1. WRITE A MONGO QUERY TO DISPLAY THE LOWEST PAID EMPLOYEE'S (NAME , SALARY , DEPARTMENT NAME)

**db.employee.aggregate([**

**{"$project" : {name: "$EMPNAME",salary: "$SALARY"}}**

**,**

**{$group: { \_id: "$salary",names: {$push:{ name:"$name" }}}}**

**,**

**{$sort: {\_id: 1}}**

**,**

**{$limit: 1}**

**])**

|  |  |  |
| --- | --- | --- |
| **ENAME** | **SAL** | **DNAME** |
| **SMITH** | 800 | RESEARCH |

1. LIST MINIMUM SALARY FOR EACH DEPARTMENT

**db.employee.aggregate([**

**{"$project" : {name: "$EMPNAME",salary: "$SALARY",dept: "$DEPARTMENT.DEPARTMENT\_NAME"}}**

**,**

**{$group: { \_id: "$dept",min: {$min: "$salary"}}}**

**])**

|  |  |
| --- | --- |
| **DEPTNO** | **MIN(SAL)** |
| **10** | 1300 |
| **20** | 800 |
| **30** | 950 |

1. WRITE A QUERY BASED ON FOLLOWING RESULT.

**db.employee.find({**

**EMPNAME: "ABC"**

**}).forEach(**

**function(doc)**

**{**

**data=db.employee.aggregate({**

**$match: {"DEPARTMENT.DEPTID": doc.DEPARTMENT.DEPTID}**

**}).result**

**print(data)**

**}**

**)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **EMPNO** | **ENAME** | **JOB** | **SAL** | **DEPTNO** | **DNAME** |
| **7369** | SMITH | CLERK | 800 | 20 | RESEARCH |
| **7900** | JAMES | CLERK | 950 | 30 | SALES |
| **7934** | MILLER | CLERK | 1300 | 10 | ACCOUNTING |

1. LIST ALL THE EMPLOYEES WHO ARE WORKING IN FORD’S DEPARTMENT.

**db.employee.aggregate([**

**{**

**"$project" : {**

**name: "$EMPNAME",**

**salary: "$SALARY",**

**deptid: "$DEPARTMENT.DEPTID",**

**dept: "$DEPARTMENT.DEPARTMENT\_NAME",**

**empid: "$EMPID",**

**job: "$JOB"**

**}**

**},**

**{ "$match" : { dept: "Support" } }**

**])**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **EMPNO** | **ENAME** | **JOB** | **MGR** | **HIREDATE** | **SAL** | **DEPTNO** |
| **7369** | SMITH | CLERK | 7902 | 17-Dec-00 | 800 | 20 |
| **7566** | JONES | MANAGER | 7839 | 02-Apr-01 | 2975 | 20 |
| **7788** | SCOTT | ANALYST | 7566 | 19-Apr-07 | 3000 | 20 |
| **7876** | ADAMS | CLERK | 7788 | 23-May-07 | 1100 | 20 |
| **7902** | FORD | ANALYST | 7566 | 03-Dec-01 | 3000 | 20 |
|  |  |  |  |  |  |  |

1. LIST ALL EMPLOYEE WHO ARE WORKING IN WARD'S DEPARTMENT AND

EARNING MORE THEN MARTIN

**db.employee.find({**

**EMPNAME: "ABC"**

**}).forEach(**

**function(doc)**

**{**

**saldata=db.employee.find({EMPNAME: "Rohit"})**

**saldata.forEach(**

**function(doc2)**

**{**

**data=db.employee.aggregate(**

**{"$match" : { "DEPARTMENT.DEPTID": doc.DEPARTMENT.DEPTID , "SALARY":{$lt:doc2.SALARY} }**

**}**

**).result**

**print(data)**

**}**

**)**

**}**

**)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **EMPNO** | **ENAME** | **JOB** | **MGR** | **HIREDATE** | **SAL** | **DEPTNO** |
| **7369** | SMITH | CLERK | 7902 | 17-Dec-00 | 800 | 20 |
| **7566** | JONES | MANAGER | 7839 | 02-Apr-01 | 2975 | 20 |
| **7788** | SCOTT | ANALYST | 7566 | 19-Apr-07 | 3000 | 20 |

