# **MongoDB solutions**

Name : Vinay Shinde.

ID – 230340325076

* 1. To retrieve all documents in the restaurant collection where the name field starts with the letter P

**Command:**

**db.rest.find({name:{$regex:/\P/}})**

**Note: The command gives output of names that contain letter P.**

1.2 To retrieve all documents in the restaurant collection where the cuisine field is equal to Bakery or Chinese.

**Command:**

**hotel> db.rest.find({$in:[{cuisine:'Bakery'},{cuisine:'Chinese'}]})**

1.3 To retrieve all documents in the restaurant collection where the ‘grades.score’ field is greater than or equal to 20.

**Command:**

**hotel> db.rest.find({'grades.score':{$gte:20}})**

(OR)

**hotel> db.rest.find({grades:{$elemMatch:{"score":{$gte:20}}**

1.4 To retrieve all documents in the restaurant collection sorted by the name field in ascending order.

**Command:**

**hotel> db.rest.find().sort({name:1})**

1.5 To retrieve all documents in the restaurant collection where the grades array contains at least one element where the grade field is equal to A.

**Command:**

**hotel> db.rest.find({grades:{$elemMatch:{grade:"A"}}})**

# **MYSQL Solutions:**

**Q1.** Create a customers tables and having 4 constraints in the table(3 Marks)

**Q2**. Insert minimum three records in the table(2 Marks)

create database DBMS;

USE DBMS;

DROP TABLE IF EXISTS `customers`;

create TABLE customers (

customer\_Id int(20) not null,

First\_Name varchar(50) not null,

Last\_Name varchar(50) not null,

contact\_details varchar(50) not null,

PRIMARY KEY (`customer\_Id`)

);

insert into customers(customer\_Id,First\_Name,Last\_Name,contact\_details)

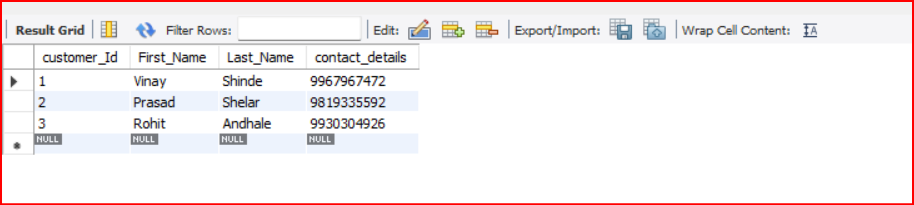
values

(1,'Vinay','Shinde','9967967472'),

(2,'Prasad','Shelar','9819335592'),

(3,'Rohit','Andhale','9930304926');

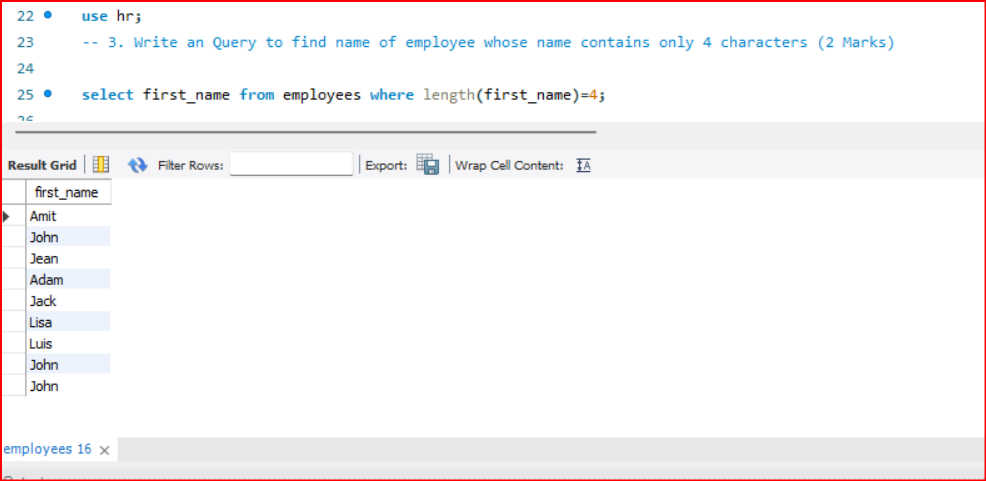
select \* from customers;



