

**GYM MANAGEMENT SYSTEM**

**GROUP MEMBERS:**

**HASAN, MD. SIAM 14-27460-2**

**NOMAN, ABDULLAH AL 15-30873-3**

**HOSSAIN, MD. MANWAR 16-32092-2**

**FAHIM, SADMAN CHOWDHURY 16-32374-2**

**DAS, DIP 16-32482-2**

**COURSE NAME: INTRODUCTION TO DATA BASE**

**SECTION: I**

**FACULTY NAME: JUENA AHMED NOSHIN**

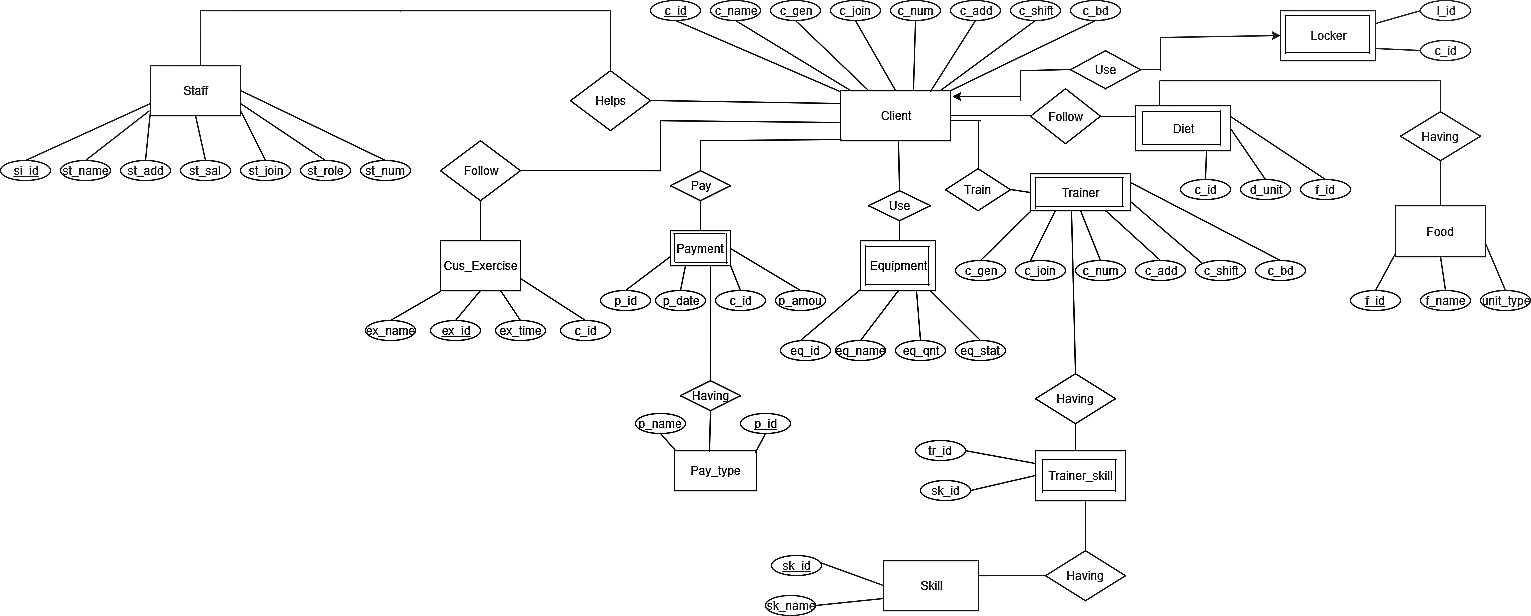
**Introduction:**

Our project is about a gym management system. We know in a gym there are various kinds of data that is need to keep safe. If we use a digital system for that then it will be very efficient. We know in a normal gym there are gym customer, staff, instrument etc. are exist. So in our data base we will keep the information about them. Additionally we will save the data about income and expense. So at last we hope that it will help a gym to maintain its activity efficiently.

**Scenario Description:**

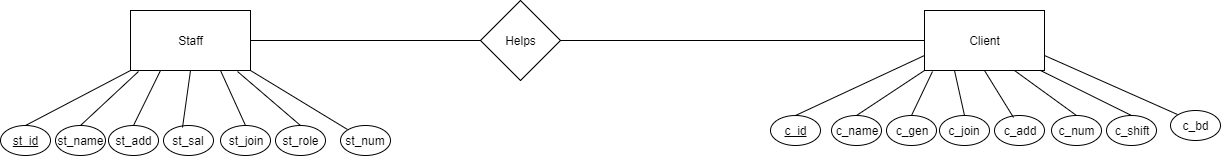
In our data base system we will work about the client, equipment, staffs, trainers, locker, food chart, payment and skill of that trainer. In the client section there are name, gender, joining date, shift, mobile number, birthday and a unique client id. In the staff section there are also staff name, address, salary, joining date, staff role, mobile number and a unique staff id. In the equipment section there are equipment name, equipment quantity, status and a unique equipment id. A client can use more than one equipment and an equipment can be used by more than one client. In the payment section there are payment is, date, amount and the customer id as a foreign key. In payment type there are payment name and a unique payment id. Payment can be paid by BKASH, CASH and CREDIT CARD. The unique payment id will be used as a foreign key in the Payment table. In the Customer exercise table have exercise name, time and a unique exercise id. The client id is the foreign key in this table. In the trainer section there are trainer name, mobile number, and address, joining date, shift and a unique trainer id. There are a Trainer skill table related to trainer table. A trainer can have more than one skill so in the table there will be trainer id and skill id. There will also be a skill table where skill name and a unique skill id will be available. In a quality gym locker system is important. In our data base there are a section for locker system which is containing locker id and client id. Locker table is related with client table and the client id is will be used as foreign key. A locker can be used by only one client. Diet routine is very important for the gym client so we keep a diet table in our data base. In the diet table there are three attributes which are diet id, unit and the client id is the foreign key. Food table is related with diet table which is containing food name, unit type and a unique food id. A staff or a trainer can help more than one client and a client can get help from more than one staff or trainer.

**ER Diagram:**



**Normalization:**

The Normalization have been done below from 1NF to 3NF.



**1NF**

Helps (c\_id, c\_name, c\_gen, c\_join, c\_add, c\_num, c\_shift, c\_bd, st\_id, st\_name, st\_add, st\_sal, st\_join, st\_role, st\_num)

**2NF**

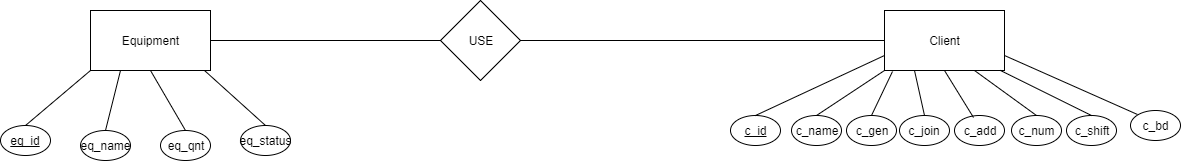
(c\_id, c\_name, c\_gen, c\_join, c\_add, c\_num, c\_shift, c\_bd)

(st\_id, st\_name, st\_add, st\_sal, st\_join, st\_role, st\_num)

(c\_id, st\_id)

**3NF**

Already in 2NF



**1NF**

Use (c\_id, c\_name, c\_gen, c\_join, c\_add, c\_num, c\_shift, c\_bd, eq\_id, eq\_name, eq\_qnt, eq\_status)

**2NF**

(c\_id, c\_name, c\_gen, c\_join, c\_add, c\_num, c\_shift, c\_bd)

(eq\_id, eq\_name, eq\_qnt, eq\_status)

(c\_id, eq\_id)

**3NF**

Already in 2NF

**Final Normalization:**

**#1NF:**

Client (c\_id, c\_name, c\_gender, c\_join, c\_num, c-add, c\_shift, c\_bd)

Staff (st\_id, st\_name, st\_add, st\_salary, st\_join, st\_role, st\_num)

Trainer (t\_id, t\_name, t\_num, t\_add, t\_join, t\_shift)   
  
**#2NF:**

Client (c\_id, c\_name, c\_gender, c\_join, c\_num, c-add, c\_shift, c\_bd)

Staff (st\_id, st\_name, st\_add, st\_salary, st\_join, st\_role, st\_num)

Trainer (t\_id, t\_name, t\_num, t\_add, t\_join, t\_shift)

Payment (p\_id, p\_date, c\_id, p\_amount)

Cus\_Exercise (ex\_id, ex\_name, cex\_time, c\_id)

Equipment (eq\_id, eq\_name, eq\_qnt, eq\_status)

Trainer\_skill (tr\_id, sk\_id)  
Diet (c\_id, d\_unit, d\_id)  
  
**#3NF:**Client (c\_id, c\_name ,c\_gender, c\_join, c\_num, c-add, c\_shift, c\_bd)

Staff (st\_id, st\_name, st\_add, st\_salary, st\_join, st\_role, st\_num)

Trainer (t\_id, t\_name, t\_num, t\_add, t\_join, t\_shift)

Payment (p\_id, p\_date, c\_id, p\_amount)

Payment\_type (p\_id, p\_name)

Cus\_exer (ex\_id, ex\_time, ex\_\_name, c\_id)

Equipment (eq\_id, eq\_name, eq\_qnt, eq\_status)

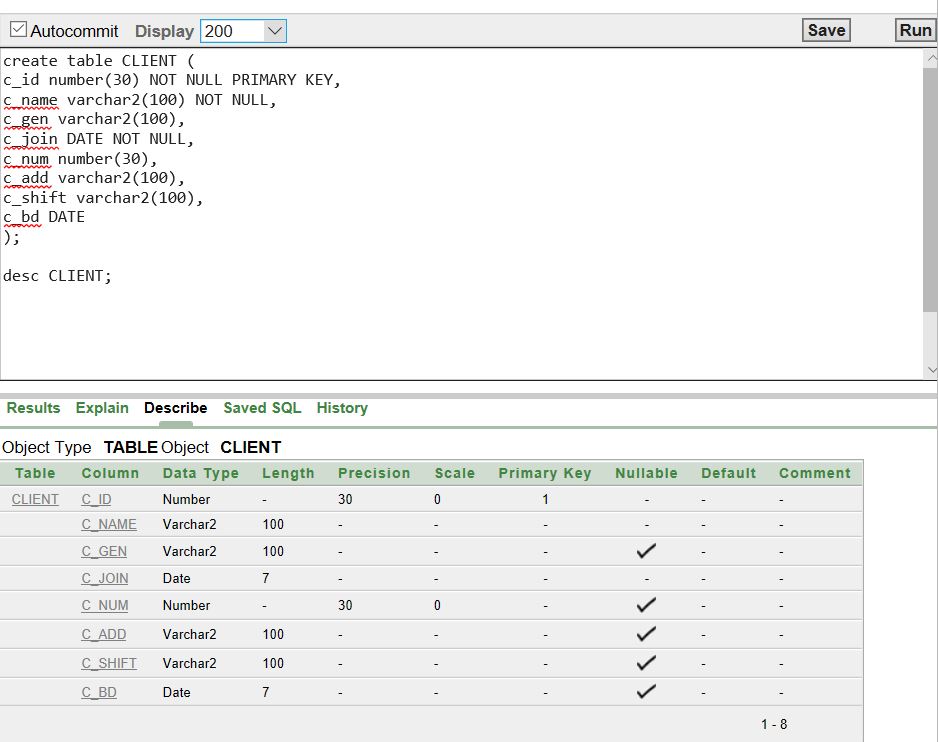
Trainer\_skill (tr\_id, sk\_id)   
Diet (c\_id, d\_unit, d\_id)

