MongoDB exercise

Load db dump into MongoDB

Create useful folders

```
mkdir mongodb
cd mongodb

mkdir data
```

Create and run MongoDB on Docker

```
docker run \
        --publish=27017:27017 \
        --volume=./data:/data/db \
        --name=mongo \
        -d \
        mongo
```

Download test data from Moodle and unzip it to any location

Import the data

Download mongoDB Compass

Create new database and new collection

(call it *restaurants* or anything you like)

Go to «ADD DATA», then «Import JSON» and select the file *restaurants.json*

Querying the data

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

```
db.restaurants.find();
```

Write a MongoDB query to display the fields restaurant_id, name, borough and cuisine for all the documents in the collection restaurant

```
db.restaurants.find({},{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1});
```

Write a MongoDB query to display the fields restaurant_id, name, borough and cuisine, but exclude the field _id for all the documents in the collection restaurant.

```
db.restaurants.find({},{"restaurant_id" : 1,"name":1,"borough":1,"cuisine" :1,"_id":0});
```

Write a MongoDB query to display all the restaurant which is in the borough Bronx.

```
db.restaurants.find({"borough": "Bronx"});
```

Write a MongoDB query to display the first 5 restaurant which is in the borough Bronx.

```
db.restaurants.find({"borough": "Bronx"}).limit(5);
```

Write a MongoDB query to display the next 5 restaurants after skipping first 5 which are in the borough Bronx.

```
db.restaurants.find({"borough": "Bronx"}).skip(5).limit(5);
```

Write a MongoDB query to find the restaurants who achieved a score more than 90.

```
db.restaurants.find({grades : { $elemMatch:{"score":{$gt : 90}}}});
```

Write a MongoDB query to find the restaurants that achieved a score is more than 80 but less than 100.

```
db.restaurants.find({grades : { $elemMatch:{"score":{$gt : 80 , $lt :100}}}});
```

Write a MongoDB query to find the restaurants that do not prepare any cuisine of 'American' and their grade score more than 70

```
db.restaurants.find(
            {$and:
                [
                  {"cuisine" : {$ne :"American "}},
                  {"grades.score" : {$gt : 70}}
                 ]
             }
                );
```

Write a MongoDB query to find the restaurants which do not prepare any cuisine of 'American' and achieved a grade point 'A' not belongs to the borough Brooklyn. The document must be displayed according to the cuisine in descending order.

```
db.restaurants.find( {
                "cuisine" : {$ne : "American"},
                "grades.grade" :"A",
                "borough": {$ne : "Brooklyn"}
             }
           ).sort({"cuisine":-1});
```

Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which contain 'Wil' as first three letters for its name.

```
db.restaurants.find(
                {name: /^Wil/},
                {
                "restaurant_id" : 1,
                "name":1,"borough":1,
                "cuisine" :1
                }
);
```

Write a MongoDB query to find the restaurants which belong to the borough Bronx and prepared either American or Chinese dish.

```
db.restaurants.find(
            {
            "borough": "Bronx" ,
            $or : [
                        { "cuisine" : "American " },
                        { "cuisine" : "Chinese" }
                        ]
            }
);
```