MONGO DB ASSIGNMENT

Step-by-Step MongoDB Installation (Windows)

Step 1: Download MongoDB

- Go to: https://www.mongodb.com/try/download/community
- Select: Windows, Version: "MSI", Package: "Current Release"
- Click Download

Step 2: Install MongoDB

- Run the downloaded .msi file.
- Choose Complete setup.
- Enable MongoDB as a Service.
- Select checkbox to install MongoDB Compass .

Step 3: Set MongoDB to Path

- Add this to environment variable PATH:

```
C:\Program Files\MongoDB\Server\<version>\bin
```

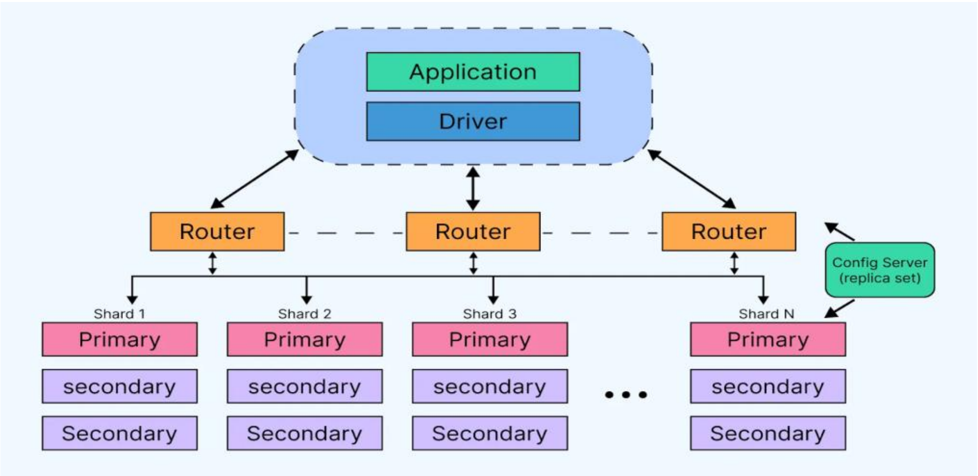
- Open Command Prompt, type:

```
mongo --version
```

Step 4: Download and Run MongoDB Shell

https://www.mongodb.com/try/download/shell ---mongoshell

MongoDB Architecture



Advantages of MongoDB

- **Flexible Schema Design**

  MongoDB does not require a predefined schema. Different documents in the same collection can have different fields and structures.
- **Document-Oriented Storage**

  Data is stored in BSON (binary JSON) format, which is easy to map with modern application data types and structures.
- **High Performance**

  Supports fast read and write operations due to efficient indexing and in-memory processing.
- **Scalability**

  Provides horizontal scalability using sharding, allowing data to be distributed across multiple servers.
- **Powerful Indexing**

  Offers multiple types of indexes (single field, compound, text, geospatial, etc.) to improve query performance.
- **Aggregation Framework**

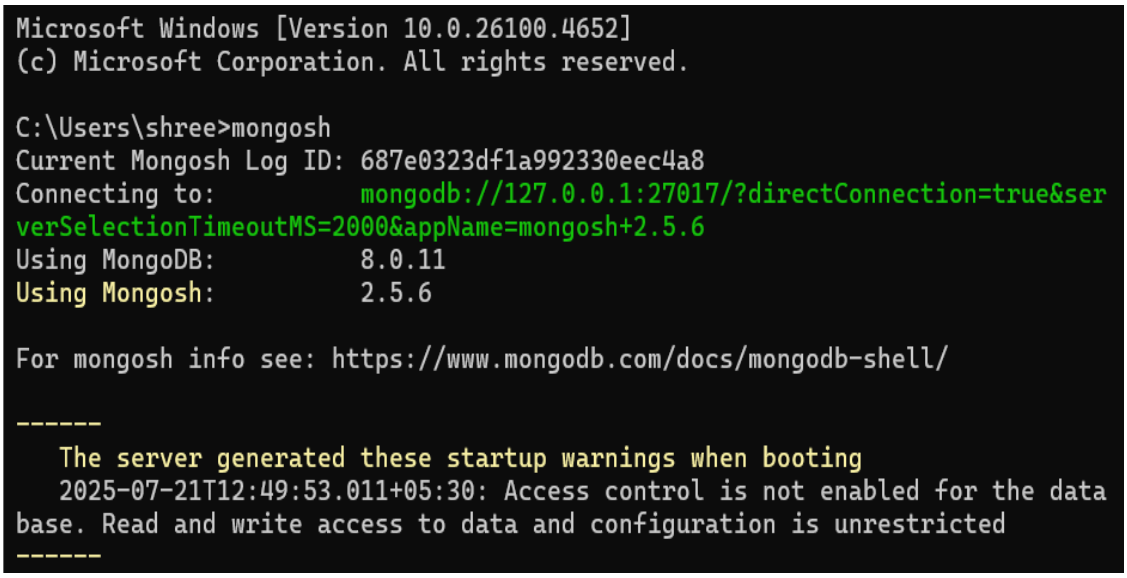
  Allows processing and transforming of data within the database using pipelines and stages (like match, group, sort).
- **Replication and High Availability**

  Uses replica sets to ensure data redundancy and failover support in case of server failure.
- **Built-in Sharding**

  Automatically splits large datasets across multiple machines, helping to handle large volumes of data efficiently.

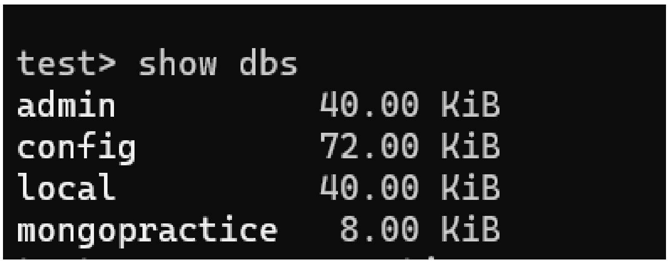
PRACTISED QUERIES IN MONGO DB:

1. **Entering into mongo shell:**



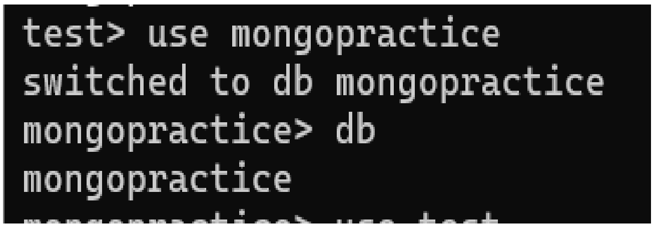
```
Microsoft Windows [Version 10.0.26100.4652]
(c) Microsoft Corporation. All rights reserved.

C:\Users\shree>mongosh
Current Mongosh Log ID: 687e0323df1a992330eec4a8
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&ser
verSelectionTimeoutMS=2000&appName=mongosh+2.5.6
Using MongoDB:          8.0.11
Using Mongosh:          2.5.6

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

------
   The server generated these startup warnings when booting
   2025-07-21T12:49:53.011+05:30: Access control is not enabled for the data
base. Read and write access to data and configuration is unrestricted
------
```

