

# Методы сбора, хранения, обработки и анализа данных

Лекция 14

NoSQL СУБД MongoDB

# NoSQL

- NoSQL – набор различных решений для построения гибких БД
- Оптимизированы для работы с большим объемом данных
- За счет отсутствия контроля целостности

# NoSQL – преимущества

- Гибкость
  - Быстрее разработка
  - Возможна поэтапная реализация
  - Частично структурированные или неструктурированные данных
- Масштабируемость
  - Использование кластеров
- Высокая производительность
  - Оптимизированы для конкретных шаблонов
- Широкие функциональные возможности

# Виды NoSQL баз данных

- Ключ-значение
- Документные базы данных
- Графовые базы данных
- InMemory базы данных
- Поисковые базы данных

# Базы данных ключ-значение

- Данные представляют собой набор ключ-значение
- Преимущества:
  - Высокая секционированность
  - Горизонтальное масштабирование
- Обычно игровые, рекламные приложения и приложения IoT
- Пример - Amazon DynamoDB

# Документные базы данных

- Данные представлены как документ в формате JSON
- Структурированный и/или иерархический характер документов
- Каталоги, пользовательские профили, системы управления контентом
- Примеры:
  - Amazon DocumentDB
  - MongoDB

# Графовые базы данных

- Данные представляют собой многоуровневый граф
- Социальные сети, сервисы рекомендаций и графы знаний
- Примеры:
  - Amazon Neptune
  - Neo4j

# InMemory базы данных

- Данные :
  - Таблицы лидеров в играх, хранение сессий и аналитика в реальном времени
- Преимущества:
  - отклик в пределах нескольких микросекунд при росте трафика
- Примеры:
  - Amazon MemoryDB для Redis
  - Amazon ElastiCache
  - Amazon DynamoDB Accelerator



# Поисковые базы данных

- Поиск в журналах приложений
- Пример:
  - Amazon OpenSearch

# MongoDB

- Документо-ориентированная система управления базами данных
- Используемая версия 6.0 (2022 )

# Особенности СУБД MongoDB

- База данных представляет собой набор коллекций
- Коллекции содержат документы
- Документы могут иметь разную структуру
- Для каждого документа имеется уникальный идентификатор, который называется `_id`
- Если явно не указать, то автоматически сгенерируется

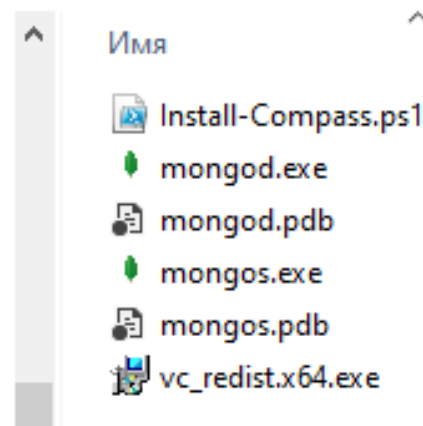
# Установка сервера СУБД MongoDB

- Кроссплатформенная:
  - Windows, Linux, MacOS, Solaris
- Бесплатный Community и платный Enterprise
- <https://www.mongodb.com/try/download/community>
- Если уже была установлена более ранняя версия, то ее необходимо удалить
- Распаковать в C:\mongodb

# Содержимое установленного сервера

- mongod: сервер
  - Обработка запросов
- mongos: служба маршрутизации
  - Определение местоположения данных в кластере

Локальный диск (C:) > mongodb > bin

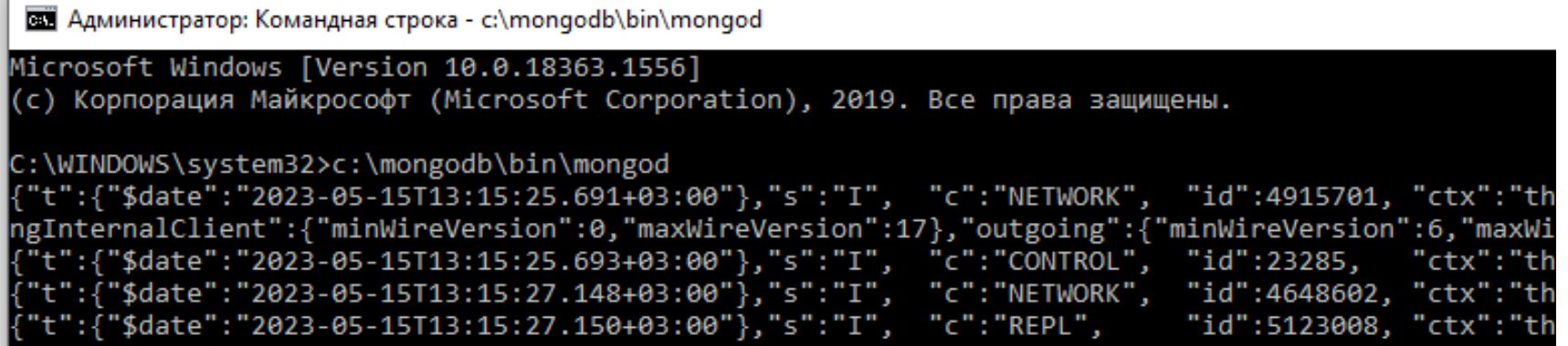


# Каталог для базы данных

- По умолчанию C:\data\db
- Если другой путь – то его передать при запуске во флаге –dbpath

