```
import pymongo
client=pymongo.MongoClient("mongodb://localhost:27017")
db=client.test

db=client['Bakadb']
```

Q1. Design a Mongodb Schema for a "Student" collection with the following fields:

```
collection = db["Student"]
```

Q2. Insert the following student data in the collection

```
student_data=[
{"RollNum": 43, "FirstName": "John","LastName":"Doe","Age":
20,"Department":"Computer Science","Mark": 78},
{"RollNum": 67, "FirstName": "Alice","LastName":"Smith","Age":
22,"Department":"Physics","Mark": 59},
{"RollNum": 23, "FirstName": "Bob","LastName":"Johnson","Age":
21,"Department":"Computer Science","Mark": 81},
{"RollNum": 18, "FirstName": "Eve","LastName":"Adams","Age":
19,"Department":"Mathematics","Mark": 56},
{"RollNum": 84, "FirstName": "Mike","LastName":"Brown","Age":
23,"Department":"Physics","Mark": 92}
]

result = collection.insert_many(student_data)
```

Q3. Write a Mongodb query to find all students

```
for i in db.Student.find({}):
    print(i)
```

```
{'_id': ObjectId('6548eeaa087e94747106aafa'), 'RollNum': 43,
'FirstName': 'John', 'LastName': 'Doe', 'Age': 20, 'Department':
'Computer Science', 'Mark': 78}
{'_id': ObjectId('6548eeaa087e94747106aafb'), 'RollNum': 67,
'FirstName': 'Alice', 'LastName': 'Smith', 'Age': 22, 'Department':
'Physics', 'Mark': 59}
{'_id': ObjectId('6548eeaa087e94747106aafc'), 'RollNum': 23,
'FirstName': 'Bob', 'LastName': 'Johnson', 'Age': 21, 'Department':
'Computer Science', 'Mark': 81}
{'_id': ObjectId('6548eeaa087e94747106aafd'), 'RollNum': 18,
'FirstName': 'Eve', 'LastName': 'Adams', 'Age': 19, 'Department':
'Mathematics', 'Mark': 56}
{'_id': ObjectId('6548eeaa087e94747106aafe'), 'RollNum': 84,
'FirstName': 'Mike', 'LastName': 'Brown', 'Age': 23, 'Department':
'Physics', 'Mark': 92}
```

Q4. Write a Mongodb query to find all students in the "Computer Science" Department

```
for i in db.Student.find({"Department": "Computer Science"}):
    print(i)
```

```
{'_id': ObjectId('6548eeaa087e94747106aafa'), 'RollNum': 43,
'FirstName': 'John', 'LastName': 'Doe', 'Age': 20, 'Department':
'Computer Science', 'Mark': 78}
{'_id': ObjectId('6548eeaa087e94747106aafc'), 'RollNum': 23,
'FirstName': 'Bob', 'LastName': 'Johnson', 'Age': 21, 'Department':
'Computer Science', 'Mark': 81}
```

Q5. Write a Mongodb query to find all students whose age is greater than or equal to 20

```
for i in db.Student.find({"Age": {"$gte": 20}}):
    print(i)
```

```
{'_id': ObjectId('6548eeaa087e94747106aafa'), 'RollNum': 43,
'FirstName': 'John', 'LastName': 'Doe', 'Age': 20, 'Department':
'Computer Science', 'Mark': 78}
{'_id': ObjectId('6548eeaa087e94747106aafb'), 'RollNum': 67,
'FirstName': 'Alice', 'LastName': 'Smith', 'Age': 22, 'Department':
'Physics', 'Mark': 59}
{'_id': ObjectId('6548eeaa087e94747106aafc'), 'RollNum': 23,
'FirstName': 'Bob', 'LastName': 'Johnson', 'Age': 21, 'Department':
'Computer Science', 'Mark': 81}
{'_id': ObjectId('6548eeaa087e94747106aafe'), 'RollNum': 84,
'FirstName': 'Mike', 'LastName': 'Brown', 'Age': 23, 'Department':
'Physics', 'Mark': 92}
```

Q6. Write a Mongodb query to find all students whose mark is less than 60

```
for i in db.Student.find({"Mark": {"$lt": 60}}):
    print(i)
```

```
{'_id': ObjectId('6548eeaa087e94747106aafb'), 'RollNum': 67,
'FirstName': 'Alice', 'LastName': 'Smith', 'Age': 22, 'Department':
'Physics', 'Mark': 59}
{'_id': ObjectId('6548eeaa087e94747106aafd'), 'RollNum': 18,
'FirstName': 'Eve', 'LastName': 'Adams', 'Age': 19, 'Department':
'Mathematics', 'Mark': 56}
```

Q7. Write a Mongodb query to show the first name and mark of all students in "Physics" department

```
for i in db.Student.find({"Department": "Physics"}, {"FirstName": 1,
"Mark": 1}):
    print(i)
```

```
{'_id': ObjectId('6548eeaa087e94747106aafb'), 'FirstName': 'Alice',
'Mark': 59}
```

```
{'_id': ObjectId('6548eeaa087e94747106aafe'), 'FirstName': 'Mike',
'Mark': 92}
```

Q8. Write a Mongodb query to find all the students in the descending order of mark

```
for i in db.Student.find().sort({"Mark": -1}):
    print(i)
```

```
{'_id': ObjectId('6548eeaa087e94747106aafe'), 'RollNum': 84,
'FirstName': 'Mike', 'LastName': 'Brown', 'Age': 23, 'Department':
'Physics', 'Mark': 92}
{'_id': ObjectId('6548eeaa087e94747106aafc'), 'RollNum': 23,
'FirstName': 'Bob', 'LastName': 'Johnson', 'Age': 21, 'Department':
'Computer Science', 'Mark': 81}
{'_id': ObjectId('6548eeaa087e94747106aafa'), 'RollNum': 43,
'FirstName': 'John', 'LastName': 'Doe', 'Age': 20, 'Department':
'Computer Science', 'Mark': 78}
{'_id': ObjectId('6548eeaa087e94747106aafb'), 'RollNum': 67,
'FirstName': 'Alice', 'LastName': 'Smith', 'Age': 22, 'Department':
'Physics', 'Mark': 59}
{'_id': ObjectId('6548eeaa087e94747106aafd'), 'RollNum': 18,
'FirstName': 'Eve', 'LastName': 'Adams', 'Age': 19, 'Department':
'Mathematics', 'Mark': 56}
```

Q9. Write a Mongodb query to find the youngest student

```
for i in db.Student.find().sort({"Age": 1}).limit(1):
    print(i)
```

```
{'_id': ObjectId('6548eeaa087e94747106aafd'), 'RollNum': 18,
'FirstName': 'Eve', 'LastName': 'Adams', 'Age': 19, 'Department':
'Mathematics', 'Mark': 56}
```

Q10. Write a Mongodb query to find all students in the "Physics" department whose Roll no is greater than or equal to 70

```
for i in db.Student.find({"Department": "Physics", "RollNum": {"$gte":
70}}):
    print(i)
```

```
{'_id': ObjectId('6548eeaa087e94747106aafe'), 'RollNum': 84,
'FirstName': 'Mike', 'LastName': 'Brown', 'Age': 23, 'Department':
'Physics', 'Mark': 92}
```