



**Project Name:** *“Beauty Parlor Management System”*

## **Group members:**

MD. AOLAD HOSSAIN ANNA (17- 33513-1)

MD. SAJID BIN FAISAL (17-33596-1)

KHONDOKER SHAHRINA (17-33461-1)

MD ASHIQUL HAQUE (17-33578-1 )

MOHAMMAD AFZAR AL MUNEM (17-33469-1)

**COURSE NAME:** INTRODUCTION TO DATABASE.

**COURSE TEACHER:** JUENA AHMED NOSHIN

**Section:** [I].

# **OBJECTIVES**

- **Introduction**
- **Scenario Description**
- **ER Diagram**
- **Normalization**
- **Table Creation**
- **Data Insertion**
- **Query Writing**
- **Conclusion**

## **Introduction:**

The title of the project is “*beauty parlor management system*”. We have made the relational tables to make the database system and their information. We have the descriptions of the query that is required to create the tables and insert the values in the tables. We had to go through the normalization process to overcome data insertion, deletion and update anomalies to decide the tables to be created.

We do believe that this project will be beneficial to a lot of new or updatable parlor management system.

# Scenario description:

The scenario is of a beauty parlor management system.

1. The parlor will be owned by 1 owner. The owner will have the properties like owner country, owner city and owner name. The parlor may have the attributes like men's section and women's arena. The parlor will have the unique attribute named branch code.
2. The parlor will hire many employees. Each employee may have properties as employee name, employee id, employee mobile number, the salary of the employee and the skill which is a multivalued attribute.
3. The parlor will offer many packages and each of the package may have the properties as follows:
  - =>Package type (multivalued attribute)
  - =>package code (unique key)
  - =>package name
  - =>discount percentage

4. The parlor will provide many services and each of the Service may have attributes as service name, service type (Multivalued attribute) and a primary key named service Code.

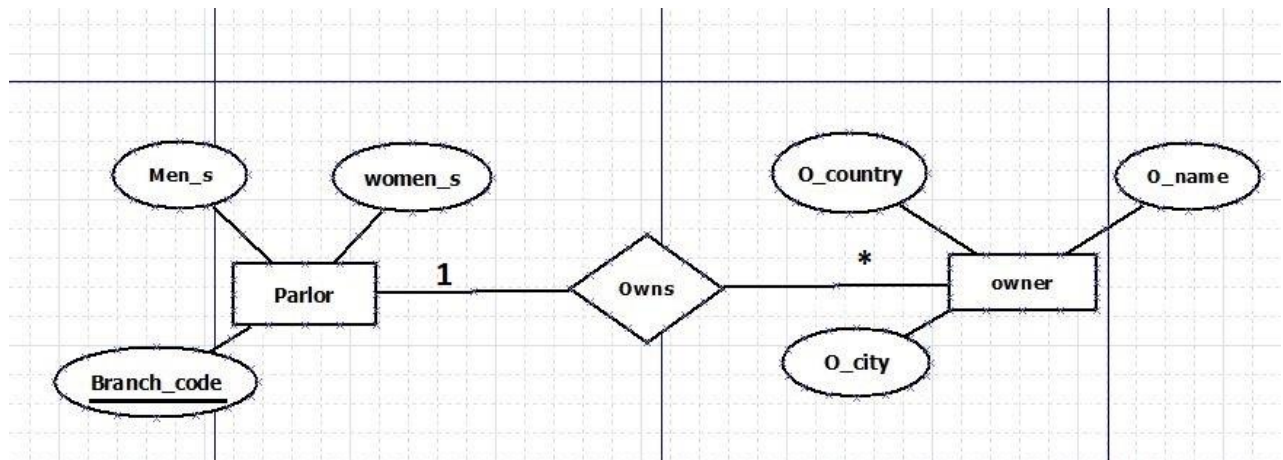
5. The services will be booked by many clients as a many to Many relationship and the clients will have these attributes  
As follows:

- =>gender
- =>client name
- =>client mobile number (multivalued)
- =>client id (unique identity)

6. In the end, the clients will be served by many employees.

This is a many to many relationship.