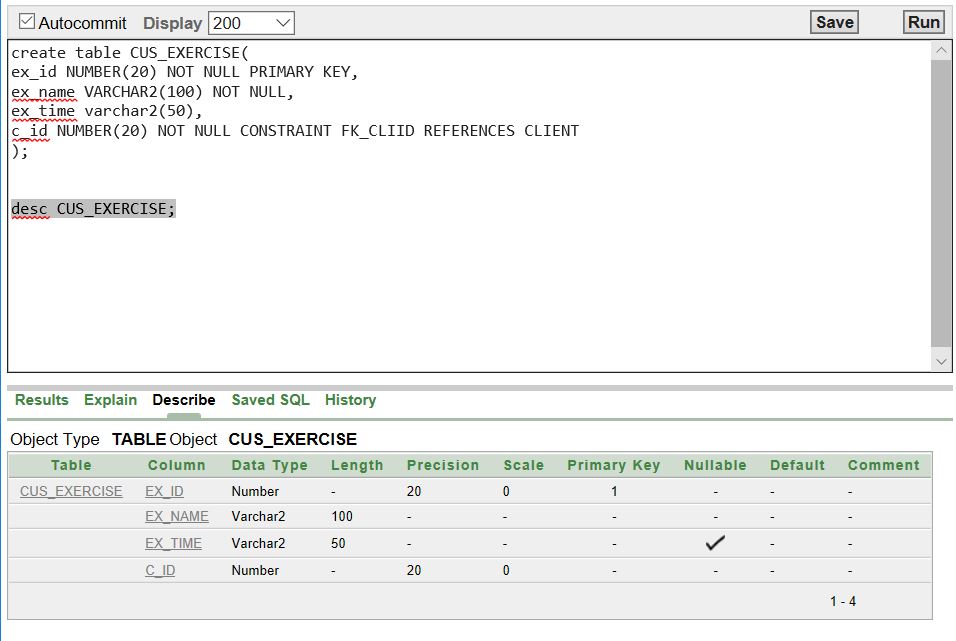
Skill (sk\_id, sk\_name)   
Locker (l\_id, c\_id)  
Food (f\_id, f\_name, unit\_type)

**Table Creation**

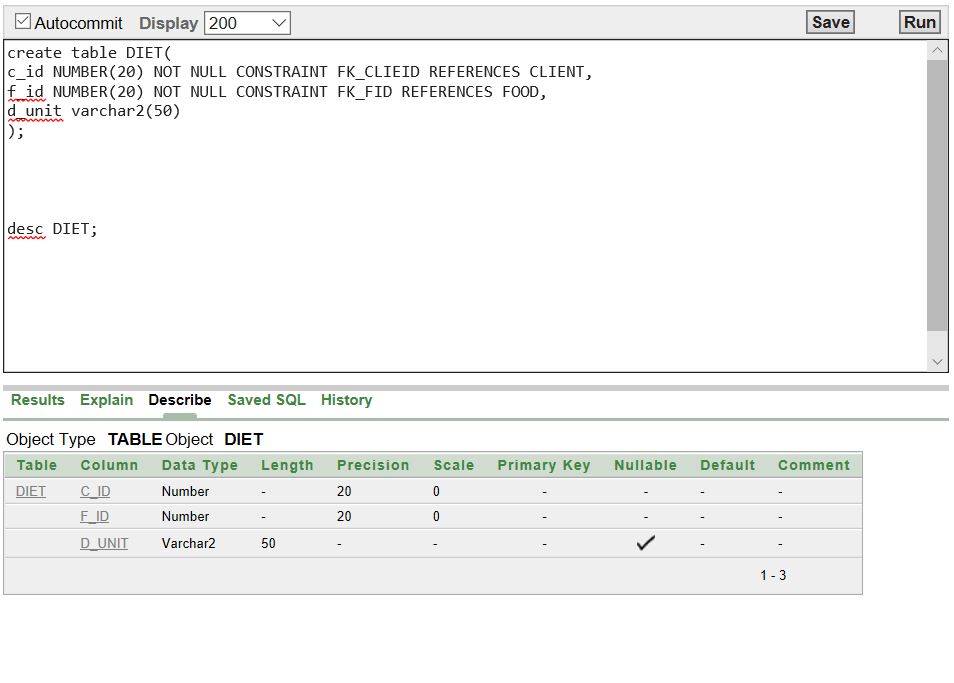
Here the screen shoot of tables for this project.

