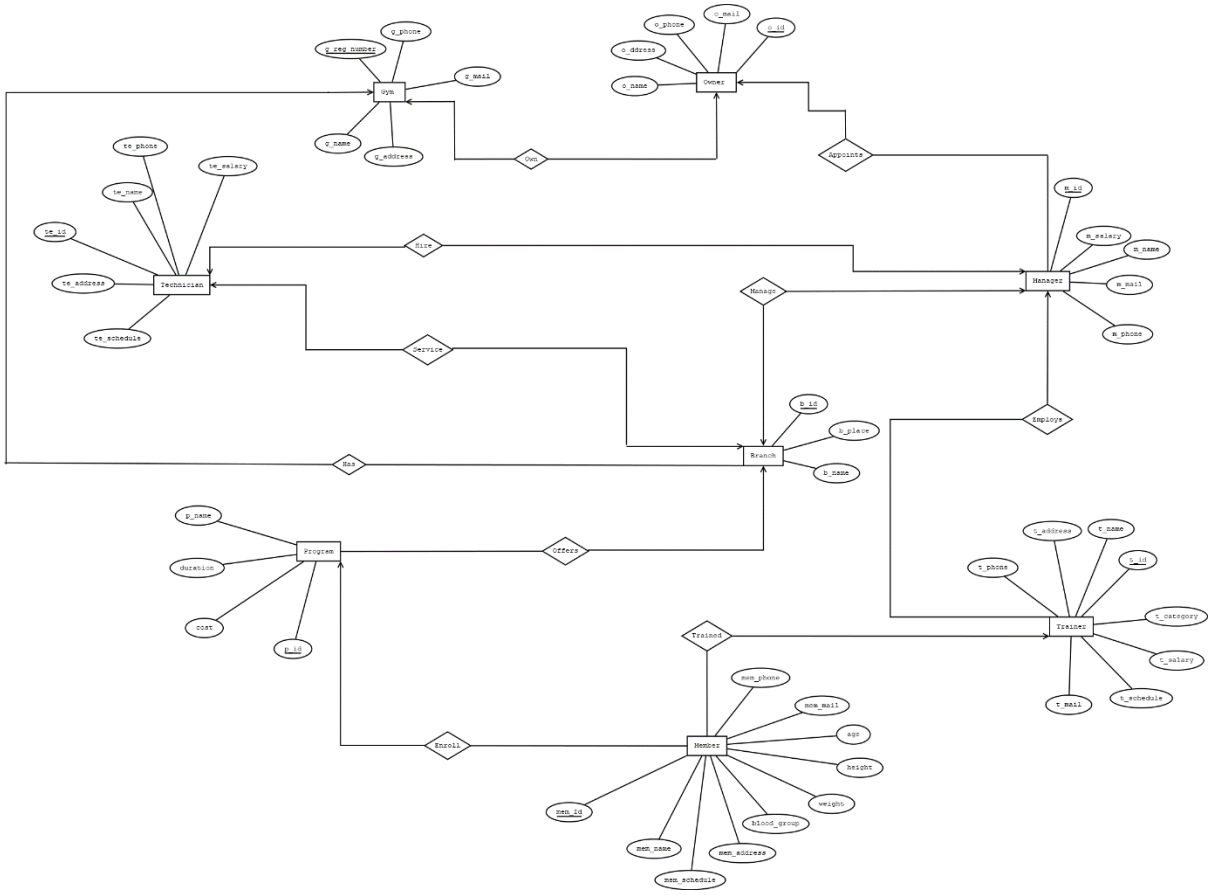
INTRODUCTION:

A database management system is (DBMS) is a computer software application that interacts with end-user, other application and the database itself to capture an analyse data. A general purpose DBMS allows the definition, creation, querying, update and administration of database.

In our project (Gym Management System) was created by the concept of DBMS. So the main purpose of our project is to know about the basic concept of database management system.

SCENARIO:

An owner wants to make a database management system for his gym where an owner can own exactly one gym where gym will be identified by its registration number, name, phone, e-mail, address will be stored. Owner will be identified by his id and name address, phone, e-mail will be also stored. A gym has many branch where branch will be identified by id and name, place will be stored. An owner can appoint many manager to manage many branch but a branch can manage by exactly one manager. Manager will be identified by his id and name, phone, e-mail, salary will also be stored. Manager can only hire a technician to provide service to his branch. Technician will be identified by his id and name, phone, address, salary, schedule will be stored. A manager can employ many trainer to train member. A trainer can be trained many member but a member can be trained by exactly one trainer. Trainer will be identified by his id and his name, address, phone, name, salary, category, e-mail, schedule will also be stored. A branch can offer many program. A program can offer by exactly one gym. Program will be identified by its id, name, duration, cost will be also stored. A member can enrol exactly one program. Member will be identified by his id, name, phone, e-mail, age, height, weight, blood group, address, schedule will also be stored.


g_phone
g_reg_number
g_mail
Gym
g_name
g_address
Own
o_phone
o_mail
o_id
o_address
Owner
o_name
Appoints
te_phone
te_salary
te_name
te_id
Hire
m_id
m_salary
m_name
te_address
Technician
Manager
m_mail
Manage
m_phone
te_schedule
Service
b_id
Employs
b_place
Branch
Has
b_name
p_name
Program
Offers
duration
t_address
t_name
t_phone
t_id
t_category
Trainer
cost
p_id
Trained
t_salary
mem_phone
t_mail
t_schedule
mem_mail
Enroll
Member
age
height
mem_id
weight
blood_group
mem_name
mem_address
mem_schedule

NORMALIZATION:

Own:

UNF:

1. Own (g_reg_number, g_phone, g_mail, g_name, g_address, o_id, o_name, o_address, o_phone, o_mail)

1NF:

There is no multivalued attribute.

