Assignment – 4

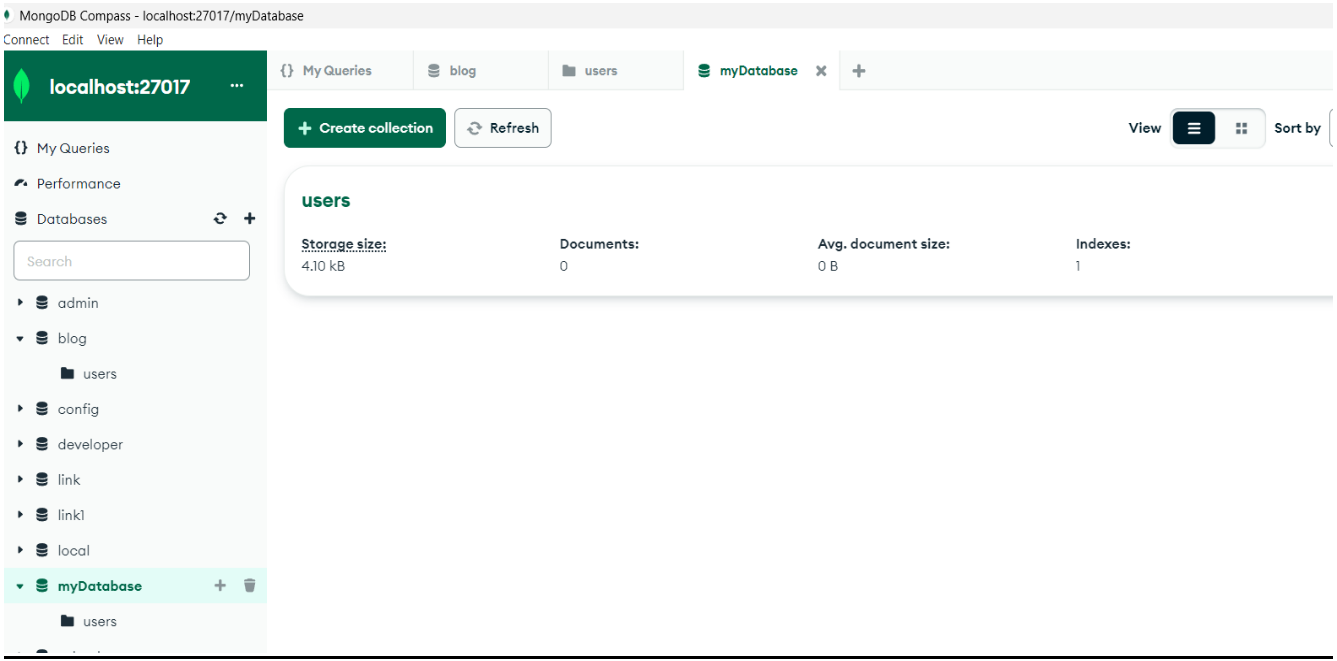
1. **Database Setup: Create a new MongoDB database called myDatabase**
2. **Collection Creation: Create a collection named users within the myDatabase database.**

   Using mongosh:

```
use myDatabase
switched to db myDatabase
db.createCollection('users')
```

{ ok: 1 }

Output:



users

Storage size: 4.10 kB

Documents: 0

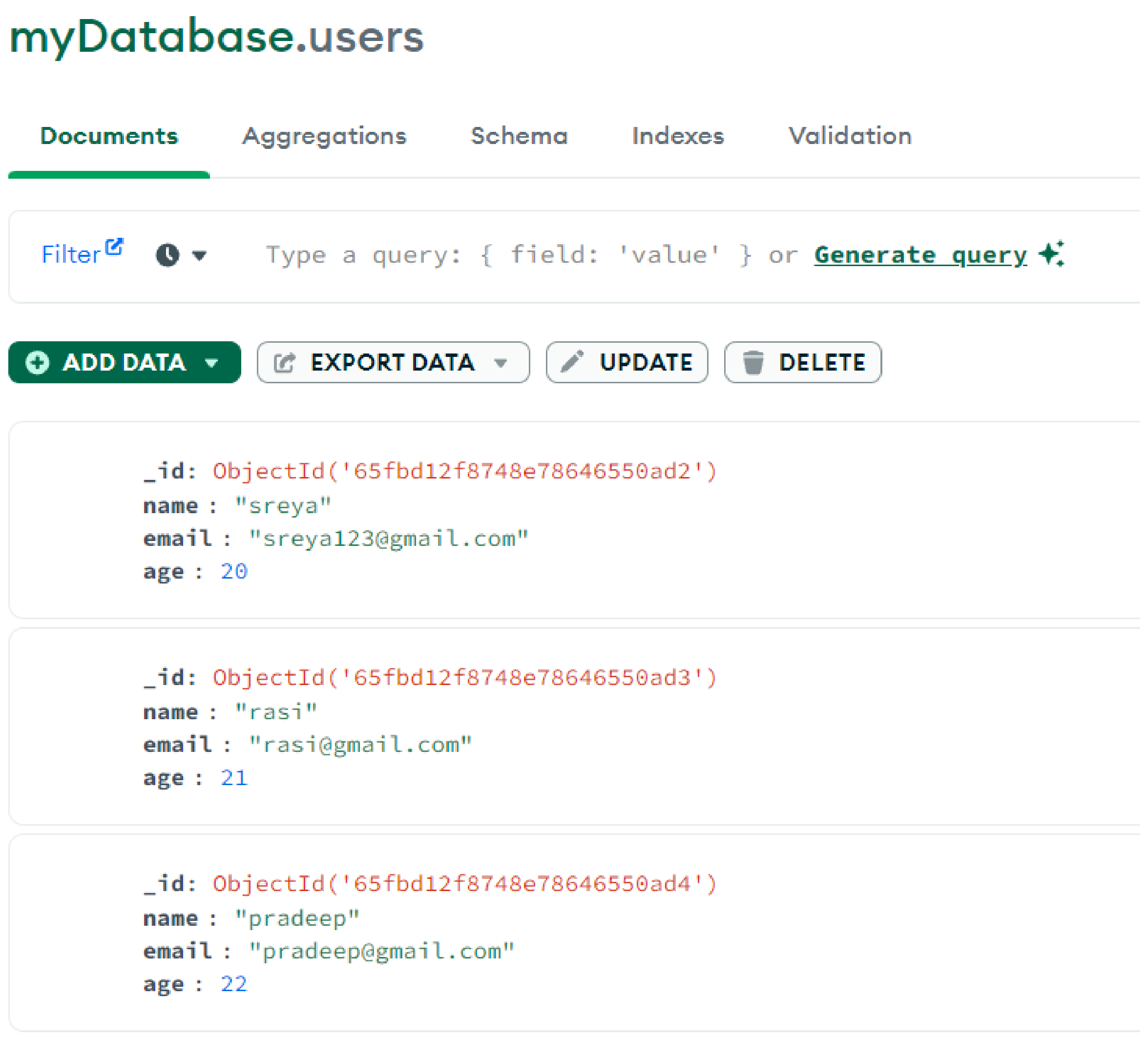
Avg. document size: 0 B

Indexes: 1

3.Document Insertion: Insert at least three documents into the users collection, each representing a user with fields such as name, email, and age.

```
Db.users.insertMany([{'name':'sreya','email':'sreya123@gmail.com','age':20},{'name':'rasi','email':'rasi@g
mail.com','age':21},{'name':'pradeep','email':'pradeep@gmail.com','age':22}])
{
 acknowledged: true,
 insertedIds: {
  '0': ObjectId('65fbcd518748e78646550acf'),
  '1': ObjectId('65fbcd518748e78646550ad0'),
  '2': ObjectId('65fbcd518748e78646550ad1')
 }
}
```

Output:



```
_id: ObjectId('65fbd12f8748e78646550ad2')
name : "sreya"
email : "sreya123@gmail.com"
age : 20
```