**Client Table:**

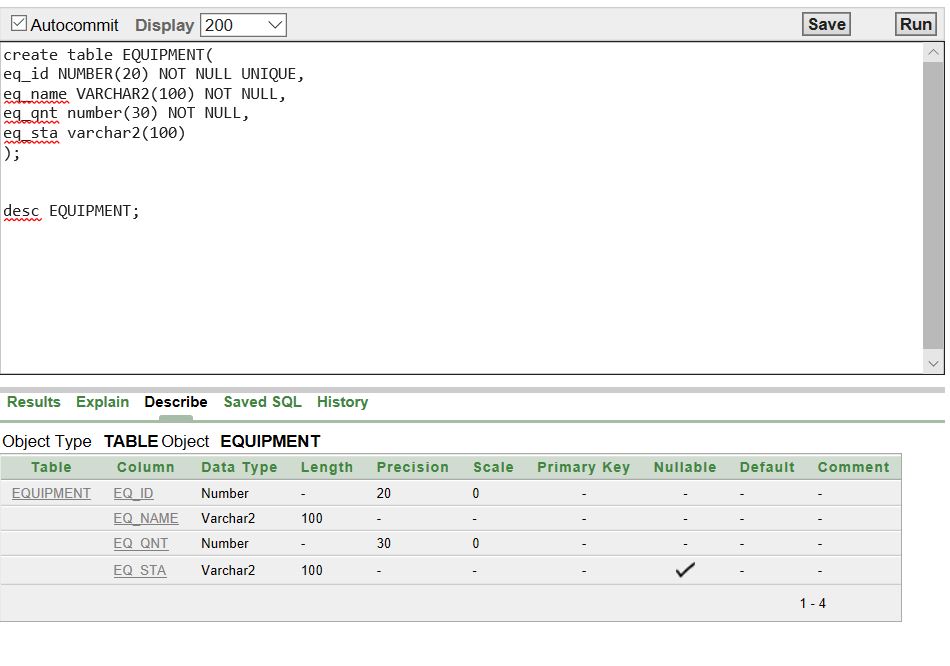
****

**Cus\_Exercise Table:** ****

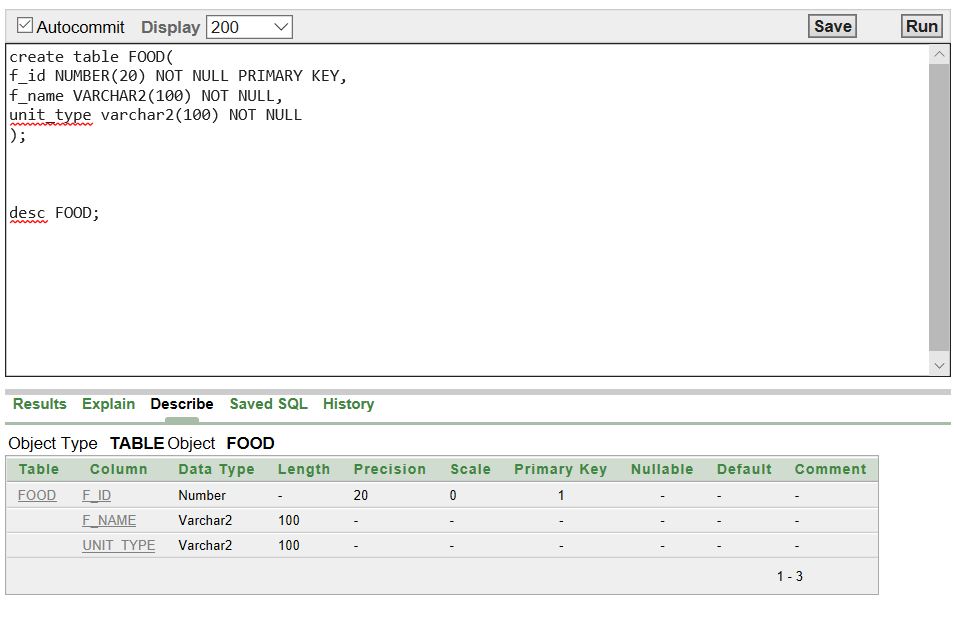
**Diet Table:**

****

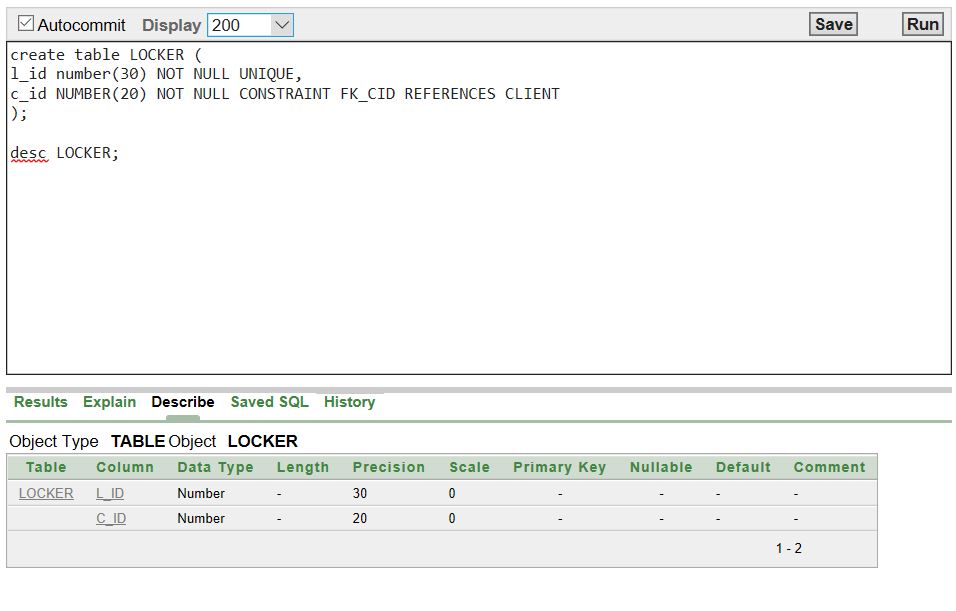
**Equipment Table:**

****

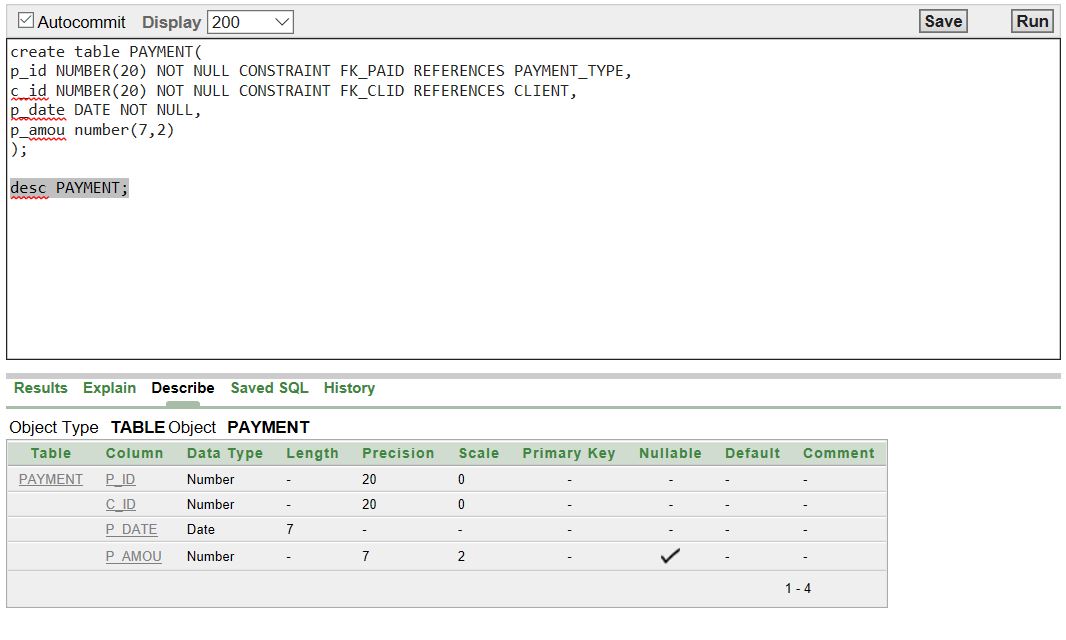
**Food Table:**

****

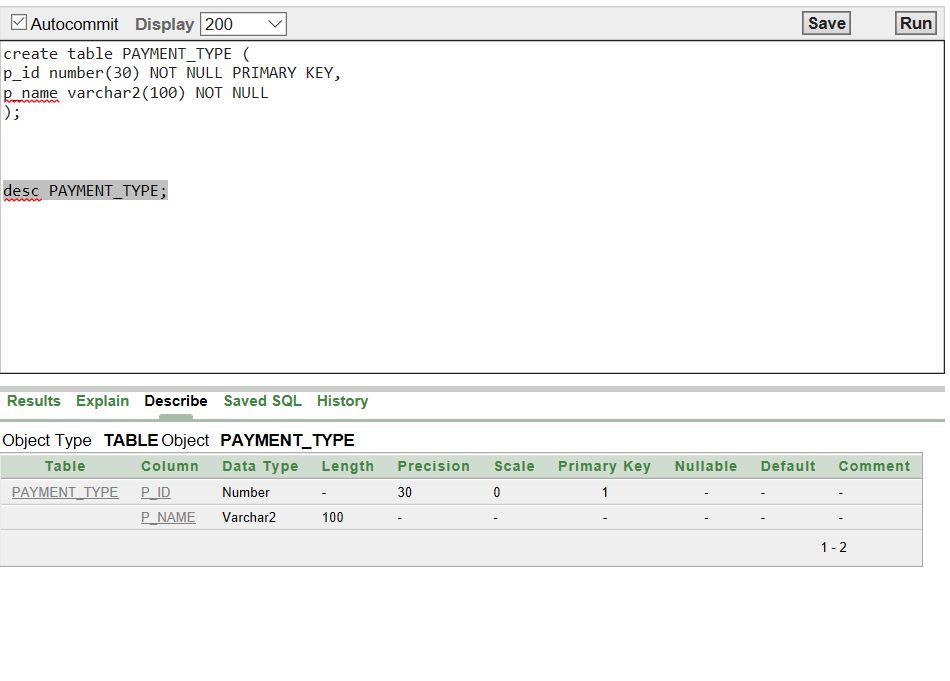
**Locker Table:**

****

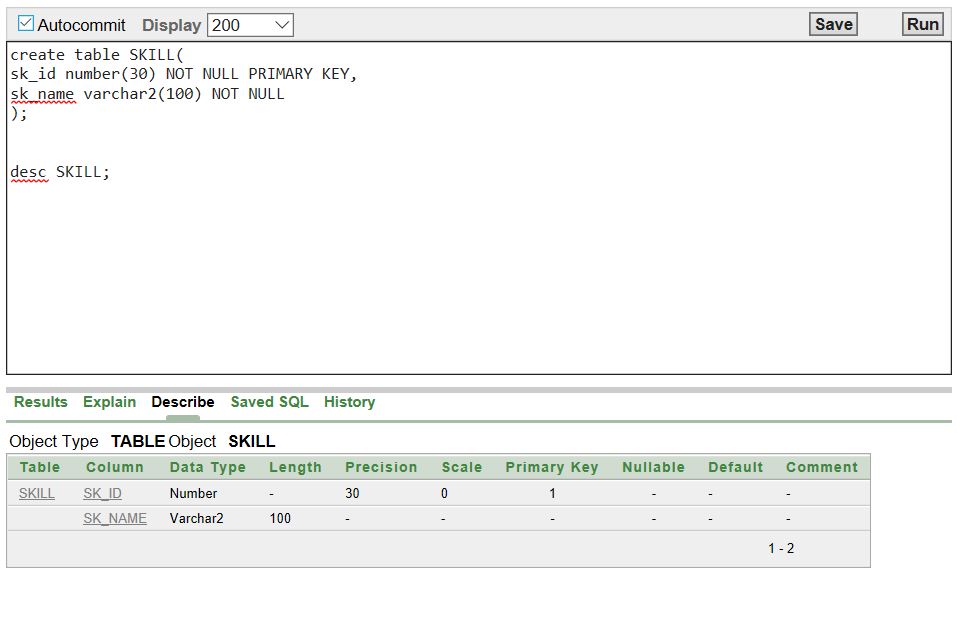
**Payment Table:**

****

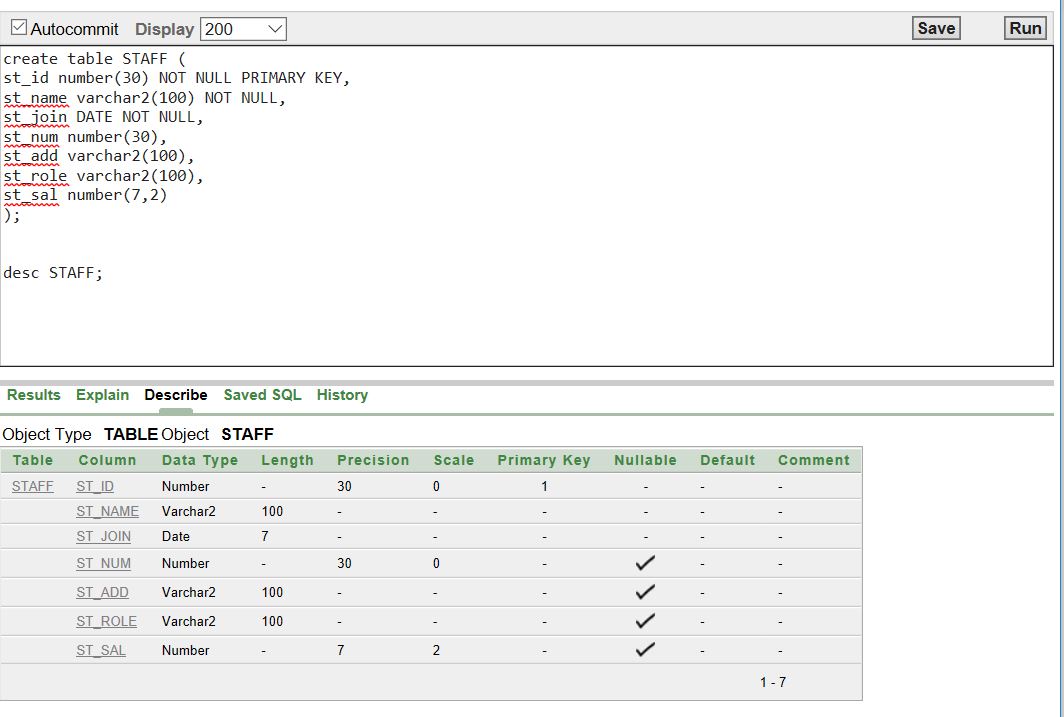
**Payment\_Type Table:**

****

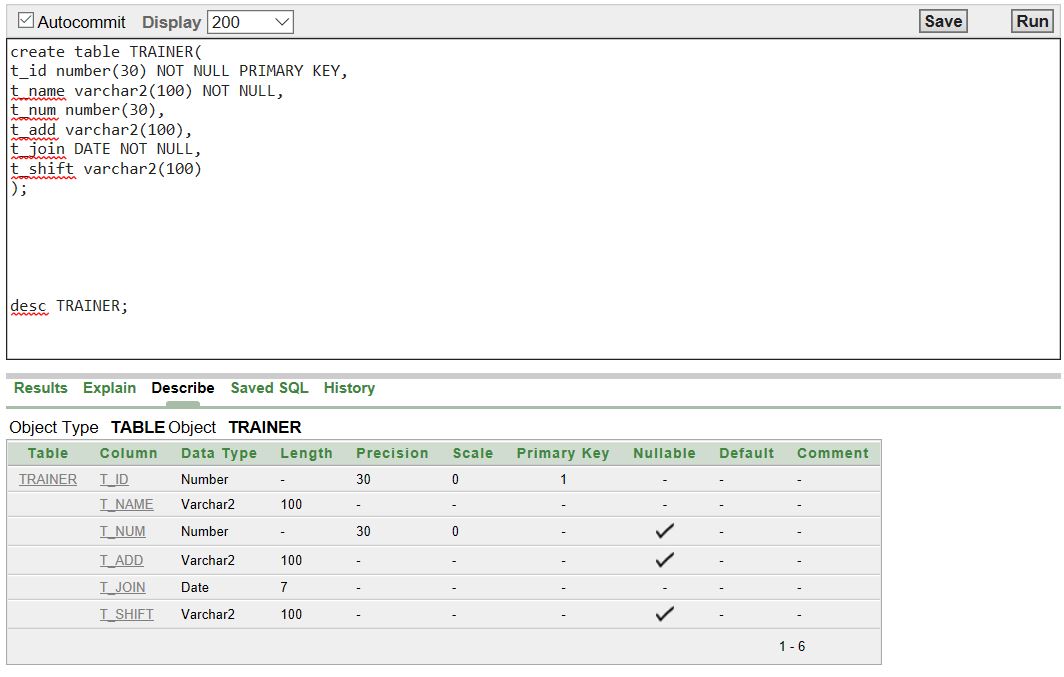
**Skill Table:**

****

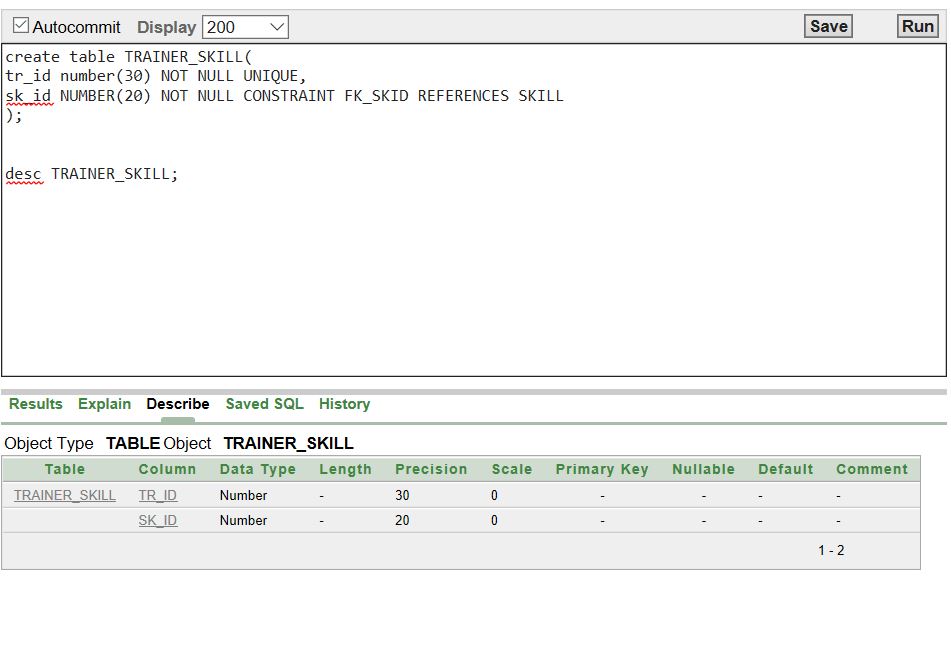
**Staff Table:**

****

**Trainer Table:**

****

**Trainer\_Skill Table:**

****

Here the qureis of those tables:

