assignment-8

November 26, 2023

[6]:
```
import pymongo
client=pymongo.MongoClient("mongodb://localhost:27017")
db=client.test
```

[7]:
```
db=client['mardb']
```

[8]:
```
collection = db["Student"]
```

Q-1 Design a MongoDB schema for an "Employee" collection with the following fields: a. EmployeeID b. FirstName c. LastName d. Age e. Department f. Salary

[9]:
```
employee_data = {
    "EmployeeID": 1,
    "FirstName": "John",
    "LastName": "Doe",
    "Age": 30,
    "Department": "IT",
    "Salary": 60000
}

result = collection.insert_one(employee_data)
print(f"Inserted document with ID: {result.inserted_id}")
```

```
Inserted document with ID: 655c4d3ba29ec3322c791dfd
```

Q-2 Insert the following employee data into the collection.

[10]:
```
employee_data = [
    {
        "EmployeeID": 121,
        "FirstName": "Emma",
        "LastName": "Johnson",
        "Age": 30,
        "Department": "Human Resources",
        "Salary": 50000
    },
    {
        "EmployeeID": 134,
        "FirstName": "David",
```

```
        "LastName": "Smith",
        "Age": 34,
        "Department": "Marketing",
        "Salary": 55000
    },
    {
        "EmployeeID": 145,
        "FirstName": "Mia",
        "LastName": "Davis",
        "Age": 28,
        "Department": "Information Tech",
        "Salary": 62000
    },
    {
        "EmployeeID": 167,
        "FirstName": "Lucas",
        "LastName": "Brown",
        "Age": 40,
        "Department": "Sales",
        "Salary": 48000
    },
    {
        "EmployeeID": 153,
        "FirstName": "Sophia",
        "LastName": "Wilson",
        "Age": 33,
        "Department": "Research",
        "Salary": 53000
    }
]

# Inserting multiple documents at once
result = collection.insert_many(employee_data)

# Print the inserted document IDs
print(f"Inserted document IDs: {result.inserted_ids}")
```

```
Inserted document IDs: [ObjectId('655c4d77a29ec3322c791dfe'),
ObjectId('655c4d77a29ec3322c791dff'), ObjectId('655c4d77a29ec3322c791e00'),
ObjectId('655c4d77a29ec3322c791e01'), ObjectId('655c4d77a29ec3322c791e02')]
```

Q-3 Write a MongoDB query to find all employees

[11]:
```
all_employees = collection.find()
for employee in all_employees:
    print(employee)
```

```
{'_id': ObjectId('6554879cb9b9bd1b857c5fce'), 'RollNum': 43, 'FirstName':
```

```
'John', 'LastName': 'Doe', 'Age': 22, 'Department': 'Computer Science', 'Mark':
88}
{'_id': ObjectId('6554879cb9b9bd1b857c5fd0'), 'RollNum': 23, 'FirstName': 'Bob',
'LastName': 'Johnson', 'Age': 23, 'Department': 'Computer Science', 'Mark': 91}
{'_id': ObjectId('6554879cb9b9bd1b857c5fd2'), 'RollNum': 84, 'FirstName':
'Mike', 'LastName': 'Brown', 'Age': 25, 'Department': 'Physical Science',
'Mark': 92}
{'_id': ObjectId('6559b026d4126227e5bf6200'), 'RollNum': 43, 'FirstName':
'John', 'LastName': 'Doe', 'Age': 22, 'Department': 'Computer Science', 'Mark':
88}
{'_id': ObjectId('6559b026d4126227e5bf6202'), 'RollNum': 23, 'FirstName': 'Bob',
'LastName': 'Johnson', 'Age': 23, 'Department': 'Computer Science', 'Mark': 91}
{'_id': ObjectId('6559b026d4126227e5bf6204'), 'RollNum': 84, 'FirstName':
'Mike', 'LastName': 'Brown', 'Age': 25, 'Department': 'Physical Science',
'Mark': 92}
{'_id': ObjectId('655c4d3ba29ec3322c791dfd'), 'EmployeeID': 1, 'FirstName':
'John', 'LastName': 'Doe', 'Age': 30, 'Department': 'IT', 'Salary': 60000}
{'_id': ObjectId('655c4d77a29ec3322c791dfe'), 'EmployeeID': 121, 'FirstName':
'Emma', 'LastName': 'Johnson', 'Age': 30, 'Department': 'Human Resources',
'Salary': 50000}
{'_id': ObjectId('655c4d77a29ec3322c791dff'), 'EmployeeID': 134, 'FirstName':
'David', 'LastName': 'Smith', 'Age': 34, 'Department': 'Marketing', 'Salary':
55000}
{'_id': ObjectId('655c4d77a29ec3322c791e00'), 'EmployeeID': 145, 'FirstName':
'Mia', 'LastName': 'Davis', 'Age': 28, 'Department': 'Information Tech',
'Salary': 62000}
{'_id': ObjectId('655c4d77a29ec3322c791e01'), 'EmployeeID': 167, 'FirstName':
'Lucas', 'LastName': 'Brown', 'Age': 40, 'Department': 'Sales', 'Salary': 48000}
{'_id': ObjectId('655c4d77a29ec3322c791e02'), 'EmployeeID': 153, 'FirstName':
'Sophia', 'LastName': 'Wilson', 'Age': 33, 'Department': 'Research', 'Salary':
53000}
```