1. (g_reg_number, g_phone, g_mail, g_name, g_address, o_id, o_name, o_address, o_phone, o_mail)

2NF:

1. (g_reg_number, g_phone, g_mail, g_name, g_address)
2. (o_id, o_name, o_address, o_phone, o_mail)

3NF:

1. (g_reg_number, g_phone, g_mail, g_name, g_address)
2. (o_id, o_name, o_address, o_phone, o_mail)

Table creation:

1. (g_reg_number, g_phone, g_mail, g_name, g_address, **o_id**)
2. (o_id, o_name, o_address, o_phone, o_mail)

Appoints:

UNF:

1. Appoints (o_id, o_name, o_address, o_phone, o_mail, m_id, m_salary, m_name, m_mail, m_phone)

1NF:

There is no multivalued attribute.

1. (o_id, o_name, o_address, o_phone, o_mail, m_id, m_salary, m_name, m_mail, m_phone)

2NF:

1. (o_id, o_name, o_address, o_phone, o_mail)
2. (m_id, m_salary, m_name, m_mail, m_phone)

3NF:

1. (o_id, o_name, o_address, o_phone, o_mail)
2. (m_id, m_salary, m_name, m_mail, m_phone)

Table creation:

1. (o_id, o_name, o_address, o_phone, o_mail)
2. (m_id, m_salary, m_name, m_mail, m_phone, **o_id**)

Hire:

UNF:

1. Hire (te_id, te_name, te_phone, te_salary, te_address, te_schedule, m_id, m_salary, m_name, m_mail, m_phone)

1NF:

There is no multivalued attribute.

1. (te_id, te_name, te_phone, te_salary, te_address, te_schedule, m_id, m_salary, m_name, m_mail, m_phone)

2NF:

1. (te_id, te_name, te_phone, te_salary, te_address, te_schedule)
2. (m_id, m_salary, m_name, m_mail, m_phone)

3NF:

1. (te_id, te_name, te_phone, te_salary, te_address, te_schedule)
2. (m_id, m_salary, m_name, m_mail, m_phone)

Table creation:

1. (te_id, te_name, te_phone, te_salary, te_address, te_schedule)
2. (m_id, m_salary, m_name, m_mail, m_phone, **te_id**)

Manage:

UNF:

1. Manage (m_id, m_salary, m_name, m_mail, m_phone, b_id, b_place, b_name)

1NF:

There is no multivalued attribute.

1. (m_id, m_salary, m_name, m_mail, m_phone, b_id, b_place, b_name)

2NF:

1. (m_id, m_salary, m_name, m_mail, m_phone)
2. (b_id, b_place, b_name)

3NF:

1. (m_id, m_salary, m_name, m_mail, m_phone)
2. (b_id, b_place, b_name)

Table creation:

1. (b_id, b_place, b_name)
2. (m_id, m_salary, m_name, m_mail, m_phone, **b_id**)

Service:

UNF:

1. Service (te_id, te_name, te_phone, te_salary, te_address, te_schedule, b_id, b_place, b_name)

1NF:

There is no multivalued attribute.

1. (te_id, te_name, te_phone, te_salary, te_address, te_schedule, b_id, b_place, b_name)

2NF:

1. (te_id, te_name, te_phone, te_salary, te_address, te_schedule)
2. (b_id, b_place, b_name)

3NF:

1. (te_id, te_name, te_phone, te_salary, te_address, te_schedule)
2. (b_id, b_place, b_name)

Table creation:

1. (te_id, te_name, te_phone, te_salary, te_address, te_schedule)
2. (b_id, b_place, b_name, **te_id**)

Has:

UNF:

1. Has (g_reg_number, g_phone, g_mail, g_name, g_address, b_id, b_place, b_name)

1NF:

There is no multivalued attribute.

1. (g_reg_number, g_phone, g_mail, g_name, g_address, b_id, b_place, b_name)

2NF:

1. (g_reg_number, g_phone, g_mail, g_name, g_address)
2. (b_id, b_place, b_name)

3NF:

1. (g_reg_number, g_phone, g_mail, g_name, g_address)
2. (b_id, b_place, b_name)

Table creation:

1. (g_reg_number, g_phone, g_mail, g_name, g_address)
2. (b_id, b_place, b_name, **g_reg_number**)

Offers:

UNF:

1. Offers (p_id, p_name, duration, cost, b_id, b_place, b_name)

1NF:

There is no multivalued attribute.

1. (p_id, p_name, duration, cost, b_id, b_place, b_name)

2NF:

1. (p_id, p_name, duration, cost)
2. (b_id, b_place, b_name)

3NF:

1. (p_id, p_name, duration, cost)
2. (b_id, b_place, b_name)

Table creation:

1. (b_id, b_place, b_name)
2. (p_id, p_name, duration, cost, **b_id**)

Employs:

UNF:

1. Employs (m_id, m_salary, m_name, m_mail, m_phone, t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail)

1NF:

There is no multivalued attribute.

1. (m_id, m_salary, m_name, m_mail, m_phone, t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail)

2NF:

1. (m_id, m_salary, m_name, m_mail, m_phone)
2. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail)

3NF:

1. (m_id, m_salary, m_name, m_mail, m_phone)
2. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail)

Table creation:

1. (m_id, m_salary, m_name, m_mail, m_phone)
2. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail, **m_id**)

Trained:

UNF:

1. Trained (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail, mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone)

1NF:

There is no multivalued attribute.

1. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail, mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone)

2NF:

1. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail)
2. (mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone)

3NF:

1. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail)
2. (mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone)

Table creation:

1. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail)
2. (mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone, **t_id**)

Enroll:

UNF:

1. Enroll (p_id, p_name, duration, cost, mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone)

1NF:

There is no multivalued attribute.

1. (p_id, p_name, duration, cost, mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone)

2NF:

1. (p_id, p_name, duration, cost)
2. (mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone)

3NF:

1. (p_id, p_name, duration, cost)
2. (mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone)

Table creation:

1. (p_id, p_name, duration, cost)
2. (mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone, **p_id**)

TABLE CREATION:

TOTAL TABLE:

3. (g_reg_number, g_phone, g_mail, g_name, g_address, **o_id**)
4. (o_id, o_name, o_address, o_phone, o_mail)
5. (o_id, o_name, o_address, o_phone, o_mail)
6. (m_id, m_salary, m_name, m_mail, m_phone, **o_id**)
7. (te_id, te_name, te_phone, te_salary, te_address, te_schedule)
8. (m_id, m_salary, m_name, m_mail, m_phone, **te_id**)
9. (b_id, b_place, b_name)
10. (m_id, m_salary, m_name, m_mail, m_phone, **b_id**)
11. (te_id, te_name, te_phone, te_salary, te_address, te_schedule)
12. (b_id, b_place, b_name, **te_id**)
13. (g_reg_number, g_phone, g_mail, g_name, g_address)
14. (b_id, b_place, b_name, **g_reg_number**)
15. (b_id, b_place, b_name)
16. (p_id, p_name, duration, cost, **b_id**)
17. (m_id, m_salary, m_name, m_mail, m_phone)
18. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail, **m_id**)
19. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail)
20. (mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone, **t_id**)
21. (p_id, p_name, duration, cost)
22. (mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone, **p_id**)

FINAL TABLE:

1. (g_reg_number, g_phone, g_mail, g_name, g_address, **o_id**)
2. (o_id, o_name, o_address, o_phone, o_mail)
3. (m_id, m_salary, m_name, m_mail, m_phone, **o_id, te_id, b_id**)
4. (te_id, te_name, te_phone, te_salary, te_address, te_schedule)
5. (b_id, b_place, b_name, **te_id, g_reg_number**)
6. (p_id, p_name, duration, cost, **b_id**)
7. (t_id, t_phone, t_address, t_name, t_category, t_salary, t_schedule, t_mail, **m_id**)
8. (mem_Id, mem_name, mem_schedule, mem_address, blood_group, weight, height, age, mem_mail, mem_phone, **t_id, p_id**)

TABLE CREATION:

1. create table owner(o_id number(2) primary key, o_name varchar2(40) not null, o_address varchar2(40) not null, o_phone number(20) not null,o_mail varchar2(40)not null);

Autocommit Display 10 Save Run

```
create table owner(o_id number(2) primary key, o_name varchar2(40) not null, o_address varchar2(40) not null, o_phone number(20) not null,o_mail varchar2(40)not null);
describe owner;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object OWNER

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OWNER	O_ID	Number	-	2	0	1	-	-	-
	O_NAME	Varchar2	40	-	-	-	-	-	-
	O_ADDRESS	Varchar2	40	-	-	-	-	-	-
	O_PHONE	Number	-	20	0	-	-	-	-
	O_MAIL	Varchar2	40	-	-	-	-	-	-

1 - 5

2. create table gym(g_reg_number number(20) primary key, g_name varchar2(40) not null, g_address varchar2(40) not null, g_phone number(20) not null,g_mail varchar2(40)not null,o_id number(2)not null);

Autocommit Display 10 Save Run

```
create table gym(g_reg_number number(20) primary key, g_name varchar2(40) not null, g_address varchar2(40) not null, g_phone number(20) not null,g_mail varchar2(40)not
null,o_id number(2)not null);
describe gym;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object GYM

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
GYM	G_REG_NUMBER	Number	-	20	0	1	-	-	-
	G_NAME	Varchar2	40	-	-	-	-	-	-
	G_ADDRESS	Varchar2	40	-	-	-	-	-	-
	G_PHONE	Number	-	20	0	-	-	-	-
	G_MAIL	Varchar2	40	-	-	-	-	-	-
	O_ID	Number	-	2	0	-	-	-	-

1 - 6

3. create table technician(te_id number(2) primary key, te_name varchar2(40) not null, te_salary number(10) not null, te_phone number(20) not null,te_address varchar2(40)not null,te_schedule number(2)not null);

Autocommit Display 10 Save Run

```
create table technician(te_id number(2) primary key, te_name varchar2(40) not null, te_salary number(10) not null, te_phone number(20) not null,te_address
varchar2(40)not null,te_schedule number(2)not null);
describe technician;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object TECHNICIAN

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
TECHNICIAN	TE_ID	Number	-	2	0	1	-	-	-
	TE_NAME	Varchar2	40	-	-	-	-	-	-
	TE_SALARY	Number	-	10	0	-	-	-	-
	TE_PHONE	Number	-	20	0	-	-	-	-
	TE_ADDRESS	Varchar2	40	-	-	-	-	-	-
	TE_SCHEDULE	Number	-	2	0	-	-	-	-

1 - 6

4. create table branch(b_id number(2) primary key, b_name varchar2(40) not null, b_place varchar2(40) not null, te_id number(2) not null,g_reg_number number(20)not null);

Autocommit Display 10 Save Run

