

Отчет по лабораторной работе 8, вариант 7.

Выполнил: Золотухин Андрей Александрович КС-36

Преподаватель: Семёнов Геннадий Николаевич

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.

Таблица

Общезнание:

Дата предоставления	Группа	Студент	Сосед по комнате	Удалённость собственного жилья от института, км.	№ комнаты	Стипендия, руб.
3.08.05	КС-10	Васильев	Гришин	200	14	2000
15.08.05	КС-14	Петрова	Сидорова	435	37	2570
2.08.05	КС-14	Сидоров	Иванов	112	25	2000
2.08.05	КС-10	Иванов	Сидоров	240	25	2240
14.08.05	КС-10	Сидорова	Петрова	1200	37	4500
4.08.05	КС-14	Гришин	Васильев	780	14	4000

Выполнение задания

1) Сформируйте запросы для вывода списков каких-либо объектов коллекции вашей БД. Ограничьте список:

Отсортируйте списки по имени.

Формирование списка студентов, согласно сортировке по имени студента (student_name) по возрастанию алфавита:

```
db.dormitory.find({}, { student_name: 1, _id: 0 }).sort({ student_name: 1 })
```

```
Zolotukhin_ks36> db.dormitory.find({}, { student_name: 1, _id: 0 }).sort({ student_name: 1 })
[
  { student_name: 'Васильев' },
  { student_name: 'Гришин' },
  { student_name: 'Иванов' },
  { student_name: 'Петрова' },
  { student_name: 'Сидоров' },
  { student_name: 'Сидорова' }
]
```

2) Ограничьте этот список с помощью функций findOne и limit:

Формирование имени одного студента:

```
db.dormitory.findOne({}, { student_name: 1, _id: 0 })
```

```
Zolotukhin_ks36> db.dormitory.findOne({}, { student_name: 1, _id: 0 })
{ student_name: 'Васильев' }
```

Формирование списка из 3 студентов, согласно сортировке по имени студента по возрастанию алфавита:

```
db.dormitory.find({}, {student_name: 1, _id: 0}).sort({
student_name: 1}).limit(3)
```

```
Zolotukhin_ks36> db.dormitory.find({}, { student_name: 1, _id: 0}).sort({ student_name: 1 }).limit(3)
[
  { student_name: 'Васильев' },
  { student_name: 'Гришин' },
  { student_name: 'Иванов' }
]
```

3) Модифицируйте запрос 1 для вывода списка объектов, исключив из результата какую-либо информацию и поле:

Формирование списка информации о студентах в алфавитном порядке их имен, исключив дальность от общежития, их стипендию и id:

```
db.dormitory.find({}, { scholarship: 0, distance: 0, _id:
0}).sort({ student_name: 1})
```

```
Zolotukhin_ks36> db.dormitory.find({}, { scholarship: 0, distance: 0, _id: 0}).sort({ student_name: 1 })
[
  {
    accommodation_date: '3.08.05',
    group_name: 'КС-10',
    student_name: 'Васильев',
    neighbour_name: 'Гришин',
    room_number: 14
  },
  {
    accommodation_date: '4.08.05',
    group_name: 'КС-14',
    student_name: 'Гришин',
    neighbour_name: 'Васильев',
    room_number: 14
  },
  {
    accommodation_date: '2.08.05',
    group_name: 'КС-10',
    student_name: 'Иванов',
    neighbour_name: 'Сидоров',
    room_number: 25
  },
  {
    accommodation_date: '15.08.05',
    group_name: 'КС-14',
    student_name: 'Петрова',
    neighbour_name: 'Сидорова',
    room_number: 37
  },
  {
    accommodation_date: '2.08.05',
    group_name: 'КС-14',
    student_name: 'Сидоров',
    neighbour_name: 'Иванов',
    room_number: 25
  },
  {
    accommodation_date: '14.08.05',
    group_name: 'КС-10',
    student_name: 'Сидорова',
    neighbour_name: 'Петрова',
    room_number: 37
  }
]
```

4) Вывести список объектов в коллекции в обратном порядке добавления.

