1. Create Table Name : Student and Exam

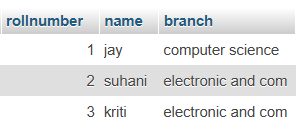
* ANS

CREATE TABLE Student ( rollnumber int PRIMARY key, name varchar ( 50),branch varchar (50));

INSERT INTO `student`(`rollnumber`, `name`, `branch`) VALUES ('1','jay','computer science'),

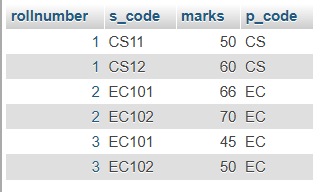
('2','suhani','electronic and com '),

('3','kriti','electronic and com');



CREATE TABLE exam (rollnumber int ,s\_code varchar (10),marks int , p\_code varchar (10), FOREIGN key (rollnumber) REFERENCES student (rollnumber));

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



INSERT INTO `exam`(`rollnumber`, `s\_code`, `marks`, `p\_code`) VALUES

('1','CS11','50','CS');

('1','CS12','60','CS');

('2','EC101','66','EC');

('2','EC102','70','EC');

('3','EC101','45','EC');

 ('3','EC102','50','EC');

1. . Create table given below: Employee and IncentiveTable

* ANS

INSERT INTO `employee`(`employee\_id`, `first\_name`, `last\_name`, `salary`, `joining\_date`, `department`) VALUES



 ('1','JOHN','ABRAHAM','1000000','2013-01-01 12:00','BANKING')

 ('2','MICHAEL','CLARKE','8000000','2013-01-01 12:00','INSURANCE')

 ('3','ROY','THOMAS','700000','2013-01-01 12:00','BANKING ')

   ('4','TOM','JOSE','8000000','2013-02-01 12:00','INSURANCE')

 ('5','JERRY','PINTO','6500000','2013-02-01 12:00','INSURANCE')

 ('6','PHILIP','MATHEW','7500000','2013-01-01 12:00','SERVICES')

 ('7','TESTNAME1','123','6500000','2013-01-01 12:00','SERVICES')

 ('8','TESTNAME2''LNAME','6000000','2013-02-01 12:00','INSURANCE');

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TABLE NAME : INCENTIVE

create table incentive(employee\_ref\_id int, incentive\_date date, incentive\_amount int, foreign key (employee\_ref\_id) REFERENCES employee (employee\_id));

INSERT INTO `incentive`(`employee\_ref\_id`, `incentive\_date`, `incentive\_amount`)

VALUES

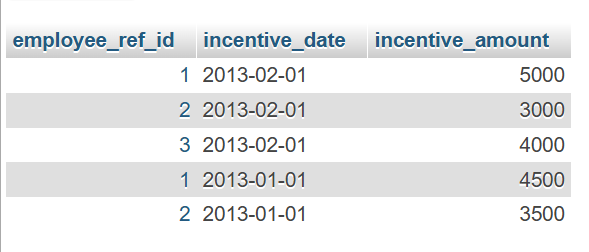
('1','2013-02-01','5000'),

('2','2013-02-01','3000'),

('3','2013-02-01','4000'),

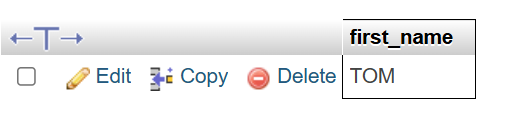
('1','2013-01-01','4500'),

('2','2013-01-01','3500');



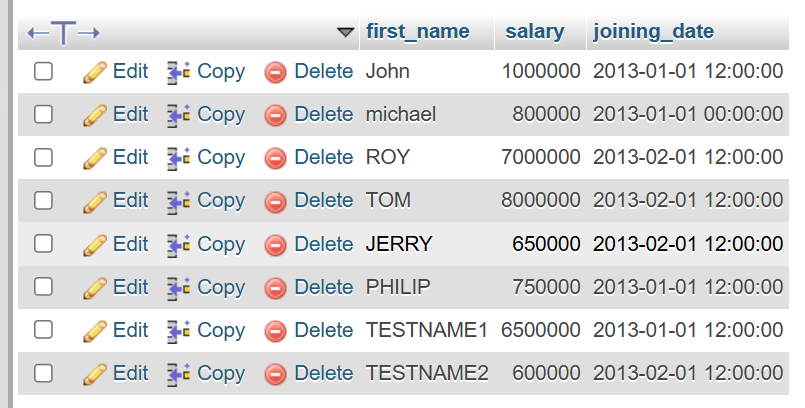
1. Get First\_Name from employee table using Tom name “Employee Name”.