**# CLIENT Table**

create table CLIENT (

c\_id number(30) NOT NULL PRIMARY KEY,

c\_name varchar2(100) NOT NULL,

c\_gen varchar2(100),

c\_join DATE NOT NULL,

c\_num number(30),

c\_add varchar2(100),

c\_shift varchar2(100),

c\_bd DATE);

**# STAFF TABLE**

create table STAFF (

st\_id number(30) NOT NULL PRIMARY KEY,

st\_name varchar2(100) NOT NULL,

st\_join DATE NOT NULL,

st\_num number(30),

st\_add varchar2(100),

st\_role varchar2(100),

st\_sal number(7,2));

**# LOCKER TABLE**

create table LOCKER (

l\_id number(30) NOT NULL UNIQUE,

c\_id NUMBER(20) NOT NULL CONSTRAINT FK\_CID REFERENCES CLIENT);

**#TRAINER TABLE**

create table TRAINER(

t\_id number(30) NOT NULL PRIMARY KEY,

t\_name varchar2(100) NOT NULL,

t\_num number(30),

t\_add varchar2(100),

t\_join DATE NOT NULL,

t\_shift varchar2(100));

**#SKILL Table**

create table SKILL(

sk\_id number(30) NOT NULL PRIMARY KEY,

sk\_name varchar2(100) NOT NULL);

**#Trainer\_Skill Table**

create table TRAINER\_SKILL(

tr\_id number(30) NOT NULL UNIQUE,

sk\_id NUMBER(20) NOT NULL CONSTRAINT FK\_SKID REFERENCES SKIL);

**#PAYMENT\_TYEP Table**

create table PAYMENT\_TYPE (

p\_id number(30) NOT NULL PRIMARY KEY,

p\_name varchar2(100) NOT NULL);

**#PAYMENT Table**

create table PAYMENT(

p\_id NUMBER(20) NOT NULL CONSTRAINT FK\_PAID REFERENCES PAYMENT\_TYPE,

c\_id NUMBER(20) NOT NULL CONSTRAINT FK\_CLID REFERENCES CLIENT,

p\_date DATE NOT NULL,

p\_amou number(7,2) );

**#EQUIPMENT Table**

create table EQUIPMENT(

eq\_id NUMBER(20) NOT NULL UNIQUE,

eq\_name VARCHAR2(100) NOT NULL,

eq\_qnt number(30) NOT NULL,

eq\_sta varchar2(100) );

**#CUS\_EXERCISE Table**

create table CUS\_EXERCISE(

ex\_id NUMBER(20) NOT NULL PRIMARY KEY,

ex\_name VARCHAR2(100) NOT NULL,

ex\_time varchar2(50),

c\_id NUMBER(20) NOT NULL CONSTRAINT FK\_CLIID REFERENCES CLIENT);

**#FOOD Table**

create table FOOD(

f\_id NUMBER(20) NOT NULL PRIMARY KEY,

f\_name VARCHAR2(100) NOT NULL,

unit\_type varchar2(100) NOT NULL);

**#DIET Table**

create table DIET(

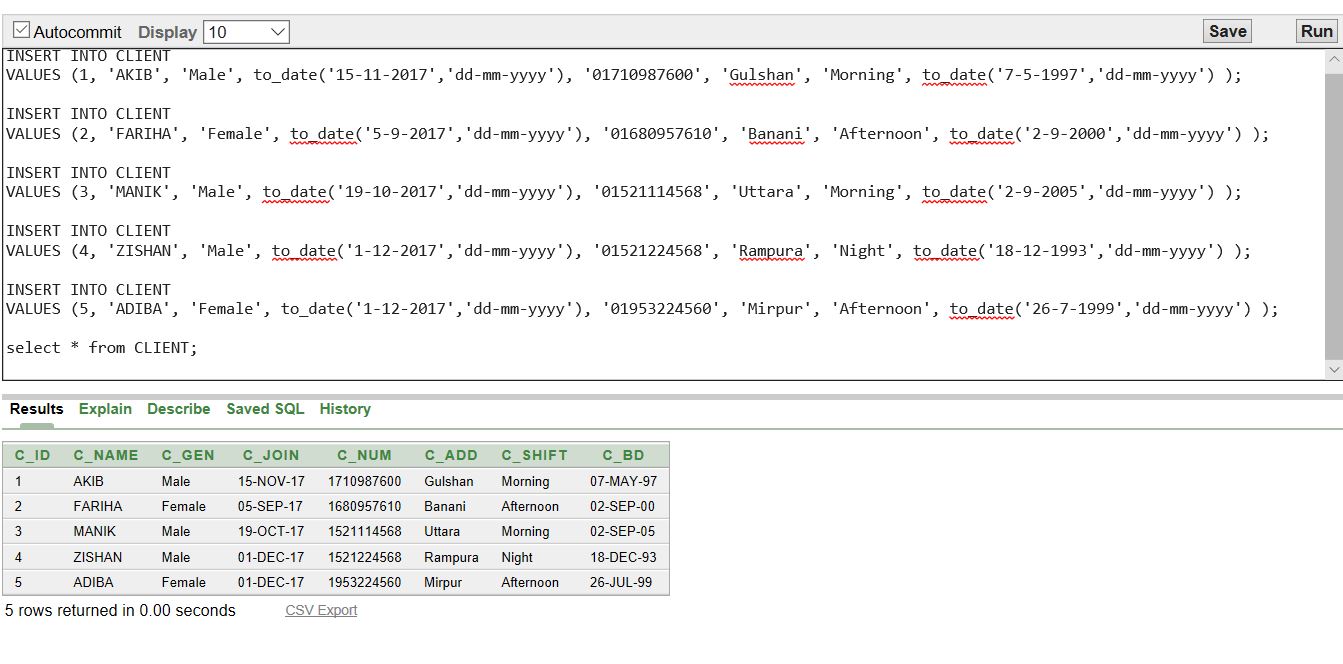
c\_id NUMBER(20) NOT NULL CONSTRAINT FK\_CLIEID REFERENCES CLIENT,

f\_id NUMBER(20) NOT NULL CONSTRAINT FK\_FID REFERENCES FOOD,

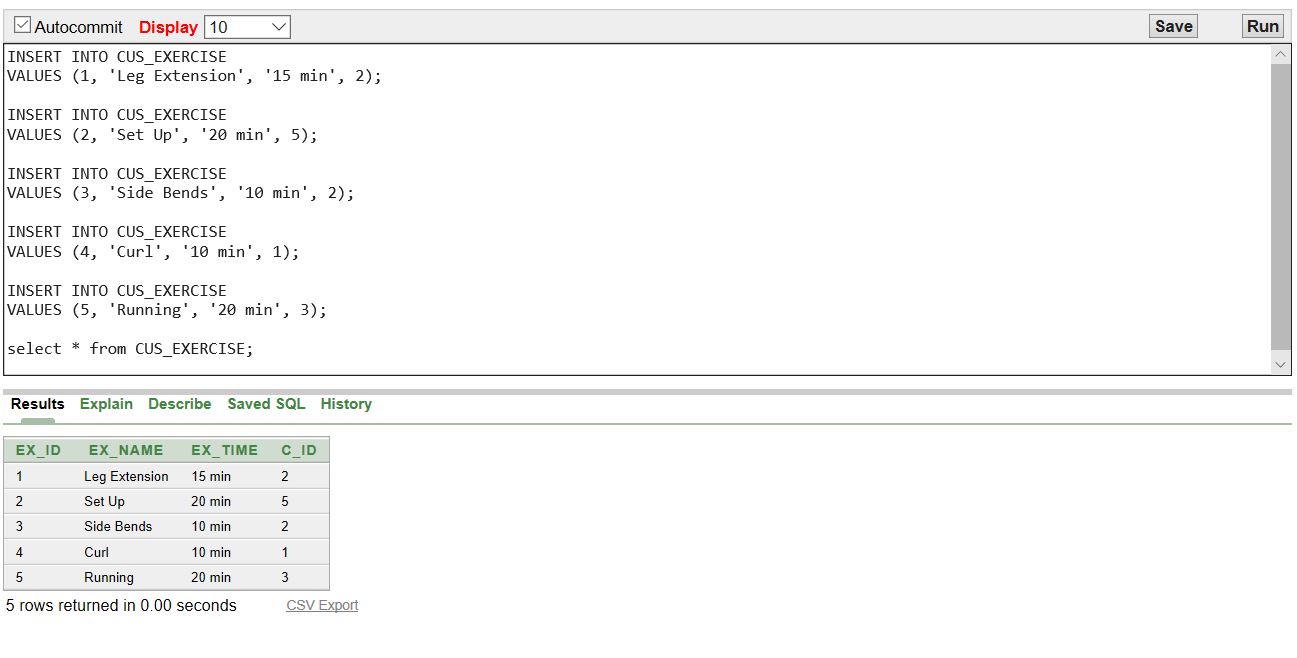
d\_unit varchar2(50));