# Normalization:



UNF:

owns(O\_name,O\_country,O\_city,men\_s,women\_s , branch\_code)

1NF:

(O\_name,O\_country,O\_city,men\_s,women\_s , branch\_code)

2NF:

(O\_name,O\_country,O\_city, Branch\_code )

(branch code, men\_s ,women\_s)

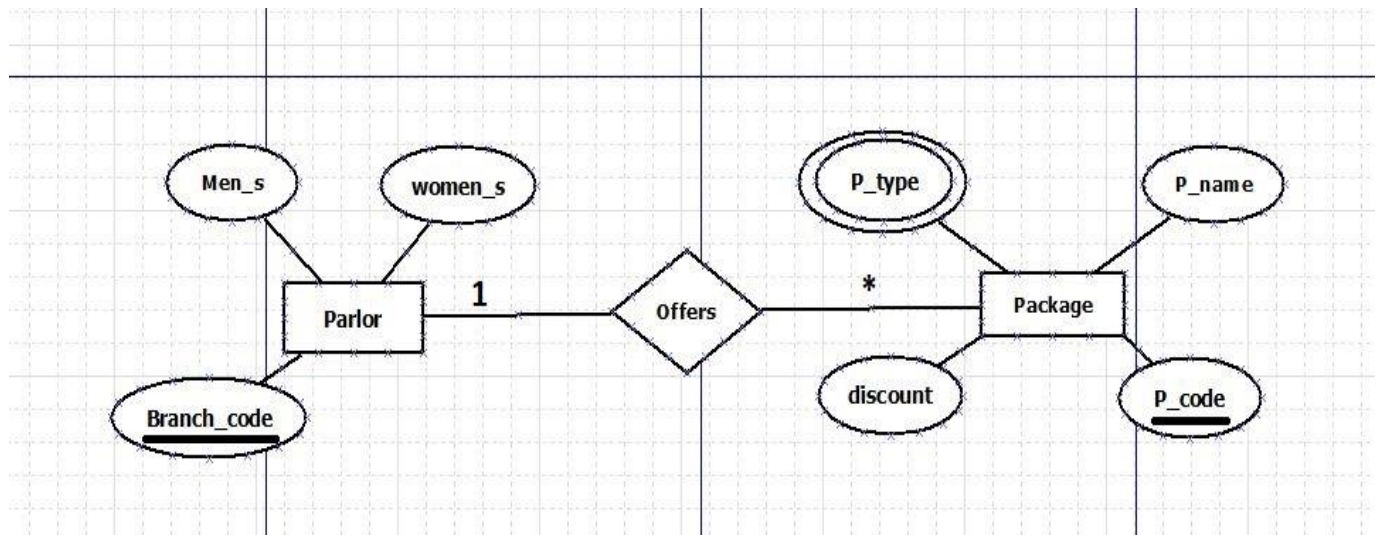
3NF:

(O city, O\_country )

(O\_name,O\_city, Branch\_code )

(branch code, men\_s ,women\_s)

# Normalization:



UNF:

Offers (branch code, men\_s, women\_s, p code, p\_name, p\_type, discount)

1NF:

(branch code, men\_s, women\_s, p code, p\_name, p\_type, discount)

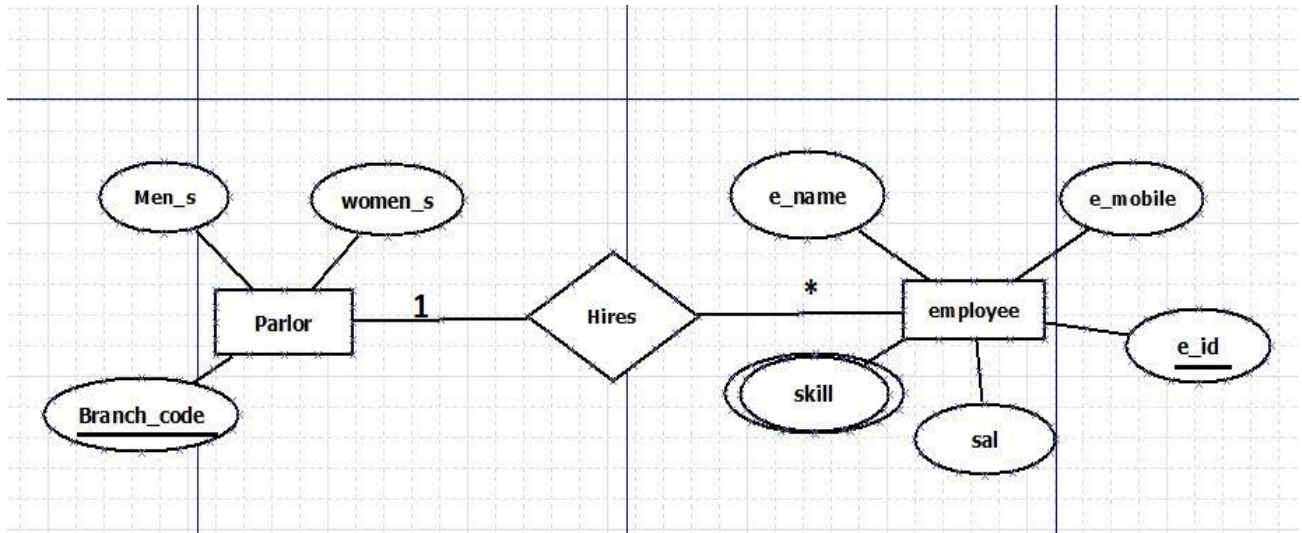
2NF:

(branch code, men\_s, women\_s)

(p code, p\_name, p\_type, discount, Branch\_code)

3NF is as same as 2NF.

# Normalization:



UNF:

Hires(branch code, men\_s, women\_s, e id, e\_name, e\_mobile, sal, skill)

1NF:

(branch code, men\_s, women\_s, e id, e\_name, e\_mobile, sal, skill)

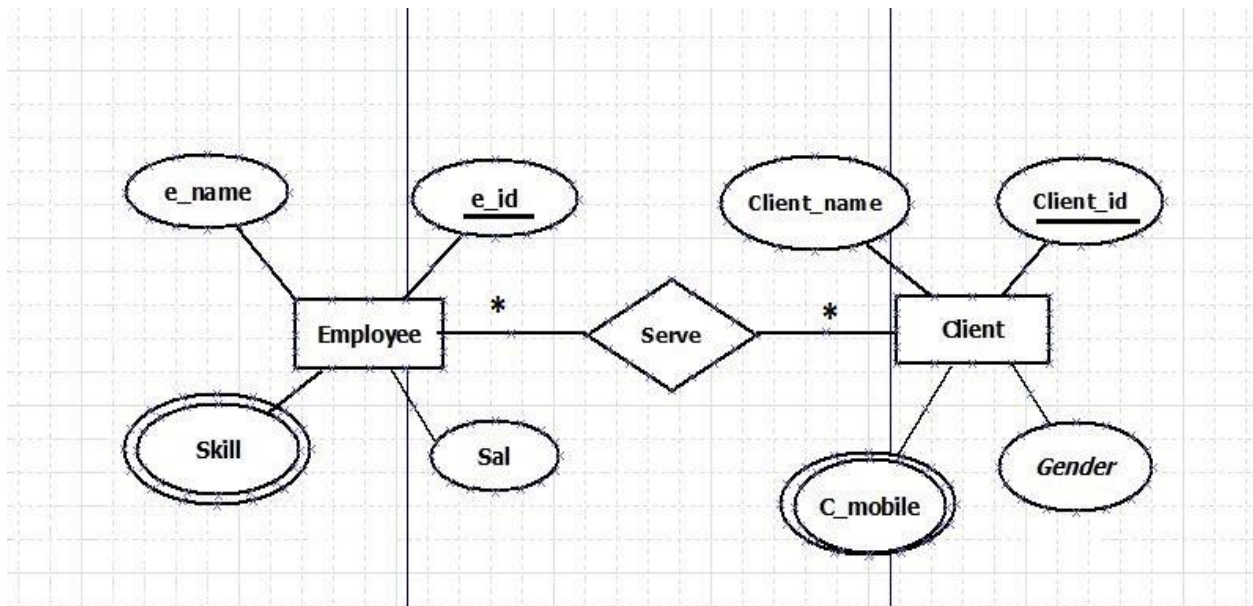
2NF:

(branch code, men\_s, women\_s)

(e id, e\_name, e\_mobile, sal, skill, Branch\_code )

3NF is as 2NF.

# Normalization:



UNF:

Serve(e\_id,e\_name,e\_mobile,sal,skill,c\_id,c\_name,c\_mobile,gender)

1NF:

(e\_id,e\_name,e\_mobile,sal,skill,c\_id,c\_name,c\_mobile,gender)

2NF:

(e\_id,e\_name,e\_mobile,sal,skill)

