

**Green University of Bangladesh**

**Department of Computer Science & Engineering**

**Project Name: Cineplex Management System**

**Course Code: CSE-210 Course Title: Database System Lab**

**Course Teacher:** MS. BABE SULTANA

Lecturer

Dept. of CSE

Sec: DA

**Group Members**

|  |  |
| --- | --- |
| Name | ID |
| Md. Shammir Ahmed | 191002271 |
| Shahriar Rahman Evan | 191002298 |
| Taslima Akter | 191002043 |

19 |

P a g

19 |

P a g

**List of Contents:**

1. Scenario:
2. ER Diagram:
3. NORMALIZATION:
4. Table Creation with screenshots
5. Data insertion:
6. Joining:
7. Query Writing on the tables:
8. Group Function
9. View

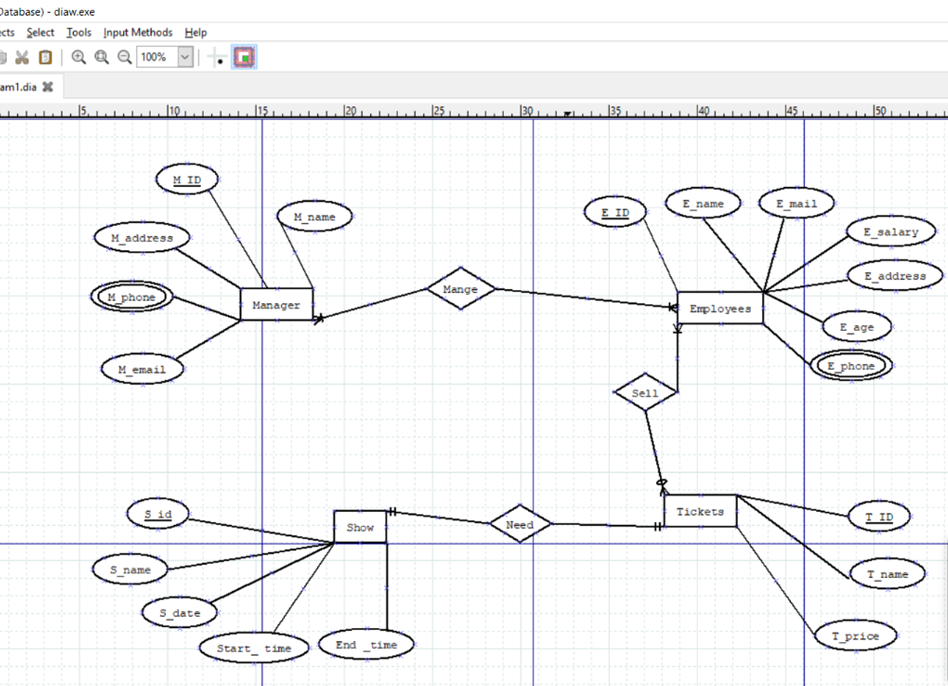
10.Using limit (ORDER BY, ASC, DESC)

11.Trigger

**Scenario:**

Star Cineplex is the first multiplex cinema theatre in Bangladesh.Star Cineplex has a strong management for maintaining the whole system. A Manager has M\_ID, M\_name, M\_address, M\_phone, M\_email. In Cineplex management system there are many employees who controlled by the manager. The attributes of employees include E\_ID, E\_name, E\_address, E\_phone, E\_mail, E\_salary, E\_age.Manager manages all employees. Tickets is must needed for watching a show in cineplex. Few employees sell ticket in ticket booth or people can buy tickets from online by visiting cineplex official website. A Ticket has T\_ID, T\_name, T\_price. One ticket can be sold from only one show. Show has some attributes include S\_id,S\_name, S\_date, Start\_ time, End \_time. One show will run at a time. One manager can manage many employees.

**ER Diagram:**

****

**NORMALIZATION:**

Manage (M\_id,M\_name,M\_address,M\_phone,M\_email,E\_id,E\_name,E\_mail,E\_salary,E\_address,

E\_age,E\_phone)

**1NF:**

M\_phone & E\_phone -> multivalued attributes

**2NF:**

M\_ID, M\_name, M\_address,M\_phone,M\_email

E\_id, E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone

**3NF:**

M\_ID, M\_name,M\_address,M\_phone,M\_email

E\_id ,E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone

No transitive dependency.

**Table:**

M\_ID,M\_name,M\_address,M\_phone,M\_email

E\_id,E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone

Sell (E\_id,E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone,T\_id,T\_name,T\_price)

**1NF:**

E\_phone -> multivalued attribute.

**2NF:**

E\_id,E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone

T\_id,T\_name,T\_price

**3NF:**

E\_id,E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone

T\_id,T\_name,T\_price

No transitive dependency.

Table:

E\_id,E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone

T\_id,T\_name,T\_price

Need (T\_id,T\_name,T\_price,S\_id,S\_name,Start\_time,End\_time)

**1NF:**

No multivalued attributes

**2NF:**

T\_id,T\_name,T\_price,S\_id

S\_id,S\_name,Start\_time,End\_time

**3NF:**

T\_id,T\_name,T\_price,S\_id

S\_id,S\_name,Start\_time,End\_time

No transitive dependency

Table:

T\_id,T\_name,T\_price,S\_id

S\_id,S\_name,Start\_time,End\_time

Total Table:

M\_ID,M\_name,M\_address,M\_phone,M\_email

E\_id,E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone

~~E\_id,E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone~~

T\_id,T\_name,T\_price

T\_id,T\_name,T\_price,S\_id

S\_id,S\_name,Start\_time,End\_time

Final Table:

1st. M\_ID,M\_name,M\_address,M\_phone,M\_email

2nd. E\_id,E\_name,E\_mail,E\_salary,E\_address,E\_age,E\_phone

3rd. T\_id,T\_name,T\_price

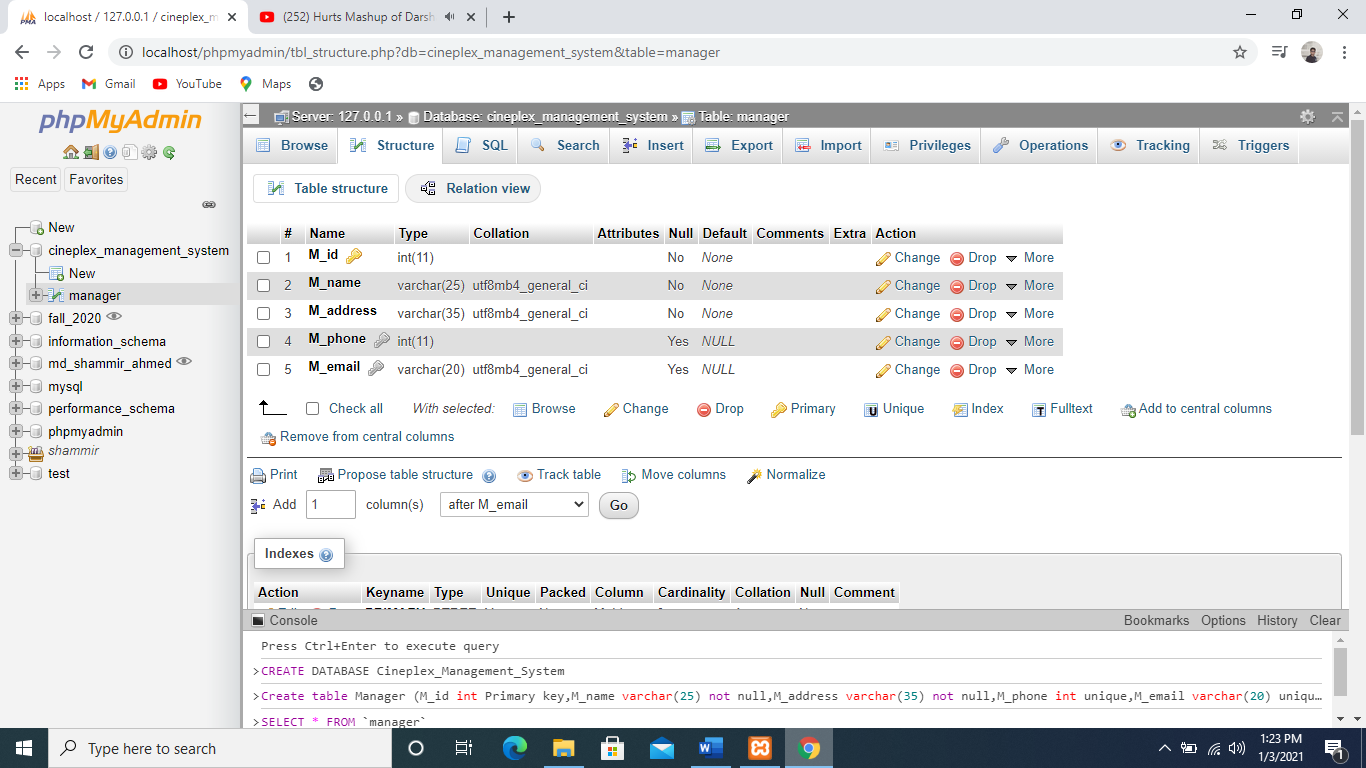
4th. T\_id,T\_name,T\_price,S\_id

5th. S\_id,S\_name,Start\_time,End\_time

**Table Creation with screenshots:**

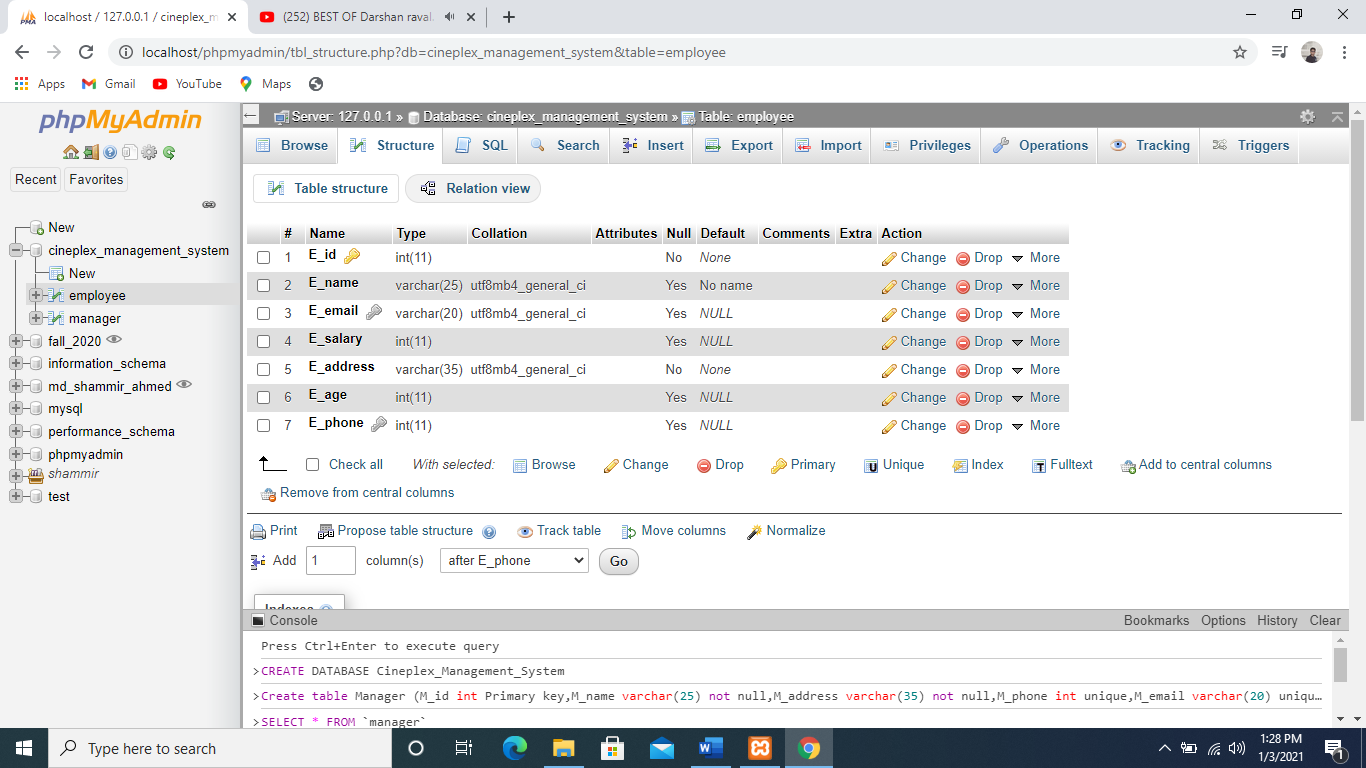
Create table Manager (M\_id int Primary key,M\_name varchar(25) not null,M\_address varchar(35) not null,M\_phone int unique,M\_email varchar(20) unique);

Desc manager



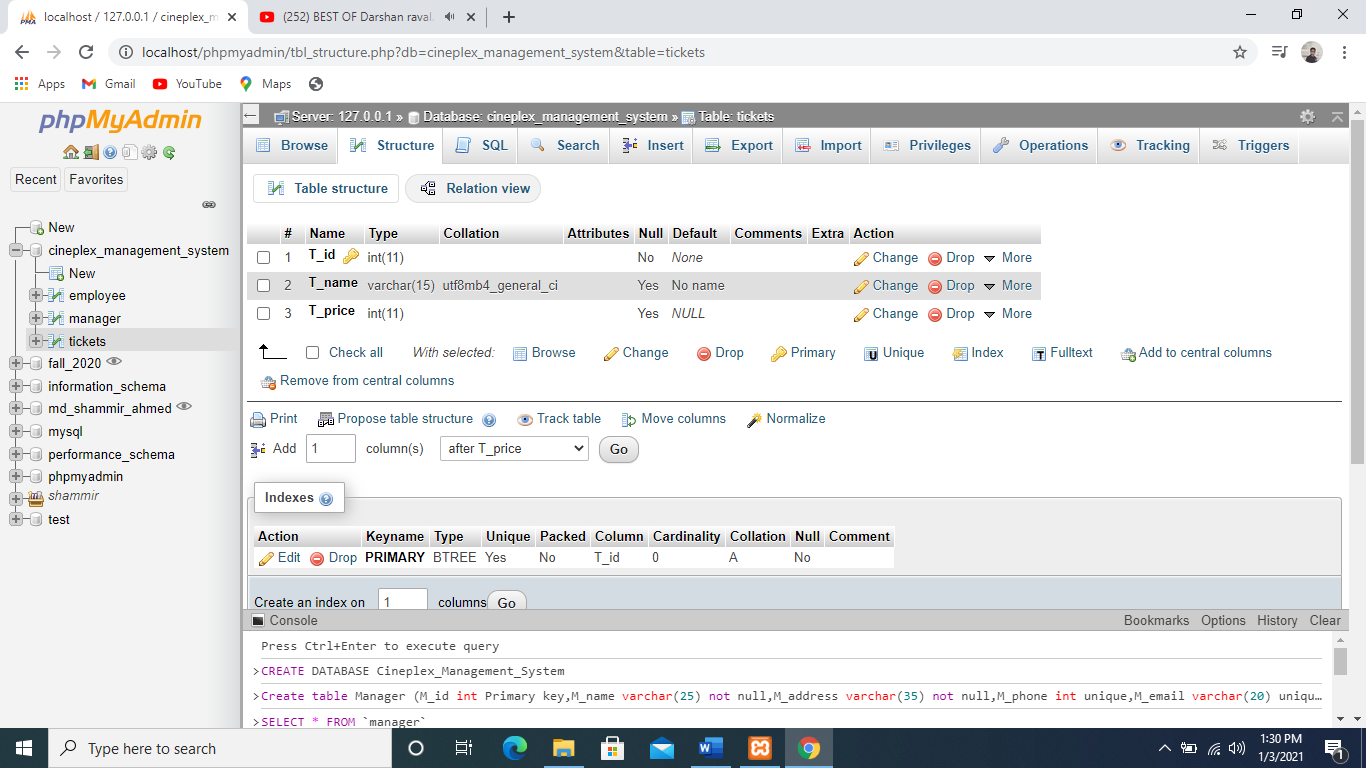
Create Table Employee(E\_id int primary key,E\_name varchar(25) default 'No name',E\_email varchar(20) unique,E\_salary int check (E\_salary>0),E\_address varchar(35) not null,E\_age int check (E\_age>16),E\_phone int unique);

Desc Employee



Create table Tickets(T\_id int primary key,T\_name varchar(15) default 'No name',T\_price int check (T\_price>300));

Desc Tickets



Create Table Shows(S\_id int primary key,S\_name varchar(15) default 'No name',Start\_time varchar(6) not null,End\_time Varchar(6) not null);

Desc Shows

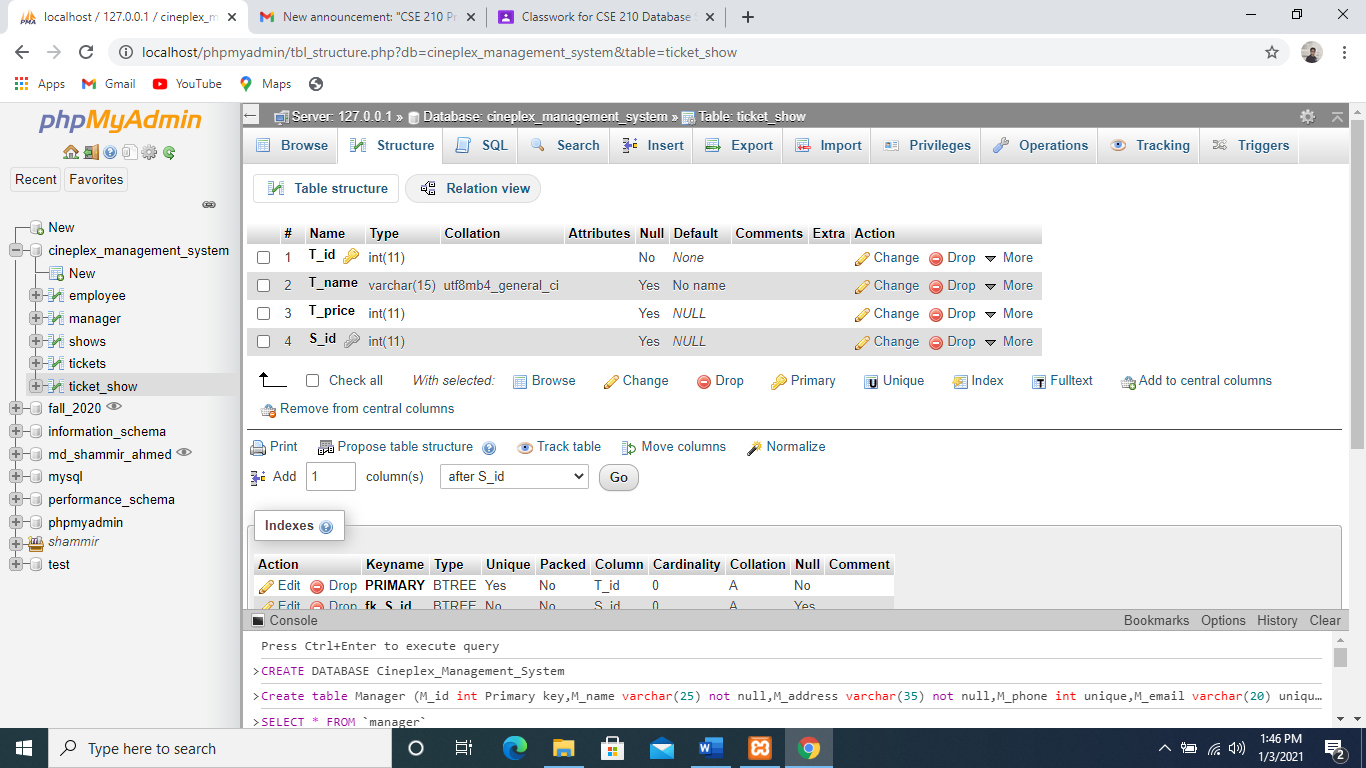


Create table Ticket\_show(T\_id int primary key,T\_name varchar(15) default 'No name',T\_price int check (T\_price>300),S\_id int ,CONSTRAINT fk\_S\_id

FOREIGN KEY (S\_id)

REFERENCES SHows(S\_id));

Desc Ticket\_show



**Data insertion:**

For Manager Table:

Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(1,'Rahman

Hasan','Mirpur-10,Dhaka',117262589,'rhasan@gmail.com');

Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(2,'Kabir

Hasan','Mirpur-2,Dhaka',113456589,'khasan@gmail.com');

Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(3,'Anik

Paul','Nikunja-2,khilkhet',517002389,'anikp@gmail.com');

Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(4,'Ashik

Rahman','Mirpur-1,Dhaka',419354729,'ashik695@gmail.com');

Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(5,'Shawon

Zaman','Nikunja-2,khilkhet',616219424,'szaman@gmail.com');

Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(6,'Chisty

Nomanuzzaman','kuratoli,Dhaka',280981723,'chisty767@gmail.com');

Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(7,'Emtiaz

Anik','Kuratoli,Dhaka',029176589,'anik876@yahoo.com');

Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(8,'Ahmed

Awsaf','Mirpur-10,Dhaka',098276523,'awsaf897@gmail.com');

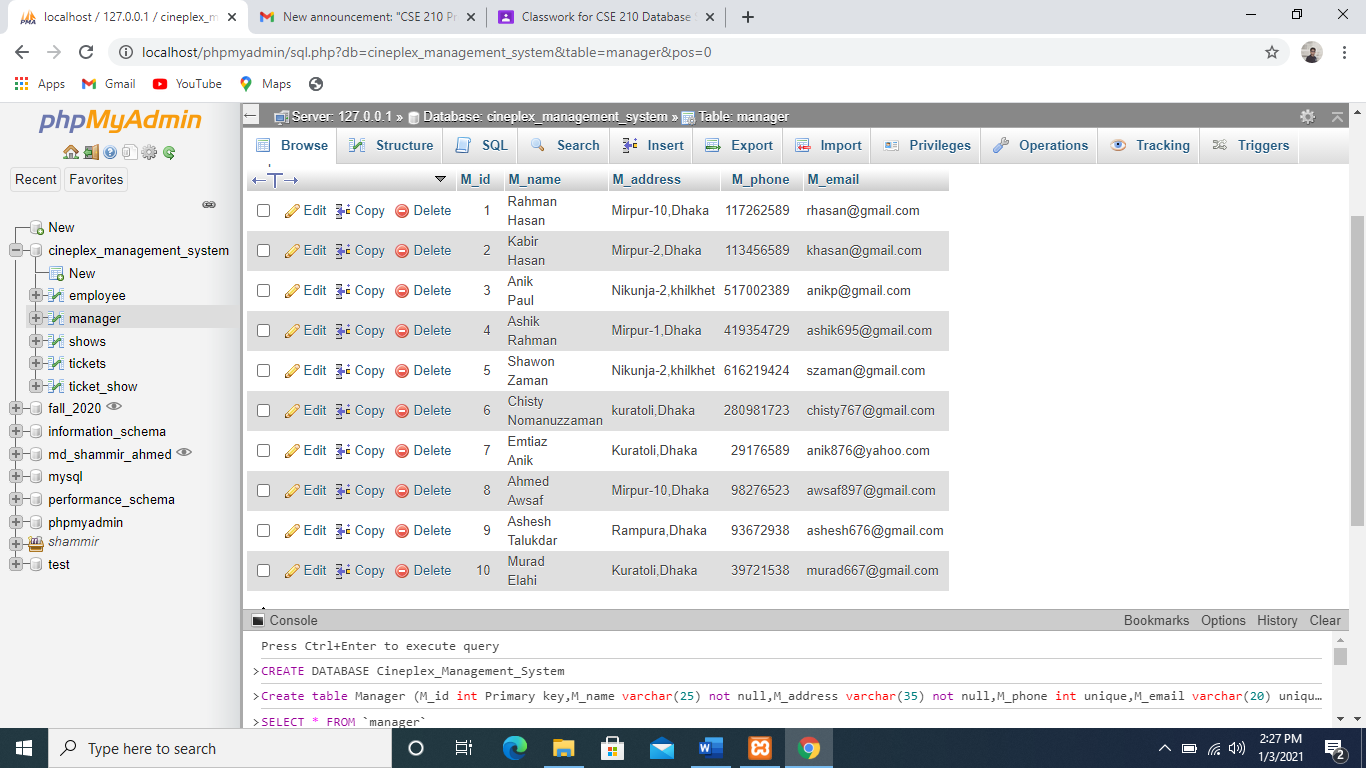
Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(9,'Ashesh

Talukdar','Rampura,Dhaka',093672938,'ashesh676@gmail.com');

Insert into Manager(M\_id,M\_name,M\_address,M\_phone,M\_email)values(10,'Murad

Elahi','Kuratoli,Dhaka',039721538,'murad667@gmail.com');

select \*from manager



For Employee Table:

Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(101,'Asif',

'asif898@gmail.com',4500,'Mirpur-2,Dhaka',17,0928238121);

Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(102,'Rakib',

'rakib3452@gmail.com',4500,'Nilkhelt,Dhaka',18,092873615);

Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(103,'Sakib',

'sakib66@yahoo.com',4200,'Panthapoth,Dhaka',20,093275634);

Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(104,'Jisan',

'jissan33@gmail.com',4000,'Gulsan,Dhaka',22,391231024);

Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(105,'Rakin',

'rakin27@gmail.com',5000,'Bananni,Dhaka',17,0391827652);

Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(106,'Turjo',

'turjo44@gmail.com',4800,'Kuratoli,Dhka',18,051234520);

Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(107,'Rouf',

'rouf78@gmail.com',3500,'Mirpur,Dhaka',24,0412783102);

Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(108,'Tutol',

'tutol44@gmail.com',4800,'Zatrabari,Dhaka',20,021234192);

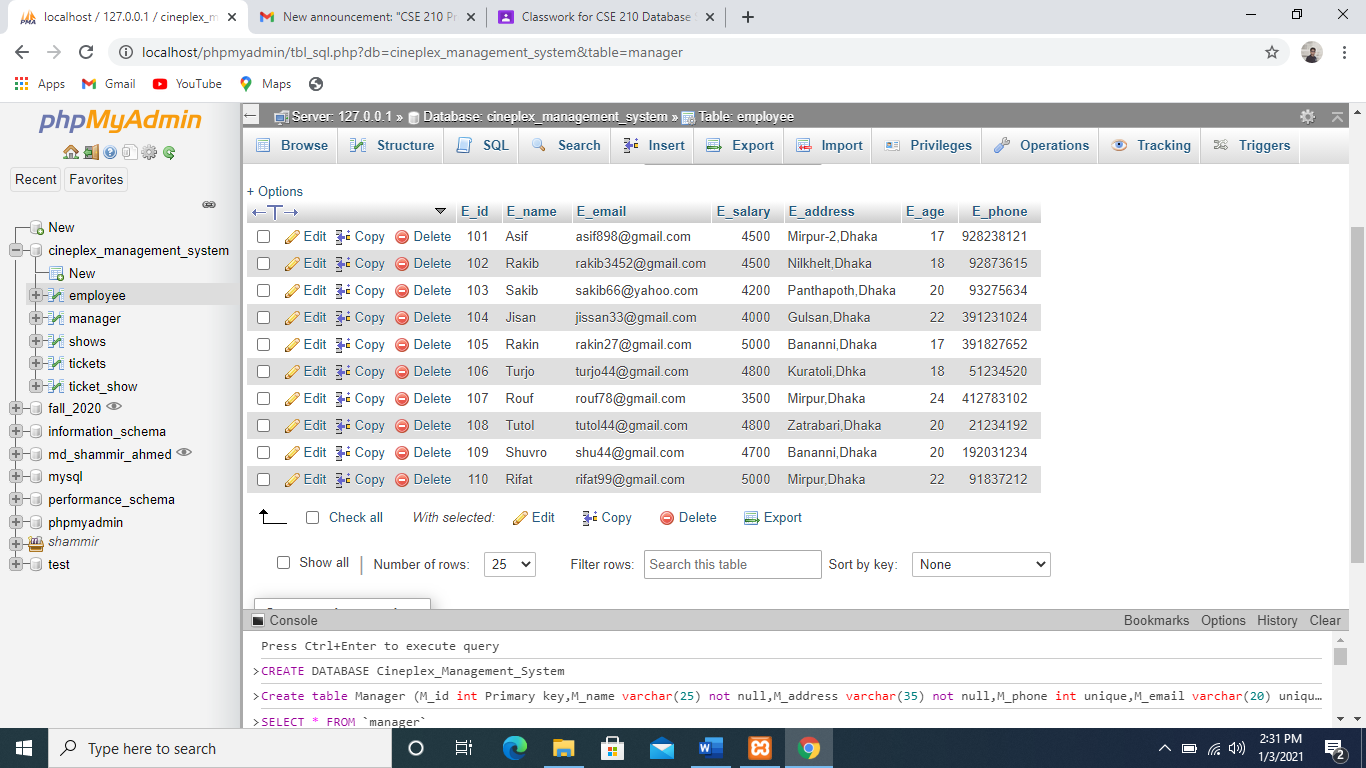
Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(109,'Shuvro',

'shu44@gmail.com',4700,'Bananni,Dhaka',20,192031234);

Insert into Employee(E\_id,E\_name,E\_email,E\_salary,E\_address,E\_age,E\_phone)Values(110,'Rifat',

'rifat99@gmail.com',5000,'Mirpur,Dhaka',22,091837212);

select \*from employee



For Tickets Table:

Insert into Tickets(T\_id,T\_name,T\_price)values(0001,'General',320);

Insert into Tickets(T\_id,T\_name,T\_price)values(0002,'VIP',500);

Insert into Tickets(T\_id,T\_name,T\_price)values(0003,'PREMIUM',400);

Insert into Tickets(T\_id,T\_name,T\_price)values(0004,'General',320);

Insert into Tickets(T\_id,T\_name,T\_price)values(0005,'PREMIUM',400);

Insert into Tickets(T\_id,T\_name,T\_price)values(0006,'General',320);

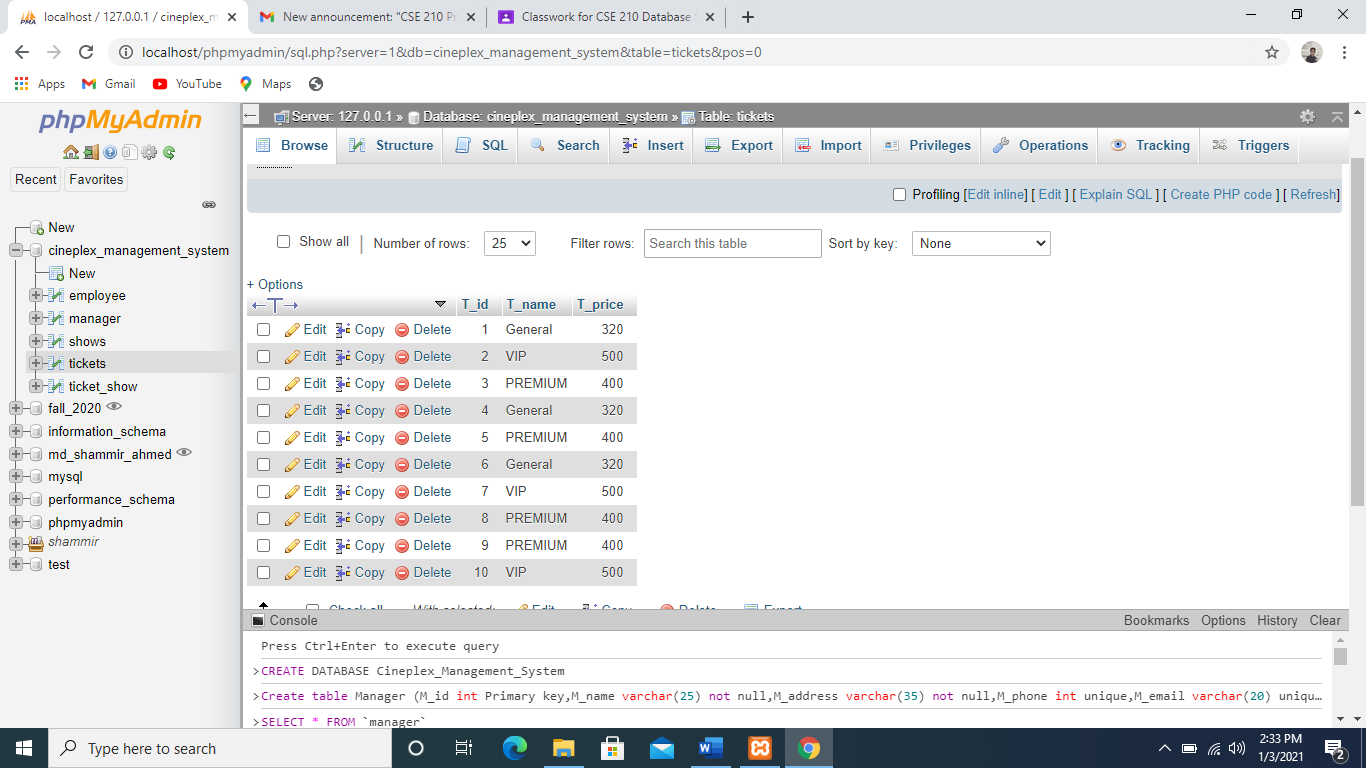
Insert into Tickets(T\_id,T\_name,T\_price)values(0007,'VIP',500);

Insert into Tickets(T\_id,T\_name,T\_price)values(0008,'PREMIUM',400);

Insert into Tickets(T\_id,T\_name,T\_price)values(0009,'PREMIUM',400);

Insert into Tickets(T\_id,T\_name,T\_price)values(0010,'VIP',500);

Select \*from Tickets



For Shows Table:

Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(01,'Movie 1','9 am','10 am');

Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(02,'Movie 2','10 am','11 am');

Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(03,'Movie 3','11 am','12 pm');

Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(04,'Movie 4','12 pm','1 pm');

Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(05,'Movie 5','1 pm','2 pm');

Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(06,'Movie 1','3 pm','4 pm');

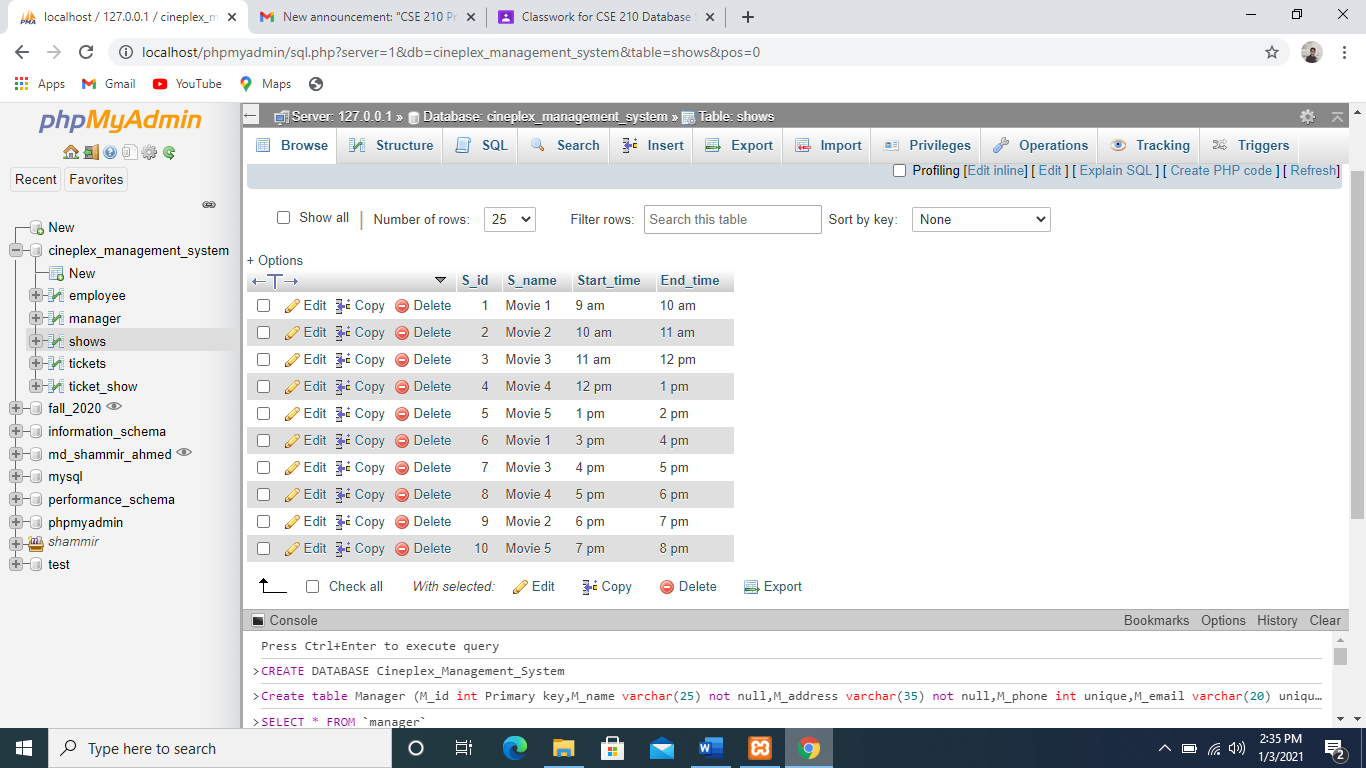
Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(07,'Movie 3','4 pm','5 pm');

Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(08,'Movie 4','5 pm','6 pm');

Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(09,'Movie 2','6 pm','7 pm');

Insert into Shows(S\_ID,S\_Name,Start\_time,End\_time)Values(10,'Movie 5','7 pm','8 pm');

Select \*from Shows



For Ticket\_Show Table:

Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0001,'General',320,01);

Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0002,'VIP',500,02);

Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0003,'PREMIUM',400,03);

Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0004,'General',320,04);

Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0005,'PREMIUM',400,05);

Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0006,'General',320,06);