**Data Insertion**

**Client Table:**



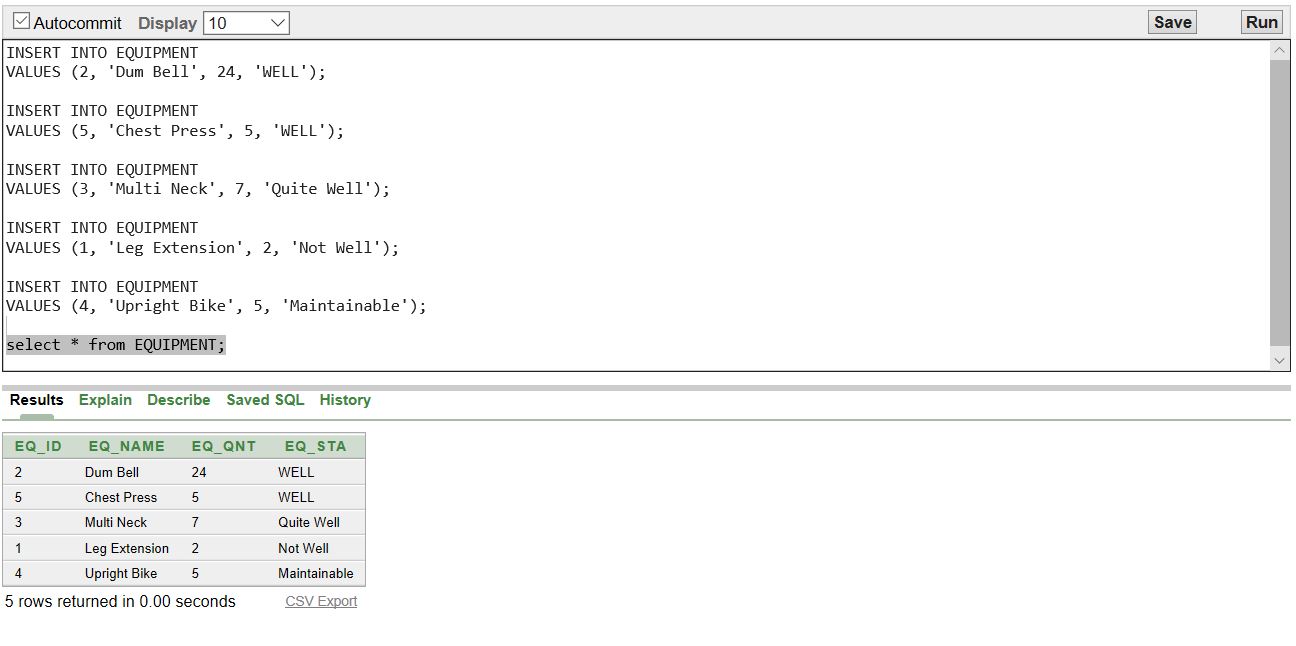
**Cus\_Exercise Table:**



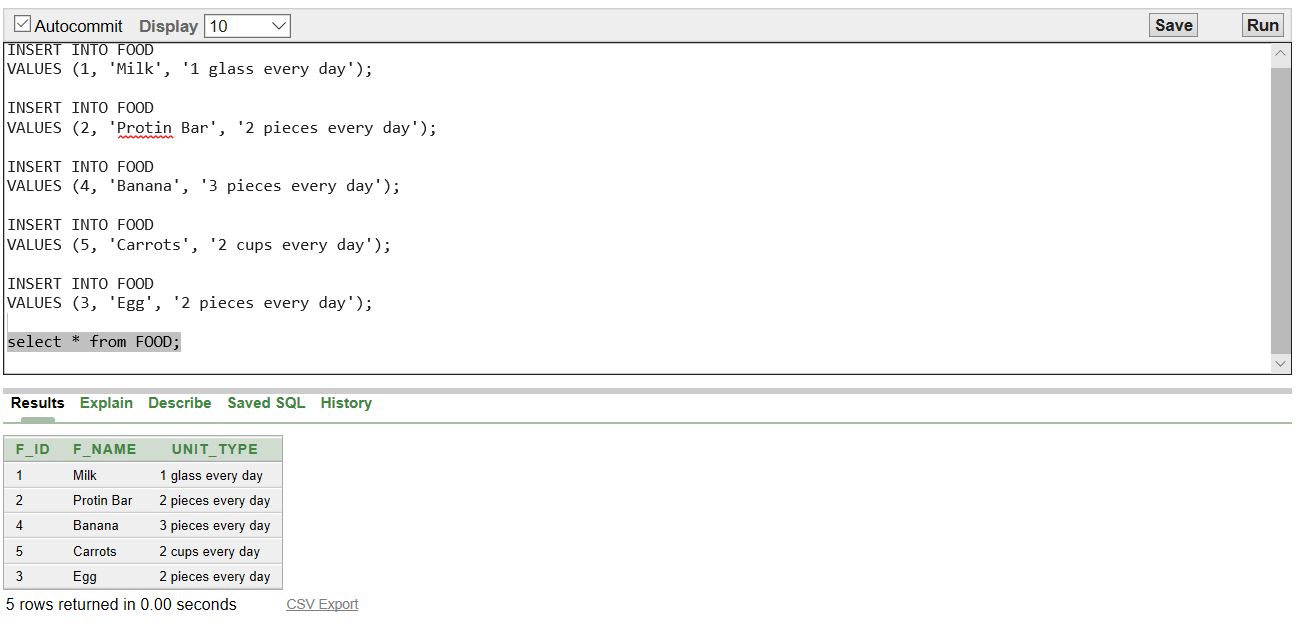
**Diet Table:**



**Equipment Table:**



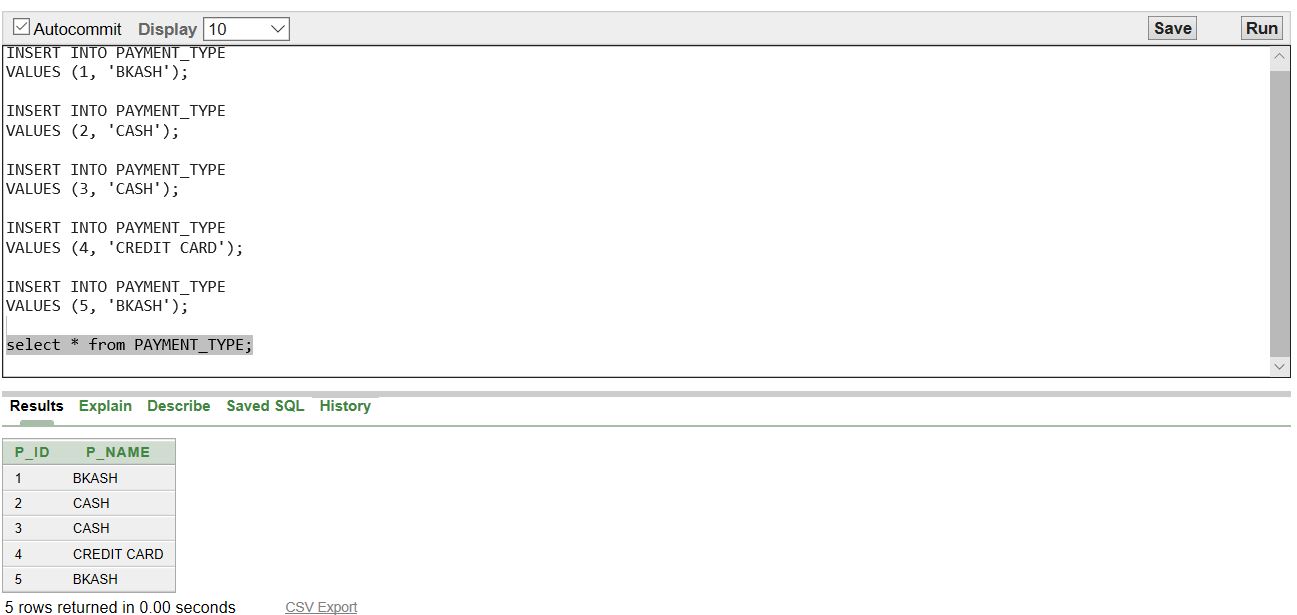
**Food Table:**

****

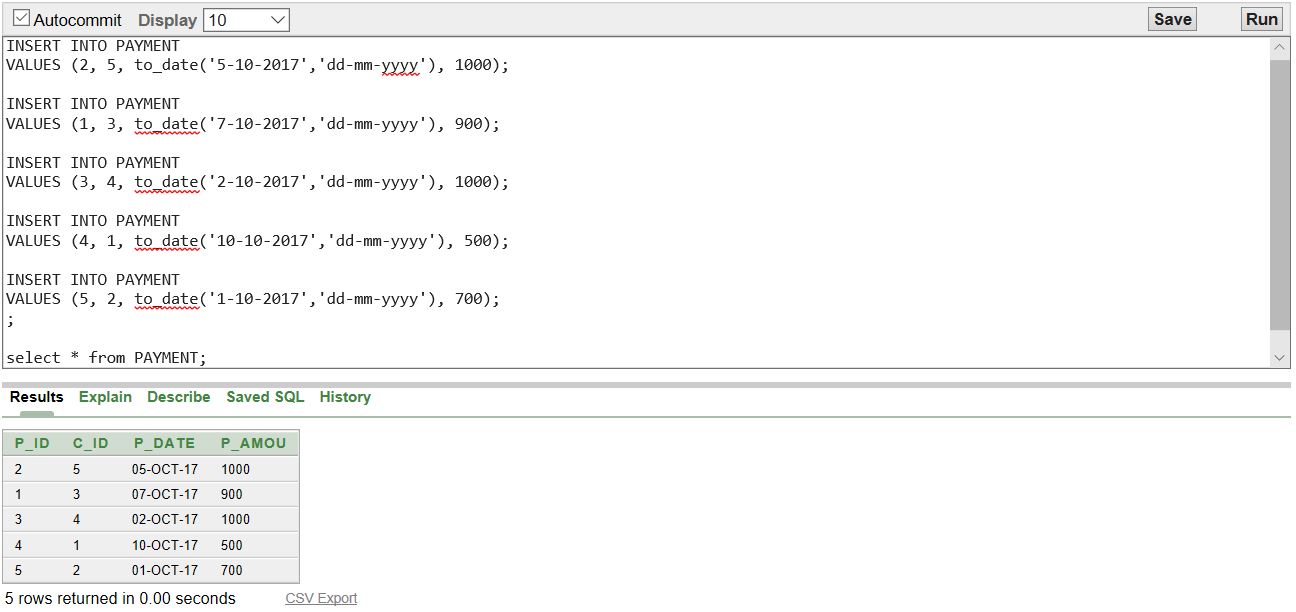
**Locker Table:**



**Payment\_Type Table:**



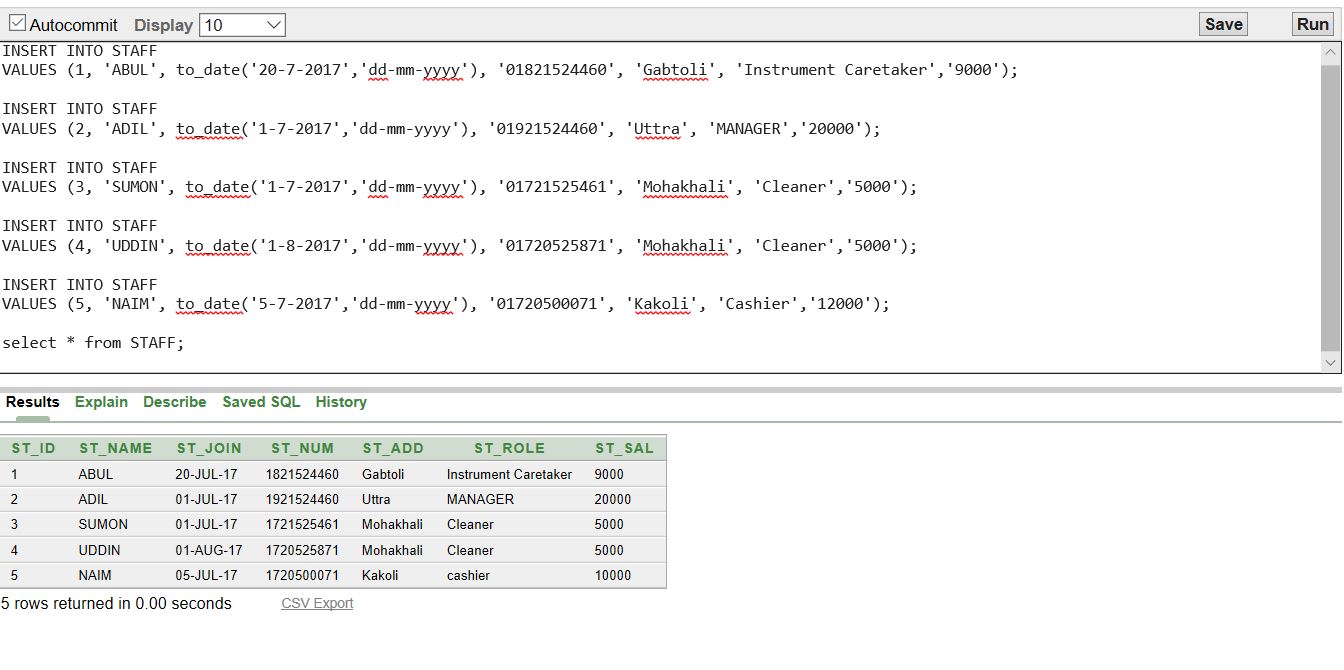
**Payment Table:**

****

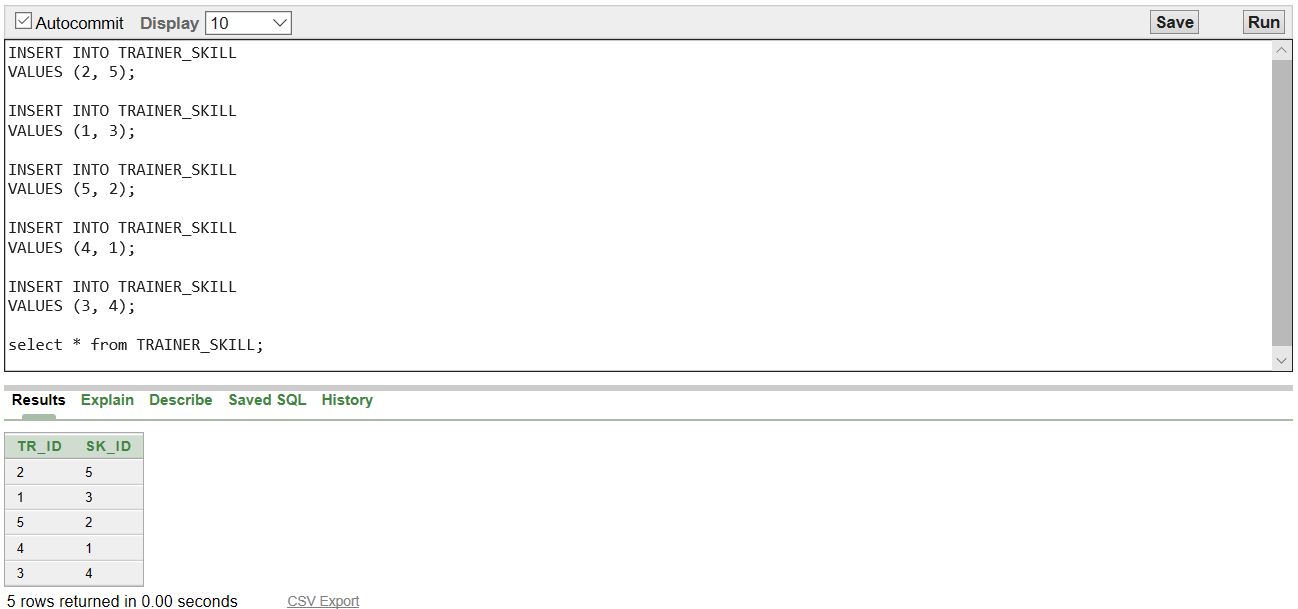
**Skill Table:**

****

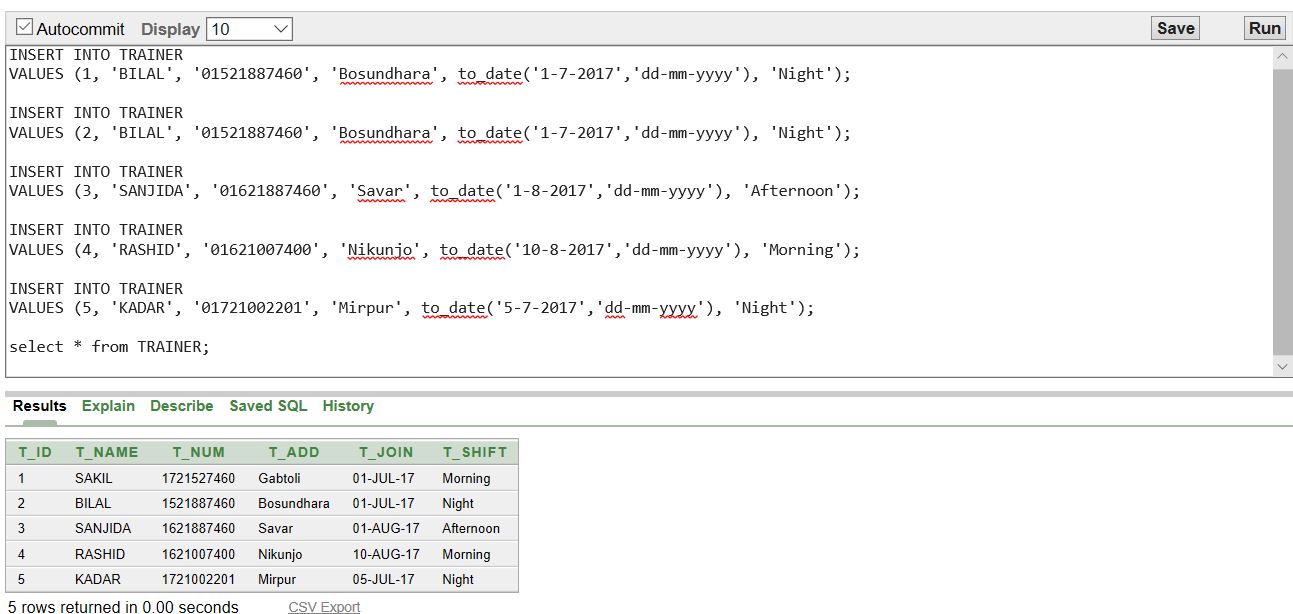
**Staff Table:**

****

**Trainer\_Skill Table:**

****

**Trainer Table:**

****

Here are the Insert queries for those tables:

**# Insert client**

INSERT INTO CLIENT

VALUES (1, 'AKIB', 'Male', to\_date('15-11-2017','dd-mm-yyyy'), '01710987600', 'Gulshan', 'Morning', to\_date('7-5-1997','dd-mm-yyyy') );

INSERT INTO CLIENT

VALUES (2, 'FARIHA', 'Female', to\_date('5-9-2017','dd-mm-yyyy'), '01680957610', 'Banani', 'Afternoon', to\_date('2-9-2000','dd-mm-yyyy') );

INSERT INTO CLIENT

VALUES (3, 'MANIK', 'Male', to\_date('19-10-2017','dd-mm-yyyy'), '01521114568', 'Uttara', 'Morning', to\_date('2-9-2005','dd-mm-yyyy') );

INSERT INTO CLIENT

VALUES (4, 'ZISHAN', 'Male', to\_date('1-12-2017','dd-mm-yyyy'), '01521224568', 'Rampura', 'Night', to\_date('18-12-1993','dd-mm-yyyy') );

INSERT INTO CLIENT

VALUES (5, 'ADIBA', 'Female', to\_date('1-12-2017','dd-mm-yyyy'), '01953224560', 'Mirpur', 'Afternoon', to\_date('26-7-1999','dd-mm-yyyy') );

**#Insert Staff**

INSERT INTO STAFF

VALUES (1, 'ABUL', to\_date('20-7-2017','dd-mm-yyyy'), '01821524460', 'Gabtoli', 'Instrument Caretaker','9000');

INSERT INTO STAFF

VALUES (2, 'ADIL', to\_date('1-7-2017','dd-mm-yyyy'), '01921524460', 'Uttra', 'MANAGER','20000');

INSERT INTO STAFF