* Ans [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) `first\_name` FROM `employee` WHERE first\_name= 'tom';



1. Get FIRST\_NAME, Joining Date, and Salary from employee table.

* [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) `first\_name`, `salary`, `joining\_date` FROM `employee`;



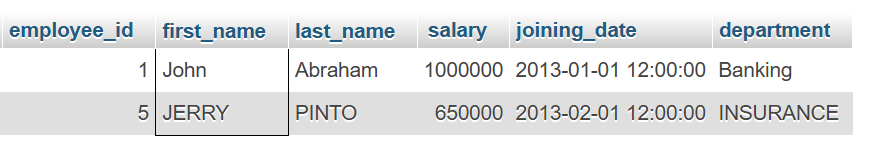
1. Get all employee details from the employee table order by First\_Name

* SELECT \* FROM `employee` order by first\_name asc, salary desc;



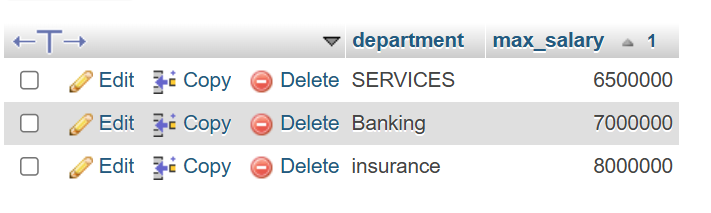
1. Get employee details from employee table whose first name contains ‘J’

* SELECT \* FROM `employee` WHERE first\_name like '%j%';



7-8.Get department wise maximum salary from employee table order by salaryascending?

* Ans [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) ‘department’, [MAX](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_max)(salary) AS max\_salary FROM ‘employee’ GROUP BY department ORDER BY max\_salary ASC;

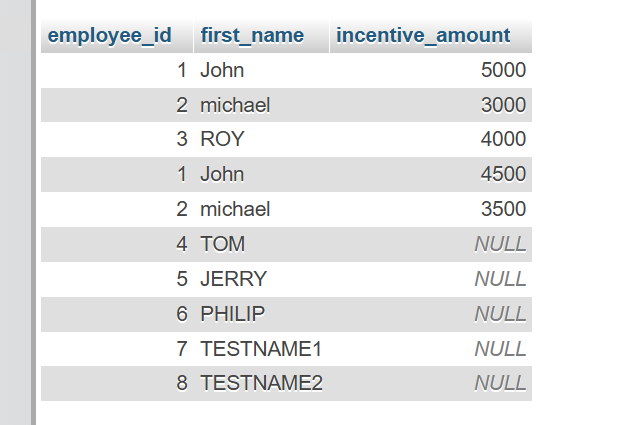


9. Select first\_name, incentive amount from employee and incentives table forthose employees who have incentives and incentive amount greater than 3000

* ANS

SELECT employee.employee\_id,employee.first\_name, incentive.incentive\_amount

from employee left join incentive on employee.employee\_id=incentive.employee\_ref\_id;



10. Create After Insert trigger on Employee table which insert records in viewtable

* Ans

CREATE TABLE ViewTable (Employee\_id INT, First\_name VARCHAR(100), Last\_name VARCHAR(100), Inserted\_at DATETIME);

DELIMITER $$

CREATE TRIGGER after\_employee\_insert AFTER INSERT ON Employee

FOR EACH ROW

BEGIN

INSERT INTO ViewTable (Employee\_id, First\_name, Last\_name, Inserted\_at)

VALUES (NEW.Employee\_id, NEW.First\_name, NEW.Last\_name, NOW());

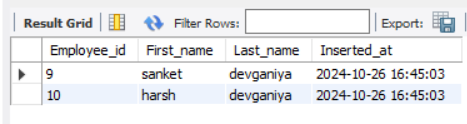
END $$

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

(9, 'sanket', 'devganiya', 50000, '2001-02-01', 'HR'),

(10, 'harsh', 'devganiya', 60000, '2002-02-01', 'sales');