# Запуск сервера

- В командной строке  
c:\mongodb\bin\mongod



Администратор: Командная строка - c:\mongodb\bin\mongod

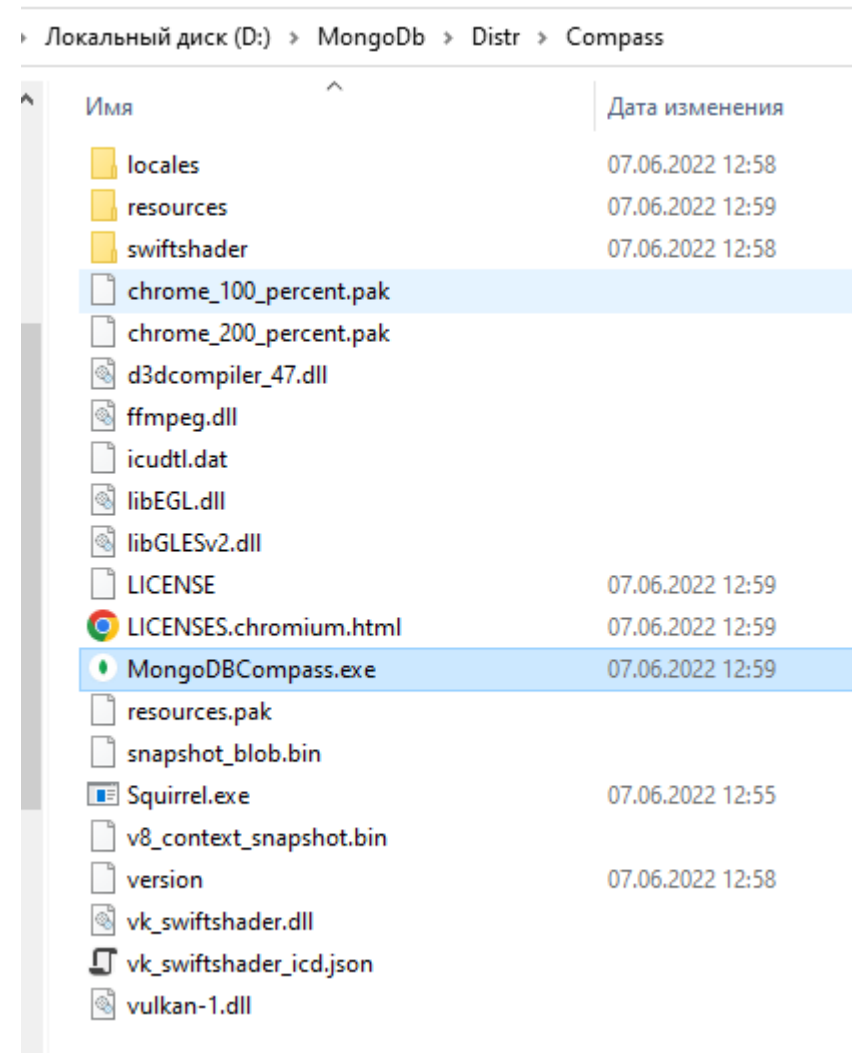
Microsoft Windows [Version 10.0.18363.1556]  
(c) Корпорация Майкрософт (Microsoft Corporation), 2019. Все права защищены.

C:\WINDOWS\system32>c:\mongodb\bin\mongod

{ "t": { "\$date": "2023-05-15T13:15:25.691+03:00" }, "s": "I", "c": "NETWORK", "id": 4915701, "ctx": "th  
ngInternalClient": { "minWireVersion": 0, "maxWireVersion": 17 }, "outgoing": { "minWireVersion": 6, "maxWi  
{ "t": { "\$date": "2023-05-15T13:15:25.693+03:00" }, "s": "I", "c": "CONTROL", "id": 23285, "ctx": "th  
{ "t": { "\$date": "2023-05-15T13:15:27.148+03:00" }, "s": "I", "c": "NETWORK", "id": 4648602, "ctx": "th  
{ "t": { "\$date": "2023-05-15T13:15:27.150+03:00" }, "s": "I", "c": "REPL", "id": 5123008, "ctx": "th

