Module: 5 Detabase

1)Create Table Name : Student and Exam:

[create](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) [table](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/create-table.html) student

(

 Rollno int primary key,

Name varchar(10),

Branch varchar(20)

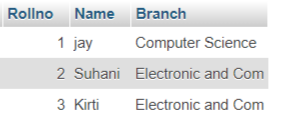
 );

insert into student values(1,"jay","Computer Science");

insert into student values(2,"Suhani","Electronic and Com");

insert into student values (3,"Kirti","Electronic and Com");

Solution:



create table Exam

(

Rollno int ,

S\_code varchar(10),

Marks int,

P\_code varchar(20),

CONSTRAINT FOREIGN KEY (Rollno)

REFERENCES student(Rollno)

);

insert into Exam values(1,"CS11",50,"CS");

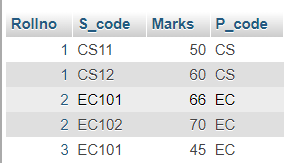
insert into Exam values(1,"CS12",60,"CS");

insert into Exam values(2,"EC101",66,"EC");

insert into Exam values(2,"EC102",70,"EC");

insert into Exam values(3,"EC101",45,"EC");

Solution:



2)Create table given below: Employee and Incentive Table:

Solution:



create table Incentive

(

Employee\_ref\_id int,

Incentive\_date varchar(10),

Incentive\_amount int

);

insert into Incentive values(1,"01-FEB-13",5000);

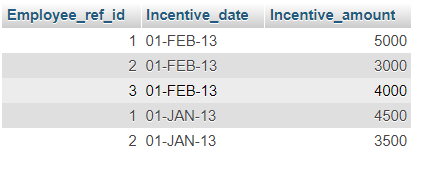
insert into Incentive values(2,"01-FEB-13",3000);

insert into Incentive values(3,"01-FEB-13",4000);

insert into Incentive values(1,"01-JAN-13",4500);

insert into Incentive values(2,"01-JAN-13",3500);

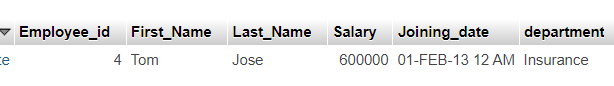
Solution:



3) Get First\_Name from employee table using Tom name “Employee Name”:

SELECT \* FROM employee WHERE First\_name= 'Tom';

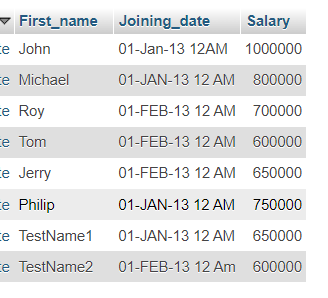
Solution:



4)Get FIRST\_NAME, Joining Date, and Salary from employee table:

SELECT First\_name,Joining\_date,Salary FROM employee ;

Solution:



5) Get all employee details from the employee table order by First\_Name Ascending and Salary descending?

SELECT \*FROM employee ORDER BY First\_Name ASC, Salary DESC;

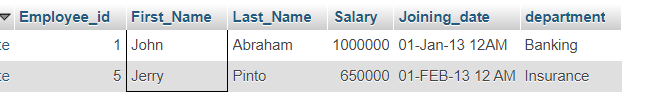
Solution:



6) Get employee details from employee table whose first name contains ‘J’:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) \* FROM employee WHERE First\_Name [LIKE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/string-comparison-functions.html%23operator_like) 'J%';

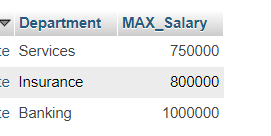
Solution:



7) Get department wise maximum salary from employee table order by 8.salary ascending?:

SELECT Department, MAX(Salary) AS MAX\_Salary FROM employee GROUP BY Department ORDER BY Max\_Salary ASC;

Solution:



10)Create After Insert trigger on Employee table which insert records in view table:

Creating View Table :