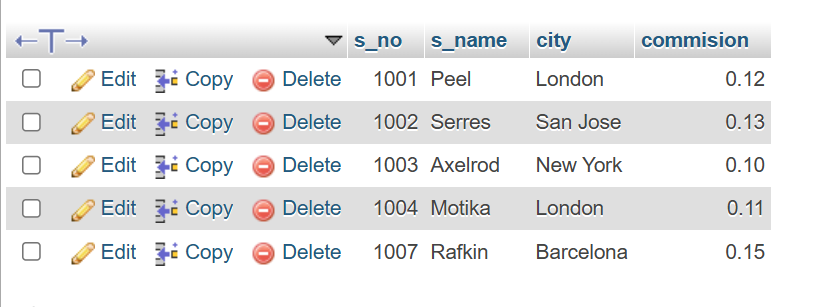
select \* from ViewTable;



11.Create table given below: Salesperson and Customer

* Ans

create table salesperson (s\_no int primary key, s\_name varchar(50), city varchar(50), commision decimal(10,2));



INSERT INTO salesperson(s\_no, s\_name, city, commision)

VALUES

('1001', 'Peel', 'London', '0.12'),

('1002', 'Serres', 'San Jose', '0.13'),

('1004', 'Motika', 'London', '0.11'),

('1007', 'Rafkin', 'Barcelona', '0.15'),

('1003', 'Axelrod', 'New York', '0.10');

* create table customer (c\_nm int primary key, c\_name varchar(50), city varchar(50),

rating int, s\_no int, foreign key(s\_no) REFERENCES salesperson(s\_no));

INSERT INTO customer(c\_nm, c\_name, city, rating, s\_no)

VALUES

('201', 'Hoffman', 'London', '100', '1001'),

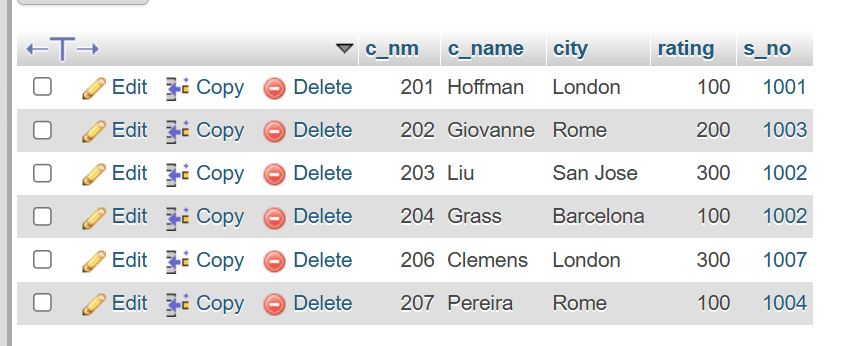
('202', 'Giovanne', 'Rome', '200', '1003'),

('203', 'Liu', 'San Jose', '300', '1002'),

('204', 'Grass', 'Barcelona', '100', '1002'),

('206', 'Clemens', 'London', '300', '1007'),

('207', 'Pereira', 'Rome', '100', '1004');



12.Retrieve the below data from above table

* Ans

Select s\_name, city from salesperson where city = ‘London’ and commision > 0.12; Select s\_name, city from salesperson where city = ‘London’ and commision > 0.12;

* --no output because there is no data where city is London and commission is above 0.12

13. All Customer name whose rating is more than 100

* Ans

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) c\_name, rating from customer where rating > 100;



14.Names and city of all salespeople in London with commission above 0.12

* Ans

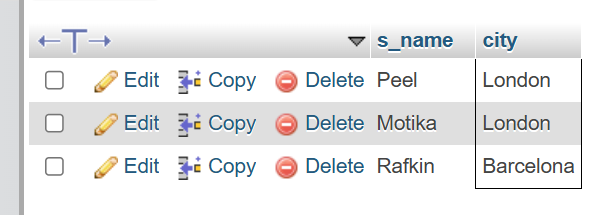
[Select](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) s\_name, city from salesperson where city = 'London' [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_and) commision > 0.12;