2. **Displaying all the databases:**



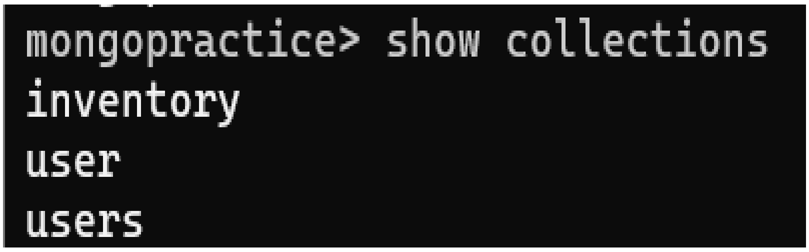
```
test> show dbs
admin          40.00 KiB
config         72.00 KiB
local          40.00 KiB
mongopractice   8.00 KiB
```

3. **Using the database:**



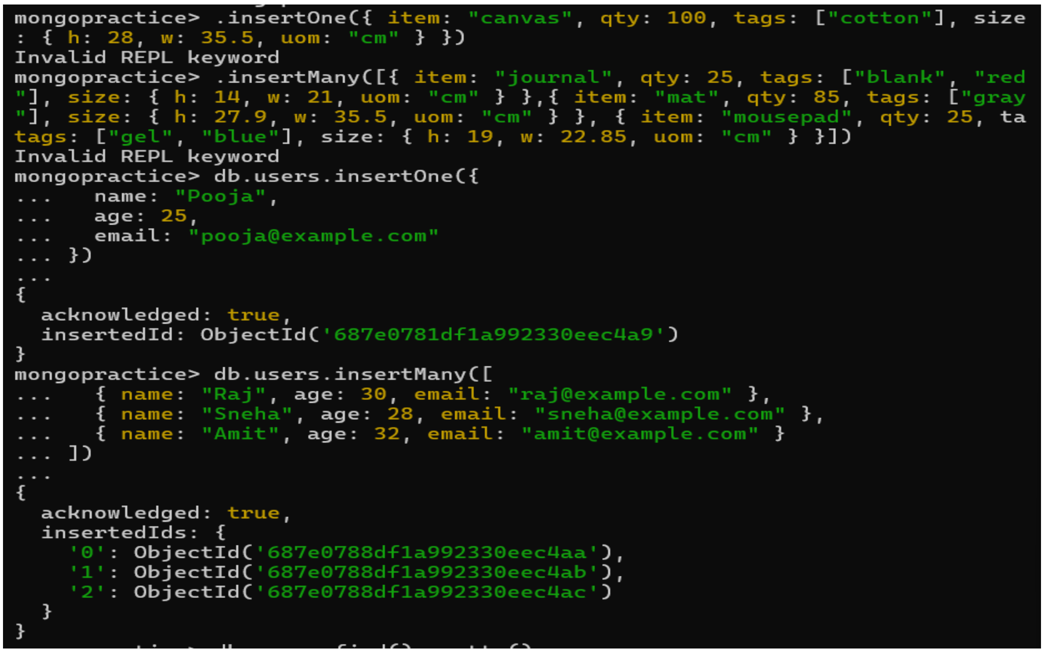
```
test> use mongopractice
switched to db mongopractice
mongopractice> db
mongopractice
```

4. **Displaying all the collections:**

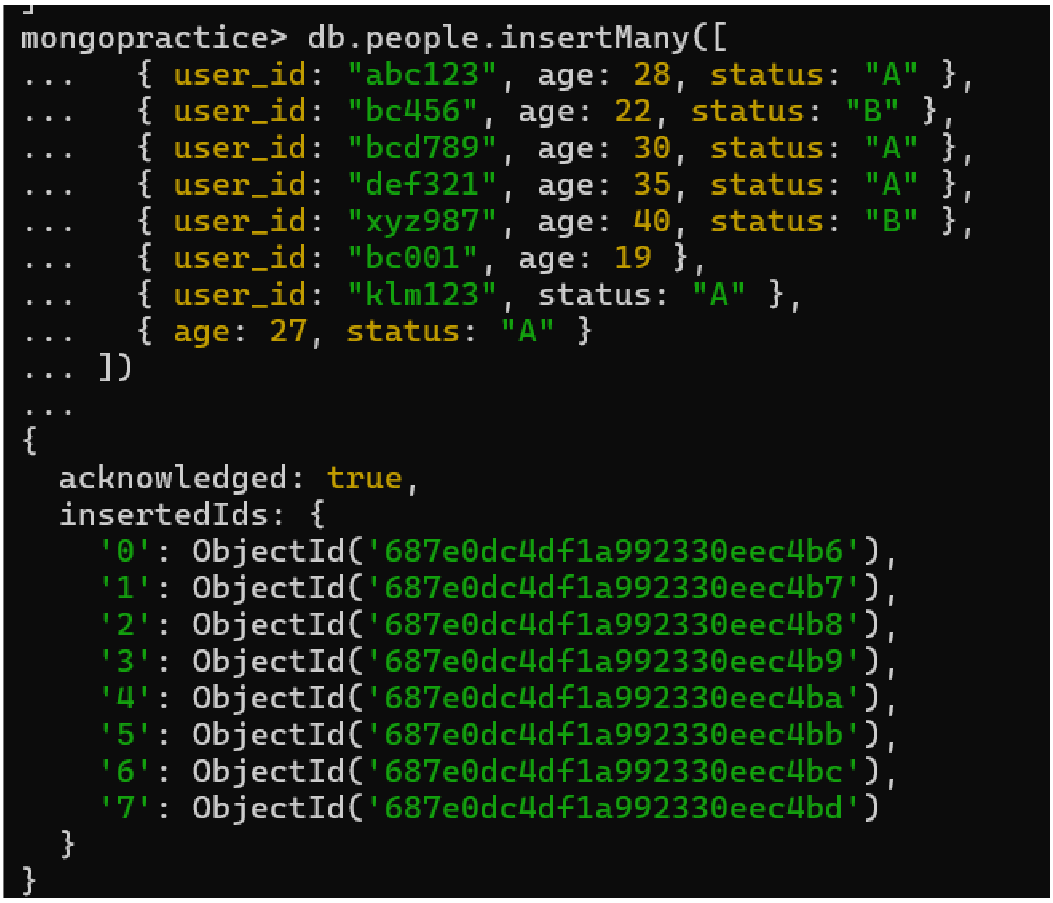


```
mongopractice> show collections
inventory
user
users
```

5. Inserting documents:



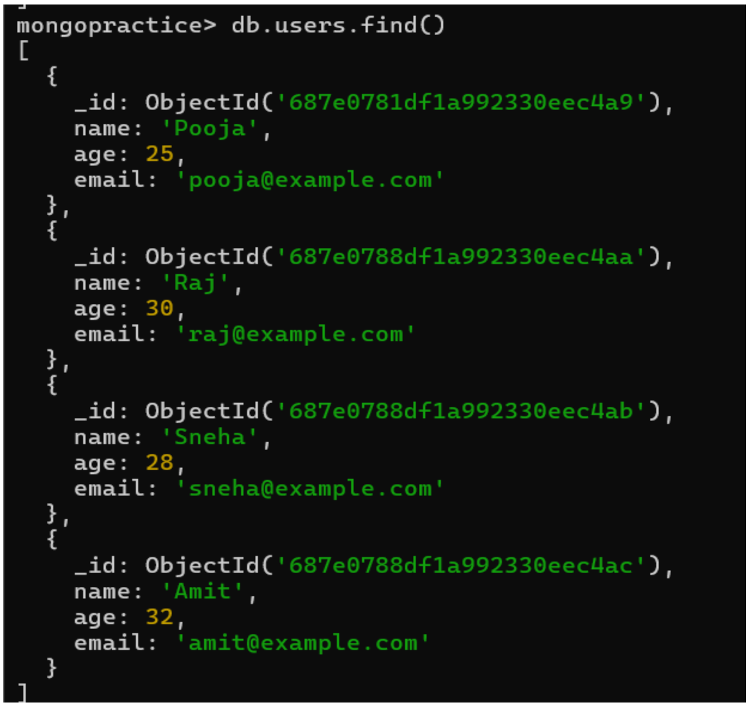
```
mongopractice> .insertOne({ item: "canvas", qty: 100, tags: ["cotton"], size
: { h: 28, w: 35.5, uom: "cm" } })
Invalid REPL keyword
mongopractice> .insertMany([{ item: "journal", qty: 25, tags: ["blank", "red
"], size: { h: 14, w: 21, uom: "cm" } },{ item: "mat", qty: 85, tags: ["gray
"], size: { h: 27.9, w: 35.5, uom: "cm" } }, { item: "mousepad", qty: 25, ta
tags: ["gel", "blue"], size: { h: 19, w: 22.85, uom: "cm" } }])
Invalid REPL keyword
mongopractice> db.users.insertOne({
...    name: "Pooja",
...    age: 25,
...    email: "pooja@example.com"
... })
...
{
  acknowledged: true,
  insertedId: ObjectId('687e0781df1a992330eec4a9')
}
mongopractice> db.users.insertMany([
...    { name: "Raj", age: 30, email: "raj@example.com" },
...    { name: "Sneha", age: 28, email: "sneha@example.com" },
...    { name: "Amit", age: 32, email: "amit@example.com" }
... ])
...
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('687e0788df1a992330eec4aa'),
    '1': ObjectId('687e0788df1a992330eec4ab'),
    '2': ObjectId('687e0788df1a992330eec4ac')
  }
}
```



```
mongopractice> db.people.insertMany([
...    { user_id: "abc123", age: 28, status: "A" },
...    { user_id: "bc456", age: 22, status: "B" },
...    { user_id: "bcd789", age: 30, status: "A" },
...    { user_id: "def321", age: 35, status: "A" },
...    { user_id: "xyz987", age: 40, status: "B" },
...    { user_id: "bc001", age: 19 },
...    { user_id: "klm123", status: "A" },
...    { age: 27, status: "A" }
... ])
...
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('687e0dc4df1a992330eec4b6'),
    '1': ObjectId('687e0dc4df1a992330eec4b7'),
    '2': ObjectId('687e0dc4df1a992330eec4b8'),
    '3': ObjectId('687e0dc4df1a992330eec4b9'),
    '4': ObjectId('687e0dc4df1a992330eec4ba'),
    '5': ObjectId('687e0dc4df1a992330eec4bb'),
    '6': ObjectId('687e0dc4df1a992330eec4bc'),
    '7': ObjectId('687e0dc4df1a992330eec4bd')
  }
}
```

6. Displaying all the documents:

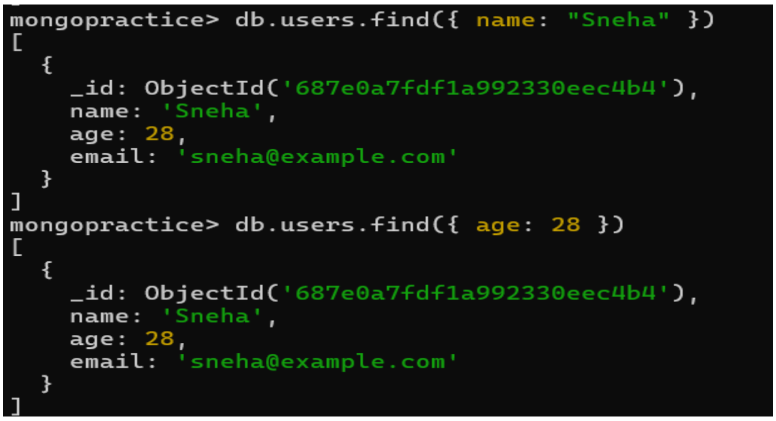
a) Find All:



```
mongopractice> db.users.find()
[
  {
    _id: ObjectId('687e0781df1a992330eec4a9'),
    name: 'Pooja',
    age: 25,
    email: 'pooja@example.com'
  },
  {
    _id: ObjectId('687e0788df1a992330eec4aa'),
    name: 'Raj',
    age: 30,
    email: 'raj@example.com'
  },
  {
    _id: ObjectId('687e0788df1a992330eec4ab'),
    name: 'Sneha',
    age: 28,
    email: 'sneha@example.com'
  },
  {
    _id: ObjectId('687e0788df1a992330eec4ac'),
    name: 'Amit',
    age: 32,
    email: 'amit@example.com'
  }
]
```

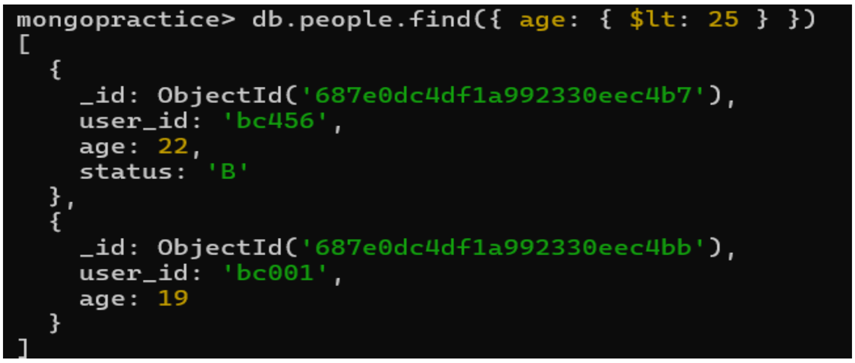
7. Filtering records based on conditions:

- ### b) Find with Filter:



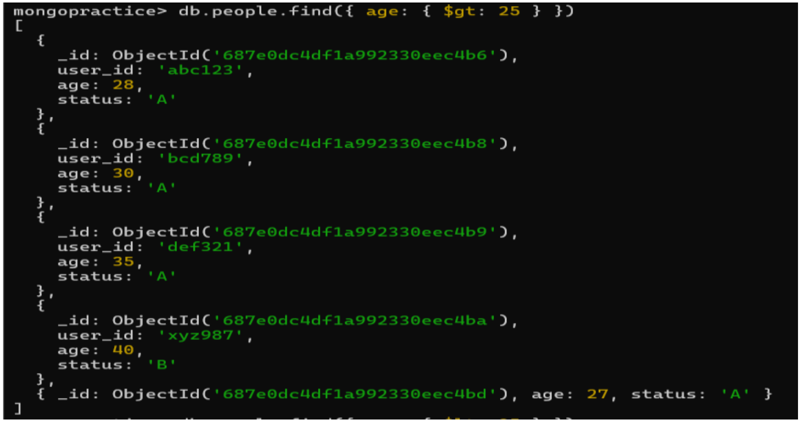
```
mongopractice> db.users.find({ name: "Sneha" })
[
  {
    _id: ObjectId('687e0a7fdf1a992330eec4b4'),
    name: 'Sneha',
    age: 28,
    email: 'sneha@example.com'
  }
]
mongopractice> db.users.find({ age: 28 })
[
  {
    _id: ObjectId('687e0a7fdf1a992330eec4b4'),
    name: 'Sneha',
    age: 28,
    email: 'sneha@example.com'
  }
]
```

c) Find all people with age lesser than 25



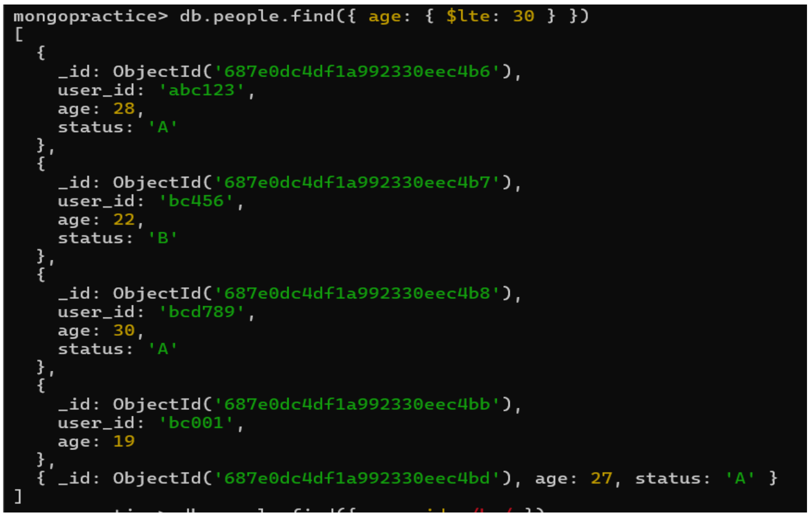
```
mongopractice> db.people.find({ age: { $lt: 25 } })
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4b7'),
    user_id: 'bc456',
    age: 22,
    status: 'B'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4bb'),
    user_id: 'bc001',
    age: 19
  }
]
```

d) Find all people with age greater than 25



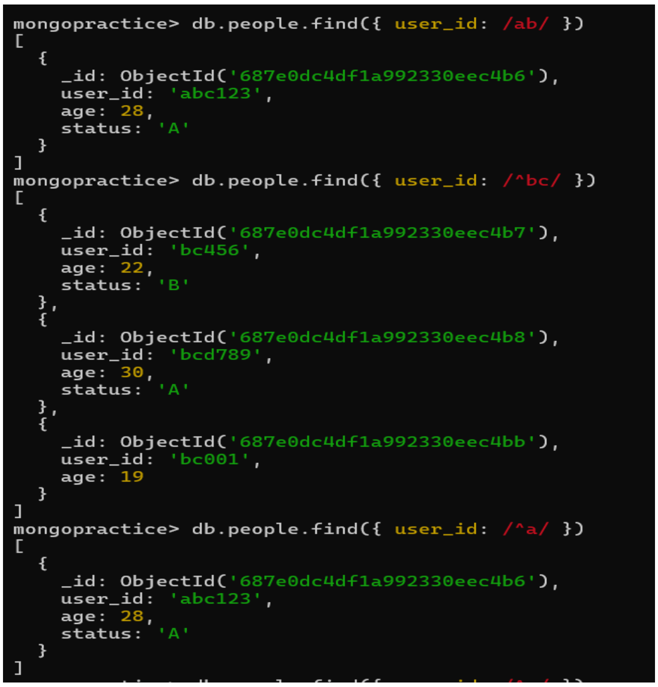
```
mongopractice> db.people.find({ age: { $gt: 25 } })
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4b6'),
    user_id: 'abc123',
    age: 28,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b8'),
    user_id: 'bcd789',
    age: 30,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b9'),
    user_id: 'def321',
    age: 35,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4ba'),
    user_id: 'xyz987',
    age: 40,
    status: 'B'
  },
  { _id: ObjectId('687e0dc4df1a992330eec4bd'), age: 27, status: 'A' }
]
```

e) Find all people with age lesser than and equal to 30



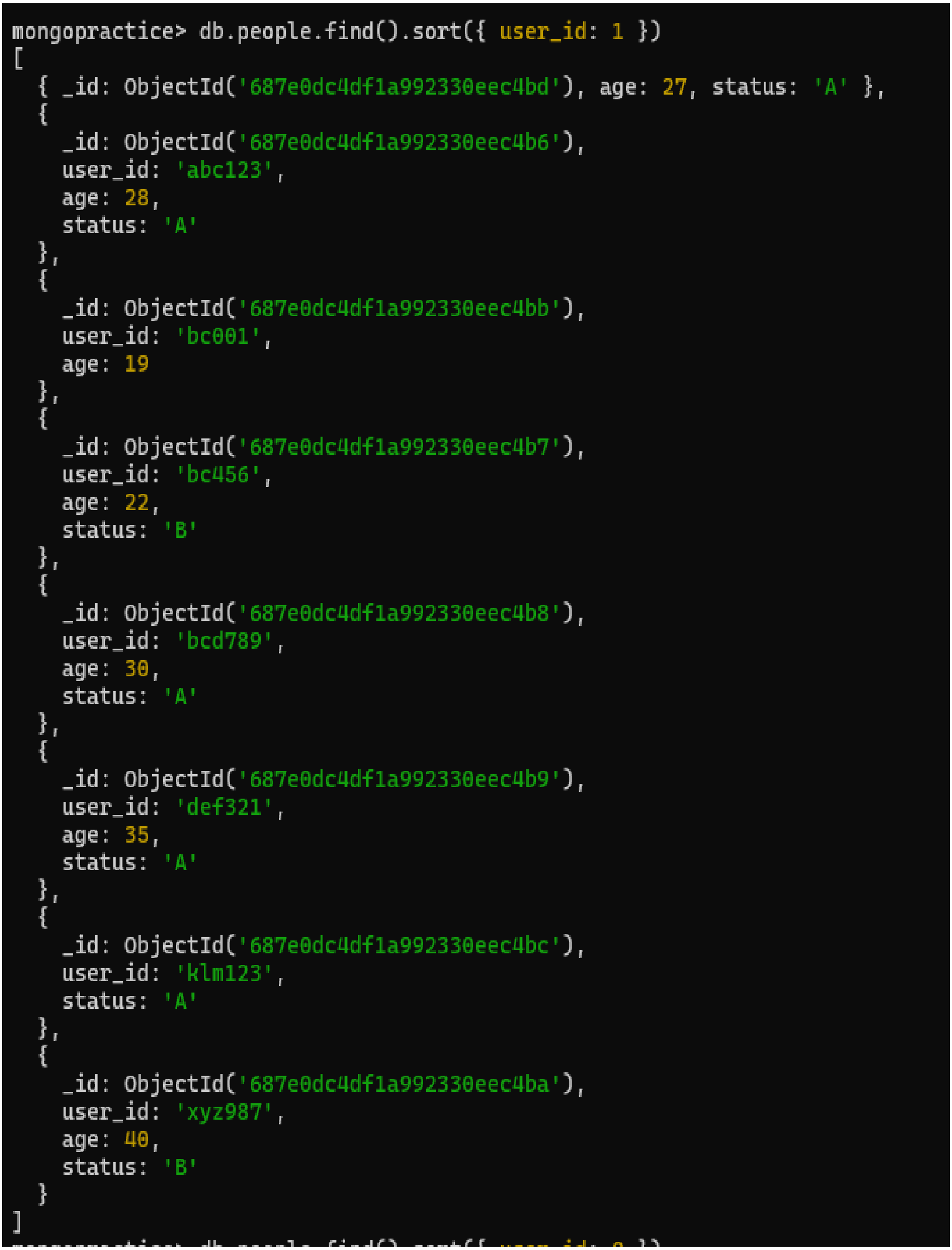
```
mongopractice> db.people.find({ age: { $lte: 30 } })
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4b6'),
    user_id: 'abc123',
    age: 28,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b7'),
    user_id: 'bc456',
    age: 22,
    status: 'B'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b8'),
    user_id: 'bcd789',
    age: 30,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4bb'),
    user_id: 'bc001',
    age: 19
  },
  { _id: ObjectId('687e0dc4df1a992330eec4bd'), age: 27, status: 'A' }
]
```

