EXERCISE 18

Structure of 'restaurants' collection:

```
{
 "address": {
   "building": "1007",
   "coord": [ -73.856077, 40.848447 ],
   "street": "Morris Park Ave",
   "zipcode": "10462"
 },
 "borough": "Bronx",
 "cuisine": "Bakery",
 "grades": [
   { "date": { "Sdate": 1393804800000 }, "grade": "A", "score": 2 },
   { "date": { "Sdate": 1378857600000 }, "grade": "A", "score": 6 },
   { "date": { "Sdate": 1358985600000 }, "grade": "A", "score": 10 },
   { "date": { "Sdate": 1322006400000 }, "grade": "A", "score": 9 },
   { "date": { "Sdate": 1299715200000 }, "grade": "B", "score": 14 }
 ],
 "name": "Morris Park Bake Shop",
 "restaurant_id": "30075445"
}
```

1. Write a MongoDB query to find the restaurant Id, name, borough and cuisine for those restaurants which prepared dish except 'American' and 'Chinees' or restaurant's name begins with letter 'Wil'. db.restaurants.find({$or:[{name:{$regex:/^Wil/i}}, {cuisine:{$nin:["American","Chinese"]}}]}, {restaurant_id:1, name:1, borough:1, cuisine:1});

2. Write a MongoDB query to find the restaurant Id, name, and grades for those restaurants which achieved a grade of "A" and scored 11 on an ISODate "2014-08-11T00:00:00Z" among many of survey dates.. db.restaurants.find({"grades":{$elemMatch:{grade:"A", score:11, date:ISODate("2014-08-11T00:00:00Z")}}}, {restaurant_id:1, name:1, grades:1});

3. Write a MongoDB query to find the restaurant Id, name and grades for those restaurants where the 2nd element of grades array contains a grade of "A" and score 9 on an ISODate "2014-08-11T00:00:00Z". db.restaurants.find({"grades.1.grade":"A", "grades.1.score":9, "grades.1.date":ISODate("2014-08-11T00:00:00Z")}, {restaurant_id:1, name:1, grades:1});

4. Write a MongoDB query to find the restaurant Id, name, address and geographical location for those restaurants where 2nd element of coord array contains a value

d

which is more than 42 and upto 52..

```
db.restaurants. find ({"address.coord.1": {$gt:42, $lte:52}},
{restaurant_id:1, name:1, address:1})
```

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

```
db.restaurants.find().sort ({name:1});
```

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

```
db.restaurants.find().sort ({name:-1}));
```

7. Write a MongoDB query to arranged the name of the cuisine in ascending order and for that same cuisine borough should be in descending order.

```
db.restaurants.find().sort ( {cuisine:1, borough : -1});
```

8. Write a MongoDB query to know whether all the addresses contains the street or not.

```
db.restaurants.find ({"address.street": {$exists: true}});
```

9. Write a MongoDB query which will select all documents in the restaurants collection where the coord field value is Double.

```
db.restaurants.find({"address.coord": {$type:"double"}});
```

10. Write a MongoDB query which will select the restaurant Id, name and grades for those restaurants which returns 0 as a remainder after dividing the score by 7.

```
db.restaurants.find ({"grades.score": {$mod: [7, 0]}}, {restaurant_id:
  1, name:1, grades:1)});
```

11. Write a MongoDB query to find the restaurant name, borough, longitude and attitude and cuisine for those restaurants which contains 'mon' as three letters somewhere in its name.

```
db.restaurant.find ({name: {$ regex:/mon/i}}, {name:1, borogh:1, "
 address.coord": 1, cuisine:1});
```

12. Write a MongoDB query to find the restaurant name, borough, longitude and latitude and cuisine for those restaurants which contain 'Mad' as first three letters of its name.

```
db.restaurants.find ({name: {$regex:/^Mad/i}},
{name:1, borough:1, "address, coord": 1, cuisine:1});
```

13. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5.

```
db. restaurants.find({"grades.score":{$ lt:5}});
```

14. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan.

```
db.restaurants.find({borough:"Manhattan", "grades.score"
 :{$lt:5}});
```

15. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn.

```
db. restaurants.find({borough:{$in:["Manhattan", "Brooklyn"]},
" grades. score":{$lt:5}});
```

16. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn, and their cuisine is not American.

```
db. restaurants.find({borough:{$in:[Manhattan", "Brooklyn"}},
cuisine:{$ne:"American"}, "grades.score":{$ lt:5}});
```

17. Write a MongoDB query to find the restaurants that have at least one grade with a score of less than 5 and that are located in the borough of Manhattan or Brooklyn, and their cuisine is not American or Chinese.

```
db.restaurants.find({borough:{$in:["Manhattan","Brooklyn"]},
cuisine:{$nin:["American", "Chinese"]}, "grades.score":{$lt:5}});
```

18. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6.

```
db. restaurants.find({"grades. scoee":{$ all:[2,6]}});
```

19. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan.

```
db. restaurants.find({borough:"Manhattan", "grades.score":
{$ all:[2,6]}});
```

20. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn.

```
db. restaurants.find({borough:{$in:["Manhattan", "Brooklyn"]},
" grades.score":{$all:[2,6]}});
```

21. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn, and their cuisine is not American.

```
db.restaurants.find ({borough : ($in : [" Manhattan ", " Brooklyn "]},
cuisine : { $ne : " American " }, "grades.score": { $all : [2,6]}});
```

22. Write a MongoDB query to find the restaurants that have a grade with a score of 2 and a grade with a score of 6 and are located in the borough of Manhattan or Brooklyn, and their cuisine is not American or Chinese.

```
db.restaurants.find ({borough : {$in : [" Manhattan". " Brooklyn "]},
cuisine : {$nin : [" American", " Chinese"]}, "grades.score": {$al: [2,6]}});
```

23. Write a MongoDB query to find the restaurants that have a grade with a score of 2 or a grade with a score of 6.

```
db.restaurants.find ({"grades.score": {$in: [2,6]}});
```

Sample document of 'movies' collection

```
{
  _id: ObjectId("573a1390f29313caabcd42e8"),
plot: 'A group of bandits stage a brazen train hold-up, only to find a determined posse hot on their heels.',
genres: [ 'Short', 'Western' ],
runtime: 11,
cast: [
    'A.C. Abadie',
    "Gilbert M. 'Broncho Billy' Anderson",
    'George Barnes',
    'Justus D. Barnes'
  ],
poster: 'https://m.media-amazon.com/images/M/MV5BMTU3NjE5NzYtYTYyNS00MDVmLWIwYjgtMmYwYWIxZDYyNzU2XkEyXkFqcGdeQXVyNzQzNzQxNzI@._V1_SY1000_SX677_AL_.jpg',
title: 'The Great Train Robbery',
fullplot: "Among the earliest existing films in American cinema - notable as the first film that presented a narrative story to tell - it depicts a group of cowboy outlaws who hold up a train and rob the passengers. They are then pursued by a Sheriff's posse. Several scenes have color included - all hand tinted.",
```

Evaluation Procedure	Marks awarded
PL/SQL Procedure(5)	5
Program/Execution (5)	5
Viva(5)	5
Total (15)	15
Faculty Signature	7/11/25

204