1. DISPLAY EMPLOYEE NUMBER, NAME,DEPT NUMBER, DEPT NAME, AND LOCATION

**db.employee.find({})**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EMPNO** | **ENAME** | **DEPTNO** | **DNAME** | **LOC** |
| **7369** | SMITH | 20 | RESEARCH | DALLAS |
| **7499** | ALLEN | 30 | SALES | CHICAGO |
| **7521** | WARD | 30 | SALES | CHICAGO |
| **7566** | JONES | 20 | RESEARCH | DALLAS |
| **7654** | MARTIN | 30 | SALES | CHICAGO |
| **7698** | BLAKE | 30 | SALES | CHICAGO |
| **7782** | CLARK | 10 | ACCOUNTING | NEW YORK |
| **7788** | SCOTT | 20 | RESEARCH | DALLAS |
| **7839** | KING | 10 | ACCOUNTING | NEW YORK |
| **7844** | TURNER | 30 | SALES | CHICAGO |
| **7876** | ADAMS | 20 | RESEARCH | DALLAS |
| **7900** | JAMES | 30 | SALES | CHICAGO |
| **7902** | FORD | 20 | RESEARCH | DALLAS |
| **7934** | MILLER | 10 | ACCOUNTING | NEW YORK |

1. DISPLAY THE FOLLOWING RESULT

**db.employee.find({},{“DEPTNO”: 1,”DEPARTMENT.DEPARTMENT\_NAME” :1, “EMPNAME”: 1}).sort({"DEPARTMENT.DEPTID": 1})**

|  |  |  |
| --- | --- | --- |
| **DEPTNO** | **DNAME** | **ENAME** |
| **10** | ACCOUNTING | CLARK |
| **10** | ACCOUNTING | KING |
| **10** | ACCOUNTING | MILLER |
| **20** | RESEARCH | JONES |
| **20** | RESEARCH | FORD |
| **20** | RESEARCH | ADAMS |
| **20** | RESEARCH | SMITH |
| **20** | RESEARCH | SCOTT |
| **30** | SALES | WARD |
| **30** | SALES | TURNER |
| **30** | SALES | ALLEN |
| **30** | SALES | JAMES |
| **30** | SALES | BLAKE |
| **30** | SALES | MARTIN |

1. LIST ALL THE EMPLOYEE WHO ARE WORKING IN NEW YORK

**db.employee.find({**

**"LOC" : 'New York'**

**},)**

|  |  |  |  |
| --- | --- | --- | --- |
| **ENAME** | **DEPTNO** | **DNAME** | **LOC** |
| **CLARK** | 10 | ACCOUNTING | NEW YORK |
| **KING** | 10 | ACCOUNTING | NEW YORK |
| **MILLER** | 10 | ACCOUNTING | NEW YORK |

1. WRITE A MONGO QUERY TO DISPLAY THE LOWEST PAID EMPLOYEE'S (NAME , SALARY , DEPARTMENT NAME) IN THE RESPECTIVE DEPARTMENT.

**db.employee.aggregate(**

**{**

**$project : {**

**name : "$EMPNAME",**

**salary : "$SALARY",**

**dept : "$DEPARTMENT.DEPARTMENT\_NAME"**

**}**

**},{**

**$sort: { salary: 1 }**

**},**

**{$group: {\_id: "$dept", name : {$first: "$name"}, salary : {$first: "$salary"}} }**

**)**

|  |  |  |
| --- | --- | --- |
| **ENAME** | **MIN(SAL)** | **DNAME** |
| **SMITH** | 800 | RESEARCH |
| **JAMES** | 950 | SALES |
| **MILLER** | 1300 | ACCOUNTING |

1. WRITE A MONGO QUERY TO DISPLAY THE HIGHEST PAID EMPLOYEE'S (NAME, JOB, MANAGER NAME, SALARY AND DEPARTMENT NAME AND DEPARTMENT NO.) IN THE RESPECTIVE DEPARTMENT.

