## Учреждение образования БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра «Проектирования информационно-компьютерных систем»

### ОТЧЕТ

по лабораторной работе №5

по дисциплине «Системы и методы управления базами данных» На тему: «Основы NOSQL СУБД»

Выполнил: студент гр.914302

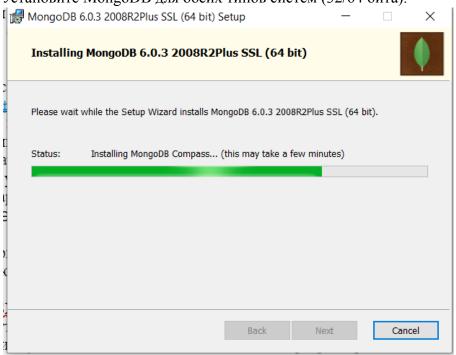
Шпаковский М.Г.

Проверила: Лукашевич А. Э.

#### Минск 2022

#### Выполнение работы:

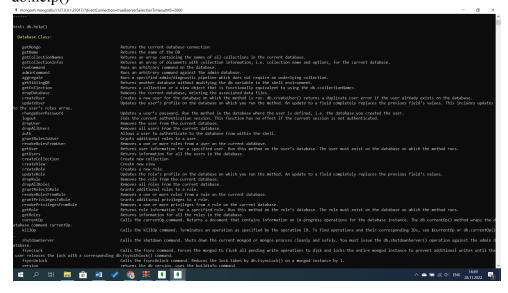
1. Установите MongoDB для обеих типов систем (32/64 бита).



2. Проверьте работоспособность системы запуском клиента mongo.

```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
                                                                                                                                Please enter a MongoDB connection string (Default: mongodb://localhost/):
Current Mongosh Log ID: 6384bbc32dd3fdeb25bb3373
Connecting to:
Jsing MongoDB:
                          6.0.3
Using Mongosh:
                           1.6.0
For mongosh info see: https://docs.mongodb.com/mongodb-shell/
To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.co
m/legal/privacy-policy).
ou can opt-out by running the disableTelemetry() command.
  The server generated these startup warnings when booting 2022-11-28T16:42:54.938+03:00: Access control is not enabled for the database. Read and write access to data and conf
guration is unrestricted
  Enable MongoDB's free cloud-based monitoring service, which will then receive and display
  metrics about your deployment (disk utilization, CPU, operation statistics, etc).
  The monitoring data will be available on a MongoDB website with a unique URL accessible to you
  and anyone you share the URL with. MongoDB may use this information to make product improvements and to suggest MongoDB products and deployment options to you.
```

# 3. Выполните методы: db.help()



## db.help

```
While affects described to the performance of the described for the performance of the pe
```

db.stats()

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

- 4. Создайте БД learn.
- 5. Получите список доступных БД.

```
test> use learn
switched to db learn
learn> show dbs
admin 40.00 KiB
config 92.00 KiB
local 72.00 KiB
```

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

```
{
    acknowledged: true,
    insertedIds: { '0': ObjectId("6384bf53bb18bba5d579cb64") }
}
learn>
```

7. Просмотрите список текущих коллекций.

```
learn> show collections
unicorns
learn>
```

8. Переименуйте коллекцию unicorns.

```
learn> db.unicorn.renameCollection("uncorn")
{ ok: 1 }
```

9. Просмотрите статистику коллекции.

10. Удалите коллекцию.

```
learn> db.uncorn.drop()
true
```

11. Удалите БД learn.

```
learn> db.dropDatabase()
{ ok: 1, dropped: 'learn' }
```

#### Практическое задание 1:

1. Создайте базу данных learn.

```
test> use learn
switched to db learn
learn> _
```

2. Заполните коллекцию единорогов unicorns:

```
learns db.unicorns.insert((name: 'Horny', dob: new Date(1992,2,13,7,47), loves: ['carrot', 'papaya'], weight: 600, gendergender: 'm', vampires: 63});
DeprecationNarning: Collection.insert() is deprecated. Use insertDne, insertDne,
```

3. Используя второй способ, вставьте в коллекцию единорогов документ: {name: 'Dunx', dob: new Date(1976, 6, 18, 18, 18), loves: ['grape', 'watermelon'], weight: 704, gender: 'm', vampires: 165}

```
learn> document = ({name: 'Dunx', dob: new Date(1976, 6, 18, 18, 18), loves: ['grape', 'watermelon'], we
weight: 704, gender: 'm', vampires: 165})
{
    name: 'Dunx',
    dob: ISODate("1976-07-18T15:18:00.000Z"),
    loves: [ 'grape', 'watermelon' ],
    weight: 704,
    gender: 'm',
    vampires: 165
}
learn> db.unicorns.insert(document)
{
    acknowledged: true,
    insertedIds: { '0': ObjectId("6385bb48a75e76e0e10c66dc") }
}
```