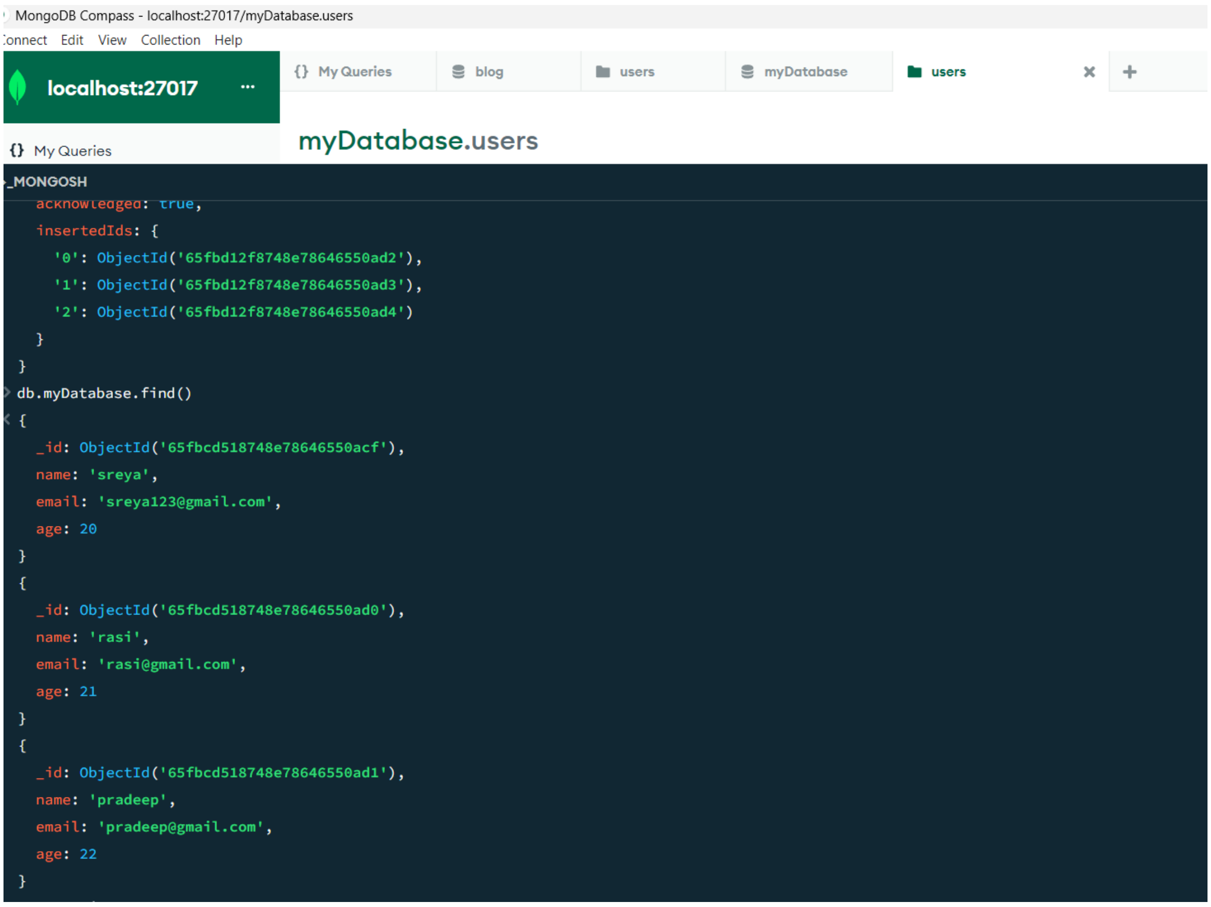
```
_id: ObjectId('65fbd12f8748e78646550ad3')
name : "rasi"
email : "rasi@gmail.com"
age : 21
```

```
_id: ObjectId('65fbd12f8748e78646550ad4')
name : "pradeep"
email : "pradeep@gmail.com"
age : 22
```

4.Querying: Write queries to retrieve:

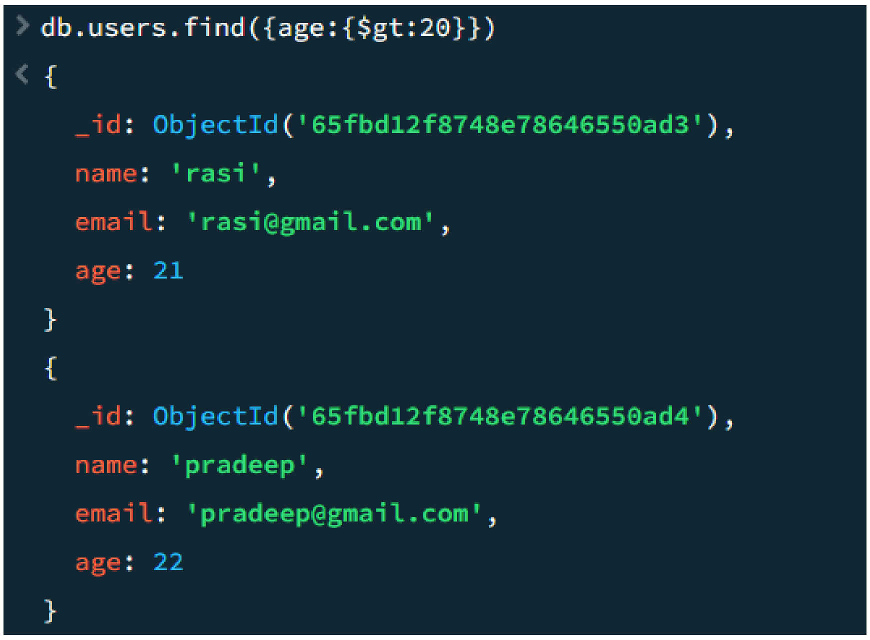
1.All users from the users collection

```
db.myDatabase.find()
```


MongoDB Compass - localhost:27017/myDatabase.users
Connect Edit View Collection Help
localhost:27017
My Queries
blog
users
myDatabase
users
My Queries
myDatabase.users
_MONGOSH

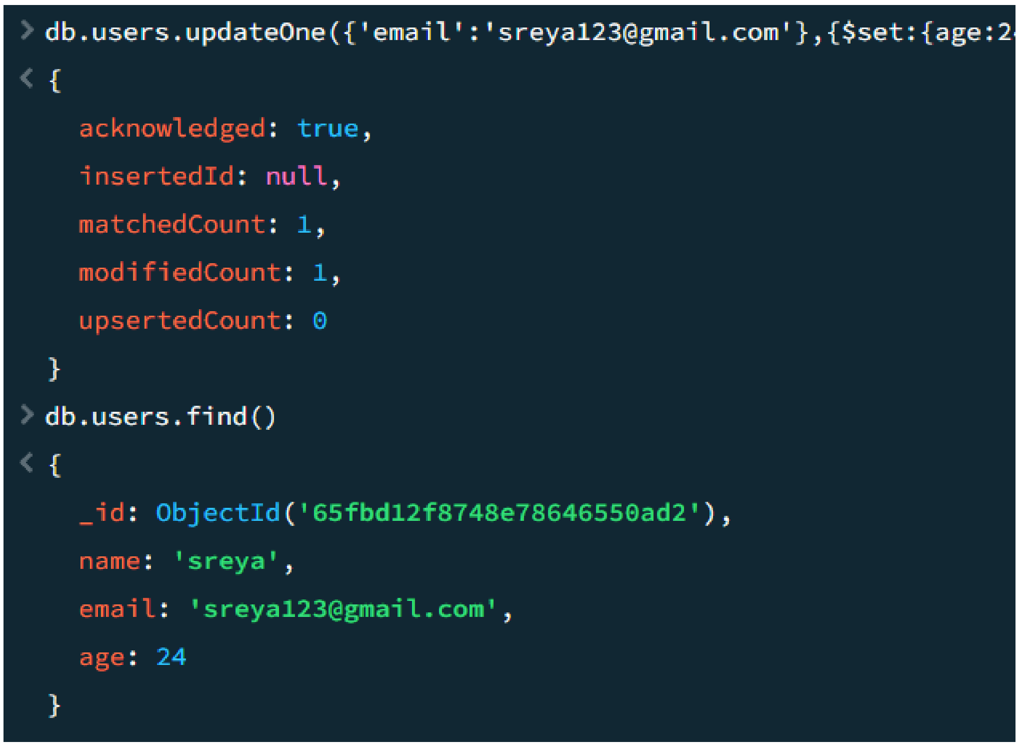
```
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('65fbd12f8748e78646550ad2'),
    '1': ObjectId('65fbd12f8748e78646550ad3'),
    '2': ObjectId('65fbd12f8748e78646550ad4')
  }
}
> db.myDatabase.find()
< {
  _id: ObjectId('65fbcd518748e78646550acf'),
  name: 'sreya',
  email: 'sreya123@gmail.com',
  age: 20
}
{
  _id: ObjectId('65fbcd518748e78646550ad0'),
  name: 'rasi',
  email: 'rasi@gmail.com',
  age: 21
}
{
  _id: ObjectId('65fbcd518748e78646550ad1'),
  name: 'pradeep',
  email: 'pradeep@gmail.com',
  age: 22
}
```

