***AMERICAN INTERNATIONAL UNIVERSITY OF BANGLADESH***

PROJECT: COFFE SHOP MANAGEMENT

COURSE: INTRODUCTION TO DATABASE

COURSE TEACHER: SIFAT RAHMAN AHONA

GROUP MEMBERS: ID

1.Md.Tohidul Alam Anik 18-37450-1

2.Md. Mostak Jakaria Sohan 18-38543-2

3.Istiuq Ahmad 18-38590-2

4.Md. Nabil Hossain 18-38585-2

Introduction:

A database management system (DBMS) is a system software for creating and managing

databases. The DBMS provides users and programmers with a systematic way to create,

retrieve, update and manage data. A DBMS makes it possible for end users to create, read,

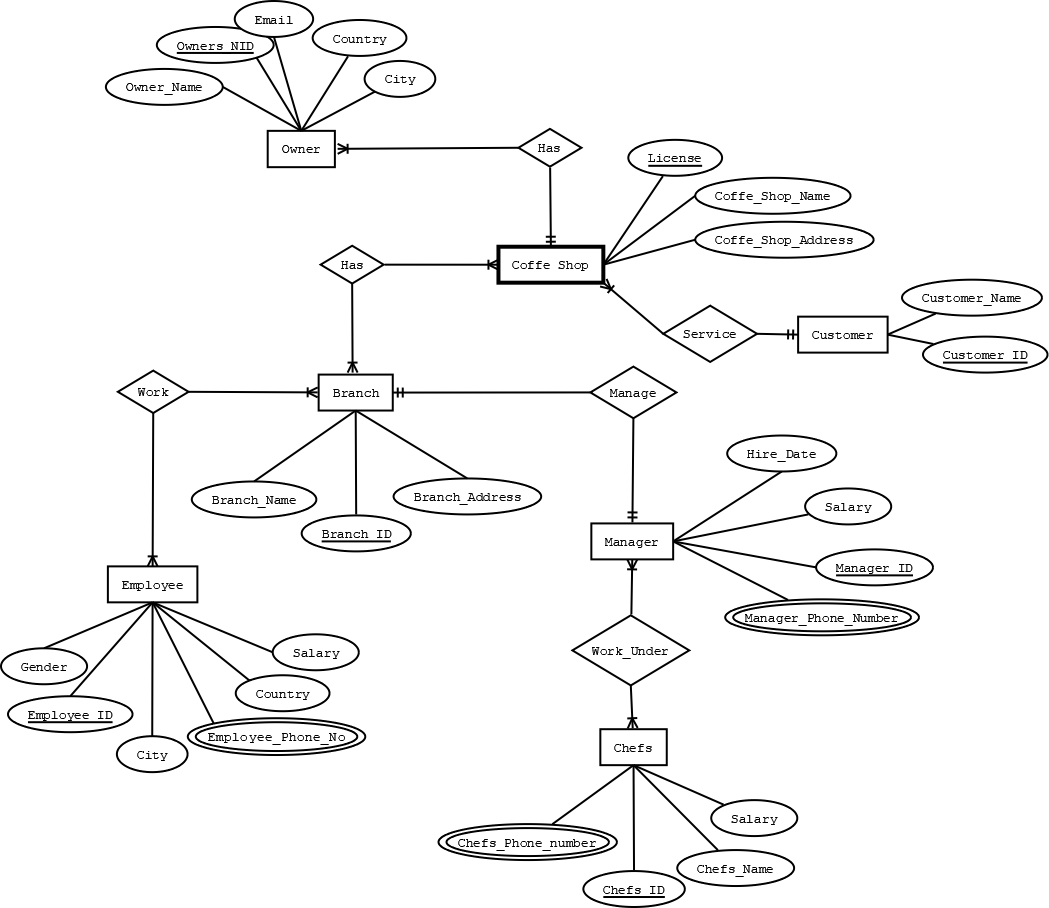
update and delete data in a database.

