NAME:SHAIK SUSHMA

USN:1NT20IS411

SUBJECT:BIGDATA LAB

1. Populate the database with at least 15 documents.

```
>>> use shaik1nt20is411
switched to db shaik1nt20is411
shaik1nt20is411>
>>> db.shaik1nt20is411.insertMany([{"Name":"abc","restid":"01","cuisine":"indian","address":"bangalore","town":"eletronicity","grades":"8"},{"Name":"ARYA
BHAVAN","restid":"02","cuisine":"indian","address":"kormangala","town":"bangalore","grades":"9"},{"Name":"A2B","restid":"03","cuisine":"italian","address":"jp
nagar","town":"bangalore","grades":"8"},{"Name":"udupi","restid":"04","cuisine":"chinese","address":"jaynagar","town":"bangalore","grades":"10"},
{"Name":"raviraj","restid":"05","cuisine":"american","address":"watertank","town":"bangalore","grades":"7"},
{"Name":"grills","restid":"06","cuisine":"itailan","address":"MG road","town":"bangalore","grades":"5"},
{"Name":"kebabacenter","restid":"07","cuisine":"indian","address":"silkboard","town":"bangalore","grades":"7"},
{"Name":"swizz","restid":"08","cuisine":"amercian","address":"hebal","town":"bangalore","grades":"4"},
{"Name":"ching","restid":"09","cuisine":"chinese","address":"rajajinagar","town":"bangalore","grades":"4"},{"Name":"raj
BHAVAN","restid":"10","cuisine":"indian","address":"vv puram","town":"bangalore","grades":"7"},{"Name":"jaya
BHAVAN","restid":"11","cuisine":"indian","address":"yelahanka","town":"bangalore","grades":"8"},
{"Name":"briyanicentre","restid":"12","cuisine":"chinese","address":"yelahanka new town","town":"bangalore","grades":"6"},{"Name":"ram
BHAVAN","restid":"13","cuisine":"american","address":"baglur","town":"bangalore","grades":"9"},
{"Name":"arunbahvan","restid":"14","cuisine":"indian","address":"kammsandra","town":"bangalore","grades":"8"},{"Name":"jaya
BHAVAN","restid":"15","cuisine":"italian","address":"electronicity","town":"bangalore","grades":"6"}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("62aee7e3222ef6297f179792"),
    '1': ObjectId("62aee7e3222ef6297f179793"),
    '2': ObjectId("62aee7e3222ef6297f179794"),
    '3': ObjectId("62aee7e3222ef6297f179795"),
    '4': ObjectId("62aee7e3222ef6297f179796"),
    '5': ObjectId("62aee7e3222ef6297f179797"),
    '6': ObjectId("62aee7e3222ef6297f179798"),
    '7': ObjectId("62aee7e3222ef6297f179799"),
    '8': ObjectId("62aee7e3222ef6297f17979a"),
    '9': ObjectId("62aee7e3222ef6297f17979b"),
    '10': ObjectId("62aee7e3222ef6297f17979c"),
    '11': ObjectId("62aee7e3222ef6297f17979d"),
    '12': ObjectId("62aee7e3222ef6297f17979e"),
    '13': ObjectId("62aee7e3222ef6297f17979f"),
    '14': ObjectId("62aee7e3222ef6297f1797a0")
  }
}
shaik1nt20is411>
```

2. Write a MongoDB query to display the field's restaurant_id, name,

town and cuisine for all the documents in the collection restaurant.

```
}
shaik1nt20is411>
>>> db.shaik1nt20is411.find({},{"restid":"01","name":"abc","cuisine":"indian"})
[
  {
    _id: ObjectId("62aee7e3222ef6297f179792"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179793"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179794"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179795"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179796"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179797"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179798"),
```

```
{
    _id: ObjectId("62aee7e3222ef6297f179797"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
},
{
    _id: ObjectId("62aee7e3222ef6297f179798"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
},
{
    _id: ObjectId("62aee7e3222ef6297f179799"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
},
{
    _id: ObjectId("62aee7e3222ef6297f17979a"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
},
{
    _id: ObjectId("62aee7e3222ef6297f17979b"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
},
{
    _id: ObjectId("62aee7e3222ef6297f17979c"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
},
{
    _id: ObjectId("62aee7e3222ef6297f17979d"),
    restid: '01',
    name: 'abc',
    cuisine: 'indian'
},
{
```

```
  restid: '01',
  name: 'abc',
  cuisine: 'indian'
},
{
  _id: ObjectId("62aee7e3222ef6297f17979b"),
  restid: '01',
  name: 'abc',
  cuisine: 'indian'
},
{
  _id: ObjectId("62aee7e3222ef6297f17979c"),
  restid: '01',
  name: 'abc',
  cuisine: 'indian'
},
{
  _id: ObjectId("62aee7e3222ef6297f17979d"),
  restid: '01',
  name: 'abc',
  cuisine: 'indian'
},
{
  _id: ObjectId("62aee7e3222ef6297f17979e"),
  restid: '01',
  name: 'abc',
  cuisine: 'indian'
},
{
  _id: ObjectId("62aee7e3222ef6297f17979f"),
  restid: '01',
  name: 'abc',
  cuisine: 'indian'
},
{
  _id: ObjectId("62aee7e3222ef6297f1797a0"),
  restid: '01',
  name: 'abc',
  cuisine: 'indian'
}
]
shaik1nt20is411>
```

