

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

Выполнил:
студент 2 курса ИКТ группа
К3241
Сизей Омар

Проверила:
Говорова Марина Михайловна

Санкт-Петербург 2021 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.

```
MongoDB shell version v5.0.9
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("994c9b38-cb38-433b-986b-9efde1ec16b3") }
MongoDB server version: 5.0.9
=====
```

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.getLastError() - just returns the err msg string
  db.getLastErrorObj() - 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 object if set
  db.hostInfo() get details about the server's host
  db.isMaster() check replica primary status
  db.hello() check replica primary status
  db.killOp(opid) kills the current operation in the db
  db.listCommands() lists all the db commands
  db.loadServerScripts() loads all the scripts in db.system.js
  db.logout()
  db.printCollectionStats()
  db.printReplicationInfo()
  db.printShardingStatus()
  db.printSecondaryReplicationInfo()
  db.rotateCertificates(message) - rotates certificates, CRLs, and CA files and logs an optional message
  db.runCommand(cmdObj) run a database command. if cmdObj is a string, turns it into {cmdObj: 1}
  db.serverStatus()
  db.setLogLevel(level,<component>)
  db.setProfilingLevel(level,slowms) 0=off 1=slow 2=all
  db.setVerboseShell(flag) display extra information in shell output
  db.setWriteConcern(<write concern doc>) - sets the write concern for writes to the db
  db.shutdownServer()
  db.stats()
  db.unsetWriteConcern(<write concern doc>) - unsets the write concern for writes to the db
  db.version() current version of the server
  db.watch() - opens a change stream cursor for a database to report on all changes to its non-system collections.
```

b. db.help

5. Получите список доступных БД.

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

После добавления данных:

```
> 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}.

7. Просмотрите список текущих коллекций.

8. Переименуйте коллекцию unicorns.

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

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
    },
    "indexSizes" : {
      "_id_" : 20480
    }
  },
  "indexes" : 1,
  "indexBuilds" : [ ],
  "totalIndexSize" : 20480,
  "totalSize" : 40960,
  "indexSizes" : {
    "_id_" : 20480
  },
  "scaleFactor" : 1,
  "ok" : 1
}
```

10. Удалите коллекцию

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

11. Удалите базу данных learn

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

ВЫВОД

В рамках моего базового знакомства с MongoDB базовые команды были реализованы на создание базы данных, коллекций и их удаление, а также проверка текущего состояния, сбор статистики и списка команд.