**Q3**. Write an Query to find name of employee whose name contains only 4 characters (2 Marks)

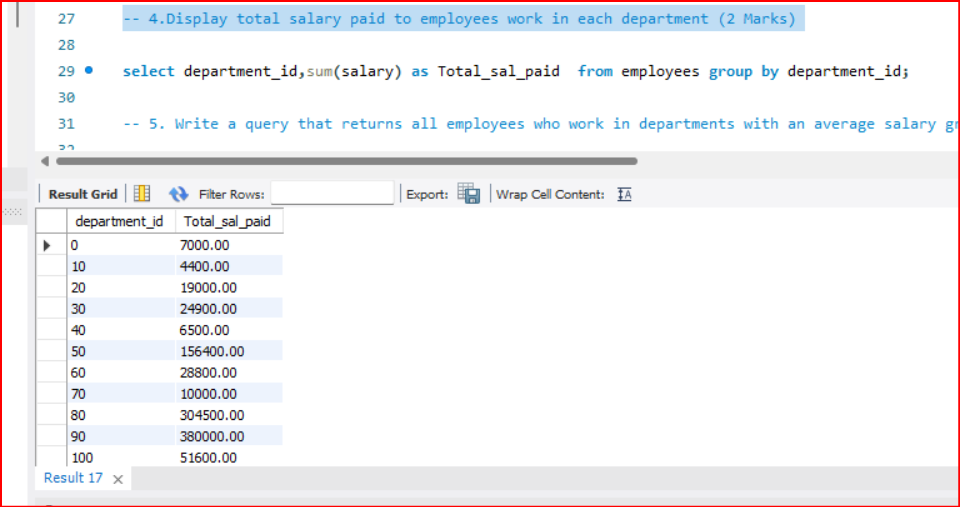
use hr;

select first\_name from employees where length(first\_name)=4;



Q 4.Display total salary paid to employees work in each department (2 Marks)

select department\_id,sum(salary) as Total\_sal\_paid from employees group by department\_id;



Q 5. Write a query that returns all employees who work in departments with an average salary greater than 75,000 per year (subquery). (3 Marks )

select concat(first\_name," ",last\_name) as full\_name from employees ;

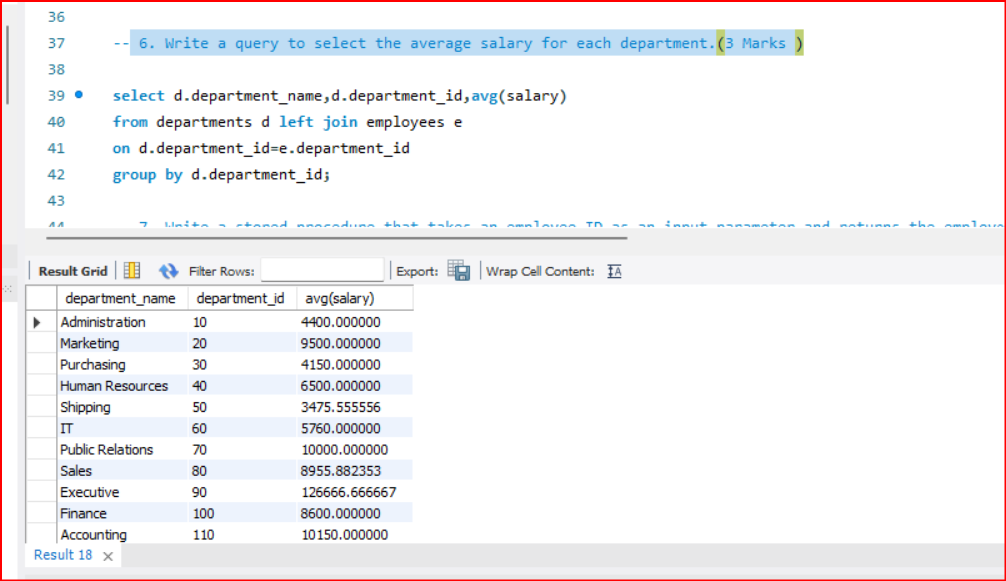
Q 6. Write a query to select the average salary for each department.(3 Marks )

select d.department\_name,d.department\_id,avg(salary)

from departments d left join employees e

on d.department\_id=e.department\_id

group by d.department\_id;