**db.employee.aggregate(**

**{**

**$project : {**

**name : "$EMPNAME",**

**salary : "$SALARY",**

**dept : "$DEPARTMENT.DEPARTMENT\_NAME"**

**}**

**},{**

**$sort: { salary: -1 }**

**},**

**{$group: {\_id: "$dept", name : {$first: "$name"}, salary : {$first: "$salary"}} }**

**)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EMPNO** | **JOB** | **MGR** | **MAX(SAL)** | **DNAME** |
| **7698** | MANAGER | 7839 | 2850 | SALES |
| **7788** | ANALYST | 7566 | 3000 | RESEARCH |
| **7839** | PRESIDENT |  | 5000 | ACCOUNTING |
| **7902** | ANALYST | 7566 | 3000 | RESEARCH |

1. WRITE A MONGO QUERY TO DISPLAY THE EMPLOYEE NAME (BOSS) AND NUMBER OF EMPLOYEE (SUBORDINATES) DIRECTLY REPORTING TO HIM?

**db.employee.aggregate(**

**{**

**$group: {\_id:"$MGR",Subordinates:{$sum:1}}**

**}).result.forEach(**

**function(doc)**

**{**

**var a=db.employee.find({EMPID: doc.\_id})**

**print(a[0])**

**if(doc.\_id!=null)**

**{**

**print(doc)**

**}**

**}**

**)**

|  |  |
| --- | --- |
| **BOSS** | **SUBORDINATES** |
| **JONES** | 2 |
| **FORD** | 1 |
| **CLARK** | 1 |
| **SCOTT** | 1 |
| **BLAKE** | 5 |
| **KING** | 3 |

1. DISPLAY THE NAMES, DESIGNATION AND SALARIES OF ALL EMPLOYEES WHO HAVE MANAGER ALONG WITH MANAGER'S NAME, DESIGNATION AND MANAGER'S SALARY.

(SELF-JOIN)

**db.employee.find({JOB: "Manager"},{MGR: 1,EMPNAME: 1,EMPID : 1}).forEach(**

**function(doc)**

**{**

**data=db.employee.aggregate(**

**{**

**$match : { "MGR" : doc.EMPID }**

**}).result**

**if(data.length>0){**

**print(data)**

**print(doc)**

**}**

**}**

**)**

1. Create the following COLLECTION:

ORDERS:

{ Id, OrderDate, OrderNumber,

ORDER\_ITEM: {Id,

PRODUCT: {Id, ProductName}

UnitPrice, Quantity},

}

Write a query to display the following output sorted by order no:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ORDER\_NO** | **ORDER\_DATE** | **PRODUCT\_NAME** | **QUANTITY** | **UNIT\_PRICE** |
| **7369** | 7/4/2012 12:00:00 AM | EASY-TRADING | 800 | 20 |
| **7900** | 2/10/2011 12:00:00 AM | BANK-ANYWHERE | 950 | 30 |
| **7934** | 9/23/2015 12:00:00 AM | TRIP-MANAGER | 1300 | 10 |

**db.Orders.find({}).forEach(**

**function(doc)**

**{**

**var data=db.Products.aggregate({$match:{PRODUCT\_ID: doc.PRODUCT\_ID}}).result**

**data.forEach(**

**function(doc2)**

**{**

**if(doc2.PRODUCT\_ID==doc.PRODUCT\_ID)**

**{**

**print(doc)**

**print(doc2)**

**}**

**}**

**)**

**}**

**)**

1. Find the 2nd minimum salary of the employee.

**db.employee.find().sort({SALARY: 1}).limit(1).skip(1)**

1. Find the max 3 salaries from employee COLLECTION.

**db.employee.find({},{SALARY: 1}).sort({SALARY: 1}).limit(3)**

1. Display department no wise total salary where more than 2 employees exist in a department.

**db.employee.aggregate({**

**$group: {\_id: "$DEPARTMENT.DEPTID",totalsal: {$sum:"$SALARY"},sum: {$sum: 1},department: {$first: "$DEPARTMENT.DEPARTMENT\_NAME"}}**

**}).result.forEach(**

**function(doc)**

**{**

**if(doc.sum>=2)**

**{**

**print(doc)**

**}**

**}**

**)**