

1. Hiển thị tất cả document trong restaurants

db.restaurants.find()

```
> db.restaurants.find()
< {
  _id: ObjectId('57506d62f57802807471dcd1'),
  grades: [
    2,
    6,
    10,
    9,
    14
  ],
  name: 'Morris Park Bake Shop',
  contact: {
    phone: '365-555-0134',
    email: 'morrisparkbakeshop@example.org',
    location: [
      -73.856077,
      40.848447
    ]
  },
  stars: 4,
  categories: [
    'Italian',
  ]
}
```

2. Đếm số lượng documents của restaurants

db.restaurants.countDocuments()

```
> db.restaurants.countDocuments()
< 10000
```

3. Hiển thị id, name, phone và categories của tất cả các document trong restaurants

```
db.restaurants.find(  
  {},  
  { _id: 1, name: 1, phone: 1, categories: 1 }  
)
```

```
> db.restaurants.find(  
  {},  
  { _id: 1, name: 1, phone: 1, categories: 1 }  
)  
< {  
  _id: ObjectId('57506d62f57802807471dcd1'),  
  name: 'Morris Park Bake Shop',  
  categories: [  
    'Italian',  
    'Polynesian',  
    'Coffee'  
  ]  
}  
{  
  _id: ObjectId('57506d62f57802807471dcd2'),  
  name: 'Dj Reynolds Pub And Restaurant',  
  categories: [  
    'Pancakes',  
    'Californian',  
    'African'  
  ]  
}
```


5. Hiển thị 5 nhà hàng tiếp theo sau khi skip 5 nhà hàng đầu tiên có stars > 3, sort theo thứ tự từ cao đến thấp

```
db.restaurants.find(  
  { stars: { $gt: 3 } }  
)  
.sort(  
  { stars: -1 }  
)  
.skip(5).limit(5)
```

6. Tìm các nhà hàng có vị trí latitude < -95.754168

```
db.restaurants.find(  
  { "location.latitude": { $lt: -95.754168 } }  
)
```

```
> db.restaurants.find(  
  { stars: { $gt: 3 } }  
)  
.sort(  
  { stars: -1 }  
)  
.skip(5).limit(5)  
< {  
  _id: ObjectId('57506d62f57802807471dd04'),  
  grades: [  
    12,  
    12,  
    7,  
    5  
  ],  
  name: 'Plaza Bagels and Deli',  
  contact: {  
    phone: '674-555-0124',  
    email: 'plazabagelsanddeli@example.net',  
    location: [  
      -74.1178949,  
      40.5734906
```

7. Tìm các nhà hàng không có món ăn Californian, điểm stars ≥ 4 , và latitude < -65.754168 , sắp xếp theo thứ tự stars giảm dần

```
db.restaurants.find(  
  {  
    categories: { $ne: "Californian" },  
    stars: { $gte: 4 },  
    "location.latitude": { $lt: -65.754168 }  
  }  
)  
.sort(  
  { stars: -1 }  
)
```

8. Hiển thị `_id`, `name`, `stars`, `categories`, `phone`, và `email` của các nhà hàng có tên bắt đầu bằng 'Wil'

```
db.restaurants.find(  
  { name: { $regex: '^Wil', $options: 'i' } },  
  { _id: 1, name: 1, stars: 1, categories: 1, phone: 1, email: 1 }  
)
```

```
> db.restaurants.find(  
  { name: { $regex: '^Wil', $options: 'i' } },  
  { _id: 1, name: 1, stars: 1, categories: 1, phone: 1, email: 1 }  
)  
< {  
  _id: ObjectId('57506d62f57802807471dcd9'),  
  name: "Wilken'S Fine Food",  
  stars: 4,  
  categories: [  
    'Hotdogs',  
    'Vietnamese',  
    'Italian',  
    'Tex-Mex',  
    'Italian'  
  ]  
}
```

9. Hiển thị `_id`, `name`, `stars`, `categories`, `phone`, và `email` của các nhà hàng có tên kết thúc bằng 'ces'

```
db.restaurants.find(  
  { name: { $regex: 'ces$', $options: 'i' } },  
  { _id: 1, name: 1, stars: 1, categories: 1, phone: 1, email: 1 }  
)
```

10. Hiển thị `_id`, `name`, `stars`, `categories`, `phone`, và `email` của các nhà hàng có tên chứa các ký tự 'Reg'.

```
db.restaurants.find(  
  { name: { $regex: 'Reg', $options: 'i' } },  
  { _id: 1, name: 1, stars: 1, categories: 1, phone: 1, email: 1 }  
)
```

```
> db.restaurants.find(  
  { name: { $regex: 'ces$', $options: 'i' } },  
  { _id: 1, name: 1, stars: 1, categories: 1, phone: 1, email: 1 }  
)  
< {  
  _id: ObjectId('57506d63f57802807471e163'),  
  name: 'Pieces',  
  stars: 1,  
  categories: [  
    'Eastern European',  
    'Cuban',  
    'Soups and Sandwiches',  
    'Asian'  
  ]  
}
```

11. Tìm các nhà hàng có cả ba món Russian, Chinese và Vietnamese

```
db.restaurants.find(  
  {  
    categories: { $all: ["Russian", "Chinese", "Vietnamese"] }  
  }  
)
```

```
> db.restaurants.find(  
  {  
    categories: { $all: ["Russian", "Chinese", "Vietnamese"] }  
  }  
)  
< {  
  _id: ObjectId('57506d63f57802807471f41c'),  
  grades: [  
    10,  
    11,  
    17,  
    27,  
    27,  
    24  
  ],  
}
```


12. Tìm các nhà hàng có số lượng phần tử trong grade bằng 5

```
db.restaurants.find(  
  { grades: { $size: 5 } }  
)
```

```
> db.restaurants.find(  
  { grades: { $size: 5 } }  
)  
< {  
  _id: ObjectId('57506d62f57802807471dcd1'),  
  grades: [  
    2,  
    6,  
    10,  
    9,  
    14  
  ],  
  name: 'Morris Park Bake Shop',  
  contact: {  
    phone: '365-555-0134',  
    email: 'morrisparkbakeshop@example.org',  
    location: [  
      -73.856077,  
      40.848447  
    ]  
  }  
}
```

13. Tìm các nhà hàng có ít nhất một phần tử trong grades lớn hơn hoặc bằng 20 và nhỏ hơn 30

```
db.restaurants.find(  
  {  
    grades: { $elemMatch: { $gte: 20, $lt: 30 } }  
  }  
)
```

```
> db.restaurants.find(  
  {  
    grades: { $elemMatch: { $gte: 20, $lt: 30 } }  
  }  
)  
< {  
  _id: ObjectId('57506d62f57802807471dcd3'),  
  grades: [  
    8,  
    23,  
    12,  
    12  
  ],  
  name: "Wendy'S",  
  contact: {  
    phone: '152-555-0146',  
    email: "wendy's@example.net",  
    location: [  
      -73.961704,  
      40.662942  
    ]  
  }  
}
```

14. Tìm các nhà hàng có số điện thoại bắt đầu bằng 770 và email kết thúc bằng '.com'

```
db.restaurants.find(  
  {  
    phone: { $regex: '^770' },  
    email: { $regex: '\.com$', $options: 'i' }  
  }  
)
```