Q-4 Write a MongoDB query to find all employees in the “Marketing” department

[12]:
```
marketing_employees = collection.find({"Department": "Marketing"})
for employee in marketing_employees:
    print(employee)
```

```
{'_id': ObjectId('655c4d77a29ec3322c791dff'), 'EmployeeID': 134, 'FirstName':
'David', 'LastName': 'Smith', 'Age': 34, 'Department': 'Marketing', 'Salary':
55000}
```

Q-5 Write a MongoDB query to find all employees whose age is greater than or equal to 30.

[13]:
```
employees_above_30 = collection.find({"Age": {"$gte": 30}})
for employee in employees_above_30:
    print(employee)
```

```
{'_id': ObjectId('655c4d3ba29ec3322c791dfd'), 'EmployeeID': 1, 'FirstName':
```

```
'John', 'LastName': 'Doe', 'Age': 30, 'Department': 'IT', 'Salary': 60000}
{'_id': ObjectId('655c4d77a29ec3322c791dfe'), 'EmployeeID': 121, 'FirstName':
'Emma', 'LastName': 'Johnson', 'Age': 30, 'Department': 'Human Resources',
'Salary': 50000}
{'_id': ObjectId('655c4d77a29ec3322c791dff'), 'EmployeeID': 134, 'FirstName':
'David', 'LastName': 'Smith', 'Age': 34, 'Department': 'Marketing', 'Salary':
55000}
{'_id': ObjectId('655c4d77a29ec3322c791e01'), 'EmployeeID': 167, 'FirstName':
'Lucas', 'LastName': 'Brown', 'Age': 40, 'Department': 'Sales', 'Salary': 48000}
{'_id': ObjectId('655c4d77a29ec3322c791e02'), 'EmployeeID': 153, 'FirstName':
'Sophia', 'LastName': 'Wilson', 'Age': 33, 'Department': 'Research', 'Salary':
53000}
```

Q-6 Write a MongoDB query to find all employees whose salary is less than 50000.

[14]:
```
employees_below_50000 = collection.find({"Salary": {"$lt": 50000}})
for employee in employees_below_50000:
    print(employee)
```

```
{'_id': ObjectId('655c4d77a29ec3322c791e01'), 'EmployeeID': 167, 'FirstName':
'Lucas', 'LastName': 'Brown', 'Age': 40, 'Department': 'Sales', 'Salary': 48000}
```

Q-7 Write a MongoDB query to show the first name and salary of all employees in the "Information Tech" department.

[15]:
```
it_department_employees = collection.find(
    {"Department": "Information Tech"},
    {"FirstName": 1, "Salary": 1, "_id": 0}
)
for employee in it_department_employees:
    print(employee)
```

```
{'FirstName': 'Mia', 'Salary': 62000}
```

Q-8 Write a MongoDB query to find all employees in descending order of salary.