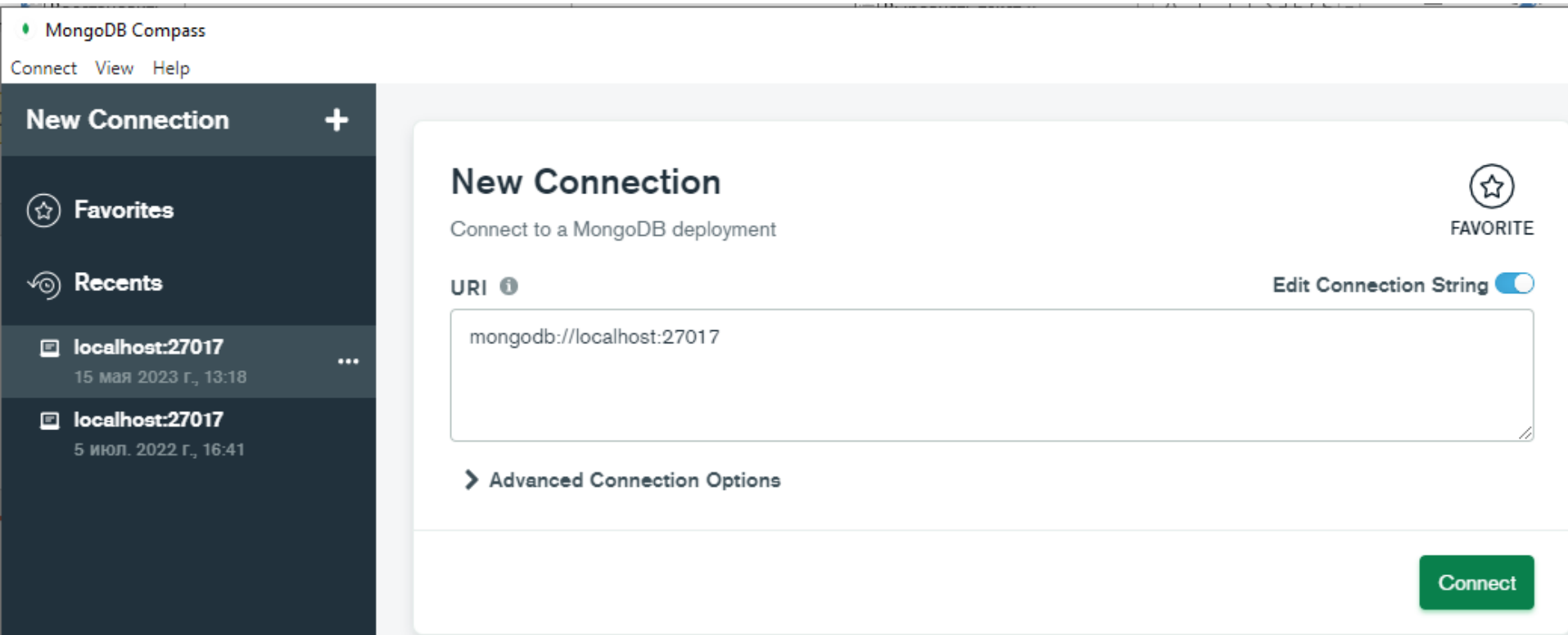
# Графический клиент Compass

- <https://www.mongodb.com/try/download/compass>
- Загрузить архив zip
- Распаковать в нужной папке





# Подключение Compass



# Подключение Compass

MongoDB Compass - localhost:27017

Connect View Help

localhost:27017

3 DBS 3 COLLECTIONS

☆ FAVORITE

HOST  
localhost:27017

CLUSTER  
Standalone

EDITION  
MongoDB 6.0.5 Community

{ } My Queries

⌵ Databases

🔍 Filter your data

- ▶ admin
- ▶ config
- ▶ local

My Queries Databases Performance

Create database View

<b>admin</b>	<b>Storage size:</b> 20.48 kB	<b>Collections:</b> 1	<b>Indexes:</b> 1
<b>config</b>	<b>Storage size:</b> 20.48 kB	<b>Collections:</b> 1	<b>Indexes:</b> 2
<b>local</b>	<b>Storage size:</b> 36.86 kB	<b>Collections:</b> 1	<b>Indexes:</b> 1

# Создание базы данных



## Create Database

Database Name

Collection Name

➤ **Advanced Collection Options** (e.g. Time-Series, Capped, Clustered collections)

---

# Создание базы данных

The screenshot displays the MongoDB Compass interface. On the left is a dark sidebar with navigation options: '4 DBS' and '2 COLLECTIONS' at the top, followed by a 'FAVORITE' button, 'My Queries', 'Databases', and a search bar 'Filter your data'. Below these are expandable sections for 'admin', 'all\_subjects' (containing a folder 'db\_subjects'), 'config', and 'local'. The main area on the right has a 'Create database' button and a 'View' toggle. It lists four databases with their respective storage sizes, collection counts, and index counts.

Database Name	Storage size	Collections	Indexes
admin	20.48 kB	0	1
all_subjects	4.10 kB	1	1
config	36.86 kB	0	2
local	36.86 kB	1	1

# Создание коллекции

MongoDB Compass - localhost:27017

Connect View Help

The screenshot shows the MongoDB Compass interface. The left sidebar displays a tree view of the database structure. The 'all\_subjects' database is selected, and a red box highlights the '+ Create collection' button. The main panel shows details for the 'admin' and 'all\_subjects' databases.