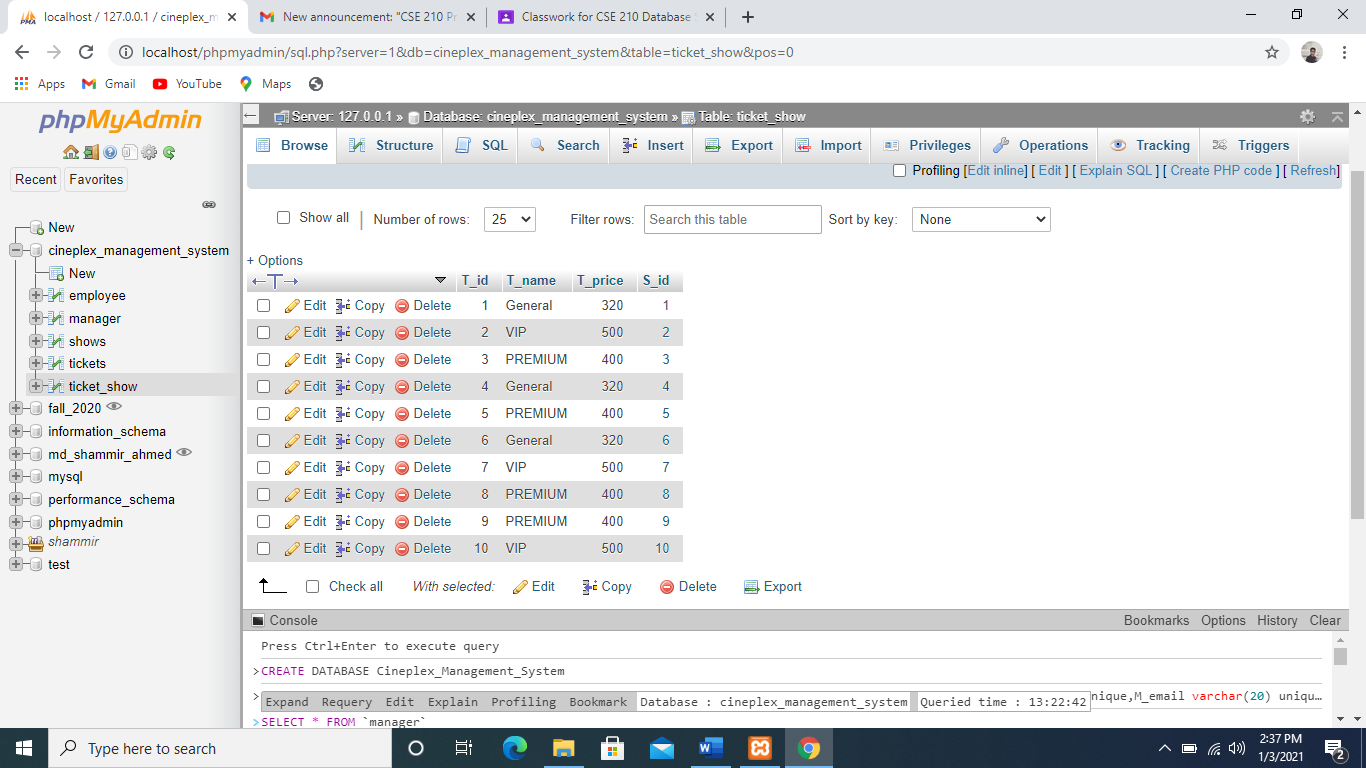
Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0007,'VIP',500,07);

Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0008,'PREMIUM',400,08);

Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0009,'PREMIUM',400,09);

Insert into Ticket\_show(T\_id,T\_name,T\_price,S\_id)Values(0010,'VIP',500,10);

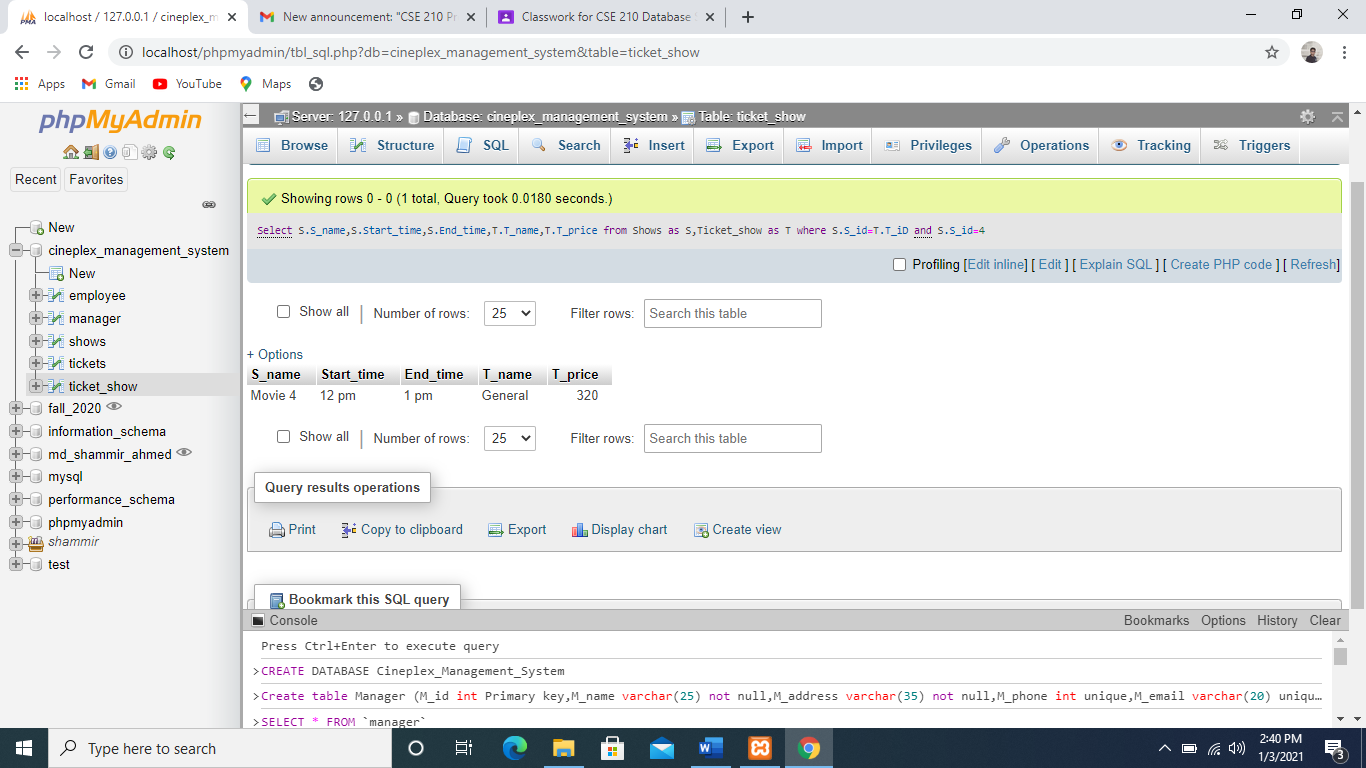
Select \*from Ticket\_show



**Joining:**

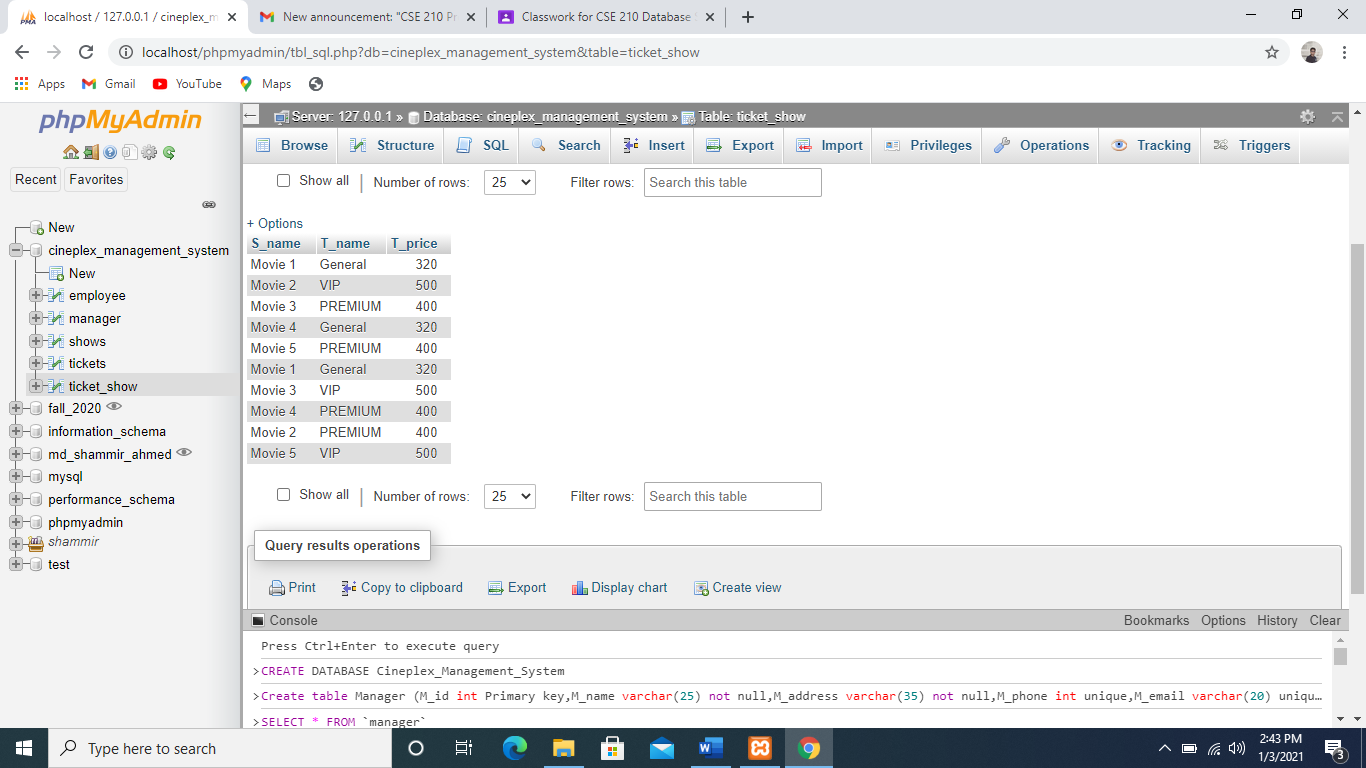
1.Write a query to display Shows name,Start and end time ,Tickets for shows price and name when there Shows number is 4.(Equi-join)

🡪 Select S.S\_name,S.Start\_time,S.End\_time,T.T\_name,T.T\_price from Shows as S,Ticket\_show as T where S.S\_id=T.T\_iD and S.S\_id=4



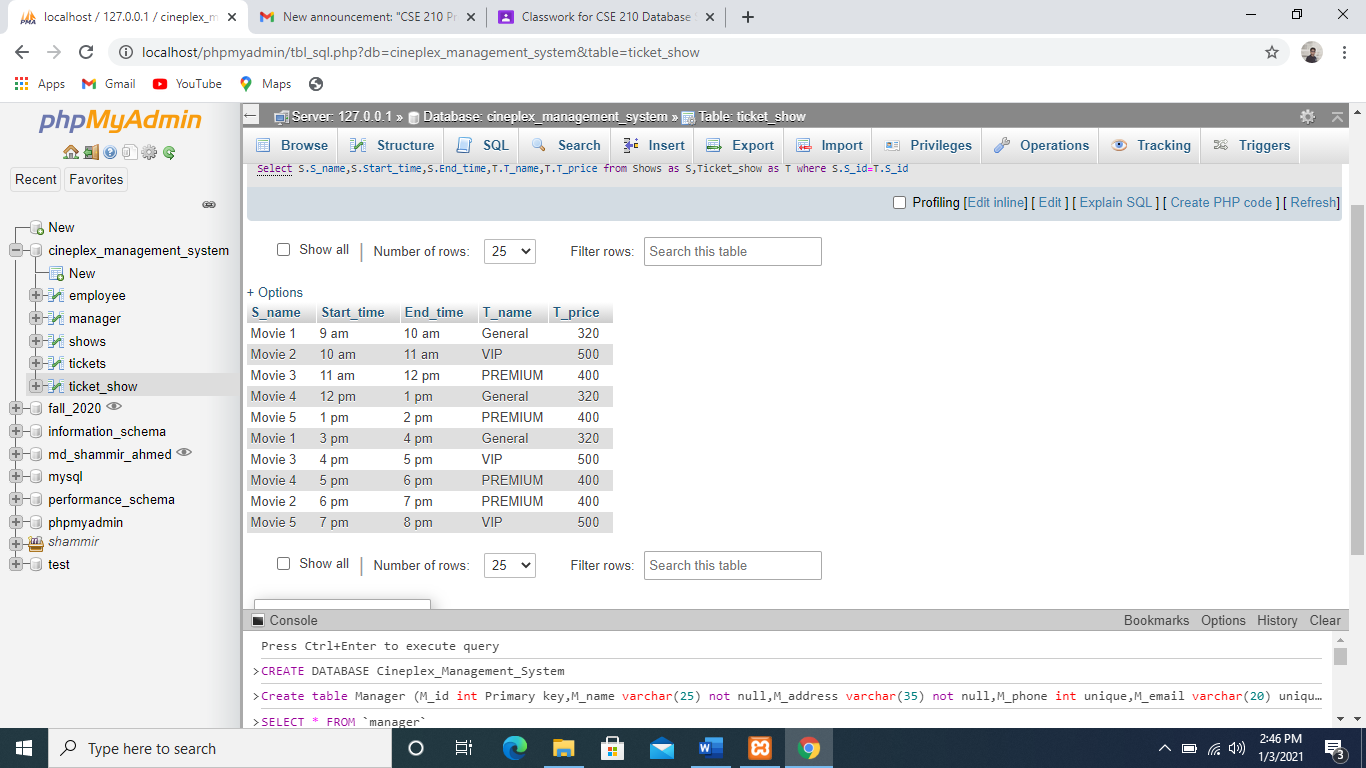
2.Write a query to display Shows name and tickets shows price and name. (Equi-join)