VALUES (3, 'SUMON', to\_date('1-7-2017','dd-mm-yyyy'), '01721525461', 'Mohakhali', 'Cleaner','5000');

INSERT INTO STAFF

VALUES (4, 'UDDIN', to\_date('1-8-2017','dd-mm-yyyy'), '01720525871', 'Mohakhali', 'Cleaner','5000');

INSERT INTO STAFF

VALUES (5, 'NAIM', to\_date('5-7-2017','dd-mm-yyyy'), '01720500071', 'Kakoli', 'Cashier','12000');

**#Insert LOCKER**

INSERT INTO LOCKER

VALUES (1, 1);

INSERT INTO LOCKER

VALUES (3, 2);

INSERT INTO LOCKER

VALUES (5, 3);

INSERT INTO LOCKER

VALUES (2, 4);

INSERT INTO LOCKER

VALUES (4, 5);

**#Insert Trainer**

INSERT INTO TRAINER

VALUES (1, 'BILAL', '01521887460', 'Bosundhara', to\_date('1-7-2017','dd-mm-yyyy'), 'Night');

INSERT INTO TRAINER

VALUES (2, 'BILAL', '01521887460', 'Bosundhara', to\_date('1-7-2017','dd-mm-yyyy'), 'Night');

INSERT INTO TRAINER

VALUES (3, 'SANJIDA', '01621887460', 'Savar', to\_date('1-8-2017','dd-mm-yyyy'), 'Afternoon');

INSERT INTO TRAINER

VALUES (4, 'RASHID', '01621007400', 'Nikunjo', to\_date('10-8-2017','dd-mm-yyyy'), 'Morning');

INSERT INTO TRAINER

VALUES (5, 'KADAR', '01721002201', 'Mirpur', to\_date('5-7-2017','dd-mm-yyyy'), 'Night');

**#Insert Skill**

INSERT INTO SKILL

VALUES (1, 'Plank');

INSERT INTO SKILL

VALUES (2, 'Split');

INSERT INTO SKILL

VALUES (3, 'Bridge');

INSERT INTO SKILL

VALUES (4, 'Dazzler');

INSERT INTO SKILL

VALUES (5, 'Air Chair');

**#Insert Trainer\_Skill**

INSERT INTO TRAINER\_SKILL

VALUES (2, 5);

INSERT INTO TRAINER\_SKILL

VALUES (1, 3);

INSERT INTO TRAINER\_SKILL

VALUES (5, 2);

INSERT INTO TRAINER\_SKILL

VALUES (4, 1);

INSERT INTO TRAINER\_SKILL

VALUES (3, 4);

**#Insert Payment\_type**

INSERT INTO PAYMENT\_TYPE

VALUES (1, 'BKASH');

INSERT INTO PAYMENT\_TYPE

VALUES (2, 'CASH');

INSERT INTO PAYMENT\_TYPE

VALUES (3, 'CASH');

INSERT INTO PAYMENT\_TYPE

VALUES (4, 'CREDIT CARD');