4. Проверьте содержимое коллекции с помощью метода find.

```
learn> db.unicorns.find()
    _id: ObjectId("6385ba85a75e76e0e10c66d1"),
    dob: ISODate("1992-03-13T04:47:00.000Z"),
    loves: [ 'carrot', 'papaya' ],
weight: 600,
    gender: 'm',
    vampires: 63
    _id: ObjectId("6385ba85a75e76e0e10c66d2"),
    dob: ISODate("1991-01-24T10:00:00.000Z"),
    loves: [ 'carrot', 'grape' ],
weight: 450,
    gender: 'f'
    vampires: 43
    _id: ObjectId("6385ba85a75e76e0e10c66d3"),
    dob: ISODate("1973-02-09T19:10:00.000Z"),
   loves: [ 'energon', 'redbull' ],
weight: 984,
gender: 'm',
    vampires: 182
    _id: ObjectId("6385ba85a75e76e0e10c66d4"),
    dob: ISODate("1979-08-18T15:44:00.000Z"),
   loves: [ 'apple' ],
weight: 575,
gender: 'm',
    vampires: 99
    _id: ObjectId("6385ba85a75e76e0e10c66d5"),
    dob: ISODate("1985-07-03T22:01:00.000Z"),
    loves: [ 'apple', 'carrot', 'chocolate' weight: 550,
```

#### Практическое задание 2:

1. Сформируйте запросы для вывода списков самцов и самок единорогов. Ограничьте список самок первыми тремя особями. Отсортируйте списки по имени.

2. Найдите всех самок, которые любят carrot. Ограничьте этот список первой особью с помощью функций findOne и limit.

```
learn> db.unicorns.find({gender:"f"}, {loves: "carrot"}).limit(1)
[ { _id: ObjectId("6385ba85a75e76e0e10c66d2"), loves: 'carrot' } ]

learn> db.unicorns.findOne({gender:"f"}, {loves: "carrot"})
{ _id: ObjectId("6385ba85a75e76e0e10c66d2"), loves: 'carrot' }
```

#### Практическое задание 3:

Модифицируйте запрос для вывода списков самцов единорогов, исключив из результата информацию о дате рождения и поле.

#### Практическое задание 4:

Вывести список единорогов в обратном порядке добавления.

```
learn> db.unicorns.find().sort({$natural: -1})
    _id: ObjectId("6385bb48a75e76e0e10c66dc"),
    dob: ISODate("1976-07-18T15:18:00.000Z"),
   loves: [ 'grape', 'watermelon' ], weight: 704, gender: 'm',
    vampires: 165
    _id: ObjectId("6385ba86a75e76e0e10c66db"),
    dob: ISODate("1999-12-20T13:15:00.000Z"),
    loves: [ 'grape', 'carrot' ],
weight: 540,
    gender: 'f'
    _id: ObjectId("6385ba85a75e76e0e10c66da"),
    dob: ISODate("1997-03-01T02:03:00.000Z"),
   loves: [ 'apple', 'watermelon'], weight: 650, gender: 'm',
    vampires: 54
    _id: ObjectId("6385ba85a75e76e0e10c66d9"),
    dob: ISODate("2001-10-08T10:53:00.000Z"),
   loves: [ 'apple', 'watermelon' ],
weight: 601,
gender: 'f',
    vampires: 33
    _id: ObjectId("6385ba85a75e76e0e10c66d8"),
    dob: ISODate("2005-05-02T20:57:00.000Z"),
```

#### Практическое задание 5а:

Вывести список единорогов с названия первого любимого фрукта, исключив идентификатор.

#### Практическое задание 56:

Вывести список самок единорогов весом от полутонны до 700 кг, исключив вывод идентификатора.

```
learn> db.unicorns.find({weight:{$gt: 500, $lt:700}, gender:"f"}, {_id: 0})
[
{
    name: 'Solnara',
    dob: ISODate("1985-07-03T22:01:00.000Z"),
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
},
{
    name: 'Leia',
    dob: ISODate("2001-10-08T10:53:00.000Z"),
    loves: [ 'apple', 'watermelon' ],
    weight: 601,
    gender: 'f',
    vampires: 33
},
{
    name: 'Nimue',
    dob: ISODate("1999-12-20T13:15:00.000Z"),
    loves: [ 'grape', 'carrot' ],
    weight: 540,
    gender: 'f'
}
```

#### Практическое задание 6:

Вывести список самцов единорогов весом от полутонны и предпочитающих grape и lemon, исключив вывод идентификатора.

```
learn> db.unicorns.find({weight:{$gt: 500}, gender:"m", $or:[{loves:"lemon", loves:'grape'}]}, {_id: 0})
[
{
    name: 'Kenny',
    dob: ISODate("1997-07-01T06:42:00.000Z"),
    loves: [ 'grape', 'lemon' ],
    weight: 690,
    gender: 'm',
    vampires: 39
},
{
    name: 'Dunx',
    dob: ISODate("1976-07-18T15:18:00.000Z"),
    loves: [ 'grape', 'watermelon' ],
    weight: 704,
    gender: 'm',
    vampires: 165
}
```

#### Практическое задание 7:

Найти всех единорогов, не имеющих ключ vampires.

#### Вывод:

В ходе данной лабораторной работы были изучены основы nosql субд mongodb, были изучены способы взаимодействия с базой данных, проведена установка mongodb, изучены CRUD-операции в данной СУБД, вставка данных и их выборка.