

NoSQL Restaurant Database

Create Database:

```
> use restaurant  
< switched to db restaurant
```

db.createCollection("Restaurant"):

Insert Values Into Collection:

```
mydb> db.Restuarant.insertMany([ { restuarant_id:"R100",name:"Meghana",town:"Bengaluru",cuisine:"Indian",score:8}, {restuarant_id:"R101",name:"Paradis  
e",town:"Mumbai",cuisine:"Chinese",score:8}, {restuarant_id:"R102",name:"Royal",town:"Delhi",cuisine:"Italian",score:10}]);  
{  
  acknowledged: true,  
  insertedIds: {  
    '0': ObjectId('6942b11d24f6d1dd001e2621'),  
    '1': ObjectId('6942b11d24f6d1dd001e2622'),  
    '2': ObjectId('6942b11d24f6d1dd001e2623')  
  }  
}
```

Queries

Write a MongoDB query to display all the documents in the collection restaurants.

```
mydb> db.Restuarant.find();
[
  {
    _id: ObjectId('6942b11d24f6d1dd001e2621'),
    restuarant_id: 'R100',
    name: 'Meghana',
    town: 'Bengaluru',
    cuisine: 'Indian',
    score: 8
  },
  {
    _id: ObjectId('6942b11d24f6d1dd001e2622'),
    restuarant_id: 'R101',
    name: 'Paradise',
    town: 'Mumbai',
    cuisine: 'Chinese',
    score: 8
  },
  {
    _id: ObjectId('6942b11d24f6d1dd001e2623'),
    restuarant_id: 'R102',
    name: 'Royal',
    town: 'Delhi',
    cuisine: 'Italian',
    score: 10
  }
]
```

Write a MongoDB query to arrange the name of the restaurants in descending along with all the columns.

```

mydb> db.Restuarant.find().sort({name:-1});
[
  {
    _id: ObjectId('6942b11d24f6d1dd001e2623'),
    restuarant_id: 'R102',
    name: 'Royal',
    town: 'Delhi',
    cuisine: 'Italian',
    score: 10
  },
  {
    _id: ObjectId('6942b11d24f6d1dd001e2622'),
    restuarant_id: 'R101',
    name: 'Paradise',
    town: 'Mumbai',
    cuisine: 'Chinese',
    score: 8
  },
  {
    _id: ObjectId('6942b11d24f6d1dd001e2621'),
    restuarant_id: 'R100',
    name: 'Meghana',
    town: 'Bengaluru',
    cuisine: 'Indian',
    score: 8
  }
]

```

Write a MongoDB query to find the restaurant Id, name, town and cuisine for those restaurants which achieved a score which is not more than 10.

```

mydb> db.Restuarant.find({score:{$lte:10}}, {_id:0,name:1,restuarant_id:1,town:1,cuisine:1});
[
  {
    restuarant_id: 'R100',
    name: 'Meghana',
    town: 'Bengaluru',
    cuisine: 'Indian'
  },
  {
    restuarant_id: 'R101',
    name: 'Paradise',
    town: 'Mumbai',
    cuisine: 'Chinese'
  },
  {
    restuarant_id: 'R102',
    name: 'Royal',
    town: 'Delhi',
    cuisine: 'Italian'
  }
]

```

Write a MongoDB query to find the average score for each restaurant

```

mydb> db.Restuarant.aggregate([{$group:{_id:"$restuarant_id",restuarantName:{$first:"$name"},averageScore:{$avg:"$score"}}}]);
[
  { _id: 'R101', restuarantName: 'Paradise', averageScore: 8 },
  { _id: 'R102', restuarantName: 'Royal', averageScore: 10 },
  { _id: 'R100', restuarantName: 'Meghana', averageScore: 8 }
]

```