Database	Storage size	Collections
admin	20.48 kB	0
all_subjects	4.10 kB	1

# Создание коллекции

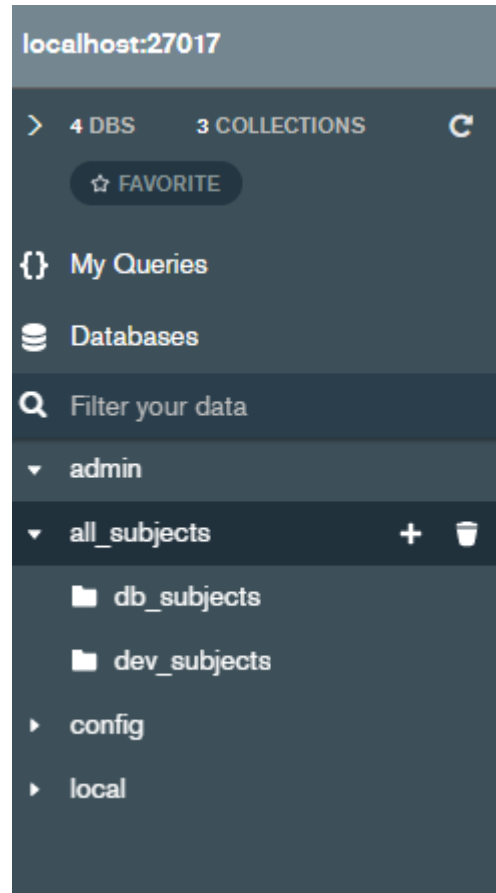
## Create Collection

Collection Name

➤ **Advanced Collection Options** (e.g. Time-Series, Capped, Clustered collections)

Cancel

Create Collection



# Создание записей в коллекции

subject_id	subject_name	semester	number_of_lectures	number_of_labs
1	Manipulating Data in SQL	3	36	36
2	RDBMS Basics	4	38	38
3	Data Analytics	5	30	30
4	Database Administration	6	36	36
5	Data Mining	7	40	52

The screenshot displays the MongoDB Compass web interface. On the left, a sidebar shows the connection 'localhost:27017' with 4 databases and 3 collections. The 'all\_subjects' database is expanded, showing collections 'db\_subjects' and 'dev\_subjects'. The main panel shows the 'all\_subjects.db\_subjects' collection. A breadcrumb trail at the top indicates the path 'Documents' > 'all\_subjects.db\_su...'. Below this, the collection name 'all\_subjects.db\_subjects' is displayed. A tabbed interface allows switching between 'Documents', 'Aggregations', 'Schema', and 'Explain Plan', with 'Documents' currently selected. A filter bar contains a 'FILTER' button and a query '{ field: 'value' }'. Below the filter bar, there is an 'ADD DATA' button with a dropdown arrow, a 'VIEW' button, and three view mode icons: list, JSON, and grid. The main content area below these controls is currently empty.

# Запросы к серверу

```
> show dbs
< admin          40.00 KiB
  all_subjects  112.00 KiB
  config        108.00 KiB
  local         72.00 KiB

> show collections
< db_subjects
  dev_subjects
```

- Встроенный клиент mongosh – написание запросов к серверу
- use **db\_name** – переход в контекст базы данных
- show dbs – сведения о базах данных
- show collections – коллекции в базе



# Статистика по базе данных


```
> db.stats
< [Function: stats] AsyncFunction {
  apiVersions: [ 0, 0 ],
  returnsPromise: true,
  serverVersions: [ '0.0.0', '999.999.999' ],
  topologies: [ 'ReplSet', 'Sharded', 'LoadBalanced', 'Standalone' ],
  returnType: { type: 'unknown', attributes: {} },
  deprecated: false,
  platforms: [ 0, 1, 2 ],
  isDirectShellCommand: false,
  acceptsRawInput: false,
  shellCommandCompleter: undefined,
  help: [Function (anonymous)] Help
}
```

```
> db.dev_subjects.stats
< [Function: stats] AsyncFunction {
  apiVersions: [ 0, 0 ],
  returnsPromise: true,
  serverVersions: [ '0.0.0', '999.999.999' ],
  topologies: [ 'ReplSet', 'Sharded', 'LoadBalanced', 'Standalone' ],
  returnType: { type: 'unknown', attributes: {} },
  deprecated: false,
  platforms: [ 0, 1, 2 ],
  isDirectShellCommand: false,
  acceptsRawInput: false,
  shellCommandCompleter: undefined,
  help: [Function (anonymous)] Help
}
```