2.Users with an age greater than or equal to 20.



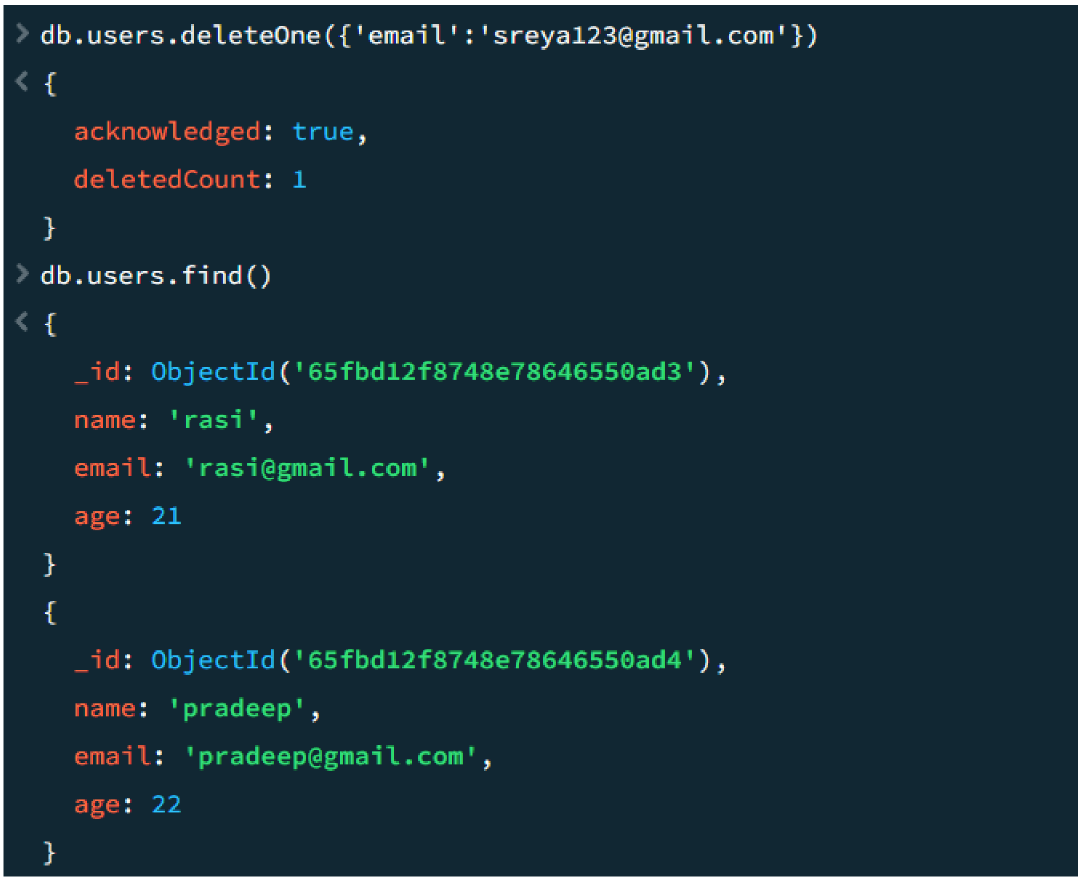
```
> db.users.find({age:{$gt:20}})
< {
    _id: ObjectId('65fbd12f8748e78646550ad3'),
    name: 'rasi',
    email: 'rasi@gmail.com',
    age: 21
  }
  {
    _id: ObjectId('65fbd12f8748e78646550ad4'),
    name: 'pradeep',
    email: 'pradeep@gmail.com',
    age: 22
  }
```