* --no output because there is no data where city is London and commission is above 0.12

15.All salespeople either in Barcelona or in London

* Ans

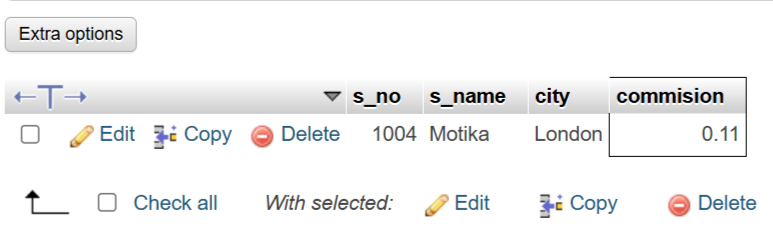
[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) s\_name,city from salesperson where city='Barcelona' [OR](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_or) city='london';



16. All salespeople with commission between 0.10 and 0.12. (Boundary valuesshould be excluded).

* Ans

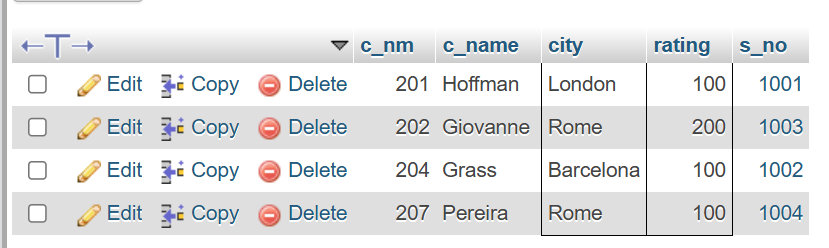
[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) \* FROM `salesperson` WHERE commision > 0.10 [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_and) commision < 0.12;



17.All customers excluding those with rating <= 100 unless they are locatedinRome

* Ans

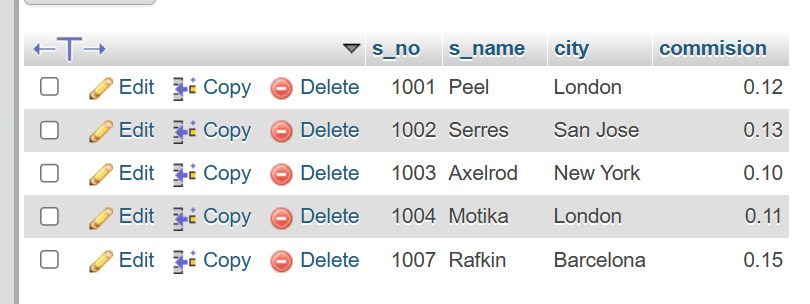
[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) \* FROM `customer` WHERE rating <= 100 [or](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_or) city = 'Rome';



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

* Ans

[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) salesman (salesman\_id int, name varchar(50), city varchar(50), commission decimal(4,2));



INSERT INTO salesman (salesman\_id, name, city, commission)

VALUES

(5001, 'James Hoog', 'New York', 0.15),

(5002, 'Nail Knite', 'Paris', 0.13),

(5005, 'Pit Alex', 'London', 0.11),

(5006, 'Mc Lyon', 'Paris', 0.14),

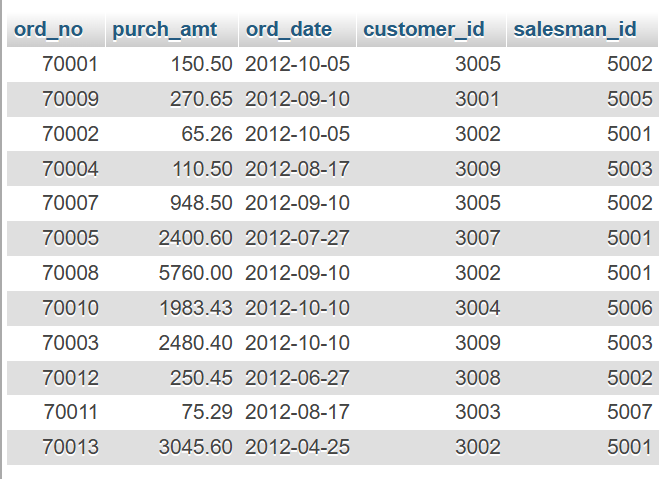
(5007, 'Paul Adam', 'Rome', 0.13),

(5003, 'Lauson Hen', 'San Jose', 0.12);

