PG-DBDA
Data Collection & DBMS
Exam Time : 3:00 PM to 4:30PM
Date : 26/04/2022

Q1 Create table employee,dept with following column and insert given data(3 Marks)

emp_id
Name - character varying
Age - Integer
hobbies character varying
salary integer
address character varying
zip integer

CONSTRAINT
Emp_id – Primary KEY
All columns not null
Salary should always greater than 0
Zip should be unique

Dept

dept_id integer
dept_name character varying
e_id integer
manager character varying

Constraints:
dept_id – Primary key
e_id - Foreign key to employee table (emp_id)

Q2 INSERT FOLLOWING DATA TO EMPLOYEE (3 Marks)

1,mohit,23,dancing, 10000, Mumbai,500049,
2,aniket,27,painting, 20000, mumbai,500149,
3,ajay,31,singing, 35000, delhi,273008,

4,priyanka,42,dancing, 55000, delhi,123876,
5,deepika,26,dancing, 10000, delhi,500786,
6,saloni,28,singing, 50000, Mumbai,400149,
7,yash,34,photography, 40000, Mumbai,450049,
8,vinay,45,painting, 70000, Mumbai,273006;

Dept table data

1,ec,8, virat,
2,cs,7, sachin,
3,it,6, rahul,
4,it,5, rahul,
5,cs,4, sachin,
6,ec,3, virat,
7,ec,2, virat,
8,ec,1, virat

Write Down Following Queries (14 Marks)

Write a Query to count No. of employees (2 Marks)

Write a Query to get unique department of employees (2 Marks)

Write a Query to get min,max,avg,sum of salary for all employees (2 Marks)

Write a Query to get highest salary of an individual based on hobbies (2 Marks)

Write a Query for sum of salary where address starts with 'M' or 'd' (2 Marks)

Write a Query to Get all employee details with their department details(2 Marks)

Write a QUERY TO FIND employees age between 20 and 30 (2 Marks)

Q3 Write a function to return name,emp_id,dept_name,hobbies,age by passing manager name (10 Marks)

MONGODB Section (10 Marks)

CREATE MONGO DB COLLECTIONS with following details and insert data
--DB = mongo exam
--Collection = assignment,inventory

--assigment data

```
  { item: "journal", qty: 25, tags: ["blank", "red"], size: { h: 14, w: 21, uom: "cm" } },
  { item: "mat", qty: 85, tags: ["gray"], size: { h: 27.9, w: 35.5, uom: "cm" } },
  { item: "mousepad", qty: 25, tags: ["gel", "blue"], size: { h: 19, w: 22.85, uom: "cm" } }
```

--inventory data

```
         { item: "journal", qty: 25, tags: ["blank", "red"], dim_cm: [ 14, 21 ] },
  { item: "notebook", qty: 50, tags: ["red", "blank"], dim_cm: [ 14, 21 ] },
  { item: "paper", qty: 100, tags: ["red", "blank", "plain"], dim_cm: [ 14, 21 ] },
  { item: "planner", qty: 75, tags: ["blank", "red"], dim_cm: [ 22.85, 30 ] },
  { item: "postcard", qty: 45, tags: ["blue"], dim_cm: [ 10, 15.25 ] }
```

1. get assignment documents having tags = gray (2 Marks)

2. Get inventory details whose dim_cm > 10 , sorted by qty descending order and print only 3 dcouments.(2 Marks)

3. Create index on inventory in descending order of qty .(2 Marks)

4. Querty to aggregare sum of qty in inventory collection(2 Marks)

5. query to update inventory collection item name where qty:75 and dim_cm > 22. (2 Marks)