🡪 Select S.S\_name,T.T\_name,T.T\_price from Shows as S,Ticket\_show as T where S.S\_id=T.T\_iD;



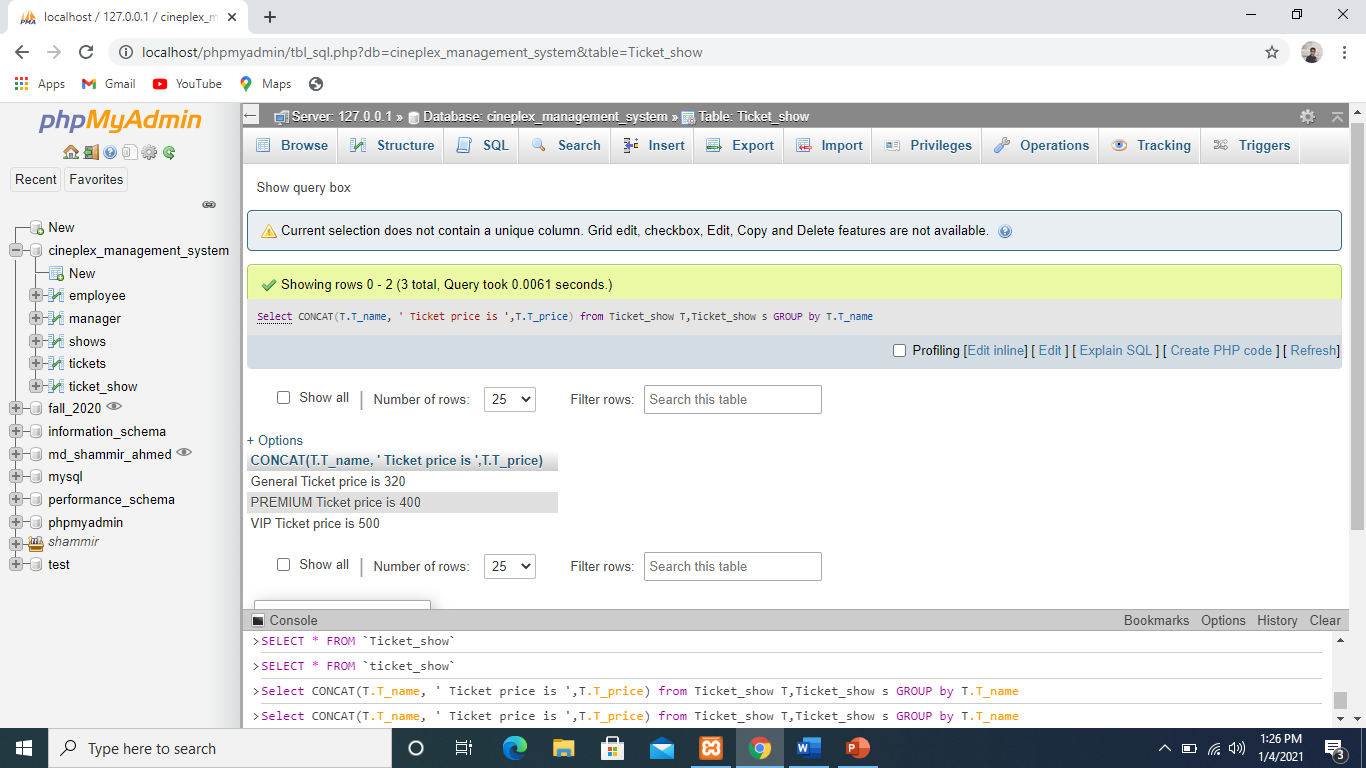
3.Write a query to display shows name id, end time and start time and tickets name and prices using outer join

🡪 Select S.S\_name,S.Start\_time,S.End\_time,T.T\_name,T.T\_price from Shows as S,Ticket\_show as T where S.S\_id=T.S\_id;



4.Write a query to display \_\_\_\_\_\_\_Tickets price is \_\_\_\_\_\_\_ using CONCAT function & join & GROUP by.

🡪 Select CONCAT(T.T\_name, ' Ticket price is ',T.T\_price) from Ticket\_show T,Ticket\_show s GROUP by T.T\_name;

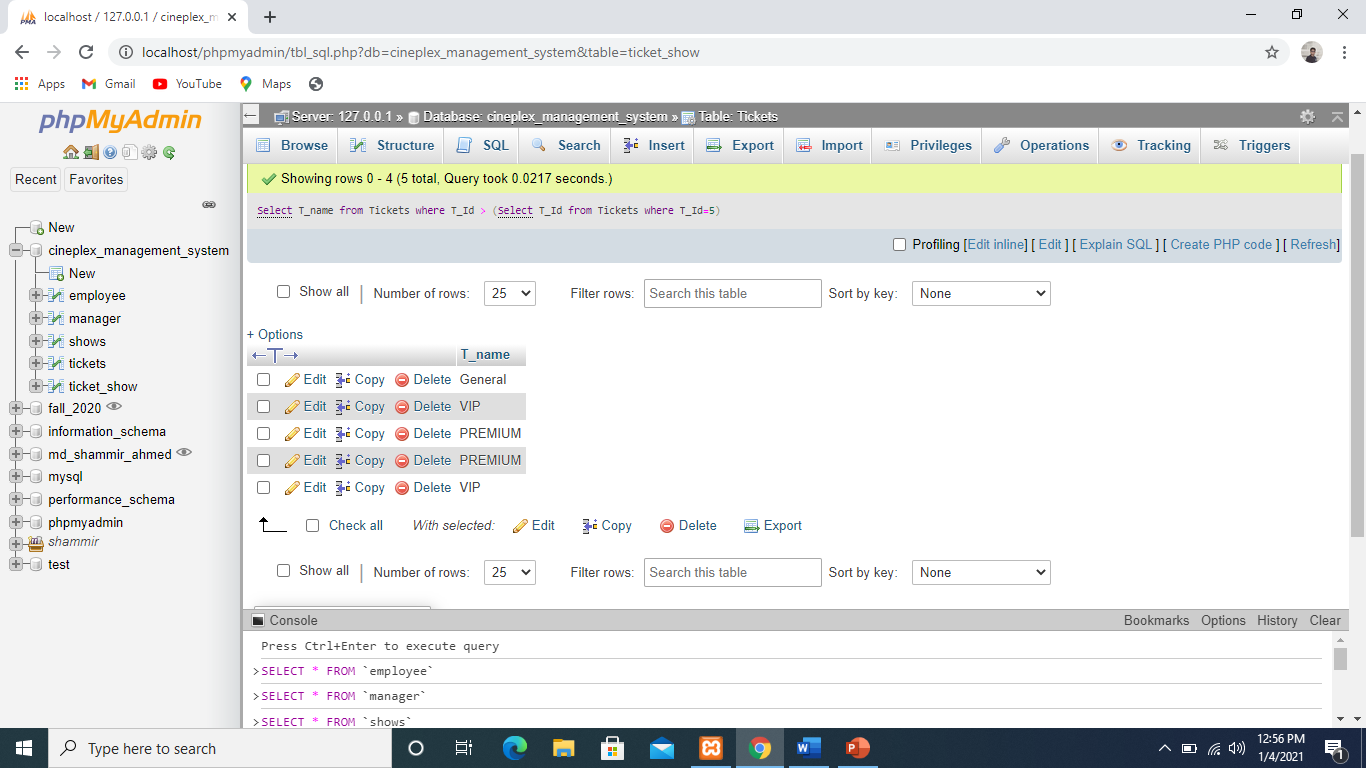


**Query Writing on the tables:**

Subquery-

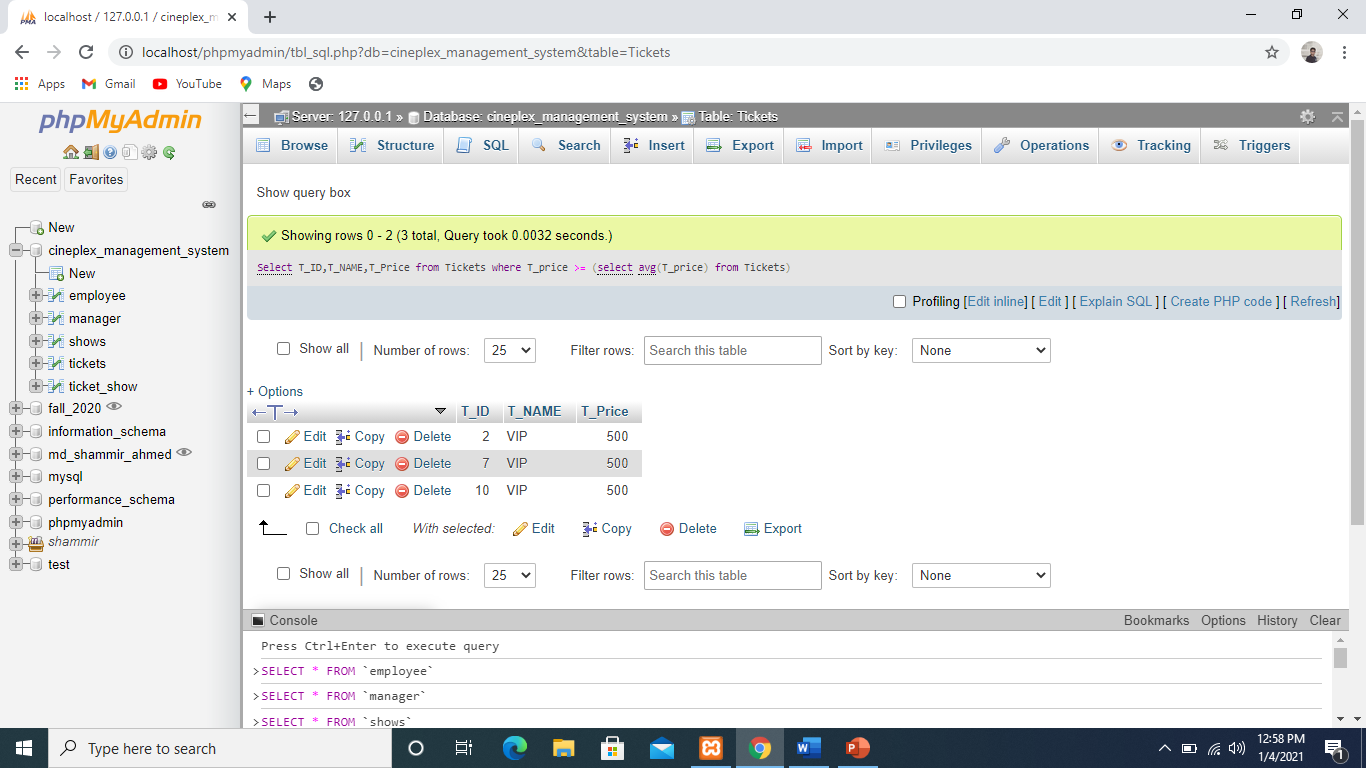
1.Write a query to display Ticket name where ticket id is greater than 5. (Tickets table)

🡪Select T\_name from Tickets where T\_Id > (Select T\_Id from Tickets where T\_Id=5);