Формирование списка последних 4 добавленных объектов в обратном порядке

```
db.dormitory.find({}).sort({ $natural: -1 }).limit(4)
```

```
Zolotukhin_ks36> db.dormitory.find({}).sort({ $natural: -1 }).limit(4)
[
  {
    _id: ObjectId('675fee4fda8a690e4da5c0be'),
    accommodation_date: '4.08.05',
    group_name: 'КС-14',
    student_name: 'Гришин',
    neighbour_name: 'Васильев',
    distance: 780,
    room_number: 14,
    scholarship: 4000
  },
  {
    _id: ObjectId('675fee2eda8a690e4da5c0bd'),
    accommodation_date: '14.08.05',
    group_name: 'КС-10',
    student_name: 'Сидорова',
    neighbour_name: 'Петрова',
    distance: 1200,
    room_number: 37,
    scholarship: 4500
  },
  {
    _id: ObjectId('675fee09da8a690e4da5c0bc'),
    accommodation_date: '2.08.05',
    group_name: 'КС-10',
    student_name: 'Иванов',
    neighbour_name: 'Сидоров',
    distance: 240,
    room_number: 25,
    scholarship: 2240
  },
  {
    _id: ObjectId('675fede9da8a690e4da5c0bb'),
    accommodation_date: '2.08.05',
    group_name: 'КС-14',
    student_name: 'Сидоров',
    neighbour_name: 'Иванов',
    distance: 112,
    room_number: 25,
    scholarship: 2000
  }
]
```

5) Вывести список объектов коллекции вашей БД с использованием логических операторов, исключив вывод идентификатора.

Формирование списка объектов, студенты которых не учатся в группе КС-14 и заселились не 02.08.2005:

```
db.dormitory.find({ $nor: [{group_name: "КС-14"},  
{accommodation_date: "2.08.05"}] }, { _id: 0 })
```

```
Zolotukhin_ks36> db.dormitory.find({ $nor: [{group_name: "КС-14"}, {accommodation_date: "2.08.05"}] }, { _id: 0 })  
[  
  {  
    accommodation_date: '3.08.05',  
    group_name: 'КС-10',  
    student_name: 'Васильев',  
    neighbour_name: 'Гришин',  
    distance: 200,  
    room_number: 14,  
    scholarship: 2000  
  },  
  {  
    accommodation_date: '14.08.05',  
    group_name: 'КС-10',  
    student_name: 'Сидорова',  
    neighbour_name: 'Петрова',  
    distance: 1200,  
    room_number: 37,  
    scholarship: 4500  
  }  
]
```

6) Вывести упорядоченный список имен объектов с информацией из коллекции вашей БД.

Формирование упорядоченного списка студентов в алфавитном порядке и их стипендий

```
db.dormitory.find({}, {student_name: 1, scholarship: 1, _id:  
0}).sort({ student_name: 1 })
```

```
Zolotukhin_ks36> db.dormitory.find({}, {student_name: 1, scholarship: 1, _id: 0}).sort({ student_name: 1 })  
[  
  { student_name: 'Васильев', scholarship: 2000 },  
  { student_name: 'Гришин', scholarship: 4000 },  
  { student_name: 'Иванов', scholarship: 2240 },  
  { student_name: 'Петрова', scholarship: 2570 },  
  { student_name: 'Сидоров', scholarship: 2000 },  
  { student_name: 'Сидорова', scholarship: 4500 }  
]
```

7) Сформировать курсор для вывода списка каких-либо объектов вашего варианта и вывести этот список.