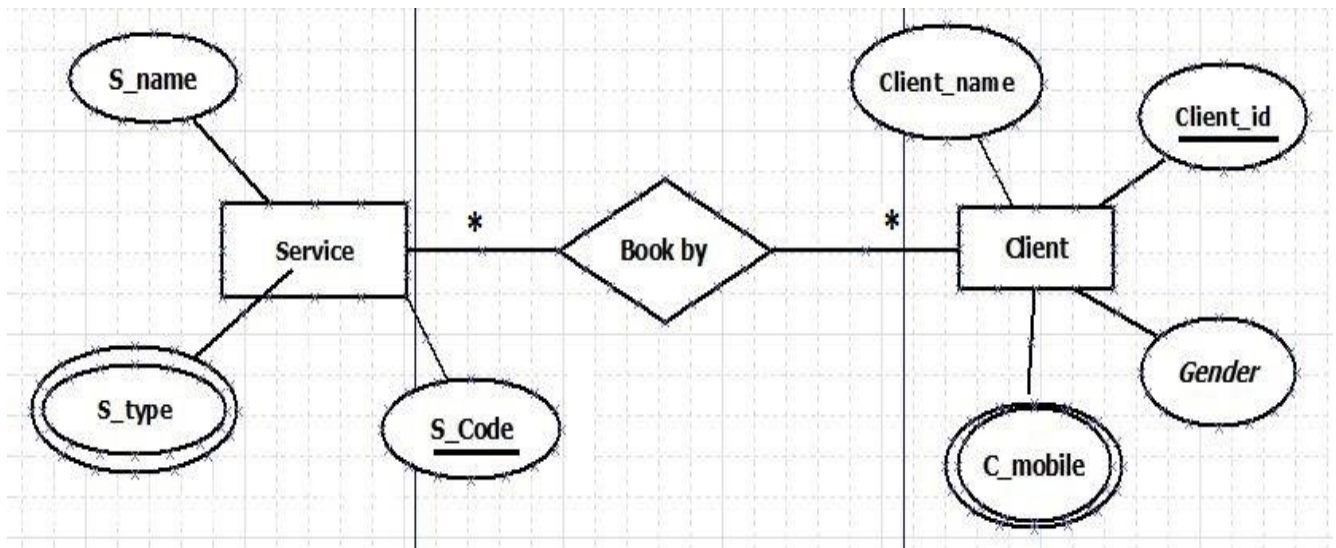
(c\_id,c\_name,c\_mobile,gender)

(e\_id, c\_id)

3NF is as same as 2NF.



# Normalization:



UNF:

Book-by(c\_id,c\_name,c\_mobile,gender,s\_code,s\_name,s\_type)

1NF:

(c\_id,c\_name,c\_mobile,gender,s\_code,s\_name,s\_type)

2NF:

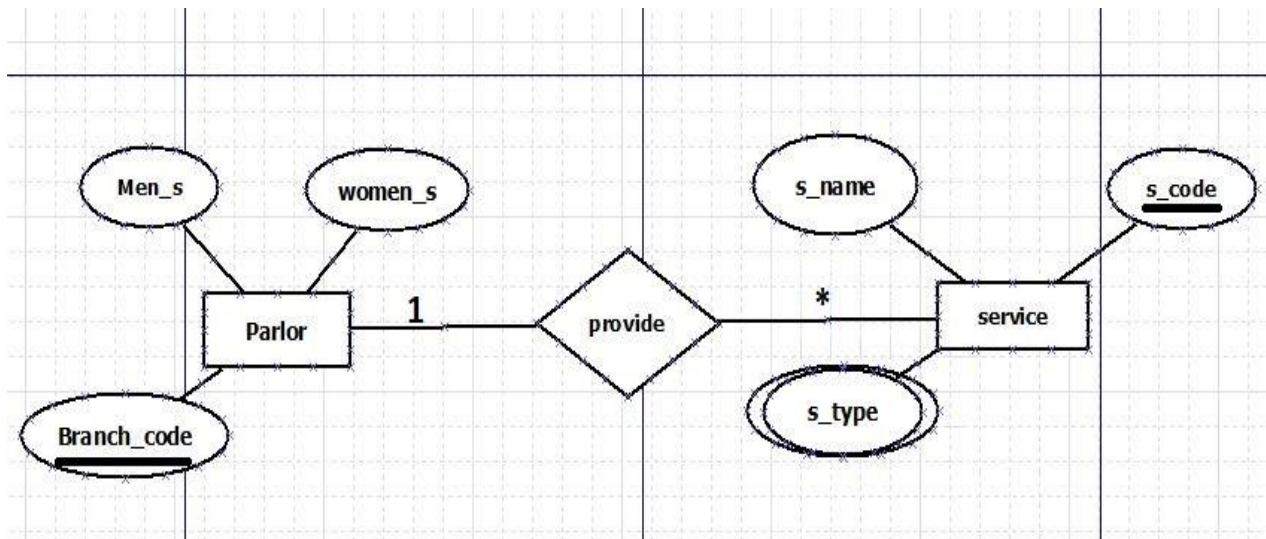
(c\_id,c\_name,c\_mobile,gender)

(s\_code,s\_name,s\_type)

(c\_id, s\_code)

3NF IS AS 2NF.