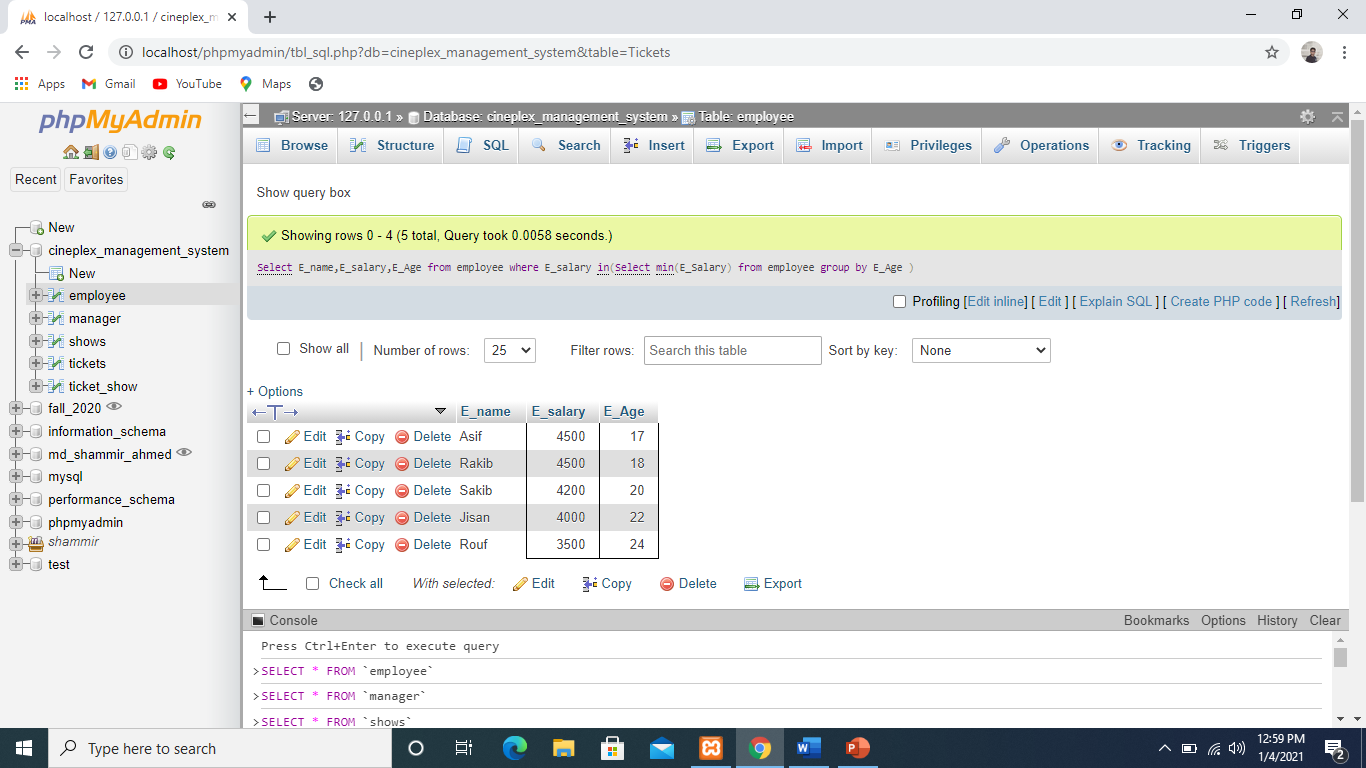
2. Write a query to display the Ticket id, Ticket name and Ticket price for all Tickets whose price more or equal than the average price. (Tickets table)

🡪Select T\_ID,T\_NAME,T\_Price from Tickets where T\_price >= (select avg(T\_price) from Tickets);



3.Write a query to display Employees name,salary and age who get the age wise lowest salary.(Employee Table)

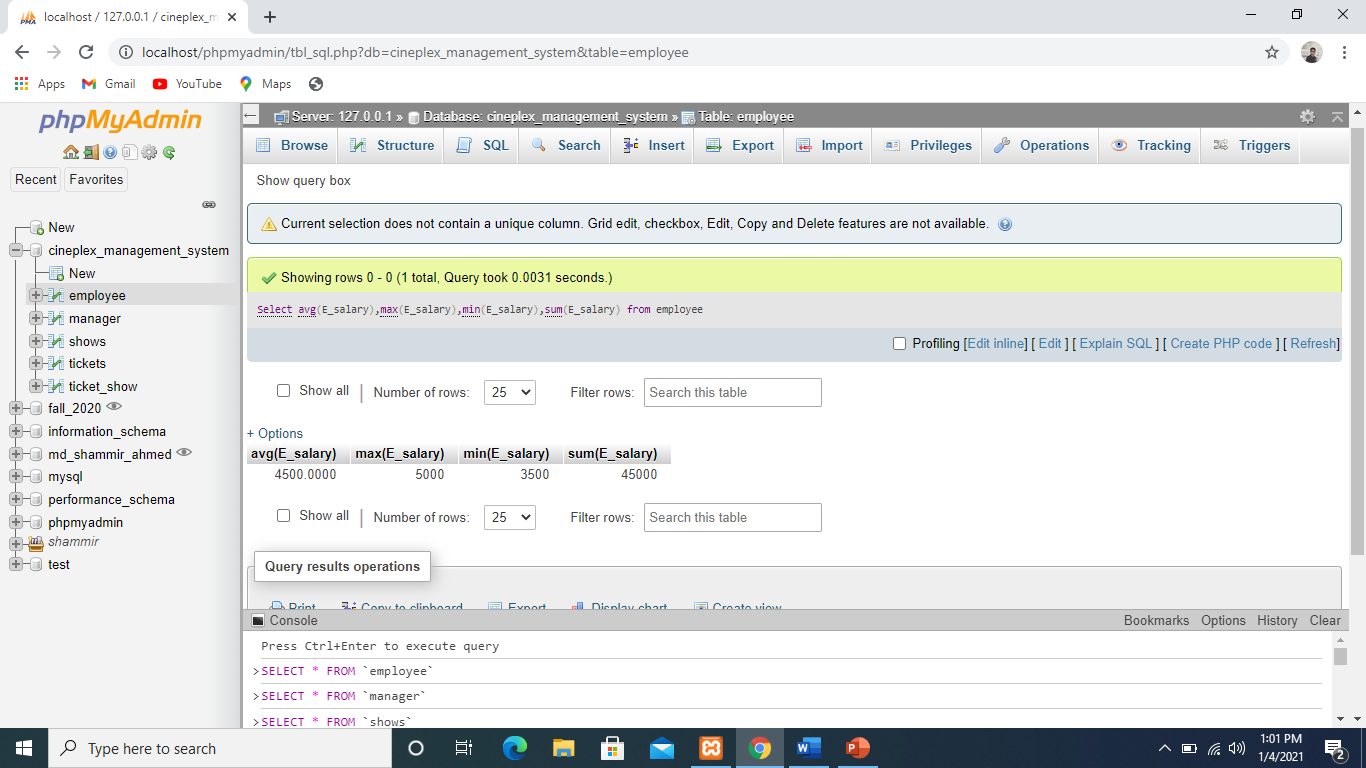
🡪Select E\_name,E\_salary,E\_Age from employee where E\_salary in(Select min(E\_Salary) from employee group by E\_Age );



**Group Function:**

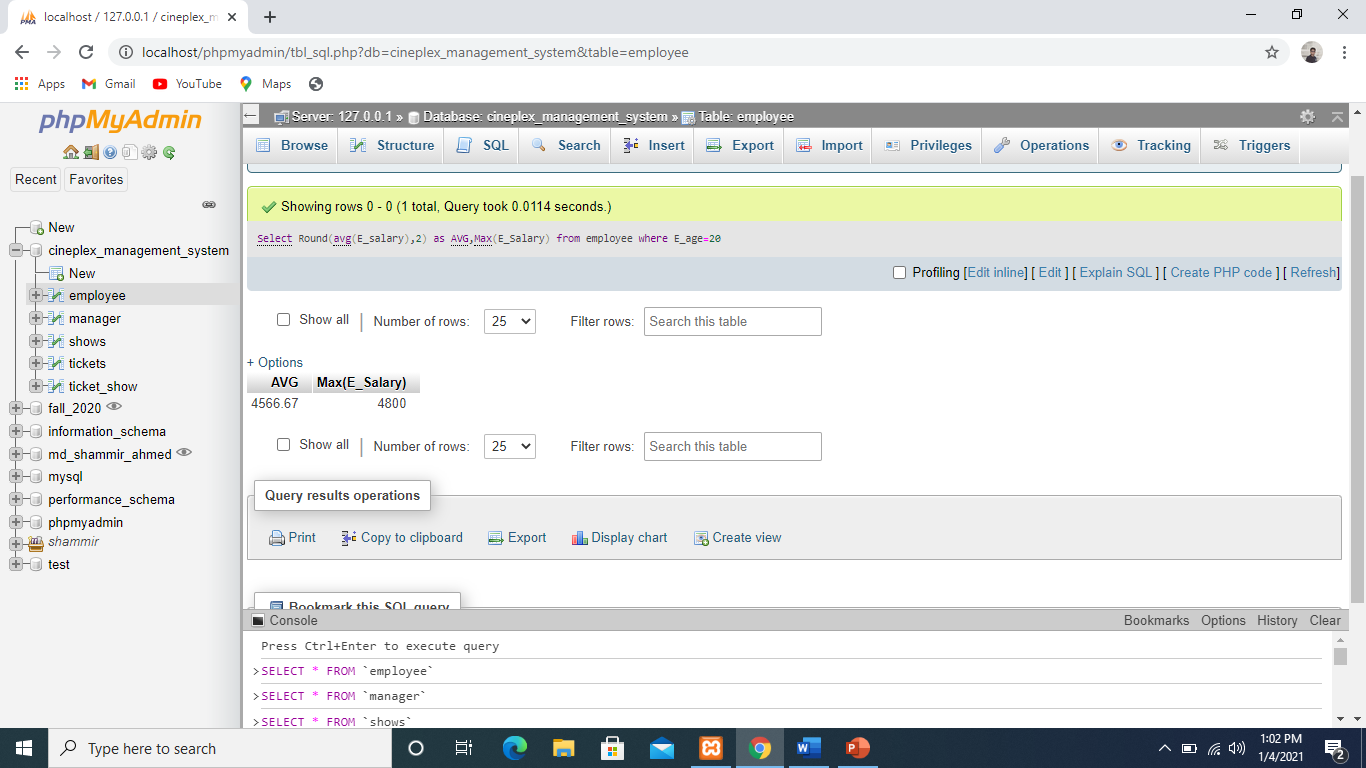
1.Write a query to display average, maximum, minimum and sum of all employees salary.(employee table)

🡪Select avg(E\_salary),max(E\_salary),min(E\_salary),sum(E\_salary) from employee;



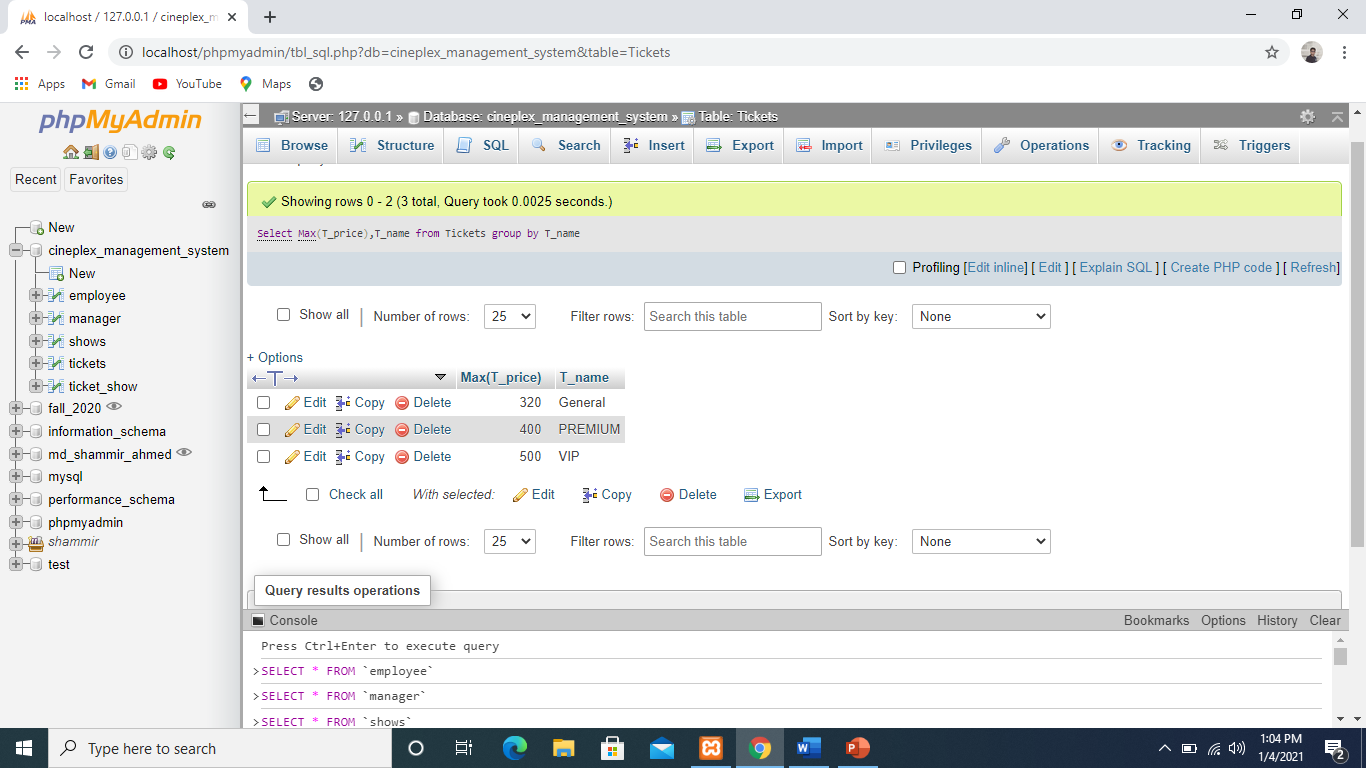
2.Find out Avg employees salary and max salary whose age is 20 and round that avg salary make its name AVG. (Employee table)