# Создание элементов в коллекции

Import To Collection all\_subjects.db\_subjects ×

Select File

 subjects.csv

Select Input File Type

JSON CSV

Options

Select delimiter COMMA ▼

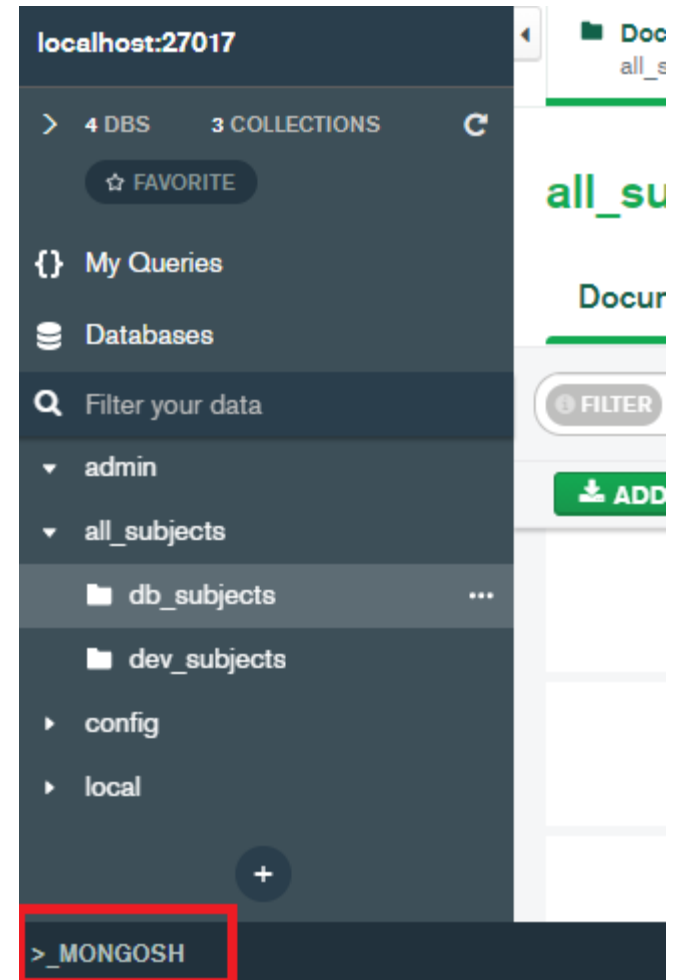
☒ Ignore empty strings

☐ Stop on errors

Specify Fields and Types

	<input checked="" type="checkbox"/> subject_id;subject_name;semester;number_of_lectures;number_of_labs <span>String</span> <span>▼</span>
1	1;Manipulating Data in SQL;3;36;36
2	2;RDBMS Basics;4;38;38
3	3;Data Analytics;5;30;30
4	4;Database Administration;6;36;36
5	5;Data Mining;7;40;52

CANCEL IMPORT



# Создание элементов в коллекции

```
> use all_subjects
< 'switched to db all_subjects'
> db.db_subjects.find()
< { _id: ObjectId("67c6cdeb29780ed6102d6720"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '1;Manipulating Data in SQL;3;36;36' }
{ _id: ObjectId("67c6cdeb29780ed6102d6721"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '2;RDBMS Basics;4;38;38' }
{ _id: ObjectId("67c6cdeb29780ed6102d6722"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '3;Data Analytics;5;30;30' }
{ _id: ObjectId("67c6cdeb29780ed6102d6723"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '4;Database Administration;6;36;36' }
{ _id: ObjectId("67c6cdeb29780ed6102d6724"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '5;Data Mining;7;40;52' }
```

# Создание элементов в коллекции

```
> db.dev_subjects.insertOne({subject_id : 1, subject_name : "Object-Oriented Programming", semester: 3, number_of_lectures : 36,number_of_labs: 36})
< { acknowledged: true,
  insertedId: ObjectId("67c6cf18033cf5a1709f1cd3") }
```

all\_subjects.dev\_subjects

Documents

Aggregations

Schema

Explain Plan

FILTER { field: 'value' }

ADD DATA



VIEW



```
_id: ObjectId('67c6cf18033cf5a1709f1cd3')
subject_id: 1
subject_name: "Object-Oriented Programming"
semester: 3
number_of_lectures: 36
number_of_labs: 36
```

- Сервер возвращает \_id добавленной записи

# Создание элементов в коллекции

```
> db.dev_subjects.insertMany([ {subject_id : 2, subject_name : "Programming Languages", semester: 4, number_of_lectures : 36, number_of_labs: 36},
```

```
{subject_id : 3, subject_name : "Programming Basics", semester: 1, number_of_lectures : 36, number_of_labs: 36} ] )
```

```
< { acknowledged: true,  
  insertedIds:  
    { '0': ObjectId("67c6d01f033cf5a1709f1cd4"),  
      '1': ObjectId("67c6d01f033cf5a1709f1cd5") } }  
all_subjects >
```

```
_id: ObjectId('67c6cf18033cf5a1709f1cd3')  
subject_id: 1  
subject_name: "Object-Oriented Programming"  
semester: 3  
number_of_lectures: 36  
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd4')  
subject_id: 2  
subject_name: "Programming Languages"  
semester: 4  
number_of_lectures: 36  
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd5')  
subject_id: 3  
subject_name: "Programming Basics"  
semester: 1  
number_of_lectures: 36  
number_of_labs: 36
```

# Обновление элементов в коллекции

```
_id: ObjectId('67c6cf18033cf5a1709f1cd3')
subject_id: 1
subject_name: "Object-Oriented Programming"
semester: 3
number_of_lectures: 36
number_of_labs: 36
```

```
_id: ObjectId('67c6cf18033cf5a1709f1cd3')
subject_id: 1
subject_name: "Object-Oriented Programming"
semester: 3
number_of_lectures: 37
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd4')
subject_id: 2
subject_name: "Programming Languages"
semester: 4
number_of_lectures: 36
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd4')
subject_id: 2
subject_name: "Programming Languages"
semester: 4
number_of_lectures: 36
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd5')
subject_id: 3
subject_name: "Programming Basics"
semester: 1
number_of_lectures: 36
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd5')
subject_id: 3
subject_name: "Programming Basics"
semester: 1
number_of_lectures: 36
number_of_labs: 36
```

```
> db.dev_subjects.updateOne({number_of_lectures : 36}, {$set: {number_of_lectures : 37}})
< { acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0 }
all_subjects>
```

