1. Consider the following restaurant database with the following attributes -

Name, address(building, street, area, pincode),id, cuisine, nearby landmarks, online

delivery- yes/no, famous for(name of the dish)

Create 10 collections with data relevant to the following questions. Write and execute MongoDB

queries:

i. List the name and address of all restaurants in Bangalore with Italian cuisine

ii. List the name, address and nearby landmarks of all restaurants in Bangalore where north

Indian thali is available

solution:

use restaurant

db.createCollection("rest")

db.rest.insert({name:"McD",building:12,street:"KR",area:"JN",pin:072,ID:1,cuisine:"fast

food",landmark:"post office",od:"yes",famousfor:"cheese burger"});

db.rest.insert({name:"Tailo",building:14, street:"MG", area:"Bangalore",pin:45,ID:2,

cuisine:"Italian",landmark:"bank",od:"yes",famousfor:"pasta"})

db.rest.insert({name:"Kesar",building:15, street:"SS",area:"Bangalore",pin:55,ID:3, cuisine:"north

indian",landmark:"milk factory",od:"no",famousfor:"north indian thali"})

//List the name and address of all restaurants in Bangalore with Italian cuisine.

db.rest.find({cuisine:"Italian",area:"Bangalore"},

{name:1,building:1,street:1,area:1,pin:1,\_id:0}).pretty()

//List the name, address and nearby landmarks of all restaurants in Bangalore where north

Indianthali is available.

db.rest.find({famousfor:"north indian thali",area:"Bangalore"},

{name:1,building:1,street:1,area:1,pin:1,landmark:1,\_id:0})

2. Create 10 collections with data relevant to the following questions. Write and execute

MongoDB queries:

i. List the name, address and nearby landmarks of all restaurants in Bangalore where north Indian

thali is available

ii. List the name and address of restaurants and also the dish the restaurant is famous for, in

Bangalore.

//List the name, address and nearby landmarks of all restaurants in Bangalore where north

Indianthali is available.

db.rest.find({famousfor:"north indian thali",area:"Bangalore"},

{name:1,building:1,street:1,area:1,pin:1,landmark:1,\_id:0})

//List the name and address of restaurants and also the dish the restaurant is famous for, in

Bangalore.

db.rest.find({area:"Bangalore"},{name:1, building:1, street:1, area:1, pin:1, famousfor:1})

3. Consider the following restaurant table with the following attributes -

Name, address (building, street, area, pincode), id, cuisine, nearby landmarks, online

delivery- (yes/no), famous for(name of the dish)

Create 10 collections with data relevant to the following questions. Write and execute MongoDB

queries:

i. List the name, address and nearby landmarks of all restaurants in Bangalore where north Indian

thali is available.

ii. List the name and address of restaurants and also the dish the restaurant is famous for, in

Bangalore where online delivery is available.

//List the name, address and nearby landmarks of all restaurants in Bangalore where north

Indianthali is available.

db.rest.find({famousfor:"north indian thali",area:"Bangalore"},

{name:1,building:1,street:1,area:1,pin:1,landmark:1,\_id:0})

//List the name and address of restaurants and also the dish the restaurant is famous for, in

Bangalore where online delivery is available.

db.rest.find({area:"Bangalore",od:"yes"},{name:1, building:1, street:1, arear:1, pin:1, famousfor:1})

4. Consider the following Tourist places table with the following attributes -

Place, address (state), id, tourist attractions,best time of the year to visit,modes of

transport(include nearest airport, railway station etc), accommodation, food - what not to miss for

sure

Create 10 collections with data relevant to the following questions. Write and execute MongoDB

queries:

i. List all the tourist places of Karnataka

ii. List the tourist attractions of Kerala. Exclude accommodation and food

db.createCollection("tourist")

db.tourist.insert([{place:"bangalore",address:"karnataka",id:1,tour\_att:"att1",time:"jan",mode:"train",acc:"acc1",food:"chicken"}])

db.tourist.insert([{place:"kochi",address:"kerala",id:2,tour\_att:"att2",time:"feb",mode:"boat",acc:"acc2",food:"fish fry"}])

db.tourist.insert([{place:"agra",address:"delhi",id:3,tour\_att:"tajmahal",time:"march",mode:"car",acc:"acc3",food:"petha"}])