In our project (Coffee Shop Management System) was created by the concept of DBMS.

Case Study:

In a coffee shop management system a coffee shop can have one or many owners and also owners can have one or many coffee shop. Owners are identified by owner nid. Shop also stores owners name, email, city, country . The shop is identified by shop license. System also stores coffee shop name, coffee shop address . Coffee shop can have one or many customers at a time . Customers are identified by customer id . System also stores customer name. A coffee shop can have many branches but branches belongs to one coffee shop. Branches are identified by branch id. System also stores branch name, branch address . Branch has employees. Branch can have one or many employees but employees can work in one shop . Employees are identified by employee id . System also stores employee name, gender, phone number, salary, city, country . A branch of a coffee shop can have one or many managers and a manager can work in one shop . Managers are identified by manager id. System also stores manager name, phone number, hire date, salary . One manager controls many chefs but chefs works under only one manager . Chefs are identified by chefs id. System also stores chefs name, salary, phone number .

ER-DIAGRAM:



Normalization:

Has (Owners\_NID,Owners\_Name,Email,Country,City,License,Coffe\_Shop\_Name,

Coffe\_Shop\_Address)

1NF: No multivalued attribute

2NF: Owners\_NID,Owners­\_Name,Email,City,Country

License,Coffe\_Shop\_Name,Coffe\_Shop\_Addres

Ol\_ID,Owners\_NID,License

3NF: Owners\_NID,Owners­\_Name,Email,~~City,Country~~,Cc\_ID

License,Coffe\_Shop\_Name,Coffe\_Shop\_Addres

Ol\_ID,Owners\_NID,License

Cc\_ID,Country,City

TABLE:

1. Owners\_NID,Owners­\_Name,Email,Cc\_ID

2.License,Coffe\_Shop\_Name,Coffe\_Shop\_Addres

3.Ol\_ID,Owners\_NID,License

4.Cc\_ID,Country,City

Service (Customer\_Name,Customer\_ID,License,Coffe\_Shop\_Name,

Coffe\_Shop\_Address)

1NF: No multivalued attribute

2NF: Customer\_ID,Customer\_Name

License,Coffe\_Shop\_Name,Coffe\_Shop\_Address,Customer\_ID

3NF: No Transitive Dependency

TABLE:  
 1. Customer\_ID,Customer\_Name

2.License,Coffe\_Shop\_Name,Coffe\_Shop\_Address,Customer\_ID

Has (License,Coffe\_Shop\_Name,Coffe\_Shop\_Address,Branch\_ID,Branch\_Address,

Branch\_Name)

1NF: No multivalued attribute

2NF: License,Coffe\_Shop\_Name,Coffe\_Shop\_Addree

Branch\_ID,Branch\_Address,Branch\_Name

Lb\_ID,License,Branch\_ID

3NF: No Transitive Dependency

TABLE:

1. License,Coffe\_Shop\_Name,Coffe\_Shop\_Addree

2. Branch\_ID,Branch\_Address,Branch\_Name

3.Lb\_ID,License,Branch\_ID

Work (Branch\_ID,Branch\_Address,Branch\_Name,Employee\_ID,Salary,Gender,

Employee\_Phone\_Number,City,Country)

1NF: Employee\_Phone\_Number multivalued attribute

2NF: Branch\_ID,Branch\_Address,Branch\_Name

Employee\_ID,Salary,Gender,Employee\_phone\_Number,City,Country

Be\_ID,Branch\_ID,Employee\_ID

3NF: Branch\_ID,Branch\_Address,Branch\_Name

Employee\_ID,Salary,Gender,Employee\_phone\_Number,~~City,Country~~,Cc\_ID

Be\_ID,Branch\_ID,Employee\_ID

Cc\_ID,City,Country

TABLE:

1. Branch\_ID,Branch\_Address,Branch\_Name

2.Employee\_ID,Salary,Gender,Employee\_phone\_Number,Cc\_ID

3.Be\_ID,Branch\_ID,Employee\_ID

4.Cc\_ID,City,Country

Manage (Manager\_ID,Hire\_Date,Salary,Manager\_Phone\_Number,Branch\_ID,

Branch\_Address,Branch\_Name)

1NF: Manager\_Phone\_Number multivalued attribute

2NF: Manager\_ID,Hire\_Date,Salary,Manager\_Phone\_Number

Branch\_ID,Branch\_Address,Branch\_Name,Manager\_ID

3NF: No Transitive Dependency

TABLE:

1.Manager\_ID,Hire\_Date,Salary,Manager\_Phone\_Number

2.Branch\_ID,Branch\_Address,Branch\_Name,Manager\_ID

Work Under (Manager\_ID,Hire\_Date,Salary,Manager\_Phone\_Number,Chefs\_ID,

Salary,Chefs\_Name,Chefs\_Phone\_Number)

1NF: Manager\_Phone\_Number,Chefs\_Phone\_Nmuber are multivalued attribute

2NF: Manager\_ID,Hire\_Date,Salary,Manager\_Phone\_Number

Chefs\_ID,Salary,Chefs\_name,Chefs\_Phone\_Number

Mc\_ID,Manager\_ID,Chefs\_ID

3NF: No Transitive Dependency

TABLE:

1.Manager\_ID,Hire\_Date,Salary,Manager\_Phone\_Number

2.Chefs\_ID,Salary,Chefs\_name,Chefs\_Phone\_Number

3.Mc\_ID,Manager\_ID,Chefs\_ID

TOTAL TABLE:

1.Owners\_NID,Owners­\_Name,Email,Cc\_ID

2.~~License,Coffe\_Shop\_Name,Coffe\_Shop\_Addres~~

3.Ol\_ID,Owners\_NID,License

4.Cc\_ID,Country,City

5.Customer\_ID,Customer\_Name

6.License,Coffe\_Shop\_Name,Coffe\_Shop\_Address,Customer\_ID

7.~~License,Coffe\_Shop\_Name,Coffe\_Shop\_Addree~~

8.Branch\_ID,Branch\_Address,Branch\_Name

9.Lb\_ID,License,Branch\_ID

10.~~Branch\_ID,Branch\_Address,Branch\_Name~~

11.Employee\_ID,Salary,Gender,Employee\_phone\_Number,Cc\_ID

12.Be\_ID,Branch\_ID,Employee\_ID

13.~~Cc\_ID,City,Country~~

14.Manager\_ID,Hire\_Date,Salary,Manager\_Phone\_Number

15.Branch\_ID,Branch\_Address,Branch\_Name,Manager\_ID

16.~~Manager\_ID,Hire\_Date,Salary,Manager\_Phone\_Number~~

17.Chefs\_ID,Salary,Chefs\_name,Chefs\_Phone\_Number

18.Mc\_ID,Manager\_ID,Chefs\_ID

FINAL TABLE:

1.Owners\_NID,Owners­\_Name,Email,Cc\_ID

2.Ol\_ID,Owners\_NID,License

3.Cc\_ID,Country,City

4.Customer\_ID,Customer\_Name

5.License,Coffe\_Shop\_Name,Coffe\_Shop\_Address,Customer\_ID

6.Branch\_ID,Branch\_Address,Branch\_Name

7.Lb\_ID,License,Branch\_ID

8.Employee\_ID,Salary,Gender,Employee\_phone\_Number,Cc\_ID

9.Be\_ID,Branch\_ID,Employee\_ID

10.Manager\_ID,Hire\_Date,Salary,Manager\_Phone\_Number

11.Branch\_ID,Branch\_Address,Branch\_Name,Manager\_ID

12.Chefs\_ID,Salary,Chefs\_name,Chefs\_Phone\_Number

13.Mc\_ID,Manager\_ID,Chefs\_ID

Table Creation :