[16]:
```
employees_sorted_by_salary = collection.find().sort("Salary", pymongo.
↪DESCENDING)
for employee in employees_sorted_by_salary:
    print(employee)
```

```
{'_id': ObjectId('655c4d77a29ec3322c791e00'), 'EmployeeID': 145, 'FirstName':
'Mia', 'LastName': 'Davis', 'Age': 28, 'Department': 'Information Tech',
'Salary': 62000}
{'_id': ObjectId('655c4d3ba29ec3322c791dfd'), 'EmployeeID': 1, 'FirstName':
'John', 'LastName': 'Doe', 'Age': 30, 'Department': 'IT', 'Salary': 60000}
{'_id': ObjectId('655c4d77a29ec3322c791dff'), 'EmployeeID': 134, 'FirstName':
'David', 'LastName': 'Smith', 'Age': 34, 'Department': 'Marketing', 'Salary':
55000}
{'_id': ObjectId('655c4d77a29ec3322c791e02'), 'EmployeeID': 153, 'FirstName':
```

```
'Sophia', 'LastName': 'Wilson', 'Age': 33, 'Department': 'Research', 'Salary':
53000}
{'_id': ObjectId('655c4d77a29ec3322c791dfe'), 'EmployeeID': 121, 'FirstName':
'Emma', 'LastName': 'Johnson', 'Age': 30, 'Department': 'Human Resources',
'Salary': 50000}
{'_id': ObjectId('655c4d77a29ec3322c791e01'), 'EmployeeID': 167, 'FirstName':
'Lucas', 'LastName': 'Brown', 'Age': 40, 'Department': 'Sales', 'Salary': 48000}
{'_id': ObjectId('6554879cb9b9bd1b857c5fce'), 'RollNum': 43, 'FirstName':
'John', 'LastName': 'Doe', 'Age': 22, 'Department': 'Computer Science', 'Mark':
88}
{'_id': ObjectId('6554879cb9b9bd1b857c5fd0'), 'RollNum': 23, 'FirstName': 'Bob',
'LastName': 'Johnson', 'Age': 23, 'Department': 'Computer Science', 'Mark': 91}
{'_id': ObjectId('6554879cb9b9bd1b857c5fd2'), 'RollNum': 84, 'FirstName':
'Mike', 'LastName': 'Brown', 'Age': 25, 'Department': 'Physical Science',
'Mark': 92}
{'_id': ObjectId('6559b026d4126227e5bf6200'), 'RollNum': 43, 'FirstName':
'John', 'LastName': 'Doe', 'Age': 22, 'Department': 'Computer Science', 'Mark':
88}
{'_id': ObjectId('6559b026d4126227e5bf6202'), 'RollNum': 23, 'FirstName': 'Bob',
'LastName': 'Johnson', 'Age': 23, 'Department': 'Computer Science', 'Mark': 91}
{'_id': ObjectId('6559b026d4126227e5bf6204'), 'RollNum': 84, 'FirstName':
'Mike', 'LastName': 'Brown', 'Age': 25, 'Department': 'Physical Science',
'Mark': 92}
```

Q-9 Write a MongoDB query to find the oldest employee.

[17]:
```
oldest_employee = collection.find().sort("Age", pymongo.DESCENDING).limit(1)
for employee in oldest_employee:
    print(employee)
```

```
{'_id': ObjectId('655c4d77a29ec3322c791e01'), 'EmployeeID': 167, 'FirstName':
'Lucas', 'LastName': 'Brown', 'Age': 40, 'Department': 'Sales', 'Salary': 48000}
```

Q-10 Write a MongoDB query to find all employees in the “Sales” department whose EmployeeID is greater than or equal to 150.

[18]:
```
sales_employees = collection.find(
    {"Department": "Sales", "EmployeeID": {"$gte": 150}}
)
for employee in sales_employees:
    print(employee)
```

```
{'_id': ObjectId('655c4d77a29ec3322c791e01'), 'EmployeeID': 167, 'FirstName':
'Lucas', 'LastName': 'Brown', 'Age': 40, 'Department': 'Sales', 'Salary': 48000}
```