INSERT INTO PAYMENT\_TYPE

VALUES (5, 'BKASH');

**#Insert Payment**

INSERT INTO PAYMENT

VALUES (2, 5, to\_date('5-10-2017','dd-mm-yyyy'), 1000);

INSERT INTO PAYMENT

VALUES (1, 3, to\_date('7-10-2017','dd-mm-yyyy'), 900);

INSERT INTO PAYMENT

VALUES (3, 4, to\_date('2-10-2017','dd-mm-yyyy'), 1000);

INSERT INTO PAYMENT

VALUES (4, 1, to\_date('10-10-2017','dd-mm-yyyy'), 500);

INSERT INTO PAYMENT

VALUES (5, 2, to\_date('1-10-2017','dd-mm-yyyy'), 700);

**#Insert Equipment**

INSERT INTO EQUIPMENT

VALUES (2, 'Dum Bell', 24, 'WELL');

INSERT INTO EQUIPMENT

VALUES (5, 'Chest Press', 5, 'WELL');

INSERT INTO EQUIPMENT

VALUES (3, 'Multi Neck', 7, 'Quite Well');

INSERT INTO EQUIPMENT

VALUES (1, 'Leg Extension', 2, 'Not Well');

INSERT INTO EQUIPMENT

VALUES (4, 'Upright Bike', 5, 'Maintainable');

**#Insert CUS\_EXERCISE**

INSERT INTO CUS\_EXERCISE

VALUES (1, 'Leg Extension', '15 min', 2);

INSERT INTO CUS\_EXERCISE

VALUES (2, 'Set Up', '20 min', 5);

INSERT INTO CUS\_EXERCISE

VALUES (3, 'Side Bends', '10 min', 2);

INSERT INTO CUS\_EXERCISE

VALUES (4, 'Curl', '10 min', 1);

INSERT INTO CUS\_EXERCISE

VALUES (5, 'Running', '20 min', 3);

**#Insert Food**

INSERT INTO FOOD

VALUES (1, 'Milk', '1 glass every day');

INSERT INTO FOOD

VALUES (2, 'Protin Bar', '2 pieces every day');

INSERT INTO FOOD

VALUES (4, 'Banana', '3 pieces every day');

INSERT INTO FOOD

VALUES (5, 'Carrots', '2 cups every day');

INSERT INTO FOOD

VALUES (3, 'Egg', '2 pieces every day');

**#Insert Diet**

INSERT INTO DIET

VALUES (1, 3, '1 pieces');

INSERT INTO DIET

VALUES (4, 1, '2 glass');

INSERT INTO DIET

VALUES (2, 5, '2 cups');

INSERT INTO DIET

VALUES (3, 2, '1 pieces');

INSERT INTO DIET

VALUES (5, 4, '2 pieces');

**Query Writing**

Here some query and screen shot of some questions for our project.

**##SINGLE ROW FUNCTION QUERIES:**

**1)** **Calculate and display the rounded salary of staff Abul and Uddin after dividing salary by 150?**

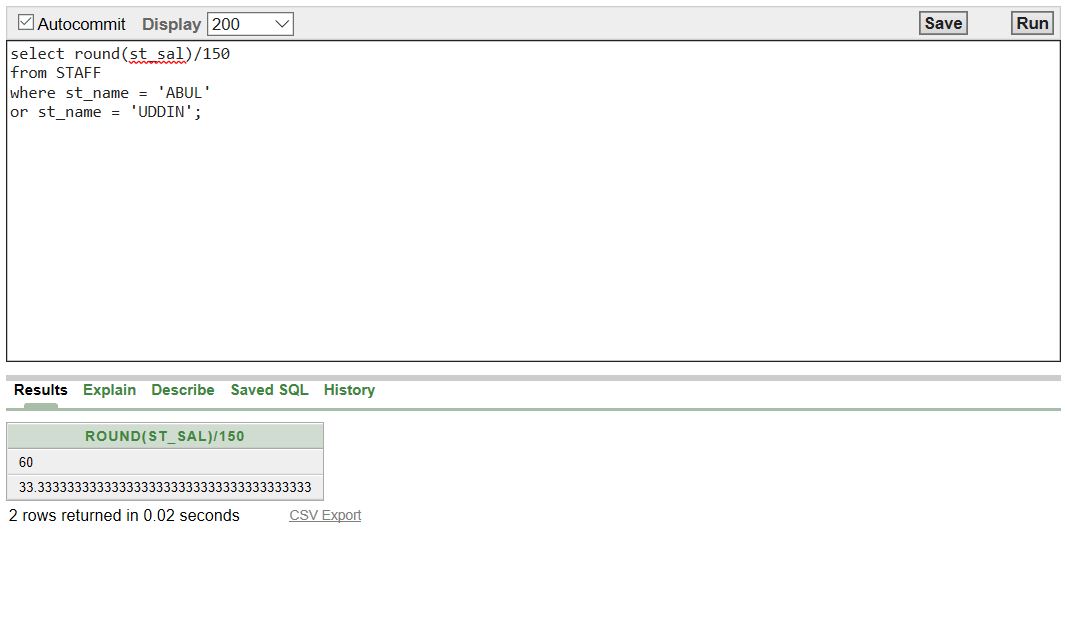
**Answer:**

**select round(st\_sal)/150**

**from STAFF**

**where st\_name = 'ABUL'**

**or st\_name = 'UDDIN'**;



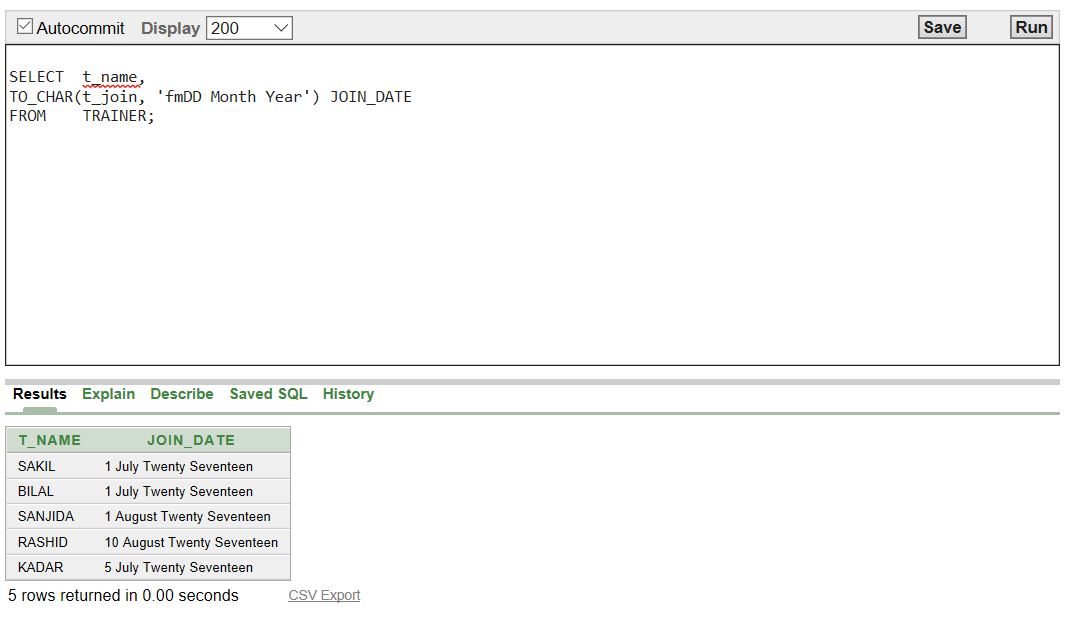
**2) Display the joindate of all trainer in the format DD-MONTH-YEAR(e.g 17 December Nineteen Eighty) and label the column JOIN\_DATE?**