# Normalization:



UNF:

Provide( branch code , men\_s,women\_s, s\_name,s\_code, s\_type)

1NF:

( branch code , men\_s,women\_s, s\_name,s\_code, s\_type)





2NF:

( branch code , men\_s,women\_s)

(s\_name,s\_code, s\_type, Branch\_code )

3NF is as 2NF.

## FINALIZATION:

1. (branch\_code, men\_s, women\_s)
2. (O\_name, O\_city, )
3. (e\_id, e\_name, e\_mobile, sal, skill, )
4. (c\_id, c\_name, c\_mobile, gender)
5. (s\_name, s\_code, s\_type, )
6. (O\_city, O\_country)
7. (p\_code, p\_name, p\_type, discount, )
8. (e\_id, c\_id)
9. (c\_id, s\_code)

### **Table Creation:**

User: PARLOR

[Home](#) > [SQL](#) > [SQL Commands](#)

☒ Autocommit    **Display** 5000    ▾

```
create table parlor(branch_code varchar2(10) primary key, men_s varchar2(10),women_s varchar2(10));
```

```
desc parlor;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **PARLOR**[illegible]

### **Table Creation:**

User: PARLOR

[Home](#) > [SQL](#) > [SQL Commands](#)

☒ Autocommit    **Display** 100000 ▾

```
create table owners(o_name varchar(50), o_city varchar(50),branch_code varchar2(50),
CONSTRAINT ky FOREIGN KEY (branch_code)REFERENCES parlor(branch_code));
```

```
desc owners;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **OWNERS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OWNERS	O_NAME	Varchar2	50	-	-	-	✓	-	-
	O_CITY	Varchar2	50	-	-	-	✓	-	-
	BRANCH_CODE	Varchar2	50	-	-	-	✓	-	-
									1 - 3

### **Table Creation:**

User: PARLOR

[Home](#) > [SQL](#) > **SQL Commands**

☒ Autocommit    **Display** 100000 ▾

```
create table employee(e_id varchar2(50) primary key,e_name varchar(50), e_mobile varchar(50),
skill varchar(50),sal varchar(50), branch_code varchar2(50),
CONSTRAINT kk FOREIGN KEY (branch_code)REFERENCES parlor(branch_code));
```

```
desc employee;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **EMPLOYEE**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE	E_ID	Varchar2	50	-	-	1	-	-	-
	E_NAME	Varchar2	50	-	-	-	✓	-	-
	E_MOBILE	Varchar2	50	-	-	-	✓	-	-
	SKILL	Varchar2	50	-	-	-	✓	-	-
	SAL	Varchar2	50	-	-	-	✓	-	-
	BRANCH_CODE	Varchar2	50	-	-	-	✓	-	-
1 - 6									

## Table Creation:

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit   Display 100000 ▾

```
create table client(c_id varchar2(50) primary key, c_name varchar2(50),  
c_mobile varchar2(50),c_gender varchar2(50));
```

```
desc client;
```

**Results**   Explain   Describe   Saved SQL   History

Object Type **TABLE** Object **CLIENT**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
<u>CLIENT</u>	<u>C_ID</u>	Varchar2	50	-	-	1	-	-	-
	<u>C_NAME</u>	Varchar2	50	-	-	-	✓	-	-
	<u>C_MOBILE</u>	Varchar2	50	-	-	-	✓	-	-
	<u>C_GENDER</u>	Varchar2	50	-	-	-	✓	-	-

1 - 4

### **Table Creation:**

User: PARLOR

[Home](#) > [SQL](#) > **SQL Commands**

☒ Autocommit    **Display** 100000 ▾

```
create table provide(s_code varchar(50) primary key,s_name varchar(50),
s_type varchar(50),branch_code varchar2(50),
CONSTRAINT oc FOREIGN KEY (branch_code)REFERENCES parlor(branch_code));
```

```
desc provide;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **PROVIDE**[illegible]



### **Table Creation:**

User: PARLOR

[Home](#) > [SQL](#) > **SQL Commands**

☒ Autocommit    Display 100000 ▾

```
create table owner_info(o_city varchar2(50) primary key,o_country varchar2(50));
desc owner_info;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **OWNER\_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OWNER INFO	<u>O_CITY</u>	Varchar2	50	-	-	1	-	-	-
	<u>O_COUNTRY</u>	Varchar2	50	-	-	-	✓	-	-
									1 - 2

### **Table Creation:**

User: PARLOR

[Home](#) > [SQL](#) > [SQL Commands](#)

☒ Autocommit    **Display** 100000 ▾