# Обновление элементов в коллекции

```
_id: ObjectId('67c6cf18033cf5a1709f1cd3')
subject_id: 1
subject_name: "Object-Oriented Programming"
semester: 3
number_of_lectures: 36
number_of_labs: 36
```

```
_id: ObjectId('67c6cf18033cf5a1709f1cd3')
subject_id: 1
subject_name: "Object-Oriented Programming"
semester: 3
number_of_lectures: 38
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd4')
subject_id: 2
subject_name: "Programming Languages"
semester: 4
number_of_lectures: 36
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd4')
subject_id: 2
subject_name: "Programming Languages"
semester: 4
number_of_lectures: 38
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd5')
subject_id: 3
subject_name: "Programming Basics"
semester: 1
number_of_lectures: 36
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd5')
subject_id: 3
subject_name: "Programming Basics"
semester: 1
number_of_lectures: 38
number_of_labs: 36
--
```

```
> db.dev_subjects.updateMany({number_of_labs : 36}, {$set: {number_of_lectures : 38}})
< { acknowledged: true,
  insertedId: null,
  matchedCount: 3,
  modifiedCount: 3,
  upsertedCount: 0 }
all_subjects>
```

# Удаление элементов из коллекции

```
> db.dev_subjects.deleteOne({number_of_labs : 36})
< { acknowledged: true, deletedCount: 1 }
all_subjects> |
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd4')
subject_id: 2
subject_name: "Programming Languages"
semester: 4
number_of_lectures: 38
number_of_labs: 36
```

```
_id: ObjectId('67c6d01f033cf5a1709f1cd5')
subject_id: 3
subject_name: "Programming Basics"
semester: 1
number_of_lectures: 38
number_of_labs: 36
```

```
> db.dev_subjects.deleteMany({number_of_labs : 36})
< { acknowledged: true, deletedCount: 2 }
all_subjects> |
```

all\_subjects.dev\_subjects

Documents Aggregations Schema Explain Plan Indexes Validations

FILTER { field: 'value' }

ADD DATA



VIEW





# Поиск элементов в коллекции

```
> db.db_subjects.find()
< { _id: ObjectId("67c6cdeb29780ed6102d6720"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '1;Manipulating Data in SQL;3;36;36' }
{ _id: ObjectId("67c6cdeb29780ed6102d6721"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '2;RDBMS Basics;4;38;38' }
{ _id: ObjectId("67c6cdeb29780ed6102d6722"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '3;Data Analytics;5;30;30' }
{ _id: ObjectId("67c6cdeb29780ed6102d6723"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '4;Database Administration;6;36;36' }
{ _id: ObjectId("67c6cdeb29780ed6102d6724"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '5;Data Mining;7;40;52' }
all_subjects>
```

- `find()` – выводит все элементы коллекции

# Поиск элементов в коллекции

```
> db.db_subjects.find().pretty()
< { _id: ObjectId("67c6cdeb29780ed6102d6720"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '1;Manipulating Data in SQL;3;36;36' }
{ _id: ObjectId("67c6cdeb29780ed6102d6721"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '2;RDBMS Basics;4;38;38' }
{ _id: ObjectId("67c6cdeb29780ed6102d6722"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '3;Data Analytics;5;30;30' }
{ _id: ObjectId("67c6cdeb29780ed6102d6723"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '4;Database Administration;6;36;36' }
{ _id: ObjectId("67c6cdeb29780ed6102d6724"),
  'subject_id;subject_name;semester;number_of_lectures;number_of_labs': '5;Data Mining;7;40;52' }
```

- `find().pretty()` – выводит все элементы коллекции в читаемом виде

# Поиск элементов в коллекции

```
> db.dev_subjects.find({subject_id : 3})
< { _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
all_subjects >
```

---

```
_id: ObjectId('67c6f221033cf5a1709f1cd6')
subject_id: 1
subject_name: "Object-Oriented Programming"
semester: 3
number_of_lectures: 36
number_of_labs: 36
```

---

```
_id: ObjectId('67c6f297033cf5a1709f1cd7')
subject_id: 2
subject_name: "Programming Languages"
semester: 4
number_of_lectures: 36
number_of_labs: 36
```

---

```
_id: ObjectId('67c6f297033cf5a1709f1cd8')
subject_id: 3
subject_name: "Programming Basics"
semester: 1
number_of_lectures: 36
number_of_labs: 36
```

# Поиск элементов в коллекции

- **\$eq** – равно
- **\$ne** – не равно
- **\$gt** – больше чем
- **\$lt** – меньше чем
- **\$gte** – больше или равно
- **\$lte** – меньше или равно

```
> db.dev_subjects.find({subject_id : {$gt : 2}})
< { _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
all_subjects> |
```

```
> db.dev_subjects.find({number_of_lectures : {$gt : 30, $lt : 40}})
< { _id: ObjectId("67c6f221033cf5a1709f1cd6"),
  subject_id: 1,
  subject_name: 'Object-Oriented Programming',
  semester: 3,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd7"),
  subject_id: 2,
  subject_name: 'Programming Languages',
  semester: 4,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
```