```
create table branch(b_id number(2) primary key, b_name varchar2(40) not null, b_place varchar2(40) not null, te_id number(2) not null,g_reg_number number(20)not null);
describe branch;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object BRANCH

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
BRANCH	B_ID	Number	-	2	0	1	-	-	-
	B_NAME	Varchar2	40	-	-	-	-	-	-
	B_PLACE	Varchar2	40	-	-	-	-	-	-
	TE_ID	Number	-	2	0	-	-	-	-
	G_REG_NUMBER	Number	-	20	0	-	-	-	-

1 - 5

5. create table manager(m_id number(2) primary key, m_name varchar2(40) not null, m_salary number(10) not null, m_phone number(20) not null,m_mail varchar2(40)not null, o_id number(2), te_id number(2), b_id number(2));

Autocommit Display 10 Save Run

```
create table manager(m_id number(2) primary key, m_name varchar2(40) not null, m_salary number(10) not null, m_phone number(20) not null,m_mail varchar2(40)not null,
o_id number(2), te_id number(2), b_id number(2));

describe manager;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object MANAGER

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
MANAGER	M_ID	Number	-	2	0	1	-	-	-
	M_NAME	Varchar2	40	-	-	-	-	-	-
	M_SALARY	Number	-	10	0	-	-	-	-
	M_PHONE	Number	-	20	0	-	-	-	-
	M_MAIL	Varchar2	40	-	-	-	-	-	-
	O_ID	Number	-	2	0	-	✓	-	-
	TE_ID	Number	-	2	0	-	✓	-	-
	B_ID	Number	-	2	0	-	✓	-	-

1 - 8

6. create table program(p_id number(2) primary key, p_name varchar2(40) not null, duration varchar2(40) not null, cost number(20) not null,b_id number(2)not null);

Autocommit Display 10 Save Run

```
create table program(p_id number(2) primary key, p_name varchar2(40) not null, duration varchar2(40) not null, cost number(20) not null,b_id number(2)not null);

describe program;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object PROGRAM

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PROGRAM	P_ID	Number	-	2	0	1	-	-	-
	P_NAME	Varchar2	40	-	-	-	-	-	-
	DURATION	Varchar2	40	-	-	-	-	-	-
	COST	Number	-	20	0	-	-	-	-
	B_ID	Number	-	2	0	-	-	-	-

1 - 5

7. create table trainer(t_id number(2) primary key, t_name varchar2(40) not null, t_address varchar2(40) not null, t_phone number(20) not null,t_mail varchar2(40)not null, t_category varchar2(40),t_salary number(10),t_schedule number(2),m_id number (2));

Autocommit Display 10 Save Run

```
create table trainer(t_id number(2) primary key, t_name varchar2(40) not null, t_address varchar2(40) not null, t_phone number(20) not null,t_mail varchar2(40)not null,
t_category varchar2(40),t_salary number(10),t_schedule number(2),m_id number (2));

describe trainer;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object TRAINER

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
TRAINER	T_ID	Number	-	2	0	1	-	-	-
	T_NAME	Varchar2	40	-	-	-	-	-	-
	T_ADDRESS	Varchar2	40	-	-	-	-	-	-
	T_PHONE	Number	-	20	0	-	-	-	-
	T_MAIL	Varchar2	40	-	-	-	-	-	-
	T_CATEGORY	Varchar2	40	-	-	-	✓	-	-
	T_SALARY	Number	-	10	0	-	✓	-	-
	T_SCHEDULE	Number	-	2	0	-	✓	-	-
	M_ID	Number	-	2	0	-	✓	-	-

1 - 9

8. create table member(mem_id number(2) primary key, mem_name varchar2(40) not null, mem_address varchar2(40) not null, mem_phone number(20) not null,mem_mail varchar2(40)not null,mem_schedule number(2), blood_group varchar2(2),weight number(2), height number(2), age number(3), t_id number(2), p_id number(2));

Autocommit Display 10 Save Run

```
create table member(mem_id number(2) primary key, mem_name varchar2(40) not null, mem_address varchar2(40) not null, mem_phone number(20) not null,mem_mail
varchar2(40)not null,mem_schedule number(2), blood_group varchar2(2),weight number(2), height number(2), age number(3), t_id number(2), p_id number(2));

describe member;
```

Results Explain Describe Saved SQL History

Object Type TABLE Object MEMBER

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
MEMBER	MEM_ID	Number	-	2	0	1	-	-	-
	MEM_NAME	Varchar2	40	-	-	-	-	-	-
	MEM_ADDRESS	Varchar2	40	-	-	-	-	-	-
	MEM_PHONE	Number	-	20	0	-	-	-	-
	MEM_MAIL	Varchar2	40	-	-	-	-	-	-
	MEM_SCHEDULE	Number	-	2	0	-	✓	-	-
	BLOOD_GROUP	Varchar2	2	-	-	-	✓	-	-
	WEIGHT	Number	-	2	0	-	✓	-	-
	HEIGHT	Number	-	2	0	-	✓	-	-
	AGE	Number	-	3	0	-	✓	-	-
	T_ID	Number	-	2	0	-	✓	-	-
	P_ID	Number	-	2	0	-	✓	-	-

1 - 12

DATA INSERTION:

1. insert into owner values('1', 'ROBIN','DHAKA','01626485694','KHAIRUL11100000@GMAIL.COM');
2. insert into owner values('2', 'xxx','DHAKA','01656485694','xxx@GMAIL.COM');
3. insert into owner values('3', 'yyy','DHAKA','01646485694','yyy@GMAIL.COM');
4. insert into owner values('4', 'abc','DHAKA','01636485694','abc@GMAIL.COM');
5. insert into owner values('5', 'def','DHAKA','01616485694','def@GMAIL.COM');

Autocommit Display 10