Press Esc to e

4. Write a MongoDB query to display the entire restaurants which is in

the town Bengaluru.

```
shaik1nt20is411>
>>> db.shaik1nt20is411.find({"town":{$in:["bangalore"]}});
[
  {
    _id: ObjectId("62aee7e3222ef6297f179793"),
    Name: 'ARYA BHAVAN',
    restid: '02',
    cuisine: 'indian',
    address: 'kormangala',
    town: 'bangalore',
    grades: '9'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179794"),
    Name: 'A2B',
    restid: '03',
    cuisine: 'italian',
    address: 'jp nagar',
    town: 'bangalore',
    grades: '8'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179795"),
    Name: 'udupi',
    restid: '04',
    cuisine: 'chinese',
    address: 'jaynagar',
    town: 'bangalore',
    grades: '10'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179796"),
    Name: 'raviraj',
    restid: '05',
    cuisine: 'american',
    address: 'watertank',
    town: 'bangalore',
    grades: '7'
  },
  {
    _id: ObjectId("62aee7e3222ef6297f179797"),
    Name: 'grills',
```

```
      address: 'yelananka',
      town: 'bangalore',
      grades: '8'
  },
  {
      _id: ObjectId("62aee7e3222ef6297f17979d"),
      Name: 'briyanicentre',
      restid: '12',
      cuisine: 'chinese',
      address: 'yelahanka new town',
      town: 'bangalore',
      grades: '6'
  },
  {
      _id: ObjectId("62aee7e3222ef6297f17979e"),
      Name: 'ram BHAVAN',
      restid: '13',
      cuisine: 'american',
      address: 'baglur',
      town: 'bangalore',
      grades: '9'
  },
  {
      _id: ObjectId("62aee7e3222ef6297f17979f"),
      Name: 'arunbahvan',
      restid: '14',
      cuisine: 'indian',
      address: 'kammsandra',
      town: 'bangalore',
      grades: '8'
  },
  {
      _id: ObjectId("62aee7e3222ef6297f1797a0"),
      Name: 'jaya BHAVAN',
      restid: '15',
      cuisine: 'italian',
      address: 'electronicity',
      town: 'bangalore',
      grades: '6'
  }
]
shaik1nt20is411>
>>>
```

7. Update the hotel by name "ARYA BHAVAN" and remove the grades

filed from documents.

```
shaik1nt20is411>
>>> db.restaurants.updateOne({name:"Arya Bhavan"},{$unset:{grades:""}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
```

8. Update the grade of all the Hotels to "A" whose score is more than 7.

```
shaik1nt20is411>
>>> db.restaurants.updateMany({"grades.0.score":{$gt:7}},{$set:{"grades.0.grade":"A"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 0
}
shaik1nt20is411>
```

9. Apply Map-reduce aggregation function to project the hotels by their

names and find the list of cuisines each hotel offers.

```
shaik1nt20is411>
>>> db.restaurants.aggregate([{$group:{_id:"$town", "maxScore":{$max:"$grades.0.score"}}}])

shaik1nt20is411>
```

10. Group all the hotels by their towns and identify the hotels with

highest score in each town.

```
shaik1nt20is411>
>>> var map=function(){emit(this.name,this.cuisine);} var reduce=function(key,val){return Array(val)} db.restaurants.mapReduce(map,reduce,{"out":"result"})
db.result.find()
```

5. Write a MongoDB query to find the restaurants which do not prepare

any cuisine of 'American' and achieved a score more than 70 and located

in the longitude less than -65.754168.

```
shaik1nt20is411>
>>> db.shaik1nt20is411.find({$and:[{cuisine:{$ne: "American"}},{"grades.0.score":{$gt: 70}},{"address.coord.longitude":{$lt: -65.754168}}]})

shaik1nt20is411>
>>> db.shaik1nt20is411.insertOne([{"name":"jase","restid":"16","cuisine":"indian","address":"airport road","town":"bangalore","grade":"80"}])
{
  acknowledged: true,
  insertedId: ObjectId("62aeed72222ef6297f1797a1")
}
shaik1nt20is411>
>>> db.shaik1nt20is411.find({$and:[{cuisine:{$ne: "American"}},{"grades.0.score":{$gt: 70}},{"address.coord.longitude":{$lt: -65.754168}}]})

shaik1nt20is411>
```

6. Write a MongoDB query to find the restaurant Id, name, borough and

cuisine for those restaurants which achieved a score which is not more

than 5.

```
shaik1nt20is411>
>>> db.shaik1nt20is411.find({"grades.0.score":{$lte:5}},{restaurant_id:1,name:1,cuisine:1})

shaik1nt20is411>
```