```
create table offers(package_code varchar2(50) primary key,package_name varchar(50),
package_type varchar(50),discount varchar(50), branch_code varchar2(50),
CONSTRAINT ou FOREIGN KEY (branch_code)REFERENCES parlor(branch_code));

desc offers;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **OFFERS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OFFERS	PACKAGE_CODE	Varchar2	50	-	-	1	-	-	-
	PACKAGE_NAME	Varchar2	50	-	-	-	✓	-	-
	PACKAGE_TYPE	Varchar2	50	-	-	-	✓	-	-
	DISCOUNT	Varchar2	50	-	-	-	✓	-	-
	BRANCH_CODE	Varchar2	50	-	-	-	✓	-	-
1 - 5									

### **Table Creation:**

User: PARLOR

[Home](#) > [SQL](#) > [SQL Commands](#)

☒ Autocommit    **Display** 100000 ▾

```
create table emp_client(e_id number(35),c_id number(35),
constraint attach primary key(e_id,c_id));
```

```
desc emp_client;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **EMP\_CLIENT**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP_CLIENT	E_ID	Number	-	35	0	1	-	-	-
	C_ID	Number	-	35	0	2	-	-	-
1 - 2									

### **Table Creation:**

User: PARLOR

[Home](#) > [SQL](#) > [SQL Commands](#)

☒ Autocommit    **Display** 100000 ▾

```
create table client_service(c_id number(35),s_code number(35),
constraint attached primary key(c_id,s_code));
desc client_service;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **CLIENT\_SERVICE**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CLIENT_SERVICE	C_ID	Number	-	35	0	1	-	-	-
	S_CODE	Number	-	35	0	2	-	-	-
1 - 2									

## Data Insertion:

User: PARLOR

Home > SQL > SQL Commands

☒ Autocommit    Display 100000 ▾

```
insert into client values('1','Nishi','015*****','Female')
insert into client values('2','Mim','016*****','Female')
insert into client values('3','shompa','017*****','Female')
insert into client values('4','Nazat','018*****','Male')
insert into client values('5','Durjoy','019*****','Male')
```

```
select *from client;
```

**Results**   Explain   Describe   Saved SQL   History

C_ID	C_NAME	C_MOBILE	C_GENDER
1	Nishi	015*****	Female
2	Mim	016*****	Female
3	shompa	017*****	Female
4	Nazat	018*****	Male
5	Durjoy	019*****	Male

5 rows returned in 0.00 seconds

[CSV Export](#)

## **Data Insertion:**

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit    Display 100000 ▾

```
insert into emp_client values('101','1')
insert into emp_client values('102','2')
insert into emp_client values('103','3')
insert into emp_client values('104','4')
insert into emp_client values('105','5')|

select *from emp_client;
```

**Results**   Explain   Describe   Saved SQL   History

E_ID	C_ID
101	1
102	2
103	3
104	4
105	5

5 rows returned in 0.00 seconds

[CSV Export](#)

## Data Insertion:

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit   **Display** 100000 ▾

```
insert into employee values('101','Nishat','016*****','hair stylist','$500','P1023')
insert into employee values('102','Anika','017*****','Beauty Job','$800','P1023')
insert into employee values('103','Tanjila','018*****','Spa Jobs','$700','P1023')
insert into employee values('104','Tanisha','019*****','Beauty Therapist','$900','P1023')
insert into employee values('105','Urmila','015*****','Nails Jobs','$400','P1023')

select *from employee ;
```

**Results**   Explain   Describe   Saved SQL   History

E_ID	E_NAME	E_MOBILE	SKILL	SAL	BRANCH_CODE
101	Nishat	016*****	hair stylist	\$500	P1023
102	Anika	017*****	Beauty Job	\$800	P1023
103	Tanjila	018*****	Spa Jobs	\$700	P1023
104	Tanisha	019*****	Beauty Therapist	\$900	P1023
105	Urmila	015*****	Nails Jobs	\$400	P1023

5 rows returned in 0.00 seconds

[CSV Export](#)

## Data Insertion:

User: PARLOR

Home > SQL > SQL Commands

☒ Autocommit   Display   100000 ▾

```
insert into offers values('P101','Spa Summer Offers','Available','15%','P1023')
insert into offers values('P102','Hair Treatment','Regular client','35%','P1023')
insert into offers values('P103','FREE Colour Treatment for the month of April','Regular client',
'100%','P1023')
insert into offers values('P104','December Hair Straightening','Regular client','25%','P1023')
insert into offers values('P105','weekly Beauty Therapy','Regular client','30%','P1023')

select *from offers;
```

