1. Display employees whose gender is Male.

db.employee.aggregate([{$match: {GENDER : "Male"}}])

1. Display employees who belong to London city.

db.employee.aggregate([{$match: {CITY : 'London'}}])

1. Display employees whose salary is greater than 3500.

db.employee.aggregate([{$match: {SALARY :{$gt:3500}}}])

1. Display employees whose joining date is before 2015-01-01.

db.employee.aggregate([{$match: {JOININGDATE : {$lt: '2015-01-01'}}}])

1. Display employees whose EID is greater than or equal to 7.

db.employee.aggregate([{$match: {EID :{$gte:7}}}])

1. Display employees whose city is Landon or New York (use:IN)

db.employee.aggregate([{$match: {CITY : {$in: ['London','New York']}}}])

1. Display employees who do not belongs to Landon or New York (use: NOT IN)

db.employee.aggregate([{$match: {CITY : {$nin: ['London','New York']}}}])

1. Display the EID of those employee who lives in city London

db.employee.find({CITY : 'New York'},{\_id:0,EID:1})

9. Display first 2 employee names who lives in New york.

db.employee.find({CITY : 'New York'},{\_id:0,ENAME:1}).limit(2)

1. Display next 2 employee after skipping first 2 whose city is London.

db.employee.find({CITY : 'London'},{\_id:0,ENAME:1}).limit(2).skip(2)

1. Display Male employees who lives Sydney

db.employee.find({CITY : 'Sydney',GENDER: 'Male'},{\_id:0,ENAME:1})

1. Display EID, ENAME, CITY and SALARY of those employees who belongs to London or Sydney.

db.employee.find({CITY : {$in:['Sydney','London']}},{\_id:0,ENAME:1,EID:1,SALARY:1})

1. Display ENAME, SALARY, and CITY of those employee whose salary is more than 7000

db.employee.find({SALARY : {$gt:7000}},{\_id:0,ENAME:1,SALARY:1,CITY:1})

1. Display documents whose name start with E

db.employee.find({ENAME:/^e/i})

26. Display Female students.

db.student.find({GENDER :'Female'})

1. Display students who belong to Rajkot city.

db.student.find({CITY: 'Rajkot'})

1. Display students studying in 7th sem

db.student.find({SEM: 7})

1. Display students not studying in 3rd sem.

db.student.find({SEM: {$nin:[3]}})

1. Display students whose roll no is greater than 107

db.student.find({ROLLNO: {$gt:107}})

1. Display students whose city is Jamnagar or Baroda (use:IN)

db.student.find({CITY: {$in:['Jamnagar','Baroda']}})

1. Display students whose fees is less than 9000.

db.student.find({FEES: {$lt: 9000}})

1. Display the roll no of those students who belongs to Mechanical department.

db.student.find({DEPARTMENT: 'Mechanical'})

1. Display first 2 students names who lives in Baroda.

db.student.find({CITY: 'Baroda'},{SNAME: 1,\_id: 0})

1. Display Male students who studying in 3rd sem

db.student.find({$and:[{SEM: 3},{GENDER: 'Male'}]},{SNAME: 1,\_id: 0})

1. Display sname and city and fees of those students whose roll no is less than 105.

db.student.find({ROLLNO :{$lt: 105}},{SNAME: 1,CITY:1,FEES:1,\_id: 0})

1. Display documents where sname start with K

db.student.find({SNAME : /^k/i})

1. Display documents where sname starts with Z or D in your collection

db.student.find({SNAME : /^[zd]/i})

1. Display documents where city starts with A to R in your collection

db.student.find({CITY : /^[a-r]/i})

1. Display students’ info whose name start with P and ends in i.

== db.student.find({$and:[{SNAME : /^P/},{SNAME : /i$/}]})

== db.student.find({SNAME : /^P.\*i$/i})

1. Display students’ info whose department name starts with ‘C’.

db.student.find({DEPARTMENT : /^C/i})

1. Display name, sem, fees, and department whose city contains ‘med’ as three letters somewhere in city name.

db.student.find({CITY : /med/i},{SNAME: 1,SEM: 1,FEES: 1,DEPARTMENT: 1,\_id:0})

1. Display name, sem, fees, and department who does not belongs to city Rajkot or Baroda

db.student.find({CITY : {$nin:['Rajkot','Baroda']}},{SNAME: 1,SEM: 1,FEES: 1,DEPARTMENT: 1,\_id:0})

1. Delete the documents whose city is Jamnagar.

db.student.deleteMany({CITY: 'Jamnagar'})

1. Update sname of Krish to ‘fenny’ and gender to ‘Female’

db.student.updateOne({SNAME:'Krish'},{$set:{SNAME: 'Fenny',GENDER: 'Female'}})

1. Display next 2 students after skipping first 2 whose city is Ahmedabad.

db.student.find({CITY: 'Ahmedabad'},{SNAME:1,\_id:0}).limit(2).skip(2)

1. Display rollno, sname, fees, and department of those students who is from Baroda and belongs to CE department.

db.student.find({$or:[{CITY: 'Baroda'},{DEPARTMENT: 'CE'}]},{SNAME:1,ROLLNO:1,FEES:1,DEPARTMENT:1,\_id:0})

48.Display documents where city name ends in ‘oda’nt.find({$or:[{CITY: 'Baroda'},{DEPARTMENT:'C

db.student.find({CITY: /oda$/i})

1. Display students’ info whose name contains v. (Both uppercase(V) and lowercase(v))

db.student.find({$and:[{SNAME: /v/},{SNAME: /V/}]})

50.Display students’ info whose name starts with V and having 4 characters db.student.find({SNAME: /^v.{3}$/i})

1. ,{SNAME:1,ROLLNO:1,FEES:1,DEPARTMENT:1,\_id:0})