🡪Select Round(avg(E\_salary),2) as AVG,Max(E\_Salary) from employee where E\_age=20;



3.Write a query to display max tickets price ,name and group by tickets name.(Tickets Table)

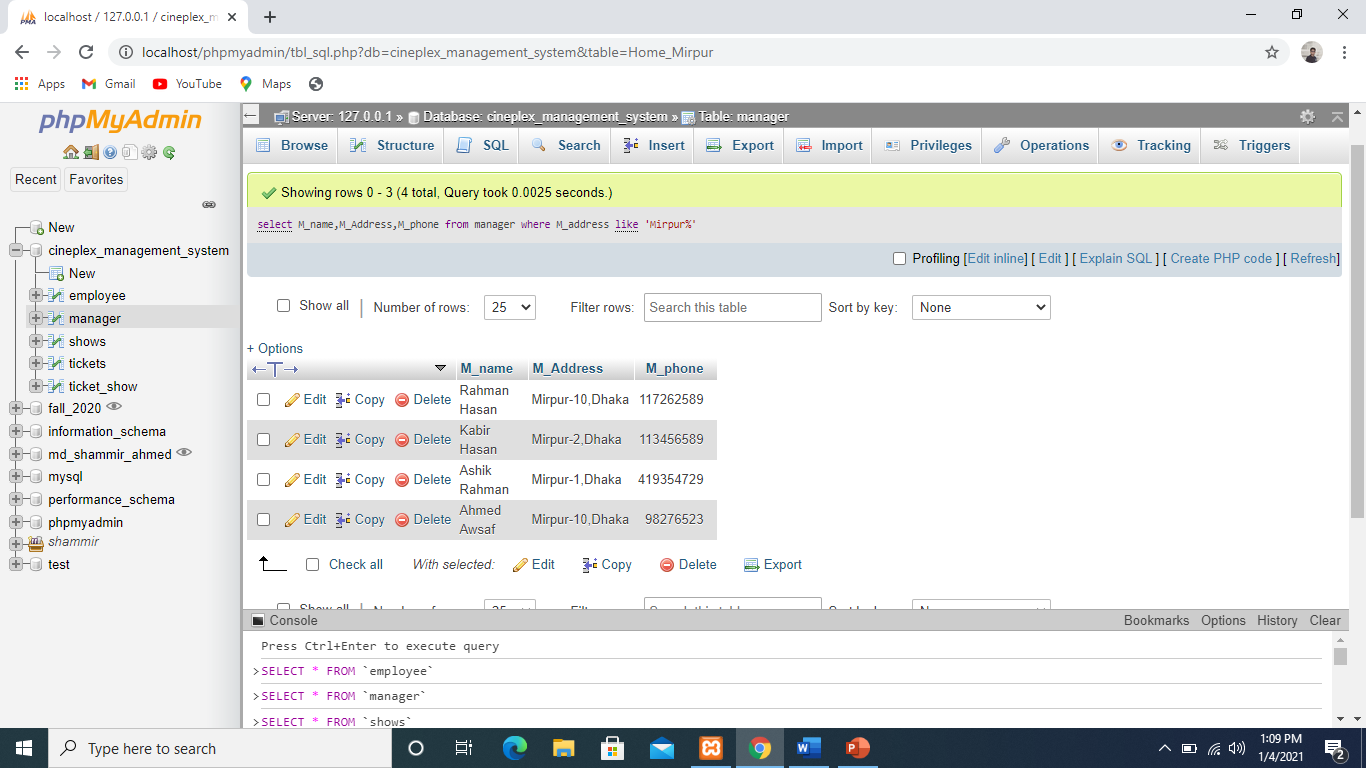
🡪Select Max(T\_price),T\_name from Tickets group by T\_name



**View:**

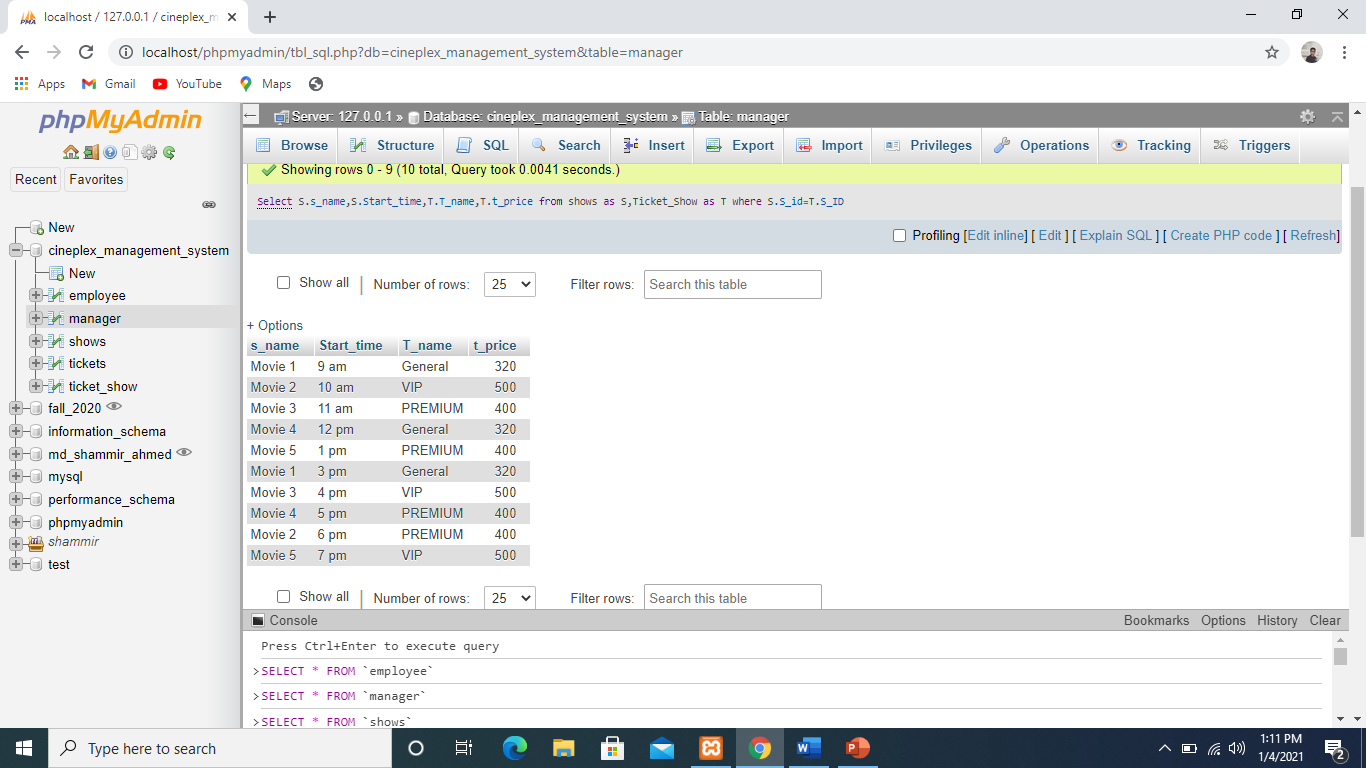
1.Create a view name Home\_Mirpur where there will be managers name,address,phone number and mangers name at least contain Mirpur.(Manager Table)

🡪 select M\_name,M\_Address,M\_phone from manager where M\_address like 'Mirpur%';



2.Create a view name Sun\_details where there will be show name, show start, ticket name, ticket price where show id=ticket show id.

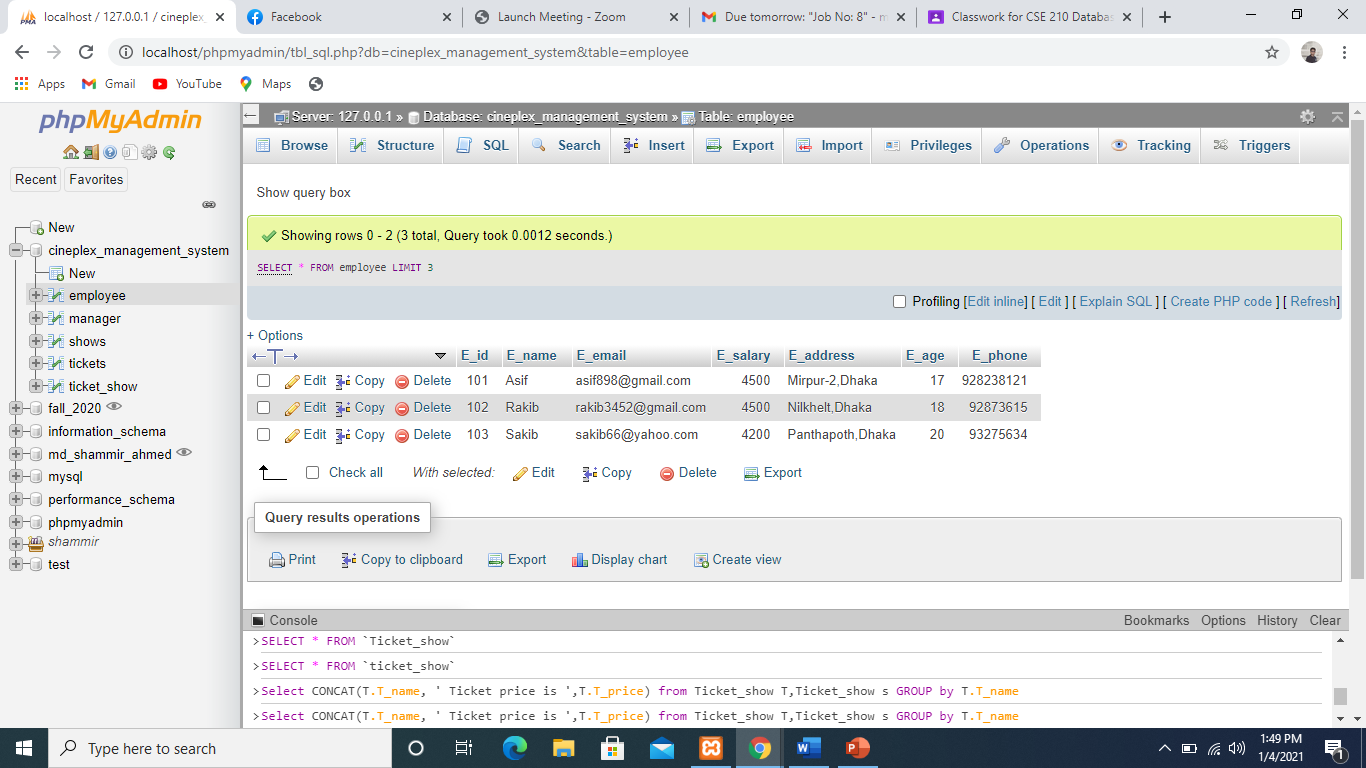
🡪 Select S.s\_name,S.Start\_time,T.T\_name,T.t\_price from shows as S,Ticket\_Show as T where S.S\_id=T.S\_ID;