*create table owner*

(owner\_Nid number(5) constraint owner\_onid\_pk primary key,

owner\_name varchar2(25) not null,

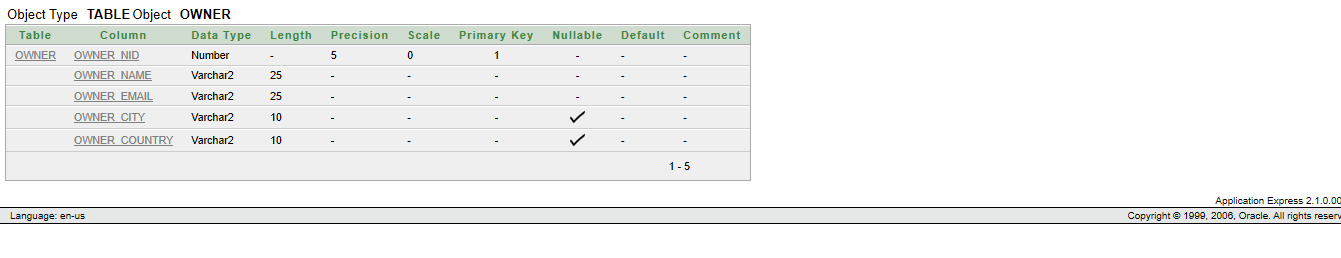
owner\_email varchar2(25) not null,

owner\_city varchar2(10) unique,

owner\_country varchar2(10) unique

)

desc owner

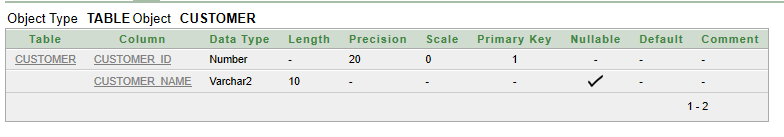


create table customer

(customer\_id number(20) constraint customer\_cid\_pk primary key,

customer\_name varchar2(10) unique)

Desc customer



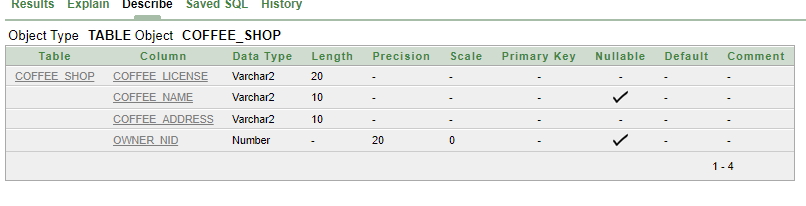
create table coffee\_shop

(coffee\_license varchar2(20) not null,

coffee\_name varchar2(10)unique,

coffee\_address varchar2(10)not null)

desc coffee\_shop



Create table branch

(

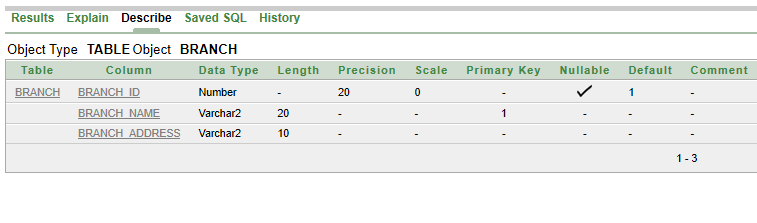
branch\_id number(20) default 1,

branch\_name varchar2(20) constraint branch\_bname\_pk primary key,

branch\_address varchar2(10) not null

)

desc branch



create table employee

(e\_id varchar2(5) constraint employee\_eid\_pk primary key,

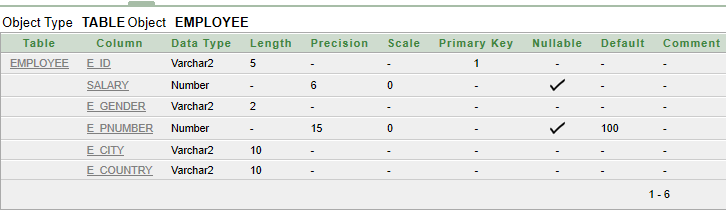
salary number(6) constraint employee\_salary\_ck check(salary<2000),