# Поиск элементов в коллекции

- Одновременные условия – через запятую

```
> db.dev_subjects.find({number_of_lectures : {$gt : 30}, number_of_labs: {$lt : 40}})
< { _id: ObjectId("67c6f221033cf5a1709f1cd6"),
  subject_id: 1,
  subject_name: 'Object-Oriented Programming',
  semester: 3,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd7"),
  subject_id: 2,
  subject_name: 'Programming Languages',
  semester: 4,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
```

# Поиск элементов в коллекции

- Одновременные условия – через запятую

```
> db.dev_subjects.find({number_of_lectures : {$gt : 30}, number_of_labs: {$lt : 40}})
< { _id: ObjectId("67c6f221033cf5a1709f1cd6"),
  subject_id: 1,
  subject_name: 'Object-Oriented Programming',
  semester: 3,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd7"),
  subject_id: 2,
  subject_name: 'Programming Languages',
  semester: 4,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
```

# Поиск элементов в коллекции

- Различные условия – через \$or

```
> db.dev_subjects.find({number_of_lectures: 36, $or: [{subject_name: 'Programming Basics'}, {semester: 4}]})
< { _id: ObjectId("67c6f297033cf5a1709f1cd7"),
  subject_id: 2,
  subject_name: 'Programming Languages',
  semester: 4,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
all_subjects> |
```

# Поиск элементов в коллекции

- \$in и \$nin

```
> db.dev_subjects.find({subject_id : {$in: [1, 2]}})
< { _id: ObjectId("67c6f221033cf5a1709f1cd6"),
  subject_id: 1,
  subject_name: 'Object-Oriented Programming',
  semester: 3,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd7"),
  subject_id: 2,
  subject_name: 'Programming Languages',
  semester: 4,
  number_of_lectures: 36,
  number_of_labs: 36 }
all_subjects> |
```

```
> db.dev_subjects.find({subject_id : {$nin: [1, 2]}})
< { _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
all_subjects> |
```



# Поиск элементов в коллекции

- null

```
> db.dev_subjects.insertOne({subject_id : 4, subject_name : "Applied Programming", number_of_lectures : 24,number_of_labs: 36})  
< { acknowledged: true,  
  insertedId: ObjectId("67c6f715033cf5a1709f1cd9") }  
all_subjects> |
```

```
> db.dev_subjects.find({semester : null})  
< { _id: ObjectId("67c6f715033cf5a1709f1cd9"),  
  subject_id: 4,  
  subject_name: 'Applied Programming',  
  number_of_lectures: 24,  
  number_of_labs: 36 }  
all_subjects> |
```

# Поиск элементов в коллекции

- \$regex – LIKE и не только
- distinct()

```
> db.dev_subjects.find({subject_name: {$regex:"A"}})
< { _id: ObjectId("67c6f715033cf5a1709f1cd9"),
  subject_id: 4,
  subject_name: 'Applied Programming',
  number_of_lectures: 24,
  number_of_labs: 36 }
all_subjects> |
```

```
> db.dev_subjects.distinct("number_of_lectures")
< [ 24, 36 ]
all_subjects> |
```

# Поиск элементов в коллекции

- Проекция – вывод только нужных полей

```
> db.dev_subjects.find({number_of_labs: 36}, {subject_name: 1})
```

```
< { _id: ObjectId("67c6f221033cf5a1709f1cd6"),  
  subject_name: 'Object-Oriented Programming' }  
{ _id: ObjectId("67c6f297033cf5a1709f1cd7"),  
  subject_name: 'Programming Languages' }  
{ _id: ObjectId("67c6f297033cf5a1709f1cd8"),  
  subject_name: 'Programming Basics' }  
{ _id: ObjectId("67c6f715033cf5a1709f1cd9"),  
  subject_name: 'Applied Programming' }
```

```
> db.dev_subjects.find({number_of_labs: 36}, {subject_name: 1, semester : 2})
```

```
< { _id: ObjectId("67c6f221033cf5a1709f1cd6"),  
  subject_name: 'Object-Oriented Programming',  
  semester: 3 }  
{ _id: ObjectId("67c6f297033cf5a1709f1cd7"),  
  subject_name: 'Programming Languages',  
  semester: 4 }  
{ _id: ObjectId("67c6f297033cf5a1709f1cd8"),  
  subject_name: 'Programming Basics',  
  semester: 1 }  
{ _id: ObjectId("67c6f715033cf5a1709f1cd9"),  
  subject_name: 'Applied Programming' }
```

```
all_subjects>
```