```
select * from owner;
```

Results Explain Describe Saved SQL History

O_ID	O_NAME	O_ADDRESS	O_PHONE	O_MAIL
1	ROBIN	DHAKA	1626485694	KHAIRUL11100000@GMAIL.COM
2	xxx	DHAKA	1656485694	xxx@GMAIL.COM
3	yyy	DHAKA	1646485694	yyy@GMAIL.COM
4	abc	DHAKA	1636485694	abc@GMAIL.COM
5	def	DHAKA	1616485694	def@GMAIL.COM

6. insert into technician values('10', 'tareq','5000','01726485694','chittagong', '12');
7. insert into technician values('11', 'robik','5000','01526485694','dhaka', '11');
8. insert into technician values('12', 'shanto','5000','01326485694','comilla', '10');
9. insert into technician values('13', 'saha','5000','01926485694','jamalpur', '9');
10. insert into technician values('14', 'shudip','5000','01626485694','chittagong', '8');

Autocommit Display 10

```
select * from technician;
```

Results Explain Describe Saved SQL History

TE_ID	TE_NAME	TE_SALARY	TE_PHONE	TE_ADDRESS	TE_SCHEDULE
10	tareq	5000	1726485694	chittagong	12
11	robik	5000	1526485694	dhaka	11
12	shanto	5000	1326485694	comilla	10
13	saha	5000	1926485694	jamalpur	9
14	shudip	5000	1626485694	chittagong	8

11. insert into gym values('102564856','rex_gym','dhaka','01254689756','rex@gmail.com','1');
12. insert into gym values('102565856','3ex_gym','dhaka','01255689756','3ex@gmail.com','2');
13. insert into gym values('102574856','mex_gym','dhaka','01354689756','mex@gmail.com','3');
14. insert into gym values('102568856','tex_gym','comilla','01754689756','tex@gmail.com','4');
15. insert into gym values('102564876','lex_gym','dhaka','01254789756','lex@gmail.com','5');

Autocommit Display 10

```
select * from gym;
```

Results Explain Describe Saved SQL History

G_REG_NUMBER	G_NAME	G_ADDRESS	G_PHONE	G_MAIL	O_ID
102564856	rex_gym	dhaka	1254689756	rex@gmail.com	1
102565856	3ex_gym	dhaka	1255689756	3ex@gmail.com	2
102574856	mex_gym	dhaka	1354689756	mex@gmail.com	3
102568856	tex_gym	comilla	1754689756	tex@gmail.com	4
102564876	lex_gym	dhaka	1254789756	lex@gmail.com	5

16. insert into branch values('15','rex_gym_br_1','mirpur','10','102564856');
17. insert into branch values('16','rex_gym_br_2','kuratoli','11','102564856');
18. insert into branch values('17','rex_gym_br_3','dhanmondi','12','102564856');
19. insert into branch values('18','rex_gym_br_4','komolapur','13','102564856');
20. insert into branch values('19','rex_gym_br_5','bashundara','14','102564856');

Autocommit Display 10

```
select * from branch;
```

Results Explain Describe Saved SQL History

B_ID	B_NAME	B_PLACE	TE_ID	G_REG_NUMBER
15	rex_gym_br_1	mirpur	10	102564856
16	rex_gym_br_2	kuratoli	11	102564856
17	rex_gym_br_3	dhanmondi	12	102564856
18	rex_gym_br_4	komolapur	13	102564856
19	rex_gym_br_5	bashundara	14	102564856

21. insert into manager values('20','mr x','20000','01565854536','x@gmail.com','1','10','15');
22. insert into manager values('21','mr y','20000','01665854536','y@gmail.com','1','11','16');
23. insert into manager values('22','mr z','20000','01765854536','z@gmail.com','1','12','17');
24. insert into manager values('23','mr a','20000','01675854536','a@gmail.com','1','13','18');
25. insert into manager values('24','mr b','20000','01875854536','b@gmail.com','1','14','19');

Autocommit Display 10

```
select * from manager;
```

Results Explain Describe Saved SQL History

M_ID	M_NAME	M_SALARY	M_PHONE	M_MAIL	O_ID	TE_ID	B_ID
20	mr x	20000	1565854536	x@gmail.com	1	10	15
21	mr y	20000	1665854536	y@gmail.com	1	11	16
22	mr z	20000	1765854536	z@gmail.com	1	12	17
23	mr a	20000	1675854536	a@gmail.com	1	13	18
24	mr b	20000	1875854536	b@gmail.com	1	14	19

26. insert into program values('30','body_bulding','2','5000','15');
27. insert into program values('31','muscle_bulding','2','7000','15');
28. insert into program values('32','6_pack','2','9000','15');
29. insert into program values('33','4_pack','2','6000','15');
30. insert into program values('34','complete_pacakge','5','15000','15');

Autocommit Display 10