e\_gender varchar2(2) not null,

e\_pnumber number(15) default 100,

e\_city varchar2(10) not null,

e\_country varchar2(10) not null)

desc employee create table manager

(m\_id number(10) not null,

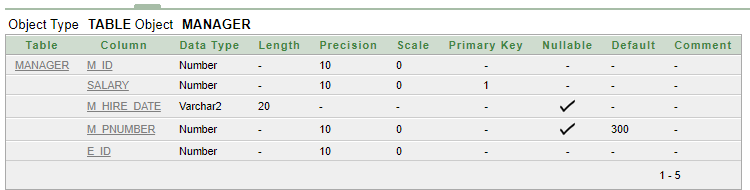
salary number(10) constraint manager\_msalary\_pk primary key,

m\_hire\_date varchar2(20) unique,

m\_pnumber number(10) default 300,

e\_id number(10) not null)

desc manager



create table chefs

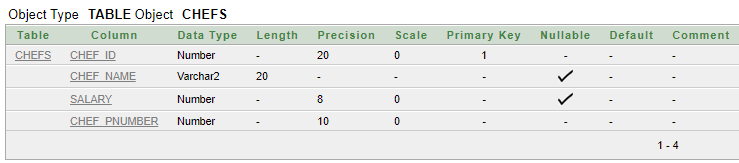
(chef\_id number(20) constraint chefs\_cid\_pk primary key,

salary number(8) constraint chefs\_salary\_ck check(salary>1000),

chef\_name varchar2(20) not null,

chef\_pnumber number(10) unique)

desc chefs



Insertion :

insert into owner

(owner\_Nid,owner\_name,owner\_email,owner\_city,owner\_country)values(101,'Anik','taanik13@gmail.com','chittagong','bangladesh')

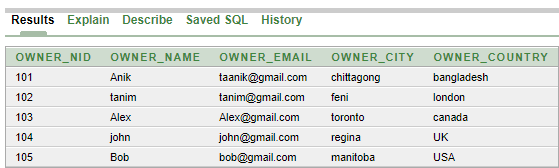
insert into owner(owner\_Nid,owner\_name,owner\_email,owner\_city,owner\_country)values(102,'tanim','tanim@gmail.com','feni','london')

insert into owner(owner\_Nid,owner\_name,owner\_email,owner\_city,owner\_country)values(103,'Alex','Alex@gmail.com','toronto','canada')

insert into owner(owner\_Nid,owner\_name,owner\_email,owner\_city,owner\_country)values(104,'john','john@gmail.com','regina','UK')

insert into owner(owner\_Nid,owner\_name,owner\_email,owner\_city,owner\_country)values(105,'Bob','bob@gmail.com','manitoba','USA')

select \* from owner



insert into customer

(customer\_id,customer\_name)values(1,'Anik')

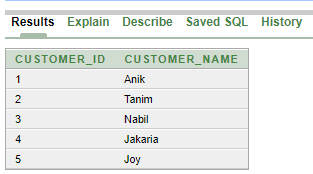
insert into customer(customer\_id,customer\_name)values(2,'Tanim')

insert into customer(customer\_id,customer\_name)values(3,'Nabil')

insert into customer(customer\_id,customer\_name)values(4,'Jakaria')

insert into customer(customer\_id,customer\_name)values(5,'Joy')

select \* from customer



insert into coffee\_shop

(coffee\_license,coffee\_name,coffee\_address,owner\_Nid)values(123,'Khanas','Dhaka',101)

insert into coffee\_shop

(coffee\_license,coffee\_name,coffee\_address,owner\_Nid)values(111,'Bachelors','banani',102)