**Answer:**

**SELECT t\_name,**

**TO\_CHAR(t\_join, 'fmDD Month Year') JOIN\_DATE**

**FROM TRAINER;**

****

**##GROUP FUNCTION QUERIES:**

**3) Display the number of client whose gender is FEMALE and label the column GENDER?**

**Answer:**

**SELECT COUNT(c\_gen) as GENDER**

**FROM CLIENT**

**WHERE c\_gen = 'Female';**

****

**4) Display the sum of salaries grouped by the satff id and role but the sum of salary must be greater than 8500?**

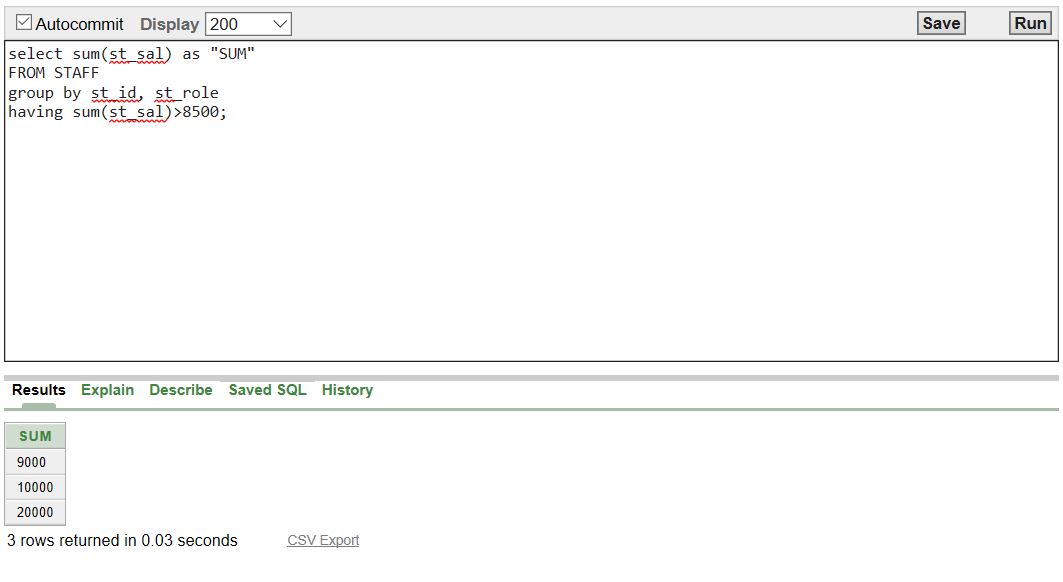
**Answer:**

**select sum(st\_sal) as "SUM"**

**FROM STAFF**

**group by st\_id, st\_role**

**having sum(st\_sal)>8500;**

****

**##SUB QUARIES:**

**5) Display the Client details who joined after MANIK?**

**Answer:**

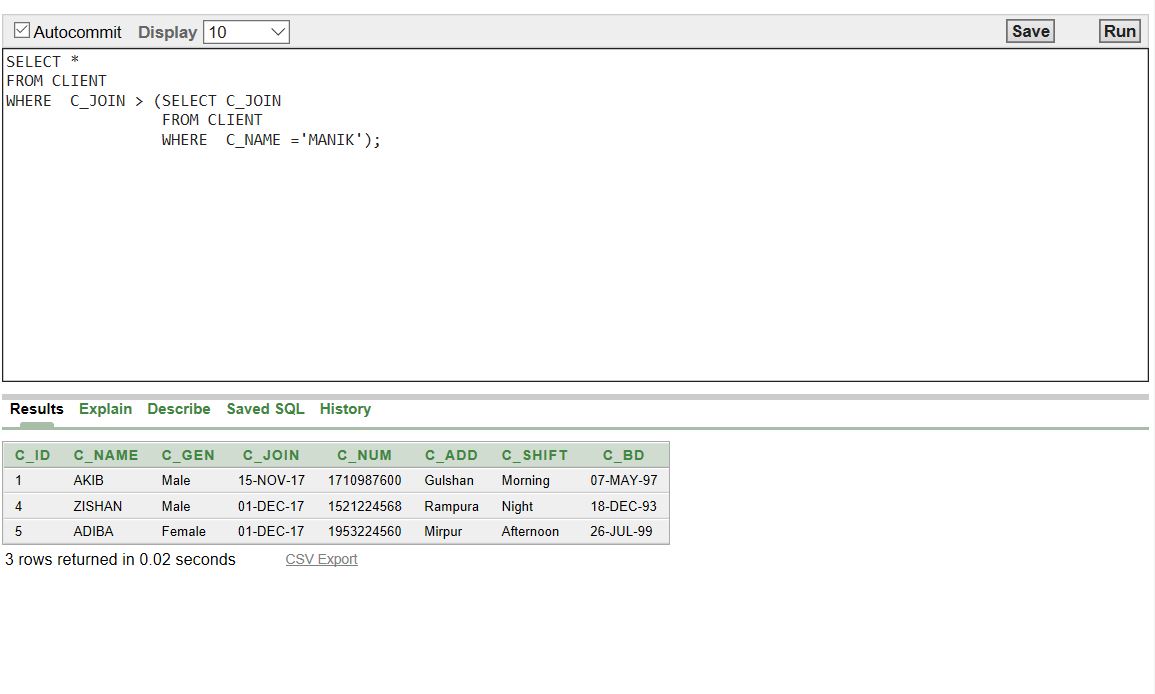
**SELECT \***

**FROM CLIENT**

**WHERE C\_JOIN > (SELECT C\_JOIN**

**FROM CLIENT**

**WHERE C\_NAME ='MANIK');**

****

**6) Find the name, address and number of the Staff who get highest paid salary?**

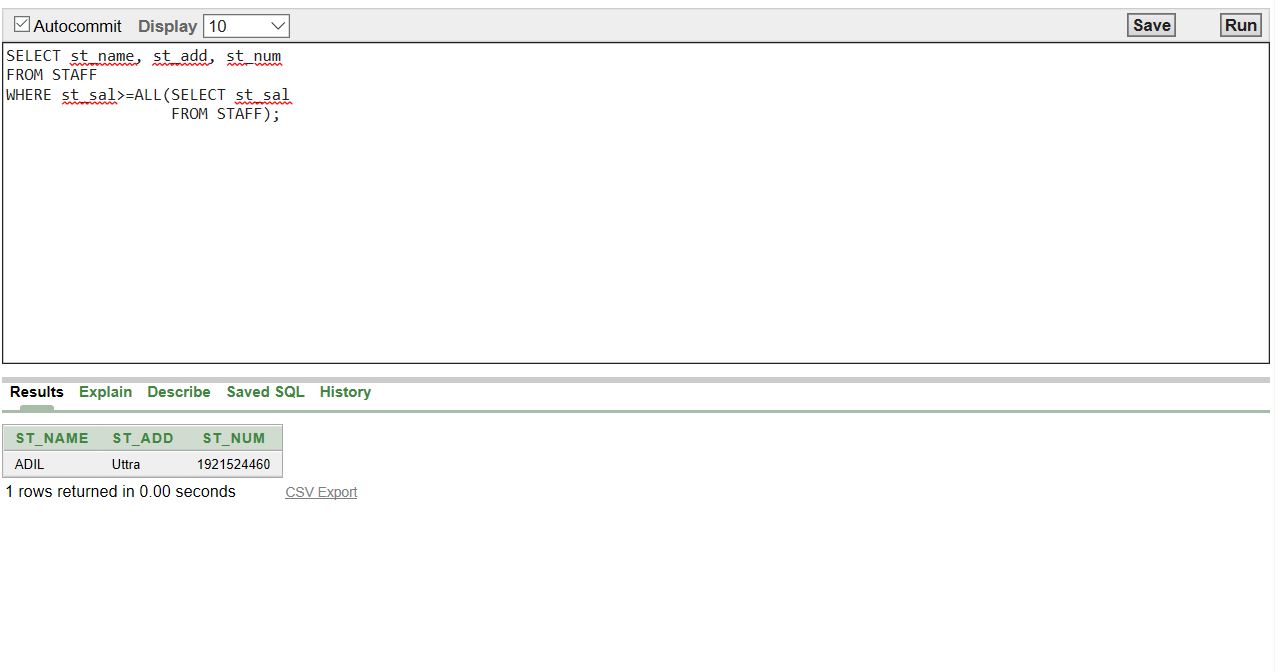
**Answer:**

**SELECT st\_name, st\_add, st\_num**

**FROM STAFF**

**WHERE st\_sal>=ALL(SELECT st\_sal**

**FROM STAFF);**

****

**7) Find the payment id and amount who of lowest payment holder?**

**Answer:**

**SELECT p\_id, p\_amou**

**FROM PAYMENT**

**WHERE p\_amou =**

**(SELECT MIN(p\_amou)**

**FROM PAYMENT);**



**##JOINING QUERIES:**

**8) Find the client name, address and shift who use locker 2?**

**Answer:**

**select client.c\_name, c\_add, c\_shift**

**from client, locker**

**where client.c\_id=locker.c\_id and client.c\_id=2;**

****

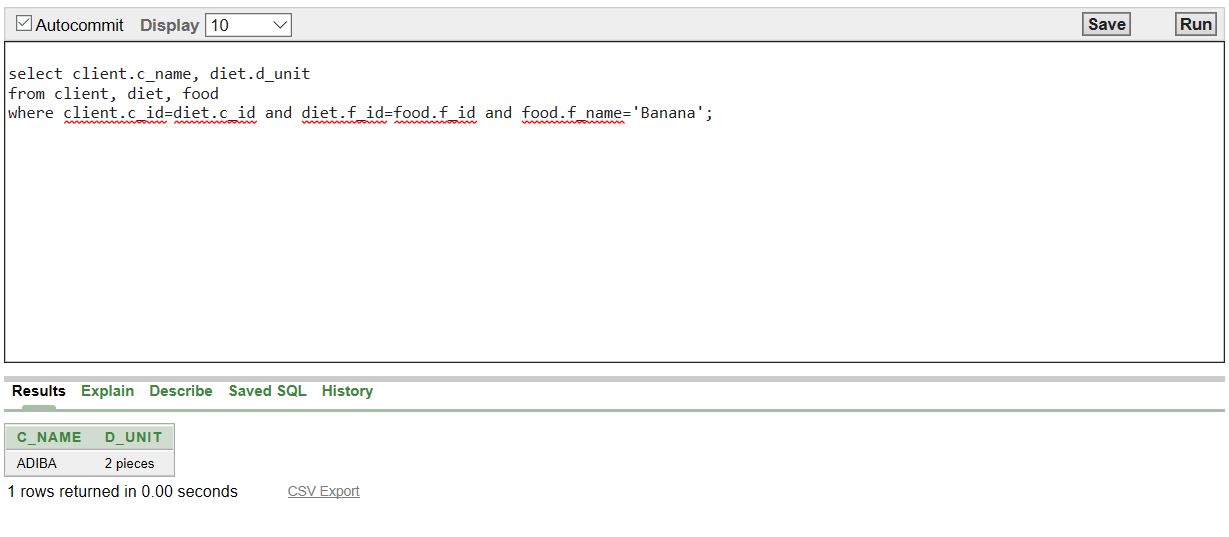
**9) Find the client name and deit unit who eat food item BANANA?**

**Answer:**

**select client.c\_name, diet.d\_unit**

**from client, diet, food**

**where client.c\_id=diet.c\_id and diet.f\_id=food.f\_id and food.f\_name='Banana';**

****

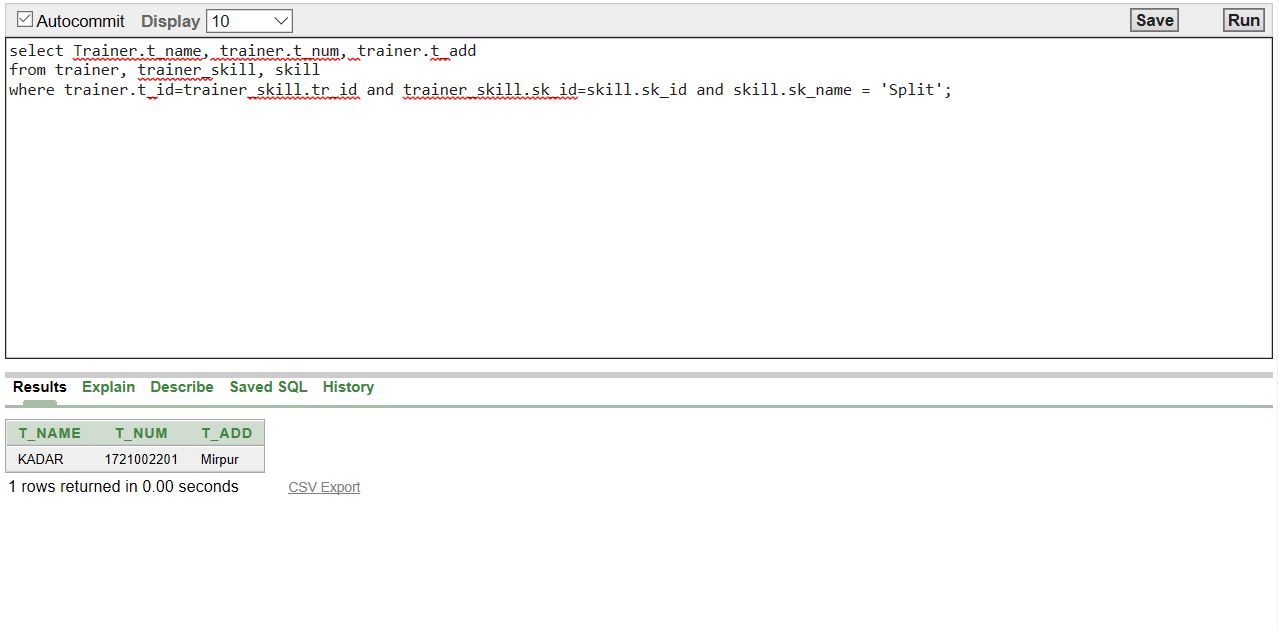
**10) Find the trainer name, address and number who is special skill for SPLIT?**

**Answer:**

**select Trainer.t\_name, trainer.t\_num, trainer.t\_add**

**from trainer, trainer\_skill, skill**

**where trainer.t\_id=trainer\_skill.tr\_id and trainer\_skill.sk\_id=skill.sk\_id and skill.sk\_name = 'Split';**

****

**Conclution:**

Our project is about a gym management system. We think our project will be very helpful to maintaining a gym. But we also have to say that it’s not complete. There are still scope for the advancement. After the advancement it will be better and better. We know that there are no perfect thing. So if we get enough chance we hope that we can make it better. We can also add it with a gym management software to complete it. We only use oracle software here so if we want to really make it useable we have to add it with a software. Then we can use the user interface and the other things. With that it will be truly ready for the use.