CREATE TABLE View\_Table(

View\_id int NOT Null AUTO\_INCREMENT PRIMARY KEY,

Employee\_id int,

First\_name varchar(40),

Last\_name varchar(40),

Salary int,

Joining\_date Datetime,

Department varchar(20)

);

Creating trigger:

DELIMITER //

CREATE TRIGGER AfterEmployeeinsert

AFTER INSERT ON employee

FOR EACH ROW

BEGIN

INSERT INTO view\_table (Employee\_id, First\_name, Last\_name, Salary, Joining\_date, Department)

VALUES (NEW.Employee\_id, NEW.First\_name, NEW.Last\_name, NEW.salary, NEW.Joining\_date, NEW.Department);

END//

DELIMITER ;

INSERT INTO employee VALUES (9,'Abc','xyz' ,40000, '2024-01-10 11:00:00' ,' Computer' );

INSERT INTO employee VALUES (10,'def','uvw' ,80000, '2024-01-6 11:00:00' ,' Computer' );

INSERT INTO employee VALUES (11,'hij','sdk' ,45000, '2024-01-8 11:00:00' ,' Computer' );

INSERT INTO employee VALUES (12,'klm','ksp' ,46000, '2024-01-2 11:00:00' ,' Computer' );

Employee table:

Select \* from employee;

Solution:



View\_table:

Select \* from view\_table;

Solution:



11)Create table given below: Salesperson and Customer:

Salesperson table:

CREATE TABLE Salseperson(

PK\_SNo int PRIMARY KEY,

SNAME varchar (40),

CITY varchar(40),

comm float

);

INSERT INTO salseperson VALUES(1001,’peel’,’london’,.1);

INSERT INTO salseperson VALUES(1002,'axeirod','new york',.1);

INSERT INTO salseperson VALUES(1003,'seeres','san jose',.13);

INSERT INTO salseperson VALUES(1004,'motika','london',.11);

INSERT INTO salseperson VALUES(1005,'rafkin','barcelona',.12);

Solution:



Create customer table:

CREATE TABLE Customer(

PK\_CNM int PRIMARY KEY,

CNAME varchar (40),

CITY varchar(40),

RATING int,

FK\_SNO int,

FOREIGN KEY(FK\_SNO) REFERENCES salseperson(PK\_SNO)

);

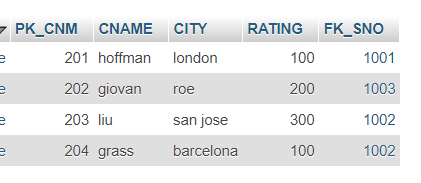
Select \* from Salseperson;

Solution:



Select \* from customer;

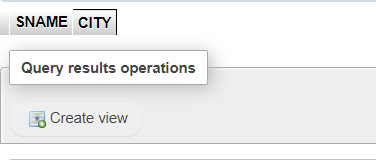
Solution:



14)Names and cities of all salespeople in London with commission above 0.12:

SELECT SNAME, CITY FROM salseperson= ' London' WHERE CITY AND comm > 0.12;

Solution:



15) All salespeople either in Barcelona or in London:

SELECT \* FROM salseperson WHERE CITY IN ('barcelona','london');

Solution:



16) All salespeople with commission between 0.10 and 0.12. (Boundary values should be excluded):

SELECT \* FROM salseperson WHERE comm>0.10 AND comm<0.12;

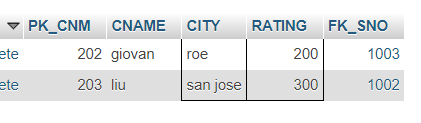
Solution:



17)All customers excluding those with rating <= 100 unless they are located in Rome:

SELECT \*FROM customer WHERE RATING > 100 OR (CITY = 'Roe');

Solution:



18) Write a SQL statement that displays all the information about all salespeople:

CREATE TABLE Salespeople

(

salesman\_id int Not Null PRIMARY KEY,

name varchar(40),

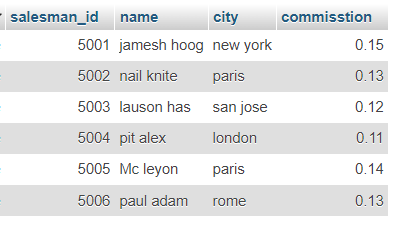
city varchar(40),

commisstion float

);