**Results**   Explain   Describe   Saved SQL   History

PACKAGE_CODE	PACKAGE_NAME	PACKAGE_TYPE	DISCOUNT	BRANCH_CODE
P101	Spa Summer Offers	Available	15%	P1023
P102	Hair Treatment	Regular client	35%	P1023
P103	FREE Colour Treatment for the month of April	Regular client	100%	P1023
P104	December Hair Straightening	Regular client	25%	P1023
P105	weekly Beauty Therapy	Regular client	30%	P1023

5 rows returned in 0.00 seconds

[CSV Export](#)



## Data Insertion:

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit    Display 100000 ▾

```
insert into owner_info values('Dhaka','Bangladesh')  
  
select *from owner_info;
```

**Results**   Explain   Describe   Saved SQL   History

O_CITY	O_COUNTRY
Dhaka	Bangladesh

1 rows returned in 0.00 seconds

[CSV Export](#)

## Data Insertion:

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit    Display

```
insert into owners values('Ananto Jalil','Dhaka','P1023')  
select *from  owners;
```

**Results**   Explain   Describe   Saved SQL   History

O_NAME	O_CITY	BRANCH_CODE
Ananto Jalil	Dhaka	P1023

1 rows returned in 0.00 seconds

[CSV Export](#)

## Data Insertion:

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit   **Display**  ▾

```
insert into parlor values('P1023','Available','Available')
|
select *from parlor;
```

**Results**   Explain   Describe   Saved SQL   History

BRANCH_CODE	MEN_S	WOMEN_S
P1023	Available	Available

1 rows returned in 0.00 seconds

[CSV Export](#)

## Data Insertion:

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit   Display

```
insert into provide values('201','Spa','Regular','P1023')
insert into provide values('202','Beauty Therapy','Regular & Home service','P1023')
insert into provide values('203','Nails','Regular','P1023')
insert into provide values('204','Hair cutting','Regular & Home service','P1023')
insert into provide values('205','beauty job','Regular','P1023')

select *from provide;
```

**Results**   Explain   Describe   Saved SQL   History

S_CODE	S_NAME	S_TYPE	BRANCH_CODE
201	Spa	Regular	P1023
202	Beauty Therapy	Regular & Home service	P1023
203	Nails	Regular	P1023
204	Hair cutting	Regular & Home service	P1023
205	beauty job	Regular	P1023

5 rows returned in 0.00 seconds

[CSV Export](#)

## **Data Insertion:**

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit    Display 100000 ▾

```
insert into client_service values('1','201')
insert into client_service values('2','202')
insert into client_service values('3','203')
insert into client_service values('4','204')
insert into client_service values('5','205')
```

```
select *from client_service;
```

**Results**   Explain   Describe   Saved SQL   History

C_ID	S_CODE
1	201
2	202
3	203
4	204
5	205

5 rows returned in 0.02 seconds

[CSV Export](#)

## Query Writing:

### Single row queries:

1. Display the package\_code, Package\_name,package\_type where discount is less than 35%.

User: PARLOR

Home > SQL > SQL Commands

☒ Autocommit   Display   100   ▾

```
select package_code, package_name, package_type
from offers
where discount < '15%';
```

**Results**   Explain   Describe   Saved SQL   History

PACKAGE_CODE	PACKAGE_NAME	PACKAGE_TYPE
P102	Hair Treatment	Regular client
P104	December Hair Straightening	Regular client
P105	weekly Beauty Therapy	Regular client

3 rows returned in 0.00 seconds

[CSV Export](#)

### Single row queries:

#### 2. Display the id and name of all the female client.

User: PARLOR

Home > SQL > SQL Commands

☒ Autocommit    Display 100 ▾

```
select c_id, c_name
from client
where c_gender='Female';|
```

**Results**   Explain   Describe   Saved SQL   History

C_ID	C_NAME
1	Nishi
2	Mim
3	shompa

3 rows returned in 0.01 seconds

[CSV Export](#)



### Group function query:

1. Display the name of the employees having the minimum salary of \$700.

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit    Display    100    ▼

```
select e_name, min(sal)
from employee
group by e_name
having min(sal) > '$700';
```