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. Sample table: orders

* ANS

CREATE TABLE orders (ord\_no INT, purch\_amt DECIMAL(10, 2), ord\_date DATE, customer\_id INT, salesman\_id INT);



INSERT INTO orders (ord\_no, purch\_amt, ord\_date, customer\_id, salesman\_id)

VALUES

(70001, 150.50, '2012-10-05', 3005, 5002),

(70009, 270.65, '2012-09-10', 3001, 5005),

(70002, 65.26, '2012-10-05', 3002, 5001),

(70004, 110.50, '2012-08-17', 3009, 5003),

(70007, 948.50, '2012-09-10', 3005, 5002),

(70005, 2400.60, '2012-07-27', 3007, 5001),

(70008, 5760.00, '2012-09-10', 3002, 5001),

(70010, 1983.43, '2012-10-10', 3004, 5006),

(70003, 2480.40, '2012-10-10', 3009, 5003),

(70012, 250.45, '2012-06-27', 3008, 5002),

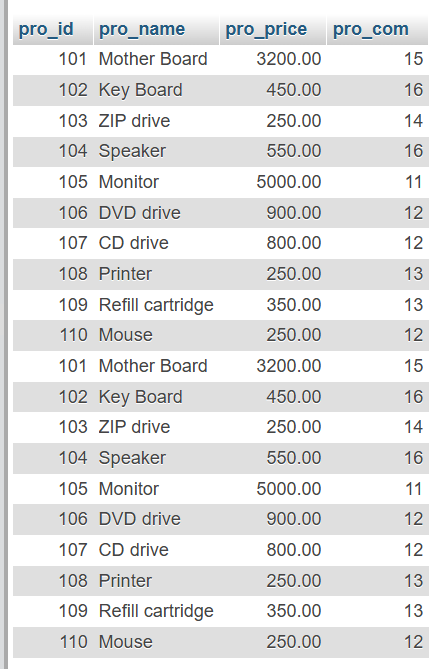
(70011, 75.29, '2012-08-17', 3003, 5007),

(70013, 3045.60, '2012-04-25', 3002, 5001);

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.

* ANS

CREATE TABLE item\_mast (pro\_id INT, pro\_name VARCHAR(50), pro\_price DECIMAL(10, 2), pro\_com INT);



INSERT INTO item\_mast (pro\_id, pro\_name, pro\_price, pro\_com)

VALUES

(101, 'Mother Board', 3200.00, 15),

(102, 'Key Board', 450.00, 16),

(103, 'ZIP drive', 250.00, 14),

(104, 'Speaker', 550.00, 16),

(105, 'Monitor', 5000.00, 11),

(106, 'DVD drive', 900.00, 12),

(107, 'CD drive', 800.00, 12),

(108, 'Printer', 250.00, 13),

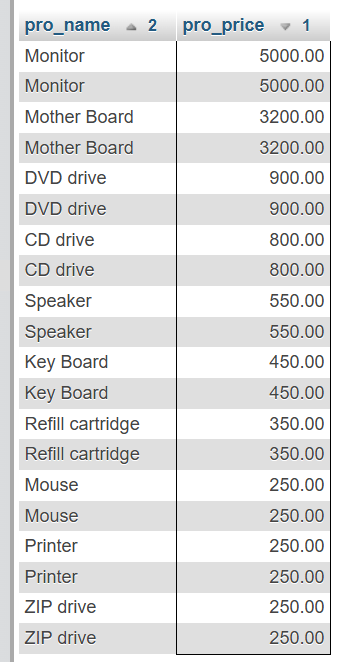
(109, 'Refill cartridge', 350.00, 13),

(110, 'Mouse', 250.00, 12);

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.

* ANS

SELECT pro\_name, pro\_price FROM `item\_mast` WHERE pro\_price >= 250 order by pro\_price desc, pro\_name asc;



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

* ANS

SELECT pro\_com, avg(pro\_price) as avg\_price FROM `item\_mast` group by pro\_com;