# Поиск элементов в коллекции

- Вложенные элементы

```
> db.dev_subjects.insertOne({subject_id : 5, subject_name : "PL/SQL Programming", book : {"name":"PL/SQL Basics", "author":"Fershtein, Pribyl"}})
< { acknowledged: true,
  insertedId: ObjectId("67c6face033cf5a1709f1cda") }
```

```
> db.dev_subjects.find({"book.name" : "PL/SQL Basics"})
< { _id: ObjectId("67c6face033cf5a1709f1cda"),
  subject_id: 5,
  subject_name: 'PL/SQL Programming',
  book: { name: 'PL/SQL Basics', author: 'Fershtein, Pribyl' } }
all_subjects>
```

```
{
  "_id": {
    "$oid": "67c6face033cf5a1709f1cda"
  },
  "subject_id": 5,
  "subject_name": "PL/SQL Programming",
  "book": {
    "name": "PL/SQL Basics",
    "author": "Fershtein, Pribyl"
  }
}
```

```
> db.dev_subjects.insertOne({subject_id : 6, subject_name : "Data Science", book : {"name":"Mathematics for Data Science", "pages":342}})
< { acknowledged: true,
  insertedId: ObjectId("67c6fc03033cf5a1709f1cdb") }
```

```
> db.dev_subjects.find({"book.author" : {"$exists":true}})
< { _id: ObjectId("67c6face033cf5a1709f1cda"),
  subject_id: 5,
  subject_name: 'PL/SQL Programming',
  book: { name: 'PL/SQL Basics', author: 'Fershtein, Pribyl' } }
all_subjects>
```

```
{
  "_id": {
    "$oid": "67c6fc03033cf5a1709f1cdb"
  },
  "subject_id": 6,
  "subject_name": "Data Science",
  "book": {
    "name": "Mathematics for Data Science",
    "pages": 342
  }
}
```

# Поиск элементов в коллекции

- Размер и сдвиг выборки

```
> db.dev_subjects.find().limit(3)
< { _id: ObjectId("67c6f221033cf5a1709f1cd6"),
  subject_id: 1,
  subject_name: 'Object-Oriented Programming',
  semester: 3,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd7"),
  subject_id: 2,
  subject_name: 'Programming Languages',
  semester: 4,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
```

```
> db.dev_subjects.find().skip(2)
< { _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f715033cf5a1709f1cd9"),
  subject_id: 4,
  subject_name: 'Applied Programming',
  number_of_lectures: 24,
  number_of_labs: 36 }
{ _id: ObjectId("67c6face033cf5a1709f1cda"),
  subject_id: 5,
  subject_name: 'PL/SQL Programming',
  book: { name: 'PL/SQL Basics', author: 'Fershtein, Pribyl' } }
{ _id: ObjectId("67c6fc03033cf5a1709f1cdb"),
  subject_id: 6,
  subject_name: 'Data Science',
  book: { name: 'Mathematics for Data Science', pages: 342 } }
all_subjects > |
```

# Поиск элементов в коллекции

- Размер и сдвиг выборки одновременно

```
> db.dev_subjects.find().limit(3).skip(1)
< { _id: ObjectId("67c6f297033cf5a1709f1cd7"),
  subject_id: 2,
  subject_name: 'Programming Languages',
  semester: 4,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f715033cf5a1709f1cd9"),
  subject_id: 4,
  subject_name: 'Applied Programming',
  number_of_lectures: 24,
  number_of_labs: 36 }
all_subjects> |
```

# Сортировка

```
> db.dev_subjects.find().sort({number_of_lectures: 1}, {subject_name : 2})
< { _id: ObjectId("67c6face033cf5a1709f1cda"),
  subject_id: 5,
  subject_name: 'PL/SQL Programming',
  book: { name: 'PL/SQL Basics', author: 'Fershtein, Pribyl' } }
{ _id: ObjectId("67c6fc03033cf5a1709f1cdb"),
  subject_id: 6,
  subject_name: 'Data Science',
  book: { name: 'Mathematics for Data Science', pages: 342 } }
{ _id: ObjectId("67c6f715033cf5a1709f1cd9"),
  subject_id: 4,
  subject_name: 'Applied Programming',
  number_of_lectures: 24,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f221033cf5a1709f1cd6"),
  subject_id: 1,
  subject_name: 'Object-Oriented Programming',
  semester: 3,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd7"),
  subject_id: 2,
  subject_name: 'Programming Languages',
  semester: 4,
  number_of_lectures: 36,
  number_of_labs: 36 }
{ _id: ObjectId("67c6f297033cf5a1709f1cd8"),
  subject_id: 3,
  subject_name: 'Programming Basics',
  semester: 1,
  number_of_lectures: 36,
  number_of_labs: 36 }
```

# Поиск элементов в коллекции

- Результат поиска помещается в курсор
- Метод `hasNext()` позволяет перебирать значения в цикле
- Метод `next()` – получить следующее значение

```
> var subject_cursor = db.dev_subjects.find()
> while(subject_cursor.hasNext()){obj = subject_cursor.next(); print(obj["subject_name"]);}
< 'Object-Oriented Programming'
< 'Programming Languages'
< 'Programming Basics'
< 'Applied Programming'
< 'PL/SQL Programming'
< 'Data Science'
all_subjects > |
```



# Поиск элементов в коллекции

- Подсчет количества

```
> db.dev_subjects.countDocuments()  
< 6  
> db.dev_subjects.find({number_of_labs:36}).count()  
< 4
```

Вопросы?