Aufgaben:

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

```
db.restaurants.find().pretty()
```

2. 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})
```

3. 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, _id:0, borough:1, cuisine:1})
```

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

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

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

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

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

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

7. 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)
```

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

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

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

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

10. Write a MongoDB query to find the restaurants which locate in latitude value less than -95.754168.

```
db.restaurants.find({"address.coord.1":
{$lt: 95.754168}})
```

11. Write a MongoDB query to find the restaurants that do not prepare any cuisine of 'American' and
    their grade score more than 70 and
    latitude less than -65.754168.

```
db.collection.find({"cuisine": {$ne:
"American "},"grades.score": {$gt:
70},"address.coord.1": {$lt: -65.754168}})
```

12. 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.
    Note : Do this query without using $and operator.

```
db.collection.find({"cuisine": {$ne:
"American "},"grades.score": {$gt:
70},"address.coord.0": {$lt: -65.754168}})
```

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

14. 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.collection.find({"name": /^Wil/},
{"restaurant_id": 1,"name": 1,"borough":
1,"cuisine": 1})
```

15. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for

those restaurants which contain 'ces' as last three letters for its name.

```
db.collection.find({"name": /ces$/},
{"restaurant_id": 1,"name": 1,"borough":
1,"cuisine": 1})
```

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

```
db.collection.find({"name": /Reg/},
{"restaurant_id": 1,"name": 1,"borough":
1,"cuisine": 1})
```

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

```
db.collection.find({"borough":
"Bronx","cuisine": {$in: ["American ",
"Chinese"]}})
```

18. Write a MongoDB query to find the restaurant Id, name, borough and cuisine
    for those restaurants which belong to the borough Staten Island or Queens or
    Bronx or Brooklyn.

```
db.collection.find({"borough": {$in:
["Staten Island", "Queens", "Bronx",
"Brooklyn"]}}, {"restaurant_id": 1,"name":
1,"borough": 1,"cuisine": 1})
```

19. Write a MongoDB query to find the restaurant Id, name, borough and cuisine
    for those restaurants which are not belonging to the borough Staten Island or
    Queens or Bronx or Brooklyn.

```
db.collection.find({"borough": {$nin:
["Staten Island", "Queens", "Bronx",
"Brooklyn"]}}, {"restaurant_id": 1,"name":
1,"borough": 1,"cuisine": 1})
```

20. 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 10.

```
db.collection.find({"grades.score": {$lte:
10}}, {"restaurant_id": 1,"name":
1,"borough": 1,"cuisine": 1})
```