```
select * from program;
```

Results Explain Describe Saved SQL History

P_ID	P_NAME	DURATION	COST	B_ID
30	body_bulding	2	5000	15
31	muscle_bulding	2	7000	15
32	6_pack	2	9000	15
33	4_pack	2	6000	15
34	complete_pacakge	5	15000	15

31. insert into trainer values('35','ab','dhaka','016258569558','ab@gmail.com','body_bulding_specialist','15000','11','20');
32. insert into trainer values('36','bc','dhaka','016258569558','bc@gmail.com','muscle_bulding_specialist','15000','9','20');
33. insert into trainer values('37','cd','dhaka','018258569558','cd@gmail.com','6_pack_specialist','15000','7','20');
34. insert into trainer values('38','de','dhaka','018298569558','de@gmail.com','4_pack_specialist','15000','02','20');
35. insert into trainer values('39','ef','dhaka','019298569558','ef@gmail.com','specialist_trainer','30000','04','20');

Autocommit Display 10

```
select * from trainer;
```

Results Explain Describe Saved SQL History

T_ID	T_NAME	T_ADDRESS	T_PHONE	T_MAIL	T_CATEGORY	T_SALARY	T_SCHEDULE	M_ID
35	ab	dhaka	16258569558	ab@gmail.com	body_bulding_specialist	15000	11	20
36	bc	dhaka	16258569558	bc@gmail.com	muscle_bulding_specialist	15000	9	20
37	cd	dhaka	18258569558	cd@gmail.com	6_pack_specialist	15000	7	20
38	de	dhaka	18298569558	de@gmail.com	4_pack_specialist	15000	2	20
39	ef	dhaka	19298569558	ef@gmail.com	specialist_trainer	30000	4	20

36. insert into member values('40','ccc','dhaka','0125648569','ccc@gmail.com','11','b+','50','5.1','30','35','30');
37. insert into member values('41','aaa','dhaka','0125648569','aaa@gmail.com','12','b+','56','5.3','32','36','31');
38. insert into member values('42','bbb','dhaka','01256485694','bbb@gmail.com','13','0+','40','5.5','33','37','32');
39. insert into member values('43','ddd','dhaka','0115648569','ddd@gmail.com','14','a+','59','5.7','34','38','33');

40. insert into member
values('44','eee','dhaka','0135648569','eee@gmail.com','02','b+','50','5.8','35','39','34');

Autocommit Display 10