Формирование списка студентов и их соседей по комнате

```
let cursor = db.dormitory.find({}, { student_name: 1,  
neighbour_name: 1, _id: 0 })  
cursor.toArray()
```

```
Zolotukhin_ks36> let cursor = db.dormitory.find({}, { student_name: 1, neighbour_name: 1, _id: 0 })
Zolotukhin_ks36> cursor.toArray()
[
  { student_name: 'Васильев', neighbour_name: 'Гришин' },
  { student_name: 'Петрова', neighbour_name: 'Сидорова' },
  { student_name: 'Сидоров', neighbour_name: 'Иванов' },
  { student_name: 'Иванов', neighbour_name: 'Сидоров' },
  { student_name: 'Сидорова', neighbour_name: 'Петрова' },
  { student_name: 'Гришин', neighbour_name: 'Васильев' }
]
```

8) Вывести количество объектов вашей БД при определенном условии.

Формирование количества студентов из группы КС-10

```
db.dormitory.count({ group_name: "КС-10" })
```

```
Zolotukhin_ks36> db.dormitory.count({ group_name: "КС-10" })
DeprecationWarning: Collection.count() is deprecated. Use countDocuments or estimatedDocumentCount.
3
```

9) Сформировать запрос на подсчет количества документов с разными именами name для коллекции вашей БД (Использование метода aggregate).

Формирование списка количества студентов в каждой из групп

```
db.dormitory.aggregate([ { "$group": { "_id": "$group_name",
"count": { "$sum": 1 } } } ] )
```

```
Zolotukhin_ks36> db.dormitory.aggregate([ { "$group": { "_id": "$group_name", "count": { "$sum": 1 } } } ] )
[ { _id: 'КС-10', count: 3 }, { _id: 'КС-14', count: 3 } ]
```

10) Создайте новую коллекцию (например, person) для часто встречающихся названий объектов в исходной коллекции, указав в качестве идентификатора кратко название объекта, далее включив полное название и описание.

Формирование новой коллекции students, состоящей из имен студентов:

```
db.students.insertMany([ { _id: "Вас", fullName: "Васильев",
description: "Студент группы КС-10" }, { _id: "Гриш",
fullName: "Гришин", description: "Студент группы КС-10" }, {
_id: "Петр", fullName: "Петрова", description: "Студентка
группы КС-14" }, { _id: "Сид", fullName: "Сидорова",
description: "Студентка группы КС-14" }, { _id: "Сид",
fullName: "Сидоров", description: "Студент группы КС-14" }, {
_id: "Ив", fullName: "Иванов", description: "Студент группы
КС-10" } ] )
```

```
Zolotukhin_ks36> db.students.insertMany([ { _id: "Вас", fullName: "Васильев", description: "Студент группы КС-10" }, { _id: "Гриш", fullName: "Гришин", description: "Студент группы КС-10" }, { _id: "Петр", fullName: "Петрова", description: "Студентка группы КС-14" }, { _id: "Сид", fullName: "Сидорова", description: "Студентка группы КС-14" }, { _id: "Сид", fullName: "Сидоров", description: "Студент группы КС-14" }, { _id: "Ив", fullName: "Иванов", description: "Студент группы КС-10" } ] );
{
  acknowledged: true,
  insertedIds: {
    '0': 'Вас',
    '1': 'Гриш',
    '2': 'Петр',
    '3': 'Сид',
    '4': 'Сид',
    '5': 'Ив'
  }
}
```

- 11) Включите в документы исходной коллекции ссылку на документы новой коллекции(например, person), используя второй способ автоматического связывания.**

Включение ссылки на id Пет и Сиды коллекции students в документы исходной коллекции соответственно для объекта студентки Петровой

```
db.dormitory.update( { student_name: "Петрова" }, { $set: {
  student_name: { $ref: "students", $id: "Петр" },
  neighbour_name: { $ref: "students", $id: "Сид" } } } );
```

```
Zolotukhin_ks36> db.dormitory.update( { student_name: "Петрова" }, { $set: { student_name: { $ref: "students", $id: "Петр" }, neighbour_name: { $ref: "students", $id: "Сид" } } } );
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

- 12) Выведете все данные исходной и новой коллекции.**

Формирование списка всех объектов

