

Name: Sakshi Jaiswal

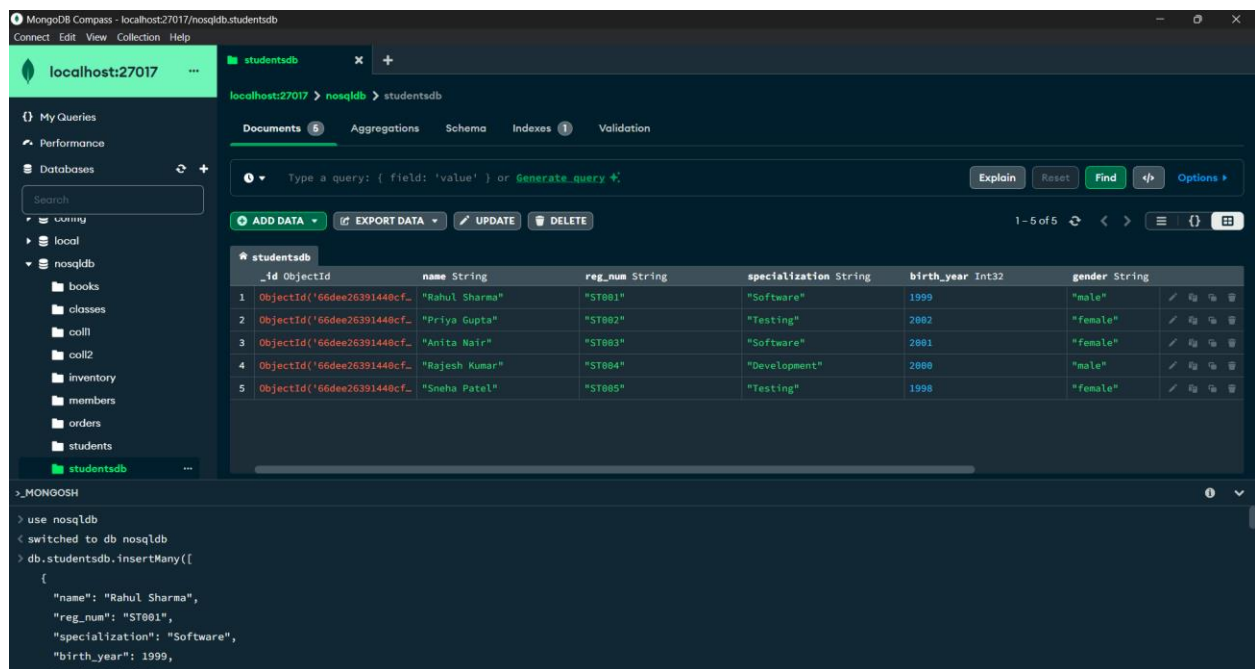
PRN: 1132230315

Class: SYMCA - C NoSQL Batch 1

Assignment 3

create a NoSQL database to store student collection data that should contain name, registration number, specialization, birth year, gender and status in the form of pass and fail.

1. write a query to insert values into the table at least 5



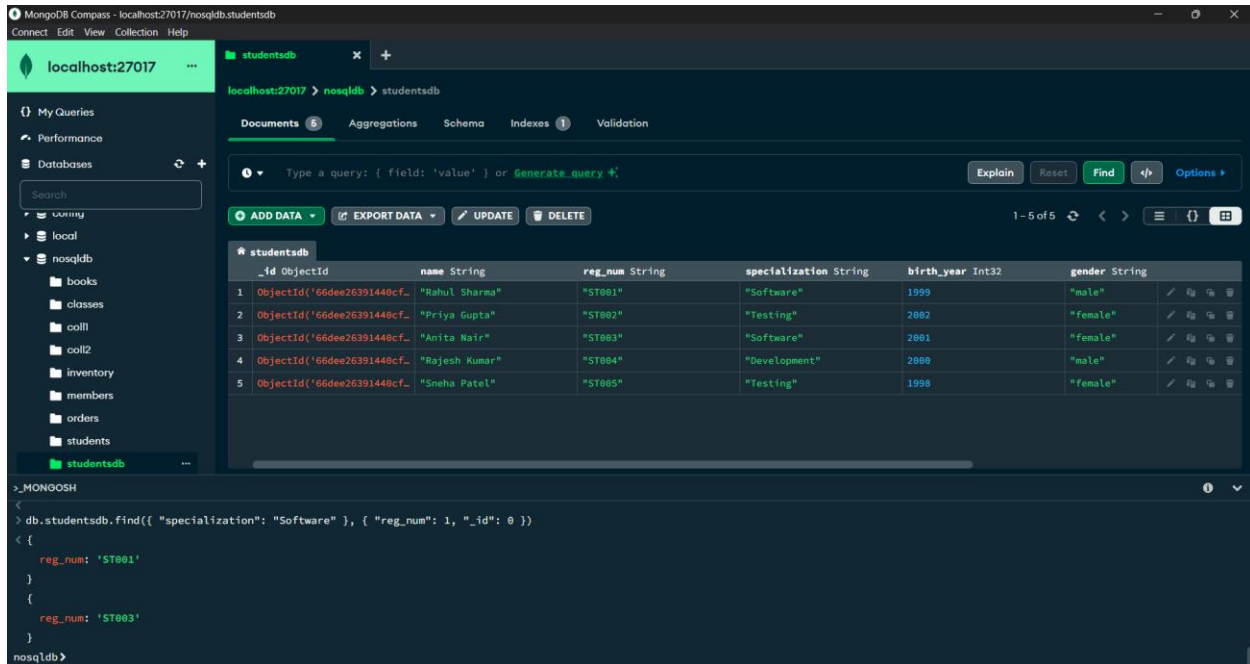
The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure, including a collection named 'students' under the 'nosql' database. The main panel shows the 'studentsdb' database with 5 documents. The documents are listed in a table with columns: _id, name, reg_num, specialization, birth_year, and gender. The documents are as follows:

_id	name	reg_num	specialization	birth_year	gender
ObjectId('66dee26391448cf...')	"Rahul Sharma"	"ST001"	"Software"	1999	"male"
ObjectId('66dee26391448cf...')	"Priya Gupta"	"ST002"	"Testing"	2002	"female"
ObjectId('66dee26391448cf...')	"Anita Nair"	"ST003"	"Software"	2001	"female"
ObjectId('66dee26391448cf...')	"Rajesh Kumar"	"ST004"	"Development"	2000	"male"
ObjectId('66dee26391448cf...')	"Sneha Patel"	"ST005"	"Testing"	1998	"female"

The bottom panel shows the MongoDB shell with the following commands:

```
> use nosql
switched to db nosql
> db.students.insertMany([
  {
    "name": "Rahul Sharma",
    "reg_num": "ST001",
    "specialization": "Software",
    "birth_year": 1999,
    "gender": "male"
  },
  {
    "name": "Priya Gupta",
    "reg_num": "ST002",
    "specialization": "Testing",
    "birth_year": 2002,
    "gender": "female"
  },
  {
    "name": "Anita Nair",
    "reg_num": "ST003",
    "specialization": "Software",
    "birth_year": 2001,
    "gender": "female"
  },
  {
    "name": "Rajesh Kumar",
    "reg_num": "ST004",
    "specialization": "Development",
    "birth_year": 2000,
    "gender": "male"
  },
  {
    "name": "Sneha Patel",
    "reg_num": "ST005",
    "specialization": "Testing",
    "birth_year": 1998,
    "gender": "female"
  }
])
```

2. retrieve student reg num where the specialisation of the student is software



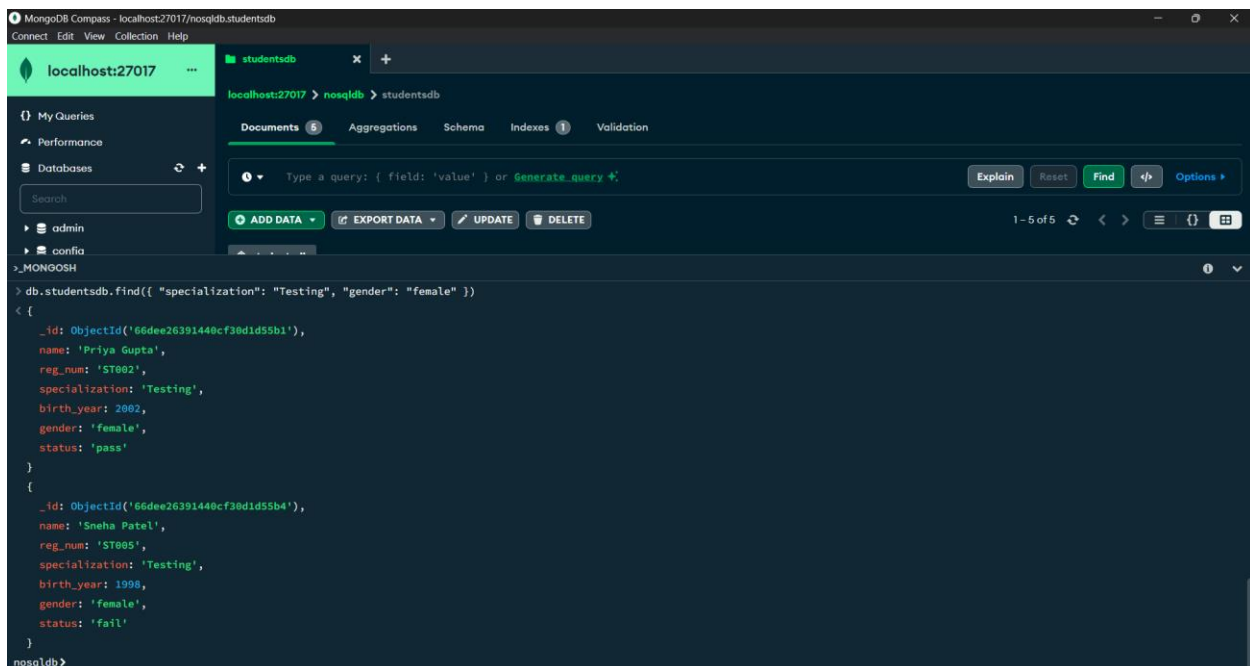
The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure with 'studentsdb' selected. The main panel shows the 'studentsdb' database with a table view of the 'students' collection. The table has columns: _id, name, reg_num, specialization, birth_year, and gender. The data is as follows:

_id	name	reg_num	specialization	birth_year	gender
1	Rahul Sharma	ST001	Software	1999	male
2	Priya Gupta	ST002	Testing	2002	female
3	Anita Nair	ST003	Software	2001	female
4	Rajesh Kumar	ST004	Development	2000	male
5	Sneha Patel	ST005	Testing	1998	female

The MongoDB Shell at the bottom shows the following query and result:

```
> db.studentsdb.find({ "specialization": "Software" }, { "reg_num": 1, "_id": 0 })
<
{
  reg_num: 'ST001'
}
{
  reg_num: 'ST003'
}
nosql>
```

3. retrieve students who have specialization in testing and whose gender is female



The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure with 'studentsdb' selected. The main panel shows the 'studentsdb' database with a table view of the 'students' collection. The data is as follows:

_id	name	reg_num	specialization	birth_year	gender
1	Rahul Sharma	ST001	Software	1999	male
2	Priya Gupta	ST002	Testing	2002	female
3	Anita Nair	ST003	Software	2001	female
4	Rajesh Kumar	ST004	Development	2000	male
5	Sneha Patel	ST005	Testing	1998	female

The MongoDB Shell at the bottom shows the following query and result:

```
> db.studentsdb.find({ "specialization": "Testing", "gender": "female" })
<
{
  _id: ObjectId('66dee26391448cf30d1d55b1'),
  name: 'Priya Gupta',
  reg_num: 'ST002',
  specialization: 'Testing',
  birth_year: 2002,
  gender: 'female',
  status: 'pass'
}
{
  _id: ObjectId('66dee26391448cf30d1d55b4'),
  name: 'Sneha Patel',
  reg_num: 'ST005',
  specialization: 'Testing',
  birth_year: 1998,
  gender: 'female',
  status: 'fail'
}
nosql>
```

4. retrieve info of student name who was born after 2000

The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure with 'nosqlldb' selected. The main panel shows the 'studentsdb' collection with a table view of 5 documents. The bottom terminal shows a MongoDB query and its results.

Query:

```
> db.studentsdb.find({ "birth_year": { "$gt": 2000 } }, { "name": 1, "_id": 0 })
```

Results:

```
< {
  name: 'Priya Gupta'
}
{
  name: 'Anita Nair'
}
```

5. retrieve student details who were born on and before 2000 those who are male and status fail

The screenshot shows the MongoDB Compass interface. The left sidebar displays the database structure with 'nosqlldb' selected. The main panel shows the 'studentsdb' collection with a table view of 2 documents. The bottom terminal shows a MongoDB query and its results.

Query:

```
> db.studentsdb.find({
  "birth_year": { "$lte": 2000 },
  "gender": "male",
  "status": "fail"
})
```

Results:

```
< {
  _id: ObjectId('66dee26391448cf30d1d55b3'),
  name: 'Rajesh Kumar',
  reg_num: 'ST004',
  specialization: 'Development',
  birth_year: 2000,
  gender: 'male',
  status: 'fail'
}
```