```
select * from member;
```

Results Explain Describe Saved SQL History

MEM_ID	MEM_NAME	MEM_ADDRESS	MEM_PHONE	MEM_MAIL	MEM_SCHEDULE	BLOOD_GROUP	WEIGHT	HEIGHT	AGE	T_ID	P_ID
40	ccc	dhaka	125648569	ccc@gmail.com	11	b+	50	5	30	35	30
41	aaa	dhaka	125648569	aaa@gmail.com	12	b+	56	5	32	36	31
42	bbb	dhaka	1256485694	bbb@gmail.com	13	0+	40	6	33	37	32
43	ddd	dhaka	115648569	ddd@gmail.com	14	a+	59	6	34	38	33
44	eee	dhaka	135648569	eee@gmail.com	2	b+	50	6	35	39	34

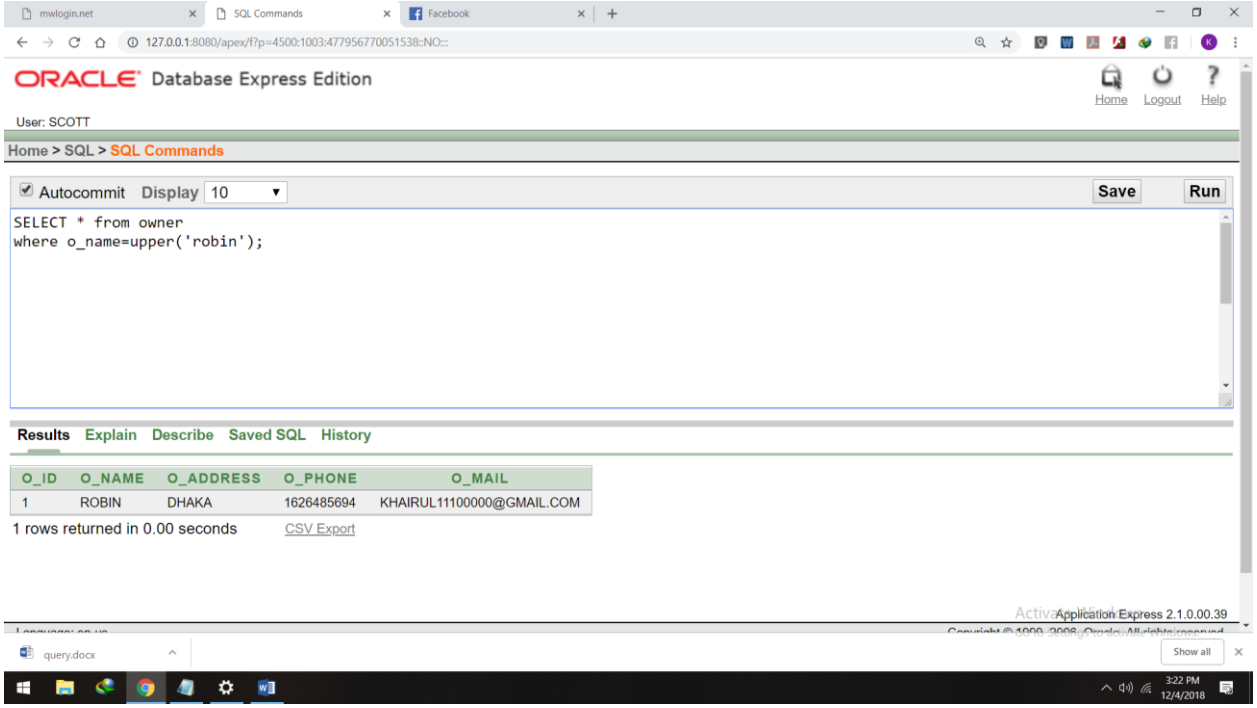
QUERY WRITING:

Single row:

1. Ques: Display all the data of owner ROBIN.

Ans: SELECT * from owner

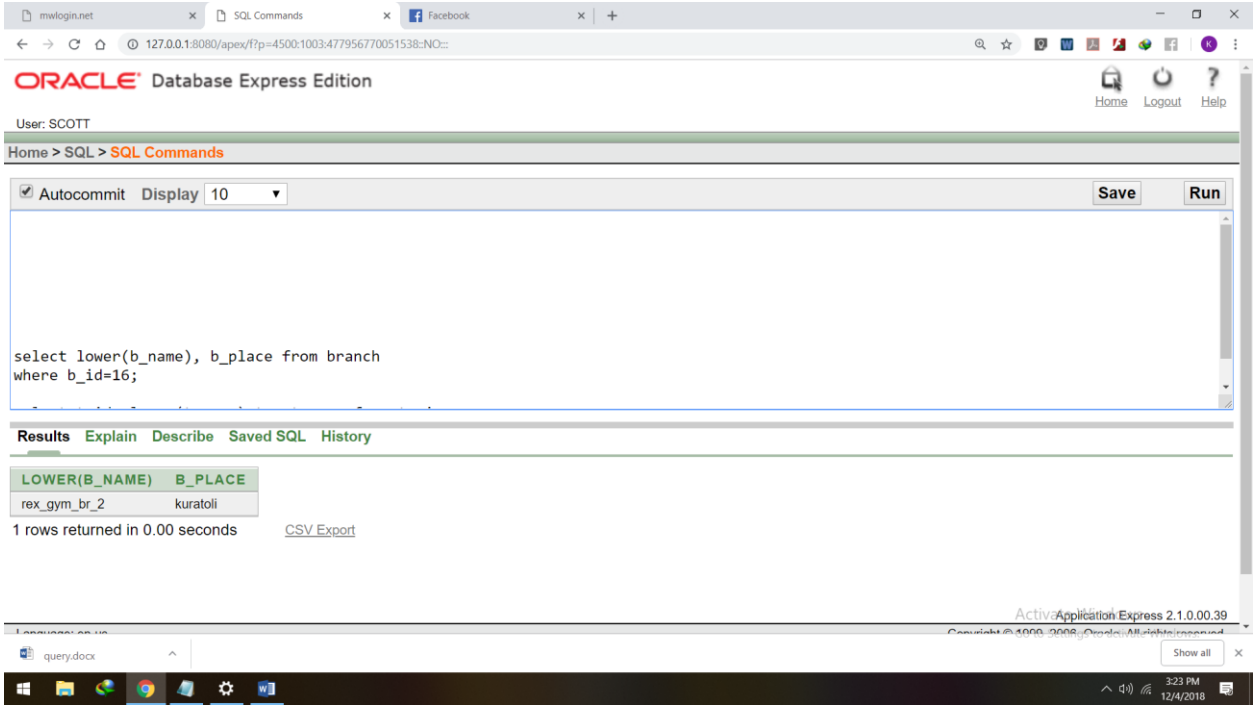
where o_name=upper('robin');



2. Ques: Display the branch name and place from branch where branch id=16.

Ans: select lower(b_name), b_place from branch

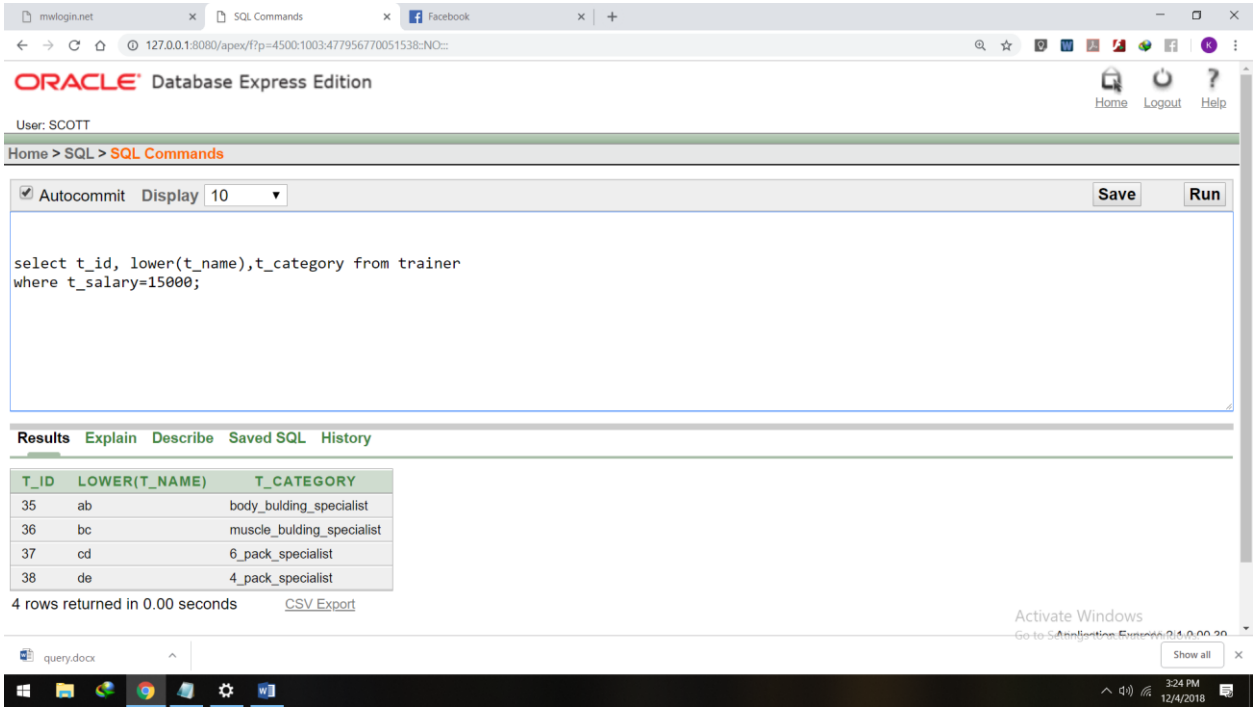
where b_id=16;



3. Ques: Display the trainer name, id, category who's salary =15000.

Ans: select t_id, lower(t_name),t_category from trainer

where t_salary=15000;



Group Function:

4. Ques: Display owner id and sum of salary of managers.

Ans: select o_id, sum(m_salary) from manager group by o_id;

Autocommit Display 10

