

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
Факультет инфокоммуникационных технологий

ОТЧЕТ
О ЛАБОРАТОРНОЙ РАБОТЕ № 5.1
по теме: Работа с БД в СУБД MongoDB
по дисциплине: Проектирование и реализация баз данных

Специальность:
09.03.03 Мобильные и сетевые технологии

Проверил:
Говорова М.М. _____
Дата: «__» _____ 2022 г.
Оценка _____

Выполнил:
студент группы К3240
Балдина Д.Д.

Санкт-Петербург 2022 г.

ЦЕЛЬ РАБОТЫ

Овладеть практическими навыками работы с CRUD-операциями, с вложенными объектами в коллекции базы данных MongoDB, агрегации и изменения данных, со ссылками и индексами в базе данных MongoDB.

ПРАКТИЧЕСКОЕ ЗАДАНИЕ:

1. Установите MongoDB для обеих типов систем (32/64 бита).
2. Проверьте работоспособность системы запуском клиента mongo.

```
C:\Program Files\MongoDB\Server\5.0\bin\mongo.exe
MongoDB shell version v5.0.8
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongod
Implicit session: session { "id" : UUID("1bc895b3-e03e-4a50-9275-35e94f46e6af") }
MongoDB server version: 5.0.8
=====
```

3. Выполните методы:

a. db.help()

```
> db.help()
DB methods:
  db.adminCommand(nameOrDocument) - switches to 'admin' db, and runs command [just calls db.runCommand(...)]
  db.aggregate([pipeline], {options}) - performs a collectionless aggregation on this database; returns a cursor
  db.auth(username, password)
  db.commandHelp(name) returns the help for the command
  db.createUser(userDocument)
  db.createView(name, viewOn, [{operator: {...}}, ...], {viewOptions})
  db.currentOp() displays currently executing operations in the db
  db.dropDatabase(writeConcern)
  db.dropUser(username)
  db.eval() - deprecated
  db.fsyncLock() flush data to disk and lock server for backups
  db.fsyncUnlock() unlocks server following a db.fsyncLock()
  db.getCollection(cname) same as db['cname'] or db.cname
  db.getCollectionInfos([filter]) - returns a list that contains the names and options of the db's collections
  db.getCollectionNames()
  db.getLastErrorMessage() - just returns the err msg string
  db.getLastErrorMessageObj() - return full status object
  db.getLogComponents()
  db.getMongo() get the server connection object
  db.getMongo().setSecondaryOk() allow queries on a replication secondary server
  db.getName()
  db.getProfilingLevel() - deprecated
  db.getProfilingStatus() - returns if profiling is on and slow threshold
  db.getReplicationInfo()
  db.getSiblingDB(name) get the db at the same server as this one
  db.getWriteConcern() - returns the write concern used for any operations on this db, inherited from server obj
```

b. db.help

```

> db.help
function() {
  print("DB methods:");
  print(
    "\tdb.adminCommand(nameOrDocument) - switches to 'admin' db, and runs command [just calls db.runCommand(...)]"

    print(
      "\tdb.aggregate([pipeline], {options}) - performs a collectionless aggregation on this database; returns a cur
  r");
  print("\tdb.auth(username, password)");
  print("\tdb.commandHelp(name) returns the help for the command");
  print("\tdb.createUser(userDocument)");
  print("\tdb.createView(name, viewOn, [{operator: {...}}, ...], {viewOptions})");
  print("\tdb.currentOp() displays currently executing operations in the db");
  print("\tdb.dropDatabase(writeConcern)");
  print("\tdb.dropUser(username)");
  print("\tdb.eval() - deprecated");
  print("\tdb.fsyncLock() flush data to disk and lock server for backups");
  print("\tdb.fsyncUnlock() unlocks server following a db.fsyncLock()");
  print("\tdb.getCollection(cname) same as db['cname'] or db.cname");
  print("\tdb.getCollectionInfos([filter]) - returns a list that contains the names and options" +
    " of the db's collections");
  print("\tdb.getCollectionNames()");
  print("\tdb.getLastError() - just returns the err msg string");
  print("\tdb.getLastErrorObj() - return full status object");
  print("\tdb.getLogComponents()");
  print("\tdb.getMongo() get the server connection object");
}

```

c. db.stats()

```

> db.stats()
{
  "db" : "test",
  "collections" : 0,
  "views" : 0,
  "objects" : 0,
  "avgObjSize" : 0,
  "dataSize" : 0,
  "storageSize" : 0,
  "totalSize" : 0,
  "indexes" : 0,
  "indexSize" : 0,
  "scaleFactor" : 1,
  "fileSize" : 0,
  "fsUsedSize" : 0,
  "fsTotalSize" : 0,
  "ok" : 1
}

```

4. Создайте БД learn.

```

> use learn
switched to db learn
> db
learn

```

5. Получите список доступных БД.

```

> show dbs
admin    0.000GB
config  0.000GB
local    0.000GB

```

Чтобы было видно нашу базу данных необходимо внести в нее какие-то данные

После вставки записи:

```
> show dbs
admin    0.000GB
config  0.000GB
learn    0.000GB
local    0.000GB
```

6. Создайте коллекцию unicorns, вставив в нее документ {name: 'Aurora', gender: 'f', weight: 450}.

```
> db.learn.insert({name:"Aurora", gender: "f", weight: 450})
WriteResult({ "nInserted" : 1 })
> show collections
learn
> db.learn.renameCollection("unicorns")
{ "ok" : 1 }
```

7. Просмотрите список текущих коллекций.

```
> show collections
unicorns
```

8. Переименуйте коллекцию unicorns (см. пункт 6).

9. Просмотрите статистику коллекции.

```
> db.unicorns.stats()
{
  "ns" : "learn.unicorns",
  "size" : 69,
  "count" : 1,
  "avgObjSize" : 69,
  "storageSize" : 20480,
  "freeStorageSize" : 0,
  "capped" : false,
  "wiredTiger" : {
    "metadata" : {
      "formatVersion" : 1
    },
    "nindexes" : 1,
    "indexBuilds" : [ ],
    "totalIndexSize" : 20480,
    "totalSize" : 40960,
    "indexSizes" : {
      "_id_" : 20480
    },
    "scaleFactor" : 1,
    "ok" : 1
  }
}
```

10. Удалите коллекцию.

```
> db.unicorns.drop()
true
> show collections
>
```

11. Удалите БД learn.

```
> db.dropDatabase()
{ "ok" : 1 }
> show dbs
admin    0.000GB
config  0.000GB
local    0.000GB
```

ВЫВОД

Произошло первое знакомство с документно-ориентированной СУБД Mongo DB. Были рассмотрены базовые команды на создание базы данных, коллекции и их удаления, а также проверки текущего состояния, сбора статистики и списка команд