//List all the tourist places of Karnataka

db.tourist.find({address:"karnataka"},{tour\_att:true}).pretty()

//List the tourist attractions of Kerala. Exclude accommodation and food.

db.tourist.find({address:"kerala"},{acc:0,food:0}).pretty()

5. Consider the following Tourist places table with the following attributes -

Place, address (state, id), tourist attractions,best time of the year to visit,modes of

transport(include nearest airport, railway station etc), accommodation, food - what not to miss for

sure

Create 10 collections with data relevant to the following questions. Write and execute MongoDB

queries:

i. List the tourist attractions of Kerala. Exclude accommodation and food.

ii. List the places sorted state wise.

//List the tourist attractions of Kerala. Exclude accommodation and food.

db.tourist.find({address:"kerala"},{acc:0,food:0}).pretty()

//List the places sorted state wise.

db.tourist.find({},{place:1}).sort({address:1})

6. Consider the following Movie table with the following attributes -

Actor\_name,Actor\_id, Actor\_birthdate, Dirctor\_name,Director\_id, Director\_birthdate, film\_title,

year of production ,type (thriller, comedy, etc.)

Create 10 collections with data relevant to the following questions. Write and execute MongoDB

queries:

i. List all the movies acted by John in the year 2018

ii. List only the actors names and type of the movie directed by Ram

db.createCollection("movie")

db.movie.insert([{act\_n:"ram",act\_id:13,act\_bdate:"2/3/1997",dir\_n:"williams",dir\_id:101,dir\_bdat

e:"12/9/1987",film:"battleship",year:2015,type:"thriller"}])

db.movie.insert([{act\_n:"john",act\_id:11,act\_bdate:"1/2/1998",dir\_n:"ram",dir\_id:100,dir\_bdate:"2/

3/1997",film:"john wick",year:2012,type:"killer"}])

db.movie.insert([{act\_n:"elly",act\_id:12,act\_bdate:"4/12/1998",dir\_n:"ram",dir\_id:100,dir\_bdate:"2

/3/1997",film:"aquaman",year:2012,type:"action"}])

db.movie.insert([{act\_n:"ram",act\_id:13,act\_bdate:"2/3/1997",dir\_n:"thomas",dir\_id:103,dir\_bdate:

"12/3/1999",film:"xxx",year:2018,type:"action"}])

db.movie.insert([{act\_n:"john",act\_id:11,act\_bdate:"1/2/1998",dir\_n:"ram",dir\_id:100,dir\_bdate:"2/

3/1997",film:"mr.bean",year:2018,type:"comedy"}])

//List all the movies acted by John in the year 2018.

db.movie.find({$and: [{act\_n:"john"},{year:2018}]},{film:1}).pretty()

//List only the actors names and type of the movie directed by Ram.

db.movie.find({dir\_n:"ram"},{act\_n:1,type:1}).pretty()

//List all the movies acted by John and Elly in the year 2012.

db.movie.find({$and : [{act\_n:{$in : ["john","elly"]}},{year:2012}]},{film:1}).pretty()

//ii. List only the name and type of the movie where Ram has acted sorted by movie names.

db.movie.find({act\_n:"ram"},{film:1,type:1}).sort({film:1}).pretty()

//List all the movies acted by John in the year 2018.

db.movie.find({$and : [{act\_n:"john"},{year:2018}]},{film:1}).pretty()

//List only the actors names and type of the movie directed by Ram.

db.movie.find({dir\_n:"ram"},{act\_n:1,type:1}).pretty()