3.Update Operation: Update the age of a user with a specific email address.



```
> db.users.updateOne({'email':'sreya123@gmail.com'},{$set:{age:2
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 1,
    modifiedCount: 1,
    upsertedCount: 0
  }
> db.users.find()
< {
    _id: ObjectId('65fbd12f8748e78646550ad2'),
    name: 'sreya',
    email: 'sreya123@gmail.com',
    age: 24
  }
```

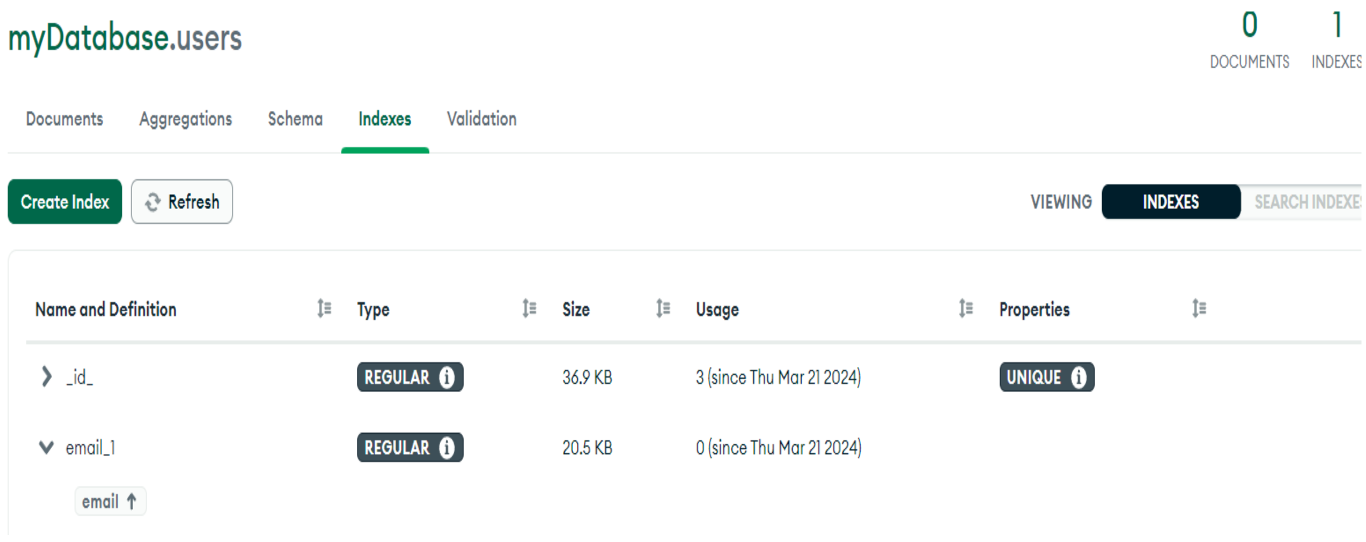
4.Deletion Operation: Delete a user document based on a specific email address.



```
> db.users.deleteOne({'email':'sreya123@gmail.com'})
< {
    acknowledged: true,
    deletedCount: 1
  }
> db.users.find()
< {
    _id: ObjectId('65fbd12f8748e78646550ad3'),
    name: 'rasi',
    email: 'rasi@gmail.com',
    age: 21
  }
  {
    _id: ObjectId('65fbd12f8748e78646550ad4'),
    name: 'pradeep',
    email: 'pradeep@gmail.com',
    age: 22
  }
```

5.Index Creation: Create an index on the email field of the users collection.

```
db.users.createIndex({ "email": 1 })
email_1
db.users.getIndexes()
[
  { v: 2, key: { _id: 1 }, name: '_id_' },
  { v: 2, key: { email: 1 }, name: 'email_1' }
]
```



Name and Definition	Type	Size	Usage	Properties
_id_	REGULAR	36.9 KB	3 (since Thu Mar 21 2024)	UNIQUE
email_1 email ↑	REGULAR	20.5 KB	0 (since Thu Mar 21 2024)	