SELECT \* FROM salespeople;

Solution:



19) . From the following table, write a SQL query to find orders that are delivered by a salesperson with ID. 5001. Return ord\_no, ord\_date, purch\_amt.:

CREATE TABLE orders(

ord\_no int Not Null PRIMARY KEY,

purch\_amt float,

ord\_date DATE,

customer\_id int,

salesman\_id int,

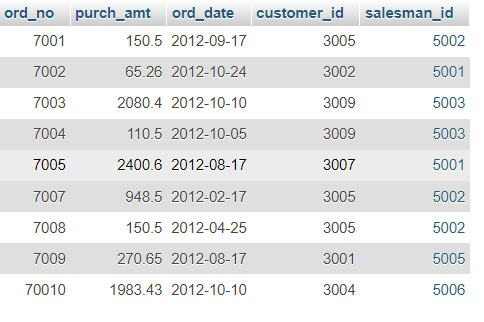
FOREIGN KEY (salesman\_id)

REFERENCES salespeople(salesman\_id)

);

SELECT \* FROM orders;

Solution:



SELECT ord\_no, ord\_date, purch\_amt FROM orders WHERE salesman\_id=5001;

Solution:



All orders for more than $1000.:

SELECT \* FROM orders WHERE purch\_amt>1000;

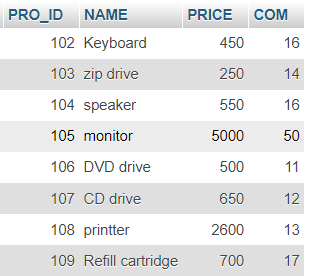
Solution:



20) From the following table, write a SQL query to select a range of products whose price is in the range Rs.200 to Rs.600. Begin and end values are included. Return pro\_id, pro\_name, pro\_price, and pro\_com.

select \* from item\_mast;

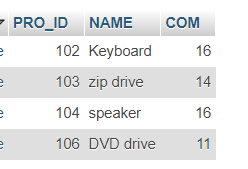
Solution:



SELECT PRO\_ID,NAME,COM FROM item\_mast

WHERE PRICE BETWEEN 200 AND 600;

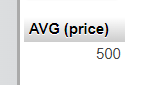
Solution:



21)From the following table, write a SQL query to calculate the average price for a manufacturer code of 16. Return avg.

SELECT AVG (price) FROM item\_mast WHERE COM=16;

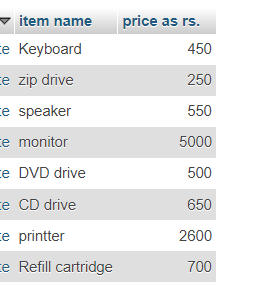
Solution:



22) . From the following table, write a SQL query to display the pro\_nameas 'Item Name' and pro\_priceas 'Price in Rs.':

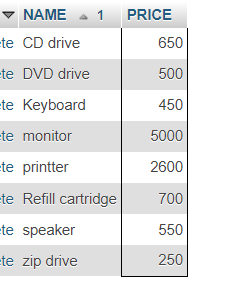
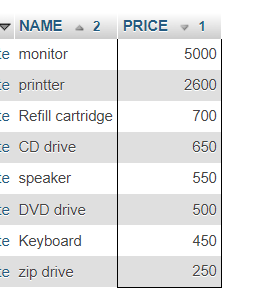
SELECT NAME AS 'item name', PRICE AS 'price as rs.' FROM item\_mast;

Solution:



23) . From the following table, write a SQL query to find the items whose prices are higher than or equal to $250. Order the result by product price in descending, then product name in ascending. Return pro\_name and pro\_price.:

Solution:

24) From the following table, write a SQL query to calculate average price of the items for each company. Return average price and company code.

SELECT AVG(PRICE) AS avg\_price, COM

FROM item\_mast

GROUP BY com;

Solution:

