MongoDB Assignment -1

Installation of MongoDB Community Server

Step 1:
Go to: https://www.mongodb.com/try/download/community
Select:

- Version: Latest LTS
- OS: Windows
- Package: MSI

Click Download.

Step 2:
Open the downloaded .msi file.

Step 3:
In the setup wizard, choose Complete setup type.

Step 4:
In the service configuration:

- Select Run service as Network Service user
- Keep default settings to start MongoDB as a service

Step 5:
Install MongoDB Compass (GUI tool).

Step 6:
Click Install, then click Finish.

Step 7:
MongoDB gets installed at:
C:\Program Files\MongoDB\Server\<version>\bin

Then add this path to the system path

Step 8:
To check installation, open Command Prompt and run:

mongod --version

```
C:\Users\Admin>mongod --version
db version v8.0.11
Build Info: {
    "version": "8.0.11",
    "gitVersion": "bed99f699da6cb2b74262aa6d473446c41476643",
    "modules": [],
    "allocator": "tcmalloc-gperf",
    "environment": {
        "distmod": "windows",
        "distarch": "x86_64",
        "target_arch": "x86_64"
    }
}
```

Installation of MongoDB Shell (mongosh)

Step 1:
Go to: https://www.mongodb.com/try/download/shell
Select:

- Version: Latest
- OS: Windows
- Package: MSI

Click Download.

Step 2:
Open the downloaded .msi file.

Step 3:
 Click Next, then Install, then Finish.

Step 4:
Shell installed at:
 C:\Program Files\MongoDB\mongosh\bin

Step 5:
To check installation, open Command Prompt and run:

mongosh

```
C:\Users\Admin>mongosh
Current Mongosh Log ID: 687e03225a68bf389feec4a8
Connecting to:          mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=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/
```

Database Creation

```
mongodb> use jisl
switched to db jisl
jisl> db
jisl
jisl> show dbs
admin      40.00 KiB
config   108.00 KiB
local      40.00 KiB
jisl> db
jisl
jisl> db.dropDatabase()
{ ok: 1, dropped: 'jisl' }
jisl> db
jisl
jisl> show db
```

Collection Operation

```
jisl> db.createCollection("user")
{ ok: 1 }
jisl> show collections
user
jisl> db.user.drop()
true
jisl> db.createCollection("inventory")
{ ok: 1 }
jisl> show collections
inventory
```

Create Operations

```
jisl> db.inventory.insertOne({ item: "canvas", qty: 100, tags: ["cotton"], size: { h: 28, w:
... 35.5, uom: "cm" } })
{
  acknowledged: true,
  insertedId: ObjectId('687e08c95a68bf389feec4a9')
}
jisl> db.inventory.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, tags: ["gel", "blue"], size: { h: 19,
... w: 22.85, uom: "cm" } }])
...
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('687e08d25a68bf389feec4aa'),
    '1': ObjectId('687e08d25a68bf389feec4ab'),
    '2': ObjectId('687e08d25a68bf389feec4ac')
  }
}
```

```
jisl> db.inventory.insertMany([
...      {
...         item: "notebook",
...         qty: 60,
...         tags: ["lined", "white"],
...         size: { h: 24, w: 18, uom: "cm" }
...      },
...      {
...         item: "diary",
...         qty: 25,
...         tags: ["hardcover", "blue"],
...         size: { h: 21, w: 15, uom: "cm" }
...      },
...      {
...         item: "stickerbook",
...         qty: 10,
...         tags: ["colorful", "fun"],
...         size: { h: 20, w: 20, uom: "cm" }
...      }
...   ])
```

Read Operations

```
jisl> db.inventory.find({$or: [{item:"mat"}, {tags:'gray'}]})
[
  {
    _id: ObjectId('687e08d25a68bf389feec4ab'),
    item: 'mat',
    qty: 85,
    tags: [ 'gray' ],
    size: { h: 27.9, w: 35.5, uom: 'cm' }
  }
]
jisl> db.inventory.find()
[
  {
    _id: ObjectId('687e08c95a68bf389feec4a9'),
    item: 'canvas',
    qty: 100,
    tags: [ 'cotton' ],
    size: { h: 28, w: 35.5, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4aa'),
    item: 'journal',
    qty: 25,
    tags: [ 'blank', 'red' ],
    size: { h: 14, w: 21, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ab'),
    item: 'mat',
    qty: 85,
    tags: [ 'gray' ],
    size: { h: 27.9, w: 35.5, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ac'),
    item: 'mousepad',
    qty: 25,
    tags: [ 'gel', 'blue' ],
```

```
jisl> db.inventory.find({ item: {$in: ["mat","journal"]}})
[
  {
    _id: ObjectId('687e08d25a68bf389feec4aa'),
    item: 'journal',
    qty: 25,
    tags: [ 'blank', 'red' ],
    size: { h: 14, w: 21, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ab'),
    item: 'mat',
    qty: 85,
    tags: [ 'gray' ],
    size: { h: 27.9, w: 35.5, uom: 'cm' }
  }
]
```

```
jisl> db.inventory.find({item:"journal", qty:25})
[
  {
    _id: ObjectId('687e08d25a68bf389feec4aa'),
    item: 'journal',
    qty: 25,
    tags: [ 'blank', 'red' ],
    size: { h: 14, w: 21, uom: 'cm' }
  }
]
```

```
jisl> db.inventory.find({$or: [{item:"mat"}, {tags:'gray'}]})
[
  {
    _id: ObjectId('687e08d25a68bf389feec4ab'),
    item: 'mat',
    qty: 85,
    tags: [ 'gray' ],
    size: { h: 27.9, w: 35.5, uom: 'cm' }
  }
]
```

MongoDB find() Statements

Retrieving Specific Fields

```
jisl> db.inventory.find({}, { item: 1, qty: 1 })
[
  {
    _id: ObjectId('687e08c95a68bf389feec4a9'),
    item: 'canvas',
    qty: 100
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4aa'),
    item: 'journal',
    qty: 25
  },
  { _id: ObjectId('687e08d25a68bf389feec4ab'), item: 'mat', qty: 85 },
  {
    _id: ObjectId('687e08d25a68bf389feec4ac'),
    item: 'mousepad',
    qty: 25
  }
]
```

```
jisl> db.inventory.find({}, { item: 1, qty: 1, _id: 0 })
[
  { item: 'canvas', qty: 100 },
  { item: 'journal', qty: 25 },
  { item: 'mat', qty: 85 },
  { item: 'mousepad', qty: 25 }
]
jisl> db.inventory.find({ qty: 25 })
[
  {
    _id: ObjectId('687e08d25a68bf389feec4aa'),
    item: 'journal',
    qty: 25,
    tags: [ 'blank', 'red' ],
    size: { h: 14, w: 21, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ac'),
    item: 'mousepad',
    qty: 25,
    tags: [ 'gel', 'blue' ],
    size: { h: 19, w: 22.85, uom: 'cm' }
  }
]
```

```
jisl> db.inventory.find({ qty: 25 }, { item: 1, qty: 1, _id: 0 })
[ { item: 'journal', qty: 25 }, { item: 'mousepad', qty: 25 } ]
jisl> db.inventory.find({ qty: { $ne: 25 } })
[
  {
    _id: ObjectId('687e08c95a68bf389feec4a9'),
    item: 'canvas',
    qty: 100,
    tags: [ 'cotton' ],
    size: { h: 28, w: 35.5, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ab'),
    item: 'mat',
    qty: 85,
    tags: [ 'gray' ],
    size: { h: 27.9, w: 35.5, uom: 'cm' }
  }
]
```

```
jisl> db.inventory.find({ qty: 25, item: "journal" })
[
  {
    _id: ObjectId('687e08d25a68bf389feec4aa'),
    item: 'journal',
    qty: 25,
    tags: [ 'blank', 'red' ],
    size: { h: 14, w: 21, uom: 'cm' }
  }
]
```

Retrieving Fields with OR operation

```
jisl> db.inventory.find({$or: [{qty:25},{item:"canvas"}]})
[
  {
    _id: ObjectId('687e08c95a68bf389feec4a9'),
    item: 'canvas',
    qty: 100,
    tags: [ 'cotton' ],
    size: { h: 28, w: 35.5, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4aa'),
    item: 'journal',
    qty: 25,
    tags: [ 'blank', 'red' ],
    size: { h: 14, w: 21, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ac'),
    item: 'mousepad',
    qty: 25,
    tags: [ 'gel', 'blue' ],
    size: { h: 19, w: 22.85, uom: 'cm' }
  }
]
```

```
jisl> db.inventory.find({size:{ h: 19, w: 22.85, uom: 'cm'}})
[
  {
    _id: ObjectId('687e08d25a68bf389feec4ac'),
    item: 'mousepad',
    qty: 25,
    tags: [ 'gel', 'blue' ],
    size: { h: 19, w: 22.85, uom: 'cm' }
  }
]
```

Retrieving Fields with gt (greater than) operation

```
jisl> db.inventory.find({qty:{$gt:25}})
[
  {
    _id: ObjectId('687e08c95a68bf389feec4a9'),
    item: 'canvas',
    qty: 100,
    tags: [ 'cotton' ],
    size: { h: 28, w: 35.5, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ab'),
    item: 'mat',
    qty: 85,
    tags: [ 'gray' ],
    size: { h: 27.9, w: 35.5, uom: 'cm' }
  }
]
```

Retrieving Fields with lt (Lesser than) operation

```
jisl> db.inventory.find({qty:{$lt:25}})
[
  {
    _id: ObjectId('687e0ee45a68bf389feec4af'),
    item: 'stickerbook',
    qty: 10,
    tags: [ 'colorful', 'fun' ],
    size: { h: 20, w: 20, uom: 'cm' }
  }
]
jisl> db.inventory.find({qty:{$gt:25}})
[
  {
    _id: ObjectId('687e08c95a68bf389feec4a9'),
    item: 'canvas',
    qty: 100,
    tags: [ 'cotton' ],
    size: { h: 28, w: 35.5, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ab'),
    item: 'mat',
    qty: 85,
    tags: [ 'gray' ],
    size: { h: 27.9, w: 35.5, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4ad'),
    item: 'notebook',
    qty: 60,
    tags: [ 'lined', 'white' ],
    size: { h: 24, w: 18, uom: 'cm' }
  }
]
```

Retrieving Fields with gt and It operation

```
jisl> db.inventory.find({qty:{$gt:25, $lt:70}})
[
  {
    _id: ObjectId('687e0ee45a68bf389feec4ad'),
    item: 'notebook',
    qty: 60,
    tags: [ 'lined', 'white' ],
    size: { h: 24, w: 18, uom: 'cm' }
  }
]
```

```
jisl> db.inventory.find({qty:{$gt:5, $lt:70}})
[
  {
    _id: ObjectId('687e08d25a68bf389feec4aa'),
    item: 'journal',
    qty: 25,
    tags: [ 'blank', 'red' ],
    size: { h: 14, w: 21, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ac'),
    item: 'mousepad',
    qty: 25,
    tags: [ 'gel', 'blue' ],
    size: { h: 19, w: 22.85, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4ad'),
    item: 'notebook',
    qty: 60,
    tags: [ 'lined', 'white' ],
    size: { h: 24, w: 18, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4ae'),
    item: 'diary',
    qty: 25,
    tags: [ 'hardcover', 'blue' ],
    size: { h: 21, w: 15, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4af'),
    item: 'stickerbook',
    qty: 10,
    tags: [ 'colorful', 'fun' ],
    size: { h: 20, w: 20, uom: 'cm' }
  }
]
```

Count with Condition

```
jisl> db.inventory.find().count()
7
jisl> db.inventory.find({item:{$exists: true}}).count()
7
```

Retrieving Fields (Pattern Match)

```
jisl> db.inventory.find({ item: /di/ })
[
  {
    _id: ObjectId('687e0ee45a68bf389feec4ae'),
    item: 'diary',
    qty: 25,
    tags: [ 'hardcover', 'blue' ],
    size: { h: 21, w: 15, uom: 'cm' }
  }
]
```

```
jisl> db.inventory.find({item: /book/})
[
  {
    _id: ObjectId('687e0ee45a68bf389feec4ad'),
    item: 'notebook',
    qty: 60,
    tags: [ 'lined', 'white' ],
    size: { h: 24, w: 18, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4af'),
    item: 'stickerbook',
    qty: 10,
    tags: [ 'colorful', 'fun' ],
    size: { h: 20, w: 20, uom: 'cm' }
  }
]
```

Sorting Data in Asec

```
jisl> db.inventory.find({tags:"blue"}).sort({qty:1})
[
  {
    _id: ObjectId('687e08d25a68bf389feec4ac'),
    item: 'mousepad',
    qty: 25,
    tags: [ 'gel', 'blue' ],
    size: { h: 19, w: 22.85, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4ae'),
    item: 'diary',
    qty: 25,
    tags: [ 'hardcover', 'blue' ],
    size: { h: 21, w: 15, uom: 'cm' }
  }
]
```

Sorting Data in Desc

```
jisl> db.inventory.find({tags:"blue"}).sort({qty:-1})
[
  {
    _id: ObjectId('687e08d25a68bf389feec4ac'),
    item: 'mousepad',
    qty: 25,
    tags: [ 'gel', 'blue' ],
    size: { h: 19, w: 22.85, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4ae'),
    item: 'diary',
    qty: 25,
    tags: [ 'hardcover', 'blue' ],
    size: { h: 21, w: 15, uom: 'cm' }
  }
```

Finding Distinct Values

```
jisl> db.inventory.distinct("item")
[
  'canvas',
  'diary',
  'journal',
  'mat',
  'mousepad',
  'notebook',
  'stickerbook'
]
```

Limiting Output to 1

```
jisl> db.inventory.find().limit(1)
[
  {
    _id: ObjectId('687e08c95a68bf389feec4a9'),
    item: 'canvas',
    qty: 100,
    tags: [ 'cotton' ],
    size: { h: 28, w: 35.5, uom: 'cm' }
  }
]
```

Skip and Limit

```
jisl> db.inventory.find().limit(5).skip(2)
[
  {
    _id: ObjectId('687e08d25a68bf389feec4ab'),
    item: 'mat',
    qty: 85,
    tags: [ 'gray' ],
    size: { h: 27.9, w: 35.5, uom: 'cm' }
  },
  {
    _id: ObjectId('687e08d25a68bf389feec4ac'),
    item: 'mousepad',
    qty: 25,
    tags: [ 'gel', 'blue' ],
    size: { h: 19, w: 22.85, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4ad'),
    item: 'notebook',
    qty: 60,
    tags: [ 'lined', 'white' ],
    size: { h: 24, w: 18, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4ae'),
    item: 'diary',
    qty: 25,
    tags: [ 'hardcover', 'blue' ],
    size: { h: 21, w: 15, uom: 'cm' }
  },
  {
    _id: ObjectId('687e0ee45a68bf389feec4af'),
    item: 'stickerbook',
    qty: 10,
    tags: [ 'colorful', 'fun' ],
    size: { h: 20, w: 20, uom: 'cm' }
  }
]
```

Explain Query Plan

```
jisl> db.people.find({item:"mat"}).explain()
{
  explainVersion: '1',
  queryPlanner: {
    namespace: 'jisl.people',
    parsedQuery: { item: { '$eq': 'mat' } },
    indexFilterSet: false,
    optimizationTimeMillis: 0,
    maxIndexedOrSolutionsReached: false,
    maxIndexedAndSolutionsReached: false,
    maxScansToExplodeReached: false,
    prunedSimilarIndexes: false,
    winningPlan: { isCached: false, stage: 'EOF' },
    rejectedPlans: []
  },
  queryShapeHash: '7CE3D40893BDB9A69232AE5705A9230F63BE9936BBB472DEB0E7D550A6B41D5C',
  command: { find: 'people', filter: { item: 'mat' }, '$db': 'jisl' },
  serverInfo: {
    host: 'LAPTOP-V1AVE5G1',
    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
}
```

Update Operations

```
jisl> db.inventory.updateMany({qty: {$lt:15}},{$set:{qty:20}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
jisl> db.inventory.find({ qty: 20 })
[
  {
    _id: ObjectId('687e0ee45a68bf389feec4af'),
    item: 'stickerbook',
    qty: 20,
    tags: [ 'colorful', 'fun' ],
    size: { h: 20, w: 20, uom: 'cm' }
  }
]
```

Delete Operation

```
jisl> db.inventory.deleteMany({item:"stickerbook"})
{ acknowledged: true, deletedCount: 1 }
jisl>
```