**Results**   Explain   Describe   Saved SQL   History

E_NAME	MIN(SAL)
Tanisha	\$900
Anika	\$800

2 rows returned in 0.01 seconds

[CSV Export](#)



**Group function query:**

2. Display the count show of the client who's name is "mim".

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit    Display  

```
select count(c_id)
from client
where c name='Mim';
```

**Results**   Explain   Describe   Saved SQL   Histor

COUNT(C_ID)
1

1 rows returned in 0.00 seconds

[CSV Export](#)

## Subquery:

1. Display the salary of the employees who's salary is greater than id of 105.

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit   Display   100   ▾

```
SELECT *FROM EMPLOYEE WHERE   SAL >(SELECT SAL FROM EMPLOYEE WHERE   E_ID='105');
```

**Results**   Explain   Describe   Saved SQL   History

E_ID	E_NAME	E_MOBILE	SKILL	SAL	BRANCH_CODE
101	Nishat	016*****	hair stylist	\$500	P1023
102	Anika	017*****	Beauty Job	\$800	P1023
103	Tanjila	018*****	Spa Jobs	\$700	P1023
104	Tanisha	019*****	Beauty Therapist	\$900	P1023

4 rows returned in 0.00 seconds

[CSV Export](#)

### Subquery:

2. Display the name and id of the client who's id is 5.

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit    Display    100    ▼

```
select c_id, c_name
from client
where c_id IN (5) ;
```

**Results**   Explain   Describe   Saved SQL   History

C_ID	C_NAME
5	Durjoy

1 rows returned in 0.00 seconds

[CSV Export](#)

### Subquery:

3. Display the name, skill and id of the employee who's salary is greater than all other employees.

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit    Display  ▾

```
SELECT e_id,e_name,skill FROM employee  
WHERE sal>=ALL(SELECT sal FROM employee);
```

**Results**   Explain   Describe   Saved SQL   History

E_ID	E_NAME	SKILL
104	Tanisha	Beauty Therapist

1 rows returned in 0.00 seconds

[CSV Export](#)

## Joining:

1. Join the employee name and client id with the employee id from employee and client table.

User: PARLOR

Home > SQL > SQL Commands

☒ Autocommit   Display 500

```
SELECT EMPLOYEE.E_ID,EMPLOYEE.E_NAME,  
EMP_CLIENT.E_ID,EMP_CLIENT.C_ID  
FROM EMPLOYEE,EMP_CLIENT  
WHERE EMPLOYEE.E ID=EMP_CLIENT.E ID;
```

**Results**   Explain   Describe   Saved SQL   History

E_ID	E_NAME	E_ID	C_ID
101	Nishat	101	1
102	Anika	102	2
103	Tanjila	103	3
104	Tanisha	104	4
105	Urmila	105	5

5 rows returned in 0.00 seconds   [CSV Export](#)

## Joining:

2. Join the service code with the client id and service type from the tables named provide and client service.

User: PARLOR

---

Home > SQL > **SQL Commands**

---

☒ Autocommit   **Display** 500 ▾

```
SELECT owners.branch_code,owners.o_name,  
parlor.branch_code,parlor.men_s, parlor.women_s  
FROM owners,parlor  
WHERE owners.branch_code=parlor.branch_code;
```

---

**Results**   Explain   Describe   Saved SQL   History

---

BRANCH_CODE	O_NAME	BRANCH_CODE	MEN_S	WOMEN_S
P1023	Ananto Jalil	P1023	Available	Available

1 rows returned in 0.00 seconds   [CSV Export](#)

## Joining:

3. Join the branch code with the owner name with the availability of men's section and female section.

User: PARLOR

Home > SQL > **SQL Commands**

☒ Autocommit    Display  ▾

```
SELECT provide.s_code,provide.s_type,  
CLIENT_service.s_code,CLIENT_service.C_ID  
FROM provide,CLIENT_service  
WHERE provide.s_code=CLIENT_service.s_code;
```

**Results**   Explain   Describe   Saved SQL   History

S_CODE	S_TYPE	S_CODE	C_ID
201	Regular	201	1
202	Regular & Home service	202	2
203	Regular	203	3
204	Regular & Home service	204	4
205	Regular	205	5

5 rows returned in 0.02 seconds

[CSV Export](#)



## **Conclusion:**

We have shown all the queries to create the tables in 'oracle 10g'. Also, we had shown the queries to insert the values and took their screen-shots. Here, we made 6 different relations among the entities.

The normalization process has made our work easier.

## **IN THE FUTURE:**

We can make the relational data base tables as a database management system as we have made this project. This job of ours can help a parlor in their data storing systems.