f) Find all people whose user_id contains “ab”,"bc" and starts with “a”



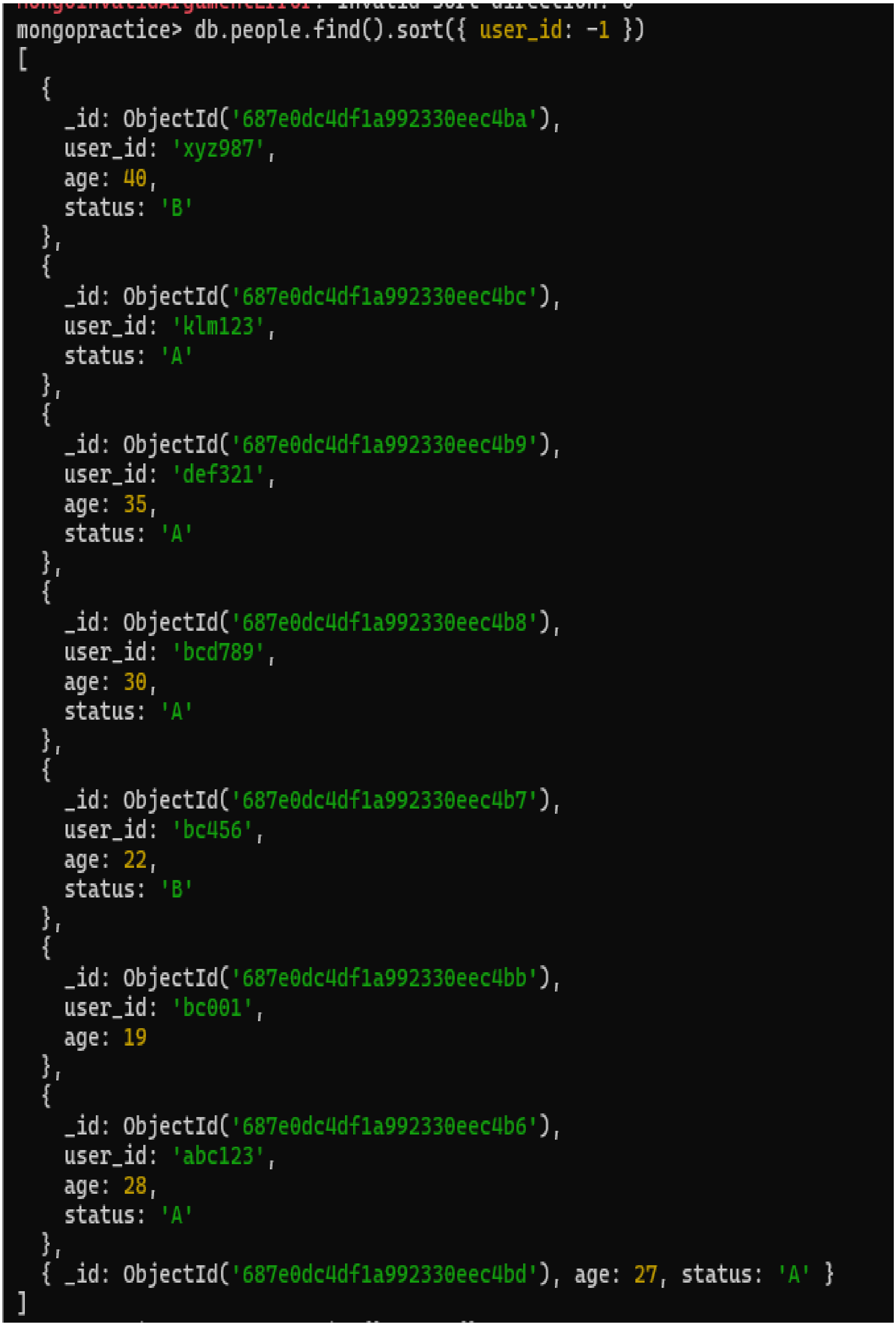
```
mongopractice> db.people.find({ user_id: /ab/ })
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4b6'),
    user_id: 'abc123',
    age: 28,
    status: 'A'
  }
]
mongopractice> db.people.find({ user_id: /^bc/ })
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4b7'),
    user_id: 'bc456',
    age: 22,
    status: 'B'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b8'),
    user_id: 'bcd789',
    age: 30,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4bb'),
    user_id: 'bc001',
    age: 19
  }
]
mongopractice> db.people.find({ user_id: /^a/ })
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4b6'),
    user_id: 'abc123',
    age: 28,
    status: 'A'
  }
]
```

g) Sort Results (Ascending age):



