Database Technologies (MongoDB) Query solving test - 1

March-24/ NoSQL/001 Time: 01:30hrs

Database Technologies (MongoDB)

Diploma in Advance Computing

March 2024

***Consider the following collection.***

*emp collection*

*{empid, ename, address: {building, street, zipcode}, gender, job, mgr, sal, deptno}*

***Solve the following queries.***

1. Write a statement to import emp.json file.
2. Write a query to display all collection names.
3. Write a query to display empid, ename and zipcode from *emp* collection.
4. Write a query to display all employees who are from 411038 zipcode.
5. Write a query to insert 2 documents in *emp* collection.
6. Write a query to display all employees who are working as ‘manager’.
7. Write a query to display first 4 documents from *emp* collection.
8. Write a query to count employee who are working as ‘manager’;
9. Write a query to display all employees who are working as ‘manager’ or ‘salesman’.
10. Write a query to display all employees whose salary greater than 2000 and less than 7000.
11. Write a query to display all employee names, give the field heading as “Employee Name”.
12. Write a query to display all employee name, his salary and new salary by adding 1000 to original salary.
13. Write a query to count document whose gender is ‘female’.
14. Write a query to print documentNumber sortedBy salary field in ascending order and also display ename, job, and salary. (use $setWindowFields stage)
15. Write a query to display last 2 documents from the *emp* collection.

**Answers**

1. mongoimport --host=192.168.100.91 --port=27017 --db=db1 --collection=e --type=json --file="C:\DBT\emp.json"
2. db.getCollectionNames();
3. db.emp.find({}, {empid:true, ename:true, "address.zipcode":true});
4. db.emp.find({"address.zipcode":411038});
5. db.emp.insertMany([{\_id:1, ename:'saleel', job:'manager'}, {\_id:2, ename:'sharmin', job:'manager' }]);
6. db.emp.find({job:'manager'});
7. db.emp.find().limit(4);
8. db.emp.find({job:'manager'}).count();
9. db.emp.find({$or:[{job:'manager'}, {job:'salesman'}]});
10. db.emp.find({$and:[{sal:{$gt:3000}}, {sal:{$lt:7000}}]});
11. db.emp.find({}, {"Employee Name" : "$ename"});
12. db.emp.find({}, {ename:true, sal:true, "New Salary" : {$add:["$sal", 1000]}});
13. db.emp.find({gender:'female'}).count();
14. db.emp.aggregate([{ $project: { \_id: false, ename: true, job: true, sal: true } }, { $setWindowFields: { sortBy: { sal: 1 }, output: { x: { $documentNumber: {} } } } }]);
15. db.emp.find().skip(db.emp.countDocuments() -2);