**Using limit (ORDER BY, ASC, DESC)**

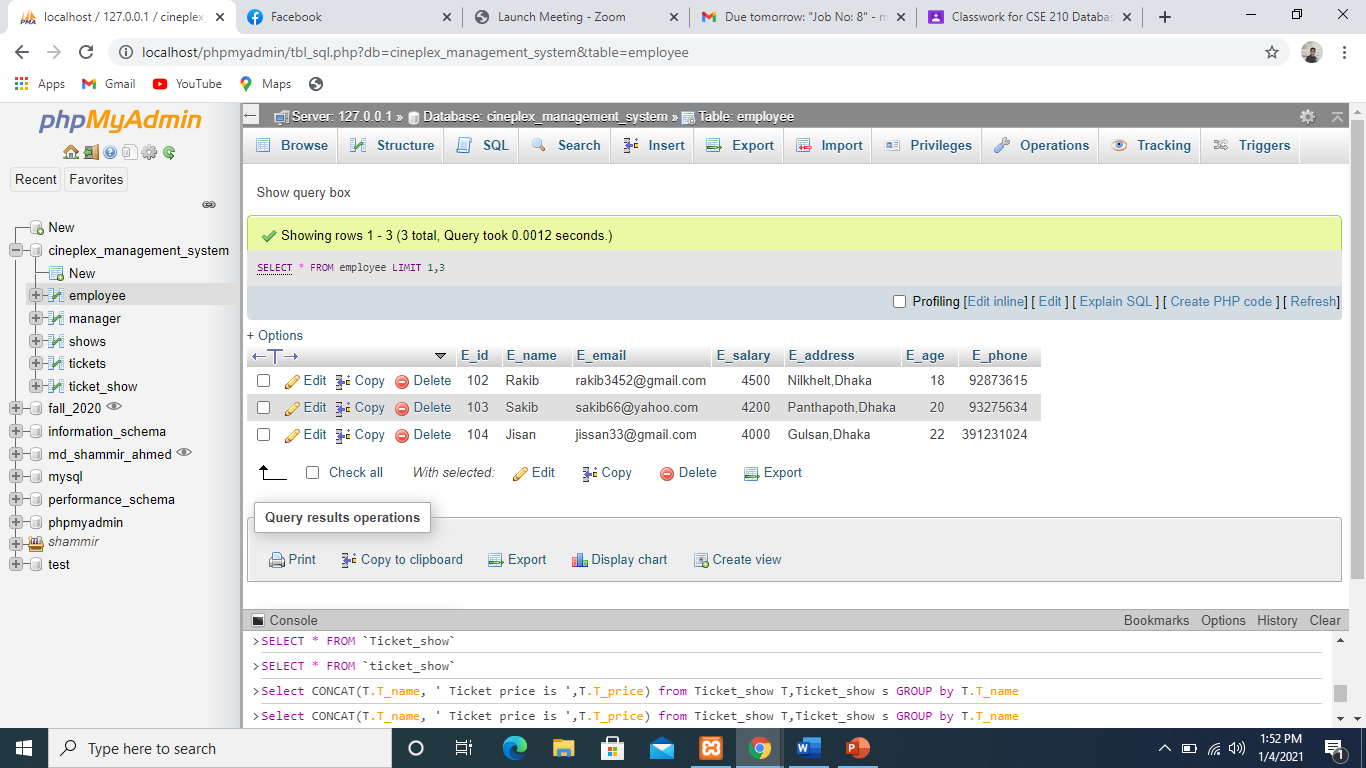
1.Select the first 3 Employee.

🡪SELECT \* FROM employee LIMIT 3;



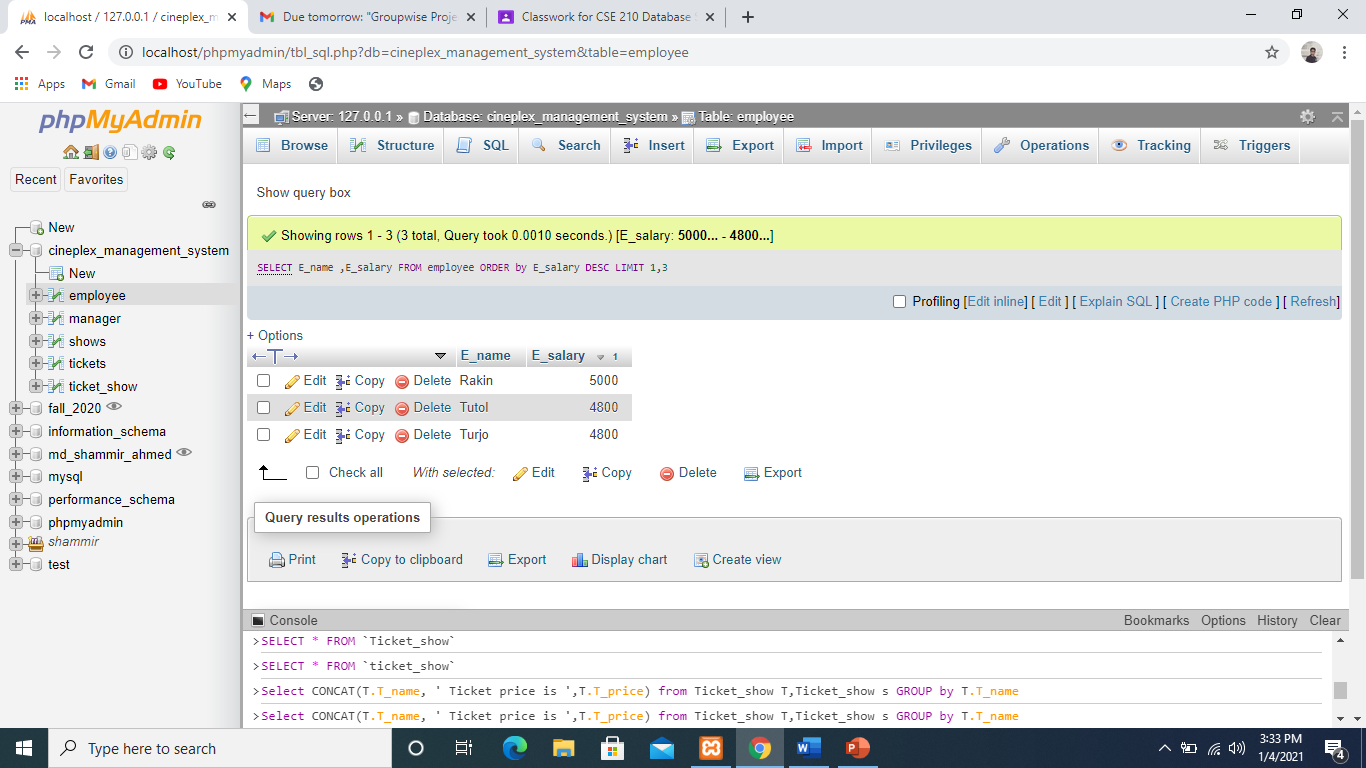
2.Select the 2nd 3 Employee.

🡪SELECT \* FROM employee LIMIT 1,3;



3. Using MySQL LIMIT to get the highest 3 values

🡪SELECT E\_name ,E\_salary FROM employee ORDER by E\_salary DESC LIMIT 1,3;

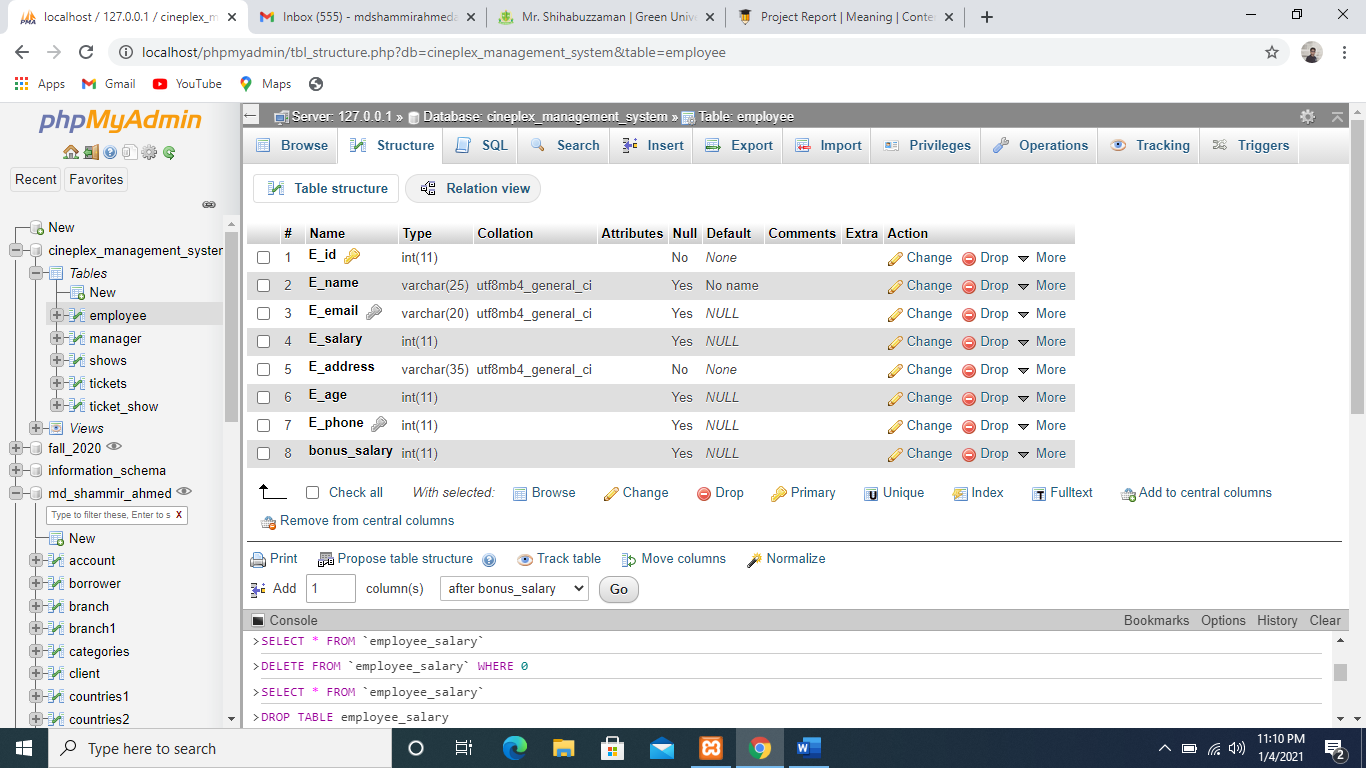


**Trigger**

1.If a employee he/she Bonus salary in a year: salary’s 50%

i) ALTER TABLE employee

ADD COLUMN bonus\_salary int;



**ii)** DELIMITER $$

CREATE TRIGGER bonus\_salary

BEFORE INSERT

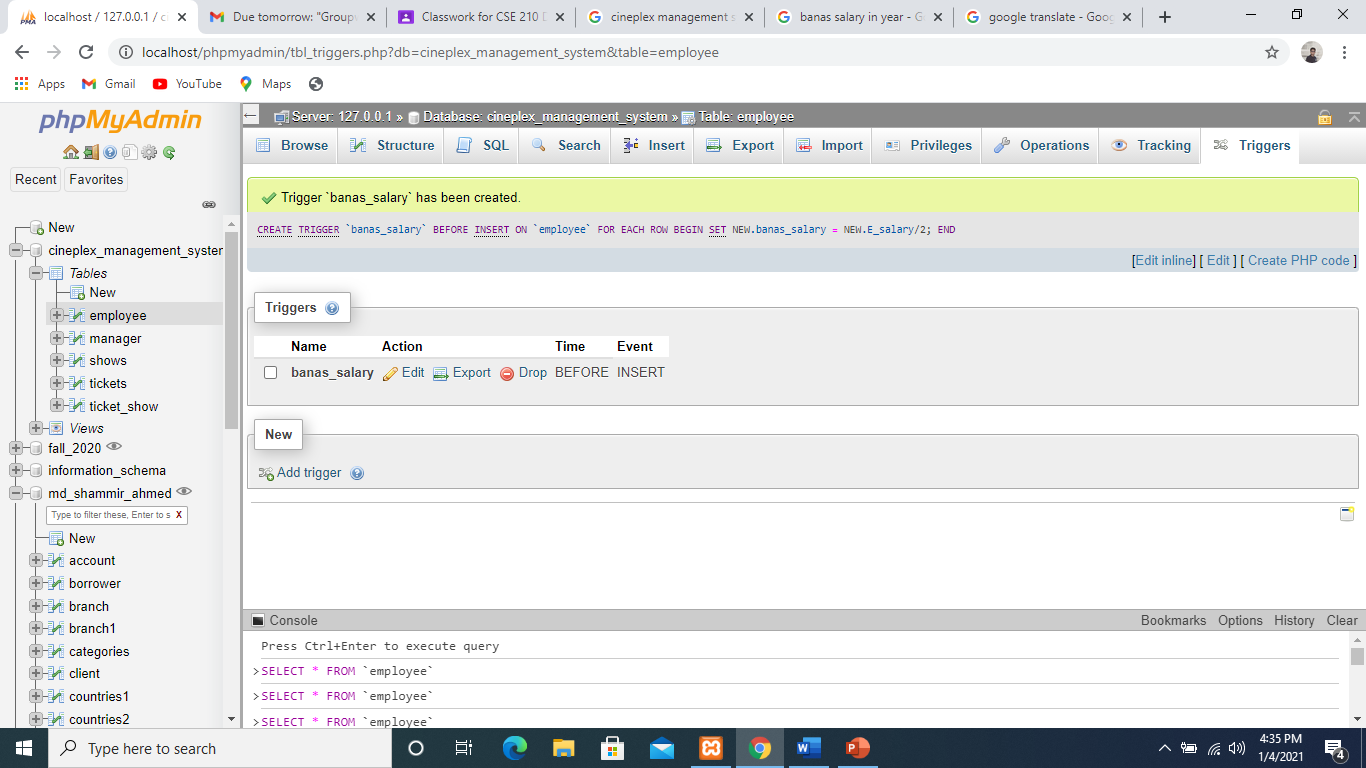
on employee FOR EACH ROW

BEGIN

SET NEW.bonus\_salary = NEW.E\_salary/2;

END$$;

DELIMITER ;



iii) select \*from employee