```
select o_id, sum(m_salary) from manager
group by o_id;
```

Results Explain Describe Saved SQL History

O_ID	SUM(M_SALARY)
1	100000

5. Ques: Display manager id and average of salary of trainers.

Ans: select m_id, avg(t_salary) from trainer group by m_id;

Autocommit Display 10

```
select m_id, avg(t_salary) from trainer
group by m_id;
```

Results Explain Describe Saved SQL History

M_ID	AVG(T_SALARY)
20	18000

6. Ques: Display manager id and sum of salary of trainers where salary > 15000.

Ans: select m_id, sum(t_salary) from trainer where t_salary>15000

group by m_id;

Autocommit Display 10

```
select m_id, sum(t_salary) from trainer where t_salary>15000
group by m_id;
```

Results Explain Describe Saved SQL History

M_ID	SUM(T_SALARY)
20	30000

SUB-QUERY:

7. QUES: Display the trainer name who gets more salary than ‘ ab‘.

Ans: select t_name from trainer

where t_salary>(select t_salary from trainer where t_name='ab');

Autocommit Display 10

```
select t_name from trainer
where t_salary>(select t_salary from trainer where t_name='ab');
```

Results Explain Describe Saved SQL History

T_NAME
ef

8. Ques: Display the id of manager who earns same as id ‘22‘.

Ans: select m_id from manager where m_salary = (select m_salary from manager where m_id='22');

Autocommit Display 10

```
select m_id from manager
where m_salary=(select m_salary from manager where m_id='22');
```

Results Explain Describe Saved SQL History

M_ID
20
21
22
23
24

9. Ques: Display the name of technician who earns same as id ‘ 10‘.

Ans: select te_name from technician where te_salary=(select te_salary from technician where te_id='10');

Autocommit Display 10

```
select te_name from technician
where te_salary=(select te_salary from technician where te_id='10');
```

Results Explain Describe Saved SQL History

TE_NAME
tareq
robik
shanto
saha
shudip

joining:

10.Ques: Display manager name ,id and owner name ,id from table.

Ans: select m.m_id,m.m_name, o.o_id, o.o_name

from manager m, owner o

where m.o_id=o.o_id;

Autocommit Display 10

```
select m.m_id,m.m_name, o.o_id, o.o_name
from manager m, owner o
where m.o_id=o.o_id;
```

Results Explain Describe Saved SQL History

M_ID	M_NAME	O_ID	O_NAME
24	mr b	1	ROBIN
23	mr a	1	ROBIN
22	mr z	1	ROBIN
21	mr y	1	ROBIN
20	mr x	1	ROBIN

11.Ques: Display manager name ,id and technician name ,id from table.

Ans: select m.m_id,m.m_name, te.te_id, te.te_name

from manager m, technician te

where m.te_id=te.te_id;

Autocommit Display 10

```
select m.m_id,m.m_name, te.te_id, te.te_name
from manager m, technician te
where m.te_id=te.te_id;
```

Results Explain Describe Saved SQL History

M_ID	M_NAME	TE_ID	TE_NAME
20	mr x	10	tareq
21	mr y	11	robik
22	mr z	12	shanto
23	mr a	13	saha
24	mr b	14	shudip

12.Ques: Display branch name ,id and technician name ,id from table.

Ans: select b.b_id,b.b_name, te.te_id, te.te_name

from branch b, technician te where b.te_id=te.te_id;

Autocommit Display 10

```
select b.b_id,b.b_name, te.te_id, te.te_name
from branch b, technician te
where b.te_id=te.te_id;
```

Results Explain Describe Saved SQL History

B_ID	B_NAME	TE_ID	TE_NAME
15	rex_gym_br_1	10	tareq
16	rex_gym_br_2	11	robik
17	rex_gym_br_3	12	shanto
18	rex_gym_br_4	13	saha
19	rex_gym_br_5	14	shudip

CONCLUSION:

We created a database gym management system that a gym owner can use for keeping track on its employees. Employees are divided into into manager, technician, trainer etc. We created the database according to the wishes of our customer gym owner so that he or she can provide better service to the members.