

QUADERNO #4 MongoDB

Carlo MIGLIACCIO

matricola: 332937

Domanda 1

```
db.bikeStations.find(
  { "extra.status": "online" },
  { "extra.status": 1 }
).count();
```

33

```
db.bikeStations.find(
  { "extra.status": "offline" },
  { "extra.status": 1 }
).count();
```

28

Domanda 2

```
db.bikeStations.find(
  {"extra.status": {$nin: ['online', 'offline']}}
).count();
```

4

Domanda 3

```
db.bikeStations.find(
  {"extra.status": {$nin: ['online', 'offline']},
   {_id: 0, "extra.status": 1 }
);
```

Domanda 4

```
db.bikeStations.find(
  {
    "extra.status": "online",
    "extra.score": {$gte: 4}
  }
).count();
```

18

```
db.bikeStations.find(
  {
    "extra.status": "online",
    "extra.score": {$gte: 4}
  },
  {_id: 0, name: 1}
).sort({name: 1});
```

Domanda 5

```
db.bikeStations.find(
  {
    "extra.status": "offline",
    $or: [
      {free_bikes: {$gt: 0}},
      {empty_slots: {$gt: 0}}
    ]
  },
  { free bikes: 1, empty slots: 1, id: 0 }
);
```

Domanda 6

```
db.bikeStations.aggregate(
  {
    $group: {
      id: null,
      Totale: {$sum: "$extra.reviews"}
    }
  }
);
```

15311**Domanda 7**

```
db.bikeStations.aggregate(
  {
    //Pipeline di aggregazione
    $group: {                                //1° stage
      id: score,
      Count: {$sum: 1}
    },
    $sort: { id: -1}                          //2° stage
  }
);
```

Domanda 8

```
db.bikeStations.aggregate(
  {
    $match: {"extra.status": {$in: ['online', 'offline']}},           //1°
stage
    $group: {_id: "$extra.status", AvgScore: {$avg: "$extra.score"}   //2°
stage
  }
);
```

Domanda 9

```
/*=====
Map: Oggetto → {punteggio, bici libere}
-----*/
var mapFunction = function(){
  var key=this.id;
  var value = {punti: this.extra.score, libere: this.freebikes};
  emit(key, value);
  return;
};

/*=====
Reduce: [] → {MediaVotate: ..., MediaDisponibili: ...}
-----*/
var reduceFunction = function(keys, values){
  var Ris = {Avg0: 0.0, AvgA: 0.0};
  var Count0=0, CountA=0;
  var Acc0=0, AccA=0;

  for(i=0; i<keys.length; i++){
    if(values[i].punti>0) AccA+=values[i].punti, CountA++;
    else Acc0+=values[i].punti, Count0++;
  }
  Ris.Avg0 = Acc0/Count0;
  Ris.AvgA = AccA/CountA;

  return Ris;
};

db.bikeStations.mapReduce(mapFunction, reduceFunction, {out: "mapReduce_Ris1"});
```

Domanda 10

```

/*=====
Map: Oggetto → {punteggio, bici libere}
-----*/
var mapFunction2 = function(){
  var key=this.id;
  var value = {punti: this.extra.score, libere: this.freebikes, status:
this.extra.status};
  emit(key, value);
  return;
};

/*=====
Reduce: [] → {MediaVote: ..., MediaDisponibili: ...}
-----*/
var reduceFunction2 = function(keys, values){
  var Ris = {Avg0: 0.0, AvgA: 0.0};
  var Count0=0, CountA=0;
  var Acc0=0, AccA=0;

  for(i=0; i<keys.length; i++){
    //Solo le stazioni attive passano alla fase di calcolo
    if(values[i].status=='online') {
      if(values[i].punti>0)   AccA+=values[i].punti, CountA++;
      else                   Acc0+=values[i].punti, Count0++;
    }
  }
  Ris.Avg0 = Acc0/Count0;
  Ris.AvgA = AccA/CountA;

  return Ris;
};

db.bikeStations.mapReduce(mapFunction2, reduceFunction2, {out: "mapReduce_Ris2"});

```

Domanda 11

```

//Occorre che ci sia un indice geospaziale per fare le query 11 e 12
db.bikeStations.createIndex({"location": "2dsphere"});

db.bikeStations.find(
  {
    location: {
      $near: {
        $geometry: { type:"Point", coordinates:[45.07456,
7.69463] }
      }
    },
    free_bikes: {$gt: 0}
  },
  {name: 1, free_bikes: 1}
).limit(3);

```

Domanda 12

```

//Occorre che ci sia un indice geospaziale per fare le query 11 e 12
db.bikeStations.createIndex({"location": "2dsphere"});

db.bikeStations.find(
  {
    location: {
      $geometry: { type: "Point",
coordinates: db.bikeStations.findOne({name: "Politecnico
4"})
.location.coordinates }
    },
    free_bikes: {$gt: 0}
  },
  {location: 1, free bikes: 1}
).limit(3);

```

```

{ free_bikes: 9, name: 'Politecnico 1' },
{ free_bikes: 5, name: 'Politecnico 3' },
{ free_bikes: 3, name: 'Tribunale' }

```