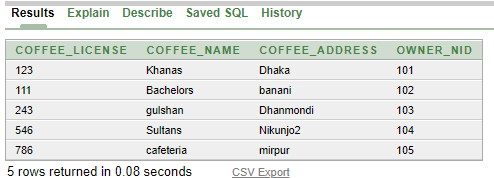
insert into coffee\_shop(coffee\_license,coffee\_name,coffee\_address,owner\_Nid)values(243,'gulshan','Dhanmondi',103)

insert into

coffee\_shop(coffee\_license,coffee\_name,coffee\_address,owner\_Nid)values(546,'Sultans','Nikunjo2',104)

insert into coffee\_shop(coffee\_license,coffee\_name,coffee\_address,owner\_Nid)values(786,'cafeteria','mirpur',105)

select \* from coffee\_shop



insert into branch(branch\_id,branch\_name,branch\_address)values(101,'Nikkunjo','Khilkhet')

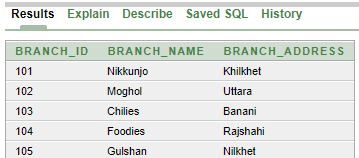
insert into branch(branch\_id,branch\_name,branch\_address)values(102,'Moghol','Uttara')

insert into branch(branch\_id,branch\_name,branch\_address)values(103,'Chilies','Banani')

insert into branch(branch\_id,branch\_name,branch\_address)values(104,'Foodies','Rajshahi')

insert into branch(branch\_id,branch\_name,branch\_address)values(105,'Gulshan','Nilkhet')

select \* from branch



insert into employee(e\_id,salary,e\_gender,e\_pnumber,e\_city,e\_country)values

(01,10000,'Male',01635942353,'mirpur' , 'Dhaka')

insert into employee(e\_id,salary,e\_gender,e\_pnumber,e\_city,e\_country)values

(02,15000,'Male',01634672353,'Dhanmondi' , 'Dhaka')

insert into employee(e\_id,salary,e\_gender,e\_pnumber,e\_city,e\_country)values

(03,20000,'female',01633842353,'uttara' , 'Dhaka')

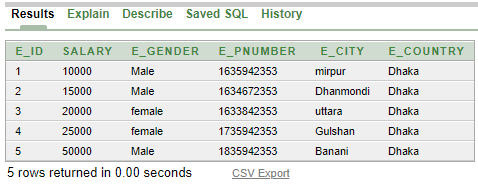
insert into employee(e\_id,salary,e\_gender,e\_pnumber,e\_city,e\_country)values

(04,25000,'female',01735942353,'Gulshan' , 'Dhaka')

insert into employee(e\_id,salary,e\_gender,e\_pnumber,e\_city,e\_country)values

(05,50000,'Male',01835942353,'Banani' , 'Dhaka')

select \* from employee



insert into manager(m\_id,salary,m\_hire\_date,m\_pnumber,e\_id)values(111,10000,'10 jan 2010',485465,1)

insert into manager(m\_id,salary,m\_hire\_date,m\_pnumber,e\_id)values(222,15000,'15 feb 2011',4845465,2)

insert into manager(m\_id,salary,m\_hire\_date,m\_pnumber,e\_id)values(333,20000,'12 may 2010',485485,3)

insert into manager(m\_id,salary,m\_hire\_date,m\_pnumber,e\_id)values(444,18000,'03 jan 2012',42346,4)

insert into manager(m\_id,salary,m\_hire\_date,m\_pnumber,e\_id)values(555,14000,'25 april 2013',485675,5)

select \*from manager



insert into chefs(chef\_id,salary,chef\_name,chef\_pnumber)values(01,5000,'Tohidul',01748739)