```
db.dormitory.find({})
```

```
Zolotukhin_ks36> db.dormitory.find({})
[
  {
    _id: ObjectId('6760387076b7903366a5c0dd'),
    accommodation_date: '3.08.05',
    group_name: 'КС-10',
    student_name: 'Васильев',
    neighbour_name: 'Гришин',
    distance: 200,
    room_number: 14,
    scholarship: 2000
  },
  {
    _id: ObjectId('6760387076b7903366a5c0de'),
    accommodation_date: '15.08.05',
    group_name: 'КС-14',
    student_name: DBRef('students', 'Петр'),
    neighbour_name: DBRef('students', 'Сид'),
    distance: 435,
    room_number: 37,
    scholarship: 2570
  },
  {
    _id: ObjectId('6760387076b7903366a5c0df'),
    accommodation_date: '2.08.05',
    group_name: 'КС-14',
    student_name: 'Сидоров',
    neighbour_name: 'Иванов',
    distance: 112,
    room_number: 25,
    scholarship: 2000
  },
  {
    _id: ObjectId('6760387076b7903366a5c0e0'),
    accommodation_date: '2.08.05',
    group_name: 'КС-10',
    student_name: 'Иванов',
    neighbour_name: 'Сидоров',
    distance: 240,
    room_number: 25,
    scholarship: 2240
  },
  {
    _id: ObjectId('6760387076b7903366a5c0e1'),
    accommodation_date: '14.08.05',
    group_name: 'КС-10',
    student_name: 'Сидорова',
    neighbour_name: 'Петрова',
    distance: 1200,
    room_number: 37,
    scholarship: 4500
  },
  {
    _id: ObjectId('6760387076b7903366a5c0e2'),
    accommodation_date: '4.08.05',
    group_name: 'КС-14',
    student_name: 'Гришин',
    neighbour_name: 'Васильев',
    distance: 780,
    room_number: 14,
    scholarship: 4000
  }
]
```

Формирование списка сокращенных имен студентов

`db.student.find()`

```
Zolotukhin_ks36> db.students.find()
[
  {
    _id: 'Вас',
    fullName: 'Васильев',
    description: 'Студент группы КС-10'
  },
  {
    _id: 'Гриш',
    fullName: 'Гришин',
    description: 'Студент группы КС-10'
  },
  {
    _id: 'Петр',
    fullName: 'Петрова',
    description: 'Студентка группы КС-14'
  },
  {
    _id: 'Сида',
    fullName: 'Сидорова',
    description: 'Студентка группы КС-14'
  },
  {
    _id: 'Сид',
    fullName: 'Сидоров',
    description: 'Студент группы КС-14'
  },
  {
    _id: 'Ив',
    fullName: 'Иванов',
    description: 'Студент группы КС-10'
  }
]
```

13) Сделайте экспорт коллекций вашей БД в файлы *.json.

*Экспорт коллекций в файлы *.json:*


```
mongoexport --db Zolotukhin_ks36 --collection dormitory --out  
dormitory.json  
mongoexport --db Zolotukhin_ks36 --collection students --out  
students.json
```

```
corgi@LAPTOP-AS2LVH3V:~/git_repositories/DataManagement_labs/lab8$ mongoexport --db Zolotukhin_ks36 --collection dormitory --out d  
ormitory.json  
2024-12-16T17:32:04.380+0300    connected to: mongodb://localhost/  
2024-12-16T17:32:04.397+0300    exported 6 records  
corgi@LAPTOP-AS2LVH3V:~/git_repositories/DataManagement_labs/lab8$ mongoexport --db Zolotukhin_ks36 --collection students --out st  
udents.json  
2024-12-16T17:32:21.882+0300    connected to: mongodb://localhost/  
2024-12-16T17:32:21.890+0300    exported 6 records
```

Выводы

В ходе лабораторной работы я овладел 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.