

```
db.createCollection('employee')
db.createCollection('department')
db.createCollection('project')
```

Inserting records into Collection Employee

```
db.employee.insertOne({
  "ssn":101,
  "name":"krupa",
  'gender':'female',
  "salary":900000,
  "depid":201,
  'qualification':['bca','mca'],
  "designation":"senior software engineer",
  "experience":9,
  "dob":"09-08-1999",
  'dependants':
  [{ 'name':'gopal','relationship':'father'},
  { 'name':'usha','relationship':'mother'}],
  'workson':[{ 'pno':301,'phour':5}]
});
```

```
db.employee.insertOne({
  "ssn":102,
  "name":"kiran",
  'gender':'male',
  "salary":9000,
  "depid":201,
  'qualification':['bsc','mca'],
  "designation":"senior software engineer",
  "experience":1,
  "dob":"29-06-2002",
  'dependants':
  [{ 'name':'gajendra','relationship':'father'}],
  'workson':[{ 'pno':301,'phour':1}]
});
```

```
});
```

```
db.employee.insertOne({  
  "ssn":103,  
  "name":"karishma",  
  'gender':'female',  
  "salary":23500,  
  "depid":201,  
  'qualification':['bca','mca'],  
  "designation":"senior software engineer",  
  "experience":5,  
  "dob":"13-04-1997",  
  'dependants':  
  [{ 'name':'varun','relationship':'fiance' }],  
  'workson':[{ 'pno':301,'phour':9},{ 'pno':302,'phour':3 }]  
});
```

```
db.employee.insertOne({  
  "ssn":104,  
  "name":"jaya",  
  'gender':'female',  
  "salary":103400,  
  "depid":202,  
  'qualification':['bsc'],  
  "designation":"junior software engineer",  
  "experience":10,  
  "dob":"29-06-1092",  
  'dependants':  
  [{ 'name':'harish','relationship':'son' }],  
  'workson':[{ 'pno':302,'phour':6 }]  
});
```

```
db.employee.insertOne({  
  "ssn":105,
```

```
"name":"ramesh",
'gender':'male',
"salary":73500,
"depid":203,
'qualification':['bcs','mca'],
"designation":"senior software engineer",
"experience":9,
"dob":"11-03-1993",
'dependants':
[{'name':'varsha','relationship':'fiance'}],
'workson': [{'pno':301,'phour':2},{'pno':302,'phour':3}]
});
```

Inserting records into Collection Department

```
db.department.insertOne({
'depid':201,
'depname':'finance',
'depmngr':101})
```

```
db.department.insertOne({
'depid':202,
'depname':'operations',
'depmngr':104})
```

```
db.department.insertOne({
'depid':203,
'depname':'customer service',
'depmngr':105})
```

Inserting records into Collection Project

```
db.project.insertOne({
'pid':301,
'pname':'Profit Report',
'depid':201,
```

```
'pstart':'2022-04-11',  
'pend':'2022-05-01'})
```

```
db.project.insertOne({  
  'pid':302,  
  'pname':'Customer Satisfaction Report',  
  'depid':201,  
  'pstart':'2022-01-19'})
```

Queries

1) Insert at least 5 values

2) Sort the employee list by ssn

Ascending : `db.employee.find().sort({"ssn":1})`

descending : `db.employee.find().sort({"ssn":-1})`

3) List the employee who are working in finance department

Shortcut method i.e. finding by Dep ID : `db.employee.find({"depid":201})`

4) Find employee who draws max salary

Works for both when more than 2 employee have same max salary

```
var sal=db.employee.distinct("salary").sort() //Created an ascending array of distinct salaries
```

```
db.employee.find({"salary":{"$eq:sal[sal.length-1]}}) //From array selected max (last) value
```

OR DO THIS IT WILL WORK SURELY

```
Var sal=db.employee.aggregate([{$group:['_id':null,MaxSal:{$max:"$salary"}]}]).toArray()[0].MaxSal
```

```
db.employee.find({salary:sal})
```

5) Update the record of employee who has worked on max projects as "Employee of the Year"

```
db.employee.update({  
  _id:db.employee.aggregate  
  ([{$project:{count:{$size:'$workson'}}},  
  {$sort:{$count:-1}}]).toArray()[0]._id,  
  {$set:{'title':'Employee of the year'}})
```

6)Delete Employee with Minimum Experience

```
db.employee.deleteOne({
experience:db.employee.aggregate
([{$group:{_id:null,
MinExp:{$min:'$experience'}}}]).toArray()[0].MinExp})
```

Additional Queries

1)Find male employee with designation senior software engineer

```
db.employee.find({$and:[{'gender':'male'},{'designation':'senior software engineer'}]})
```

2)Find employee who have specified qualification and qualification is MCA

```
db.employee.find({'qualification':'mca'})
```

3)Total number of employee in each department

```
db.employee.aggregate([{$group:{_id:"$depid",count:{$sum:1}}]})
```

4) Retrieve the details of female employees whose salary is greater than average salary of all the employees working in finance department

```
db.employee.find({
$and:[{'gender':'female'},
{'salary':{$gt:db.employee.aggregate([{$match:{'depid':201}},{$group:{_id:'$depid',avg:{$avg:'$salary'}}}]).toArray()[0].avg}}]
})
```