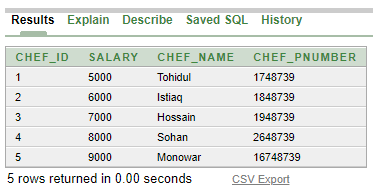
insert into chefs(chef\_id,salary,chef\_name,chef\_pnumber)values(02,6000,'Istiaq',01848739)

insert into chefs(chef\_id,salary,chef\_name,chef\_pnumber)values(03,7000,'Hossain',01948739)

insert into chefs(chef\_id,salary,chef\_name,chef\_pnumber)values(04,8000,'Sohan',02648739)

insert into chefs(chef\_id,salary,chef\_name,chef\_pnumber)values(05,9000,'Monowar',016748739)

select \* from chefs



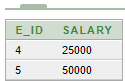
Query Writing:(Subquery)

1 Write a query to display the employee id and salary whose salary is greater than employee id = 3 ?

Ans:

select e\_id,salary from employee

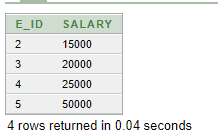
where salary > (select salary from employee where e\_id = 3)



2. Write a query to display the employee id and salary for all employees who earn more than the minimum salary (Employees table)

Ans:

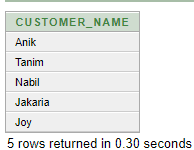
select e\_id,salary from employee where salary > (select min(salary) from employee )



3. Write a query to display the customer name?

Ans:

select customer\_name from customer where customer\_id in(select customer\_id from customer)



Join Query :

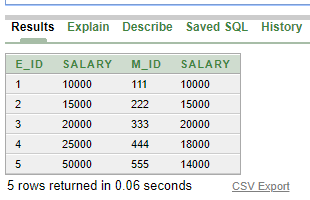
1. Write a query to display the owner name,Nid,coffeeshop name from coffee shop table?

Ans: select o.owner\_name,o.owner\_Nid,c.coffee\_name from owner o , coffee\_shop c where o.owner\_Nid = c.owner\_Nid



2. Write a query to display the employee id , employee salary , manager id , manager salary from manager table?

Ans: select e.e\_id ,e.salary, m.m\_id ,m.salary from employee e , manager m where e.e\_id = m.e\_id



3. Write a query to display the employee id , city,country,manager hiredate,phone number from manager table?

Ans: select e.e\_id ,e.e\_city, e.e\_country,m.m\_hire\_date,m.m\_pnumber from employee e , manager m

where e.e\_id = m.e\_id

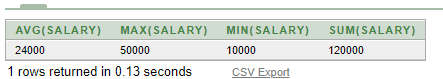


Group Function

1. Write a query to display average salary, maximum salary, minimum salary, sum of salary for

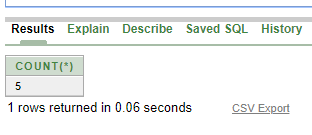
employee table?

select avg(salary),max(salary),min(salary),sum(salary) from employee



2. Write a query to display the row number from employee?

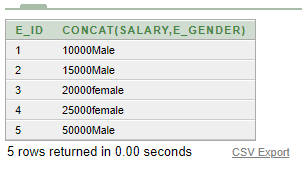
select count(\*) from employee



Single row function:

1.Write a single row function?

Ans : select e\_id,concat(salary,e\_gender)from employee



View:

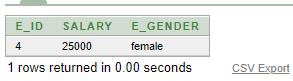
create or replace view emp90vu

as

select e\_id,salary,e\_gender from employee

where e\_id = 4

select \* from emp90vu



Conclusion:

THIS PROJECT IS ABOUT ON A COFEEE SHOP MANAGEMENT SYSTERM .THIS IS NOT AN ONLINE BASED SYSTEM. THIS SYSTEM IS CREATE FOR MANAGING A LOCAL COFFEE SHOP WHERE COFFEE SHOP , OWNER MANAGER ,EMPLOYEE DETAILS ARE INCLUDED AND CUSTOMER ACTIVITIES ARE RECORDER . THIS SYSTEM CAN BE USEFUL FOR A LOCAL COFFEE SHOP WHO WANTS TO RECORD THEIR DAILYS ACTIVITIES.