Q 7. Write a stored procedure that takes an employee ID as an input parameter and returns the employee's name, department name, and hire date..(5 Marks)

use hr;

delimiter $$

create procedure get\_details(in emp\_det int)

begin

select group\_concat(concat(e.first\_name," ",e.last\_name),",",d.department\_name,",",e.hire\_date) as details

from employees e join departments d

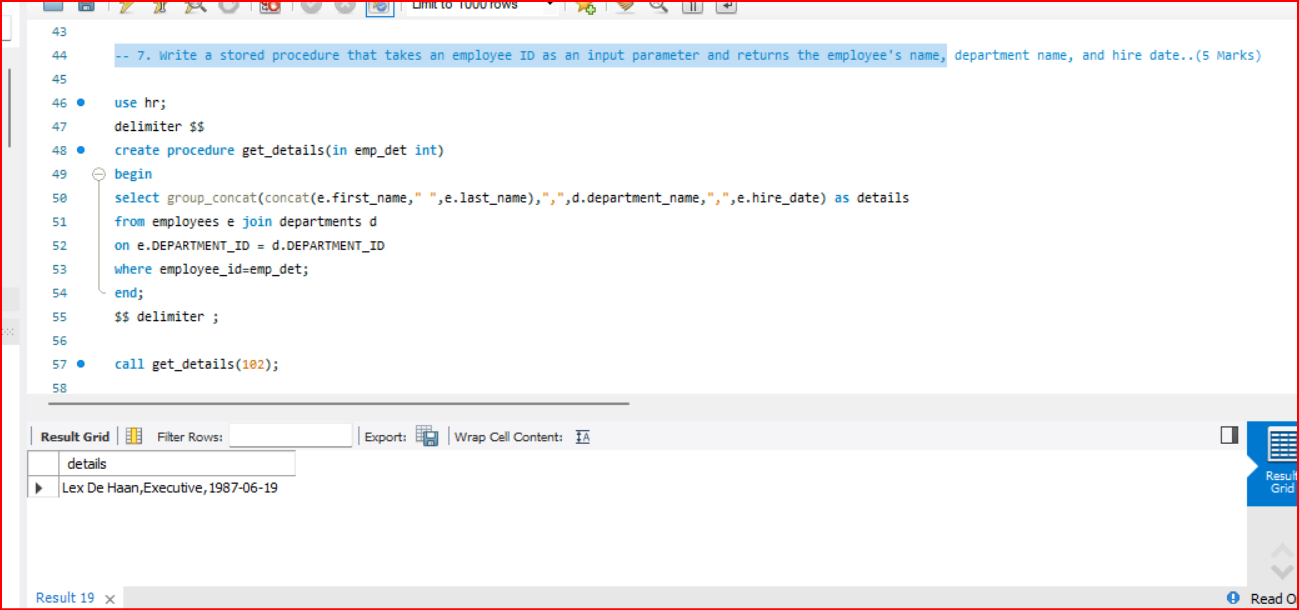
on e.DEPARTMENT\_ID = d.DEPARTMENT\_ID

where employee\_id=emp\_det;

end;

$$ delimiter ;

call get\_details(102);



Q 8. Write a stored function that takes an employee ID as an input parameter and returns the number of years the employee has been with the company.(5 Marks)

delimiter $$

create function experience(emp\_id int)

returns float(10,2)

deterministic

begin

declare experience float(10,2);

select year(end\_date)-year(start\_date) as exper into experience

from job\_history

where employee\_id=emp\_id;

return experience;

end;

$$ delimiter ;

select experience(201);