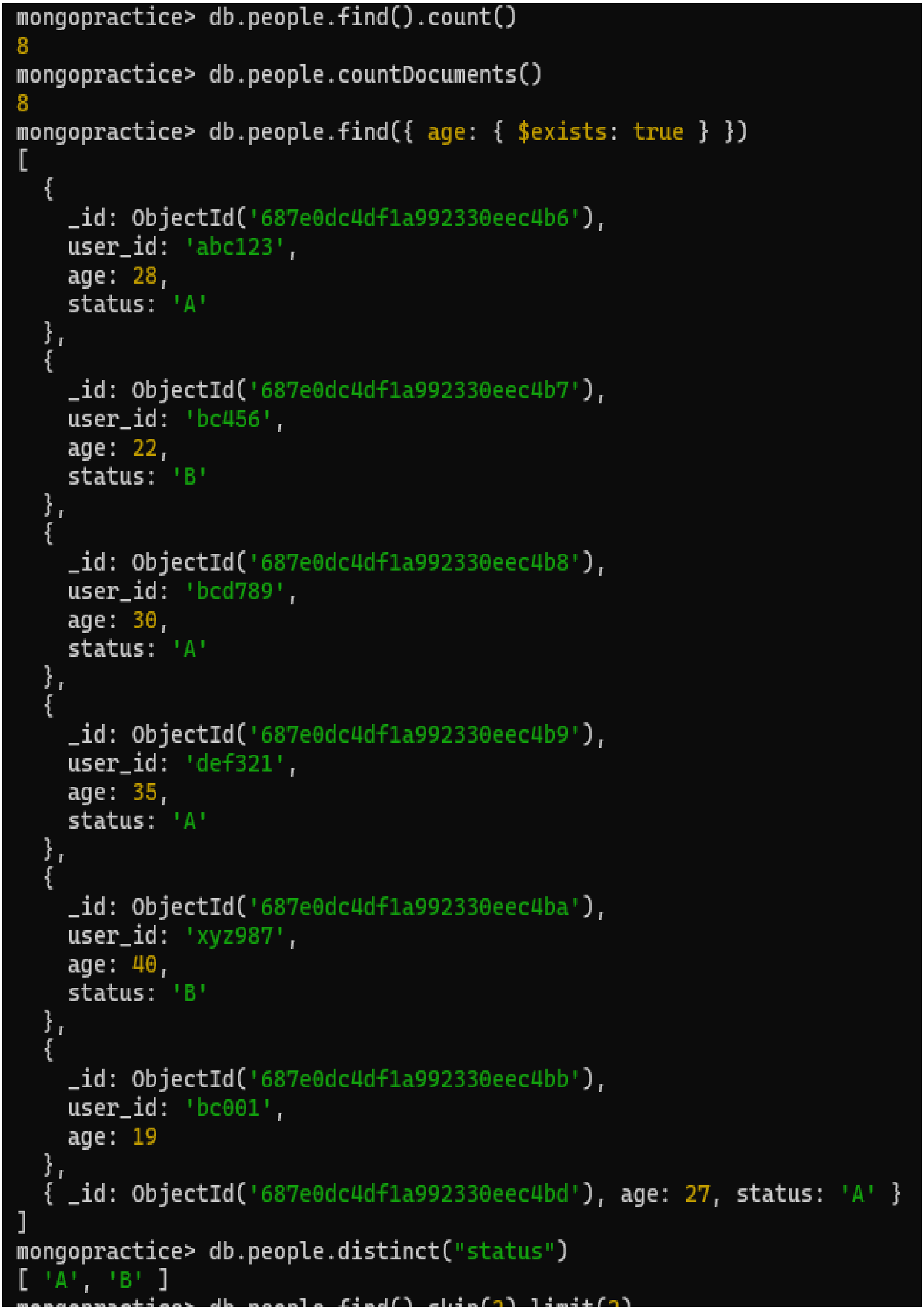
```
mongopractice> db.people.find().sort({ user_id: 1 })
[
  { _id: ObjectId('687e0dc4df1a992330eec4bd'), age: 27, status: 'A' },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b6'),
    user_id: 'abc123',
    age: 28,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4bb'),
    user_id: 'bc001',
    age: 19
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b7'),
    user_id: 'bc456',
    age: 22,
    status: 'B'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b8'),
    user_id: 'bcd789',
    age: 30,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b9'),
    user_id: 'def321',
    age: 35,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4bc'),
    user_id: 'klm123',
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4ba'),
    user_id: 'xyz987',
    age: 40,
    status: 'B'
  }
]
mongopractice> db.people.find().sort({ user_id: 0 })
```

h) Sort Results (descending age):



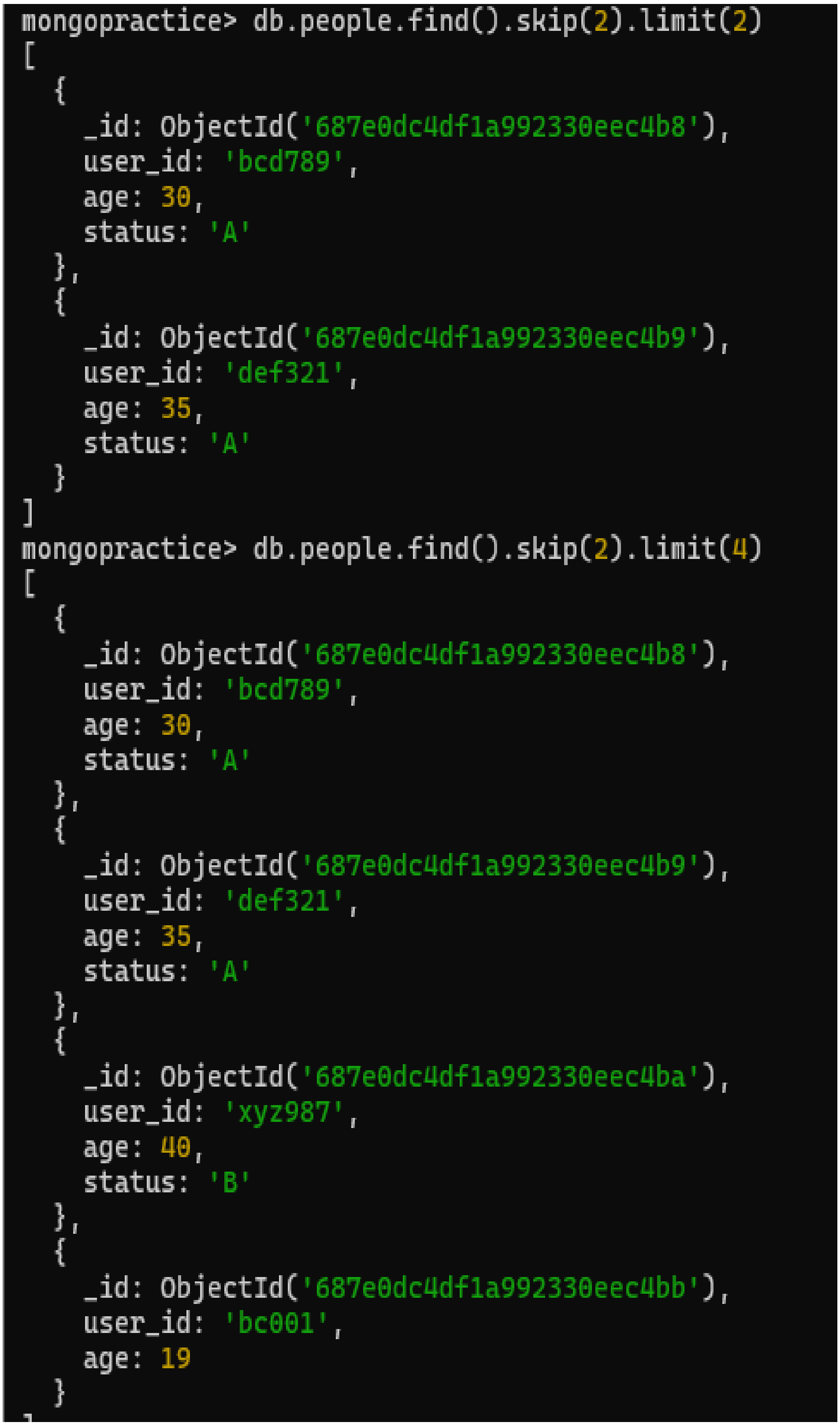
```
MongoInvalidArgumentError: Invalid sort direction: 0
mongopractice> db.people.find().sort({ user_id: -1 })
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4ba'),
    user_id: 'xyz987',
    age: 40,
    status: 'B'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4bc'),
    user_id: 'klm123',
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b9'),
    user_id: 'def321',
    age: 35,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b8'),
    user_id: 'bcd789',
    age: 30,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b7'),
    user_id: 'bc456',
    age: 22,
    status: 'B'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4bb'),
    user_id: 'bc001',
    age: 19
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b6'),
    user_id: 'abc123',
    age: 28,
    status: 'A'
  },
  { _id: ObjectId('687e0dc4df1a992330eec4bd'), age: 27, status: 'A' }
]
```

i) Count Documents and Finding all people who have an age field:



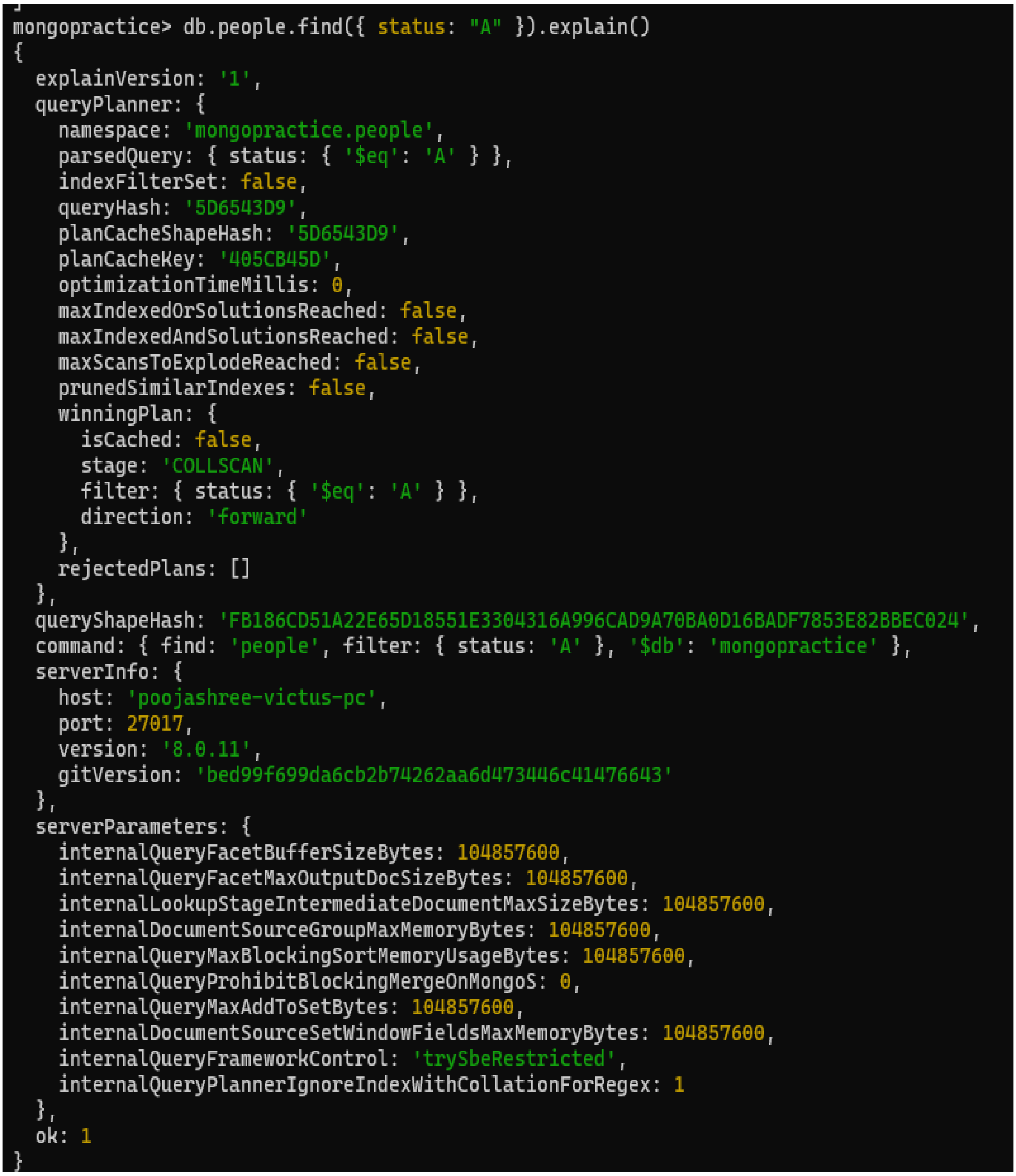
```
mongopractice> db.people.find().count()
8
mongopractice> db.people.countDocuments()
8
mongopractice> db.people.find({ age: { $exists: true } })
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4b6'),
    user_id: 'abc123',
    age: 28,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b7'),
    user_id: 'bc456',
    age: 22,
    status: 'B'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b8'),
    user_id: 'bcd789',
    age: 30,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b9'),
    user_id: 'def321',
    age: 35,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4ba'),
    user_id: 'xyz987',
    age: 40,
    status: 'B'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4bb'),
    user_id: 'bc001',
    age: 19
  },
  { _id: ObjectId('687e0dc4df1a992330eec4bd'), age: 27, status: 'A' }
]
mongopractice> db.people.distinct("status")
[ 'A', 'B' ]
mongopractice> db.people.find().skip(2).limit(2)
```

j) Using skip and limit :



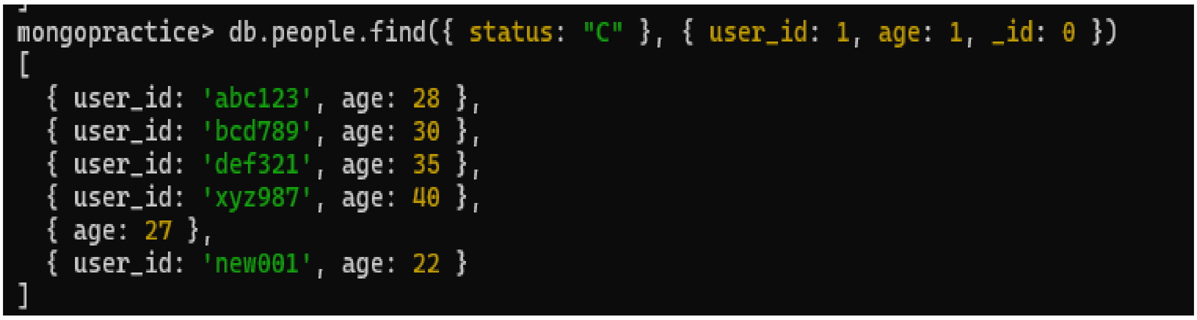
```
mongopractice> db.people.find().skip(2).limit(2)
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4b8'),
    user_id: 'bcd789',
    age: 30,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b9'),
    user_id: 'def321',
    age: 35,
    status: 'A'
  }
]
mongopractice> db.people.find().skip(2).limit(4)
[
  {
    _id: ObjectId('687e0dc4df1a992330eec4b8'),
    user_id: 'bcd789',
    age: 30,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4b9'),
    user_id: 'def321',
    age: 35,
    status: 'A'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4ba'),
    user_id: 'xyz987',
    age: 40,
    status: 'B'
  },
  {
    _id: ObjectId('687e0dc4df1a992330eec4bb'),
    user_id: 'bc001',
    age: 19
  }
```

k) Using explain:



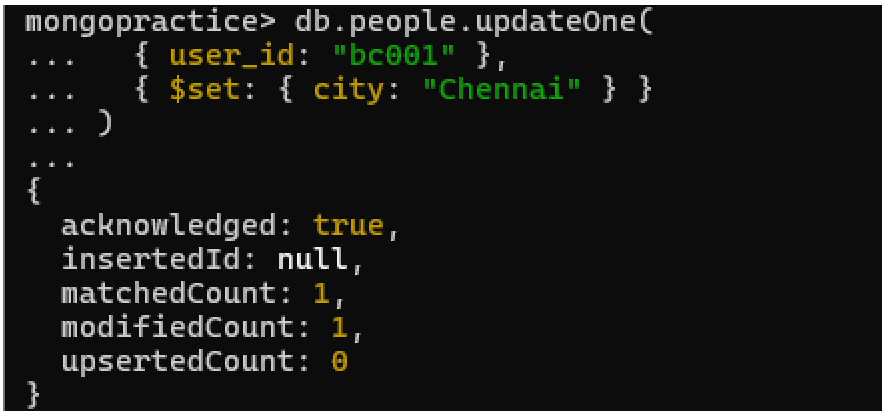
```
mongopractice> db.people.find({ status: "A" }).explain()
{
  explainVersion: '1',
  queryPlanner: {
    namespace: 'mongopractice.people',
    parsedQuery: { status: { '$eq': 'A' } },
    indexFilterSet: false,
    queryHash: '5D6543D9',
    planCacheShapeHash: '5D6543D9',
    planCacheKey: '405CB45D',
    optimizationTimeMillis: 0,
    maxIndexedOrSolutionsReached: false,
    maxIndexedAndSolutionsReached: false,
    maxScansToExplodeReached: false,
    prunedSimilarIndexes: false,
    winningPlan: {
      isCached: false,
      stage: 'COLLSCAN',
      filter: { status: { '$eq': 'A' } },
      direction: 'forward'
    },
    rejectedPlans: []
  },
  queryShapeHash: 'FB186CD51A22E65D18551E3304316A996CAD9A70BA0D16BADF7853E82BBEC024',
  command: { find: 'people', filter: { status: 'A' }, '$db': 'mongopractice' },
  serverInfo: {
    host: 'poojashree-victus-pc',
    port: 27017,
    version: '8.0.11',
    gitVersion: 'bed99f699da6cb2b74262aa6d473446c41476643'
  },
  serverParameters: {
    internalQueryFacetBufferSizeBytes: 104857600,
    internalQueryFacetMaxOutputDocSizeBytes: 104857600,
    internalLookupStageIntermediateDocumentMaxSizeBytes: 104857600,
    internalDocumentSourceGroupMaxMemoryBytes: 104857600,
    internalQueryMaxBlockingSortMemoryUsageBytes: 104857600,
    internalQueryProhibitBlockingMergeOnMongoS: 0,
    internalQueryMaxAddToSetBytes: 104857600,
    internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600,
    internalQueryFrameworkControl: 'trySbeRestricted',
    internalQueryPlannerIgnoreIndexWithCollationForRegex: 1
  },
  ok: 1
}
```

l) Find Specific Fields:



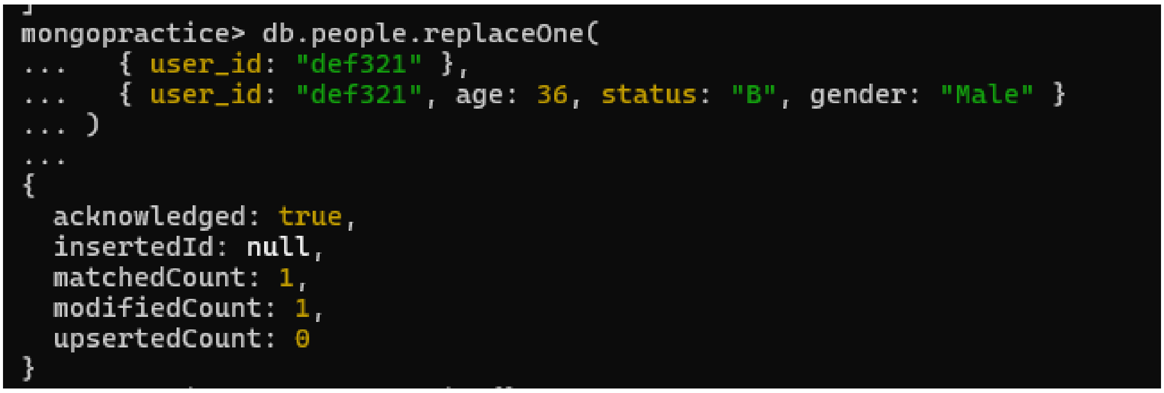
```
mongopractice> db.people.find({ status: "C" }, { user_id: 1, age: 1, _id: 0 })
[
  { user_id: 'abc123', age: 28 },
  { user_id: 'bcd789', age: 30 },
  { user_id: 'def321', age: 35 },
  { user_id: 'xyz987', age: 40 },
  { age: 27 },
  { user_id: 'new001', age: 22 }
]
```

j) Update – Modify Documents



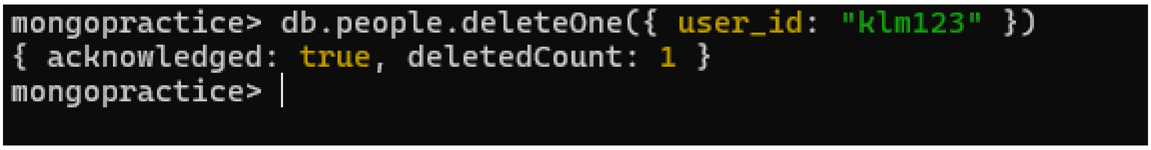
```
mongopractice> db.people.updateOne(
...   { user_id: "bc001" },
...   { $set: { city: "Chennai" } }
... )
...
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

k) Replace a Document



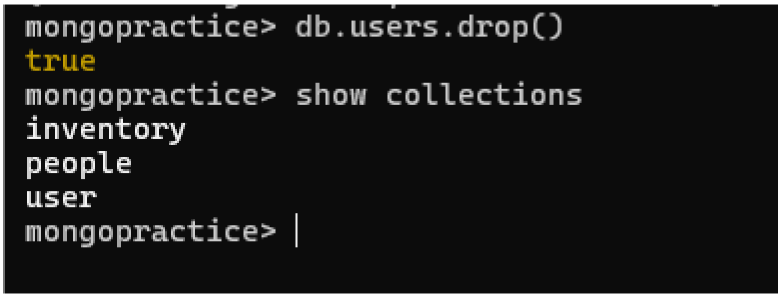
```
mongopractice> db.people.replaceOne(
...   { user_id: "def321" },
...   { user_id: "def321", age: 36, status: "B", gender: "Male" }
... )
...
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

l) Delete – Remove Documents



```
mongopractice> db.people.deleteOne({ user_id: "klm123" })
{ acknowledged: true, deletedCount: 1 }
mongopractice> |
```

m) Drop Entire Collection:



```
mongopractice> db.users.drop()
true
mongopractice> show collections
inventory
people
user
mongopractice> |
```