

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ**

Факультет «Инфокоммуникационных технологий»

Направление подготовки «45.03.04 Интеллектуальные системы в гуманитарной сфере»

О Т Ч Е Т

о выполнении лабораторной работы №6 «MongoDB»

Выполнил:

Студент **Новаторова И.И.** **К3243**
(Фамилия И.О.) номер группы

Проверил:

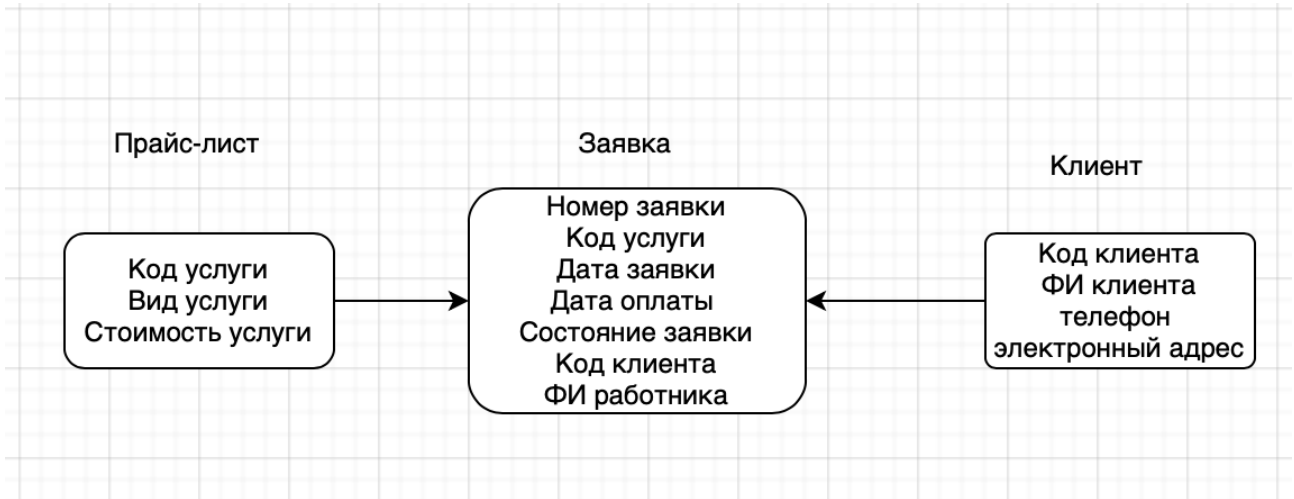
Преподаватель **Говоров А.И.**
(Фамилия И.О.)

**Санкт-Петербург
2020**

1) Цель работы

Овладеть практическими навыками и умениями реализации баз данных в MongoDB.

2) Уменьшение модели данных



3) Реализация модели

1. Таблица «прайс-лист»

```
> db.price.insert({price: 1500, type_service: 'banner'});
writeResult({ "nInserted" : 1 })
> db.price.insert({price: 2000, type_service: 'bus advertisement'});
writeResult({ "nInserted" : 1 })
> db.price.insert({price: 2500, type_service: 'metro advertisement'});
writeResult({ "nInserted" : 1 })
> db.price.insert({price: 200, type_service: 'flyer advertisement'});
writeResult({ "nInserted" : 1 })
> db.price.insert({price: 3000, type_service: 'widescreen banner'});
writeResult({ "nInserted" : 1 })
> db.price.insert({price: 1800, type_service: 'magazine advertisement'});
writeResult({ "nInserted" : 1 })
> db.price.find()
{ "_id" : ObjectId("5efe5a0dde22f61be7d55543"), "price" : 1500, "type_service" : "banner" }
{ "_id" : ObjectId("5efe5a0dde22f61be7d55544"), "price" : 2000, "type_service" : "bus advertisement" }
{ "_id" : ObjectId("5efe5a0dde22f61be7d55545"), "price" : 2500, "type_service" : "metro advertisement" }
{ "_id" : ObjectId("5efe5a0dde22f61be7d55546"), "price" : 200, "type_service" : "flyer advertisement" }
{ "_id" : ObjectId("5efe5a0dde22f61be7d55547"), "price" : 3000, "type_service" : "widescreen banner" }
{ "_id" : ObjectId("5efe5a0dde22f61be7d55548"), "price" : 1800, "type_service" : "magazine advertisement" }
```

2. Таблица «Клиент»

```
> db.client.insert({name_client: 'Ivan Ivanov', phone_client: '+79991110011', email: 'supervanya@mail.ru'});
writeResult({ "nInserted" : 1 })
> db.client.insert({name_client: 'Ksenia Efremova', phone_client: '+79876543210', email: 'ksu01@mail.ru'});
writeResult({ "nInserted" : 1 })
> db.client.insert({name_client: 'Anna Smirnova', phone_client: '+79112334045', email: 'smanya@mail.ru'});
writeResult({ "nInserted" : 1 })
> db.client.insert({name_client: 'Mikhail Efimov', phone_client: '+79213876544', email: 'efimov@mail.ru'});
writeResult({ "nInserted" : 1 })
> db.client.insert({name_client: 'Leonid Gerasimov', phone_client: '+79056789900', email: 'gerasimov@mail.ru'});
writeResult({ "nInserted" : 1 })
> db.client.find()
{ "_id" : ObjectId("5efe5addde22f61be7d55549"), "name_client" : "Ivan Ivanov", "phone_client" : "+79991110011", "email" : "supervanya@mail.ru" }
{ "_id" : ObjectId("5efe5addde22f61be7d5554a"), "name_client" : "Ksenia Efremova", "phone_client" : "+79876543210", "email" : "ksu01@mail.ru" }
{ "_id" : ObjectId("5efe5addde22f61be7d5554b"), "name_client" : "Anna Smirnova", "phone_client" : "+79112334045", "email" : "smanya@mail.ru" }
{ "_id" : ObjectId("5efe5addde22f61be7d5554c"), "name_client" : "Mikhail Efimov", "phone_client" : "+79213876544", "email" : "efimov@mail.ru" }
{ "_id" : ObjectId("5efe5addde22f61be7d5554d"), "name_client" : "Leonid Gerasimov", "phone_client" : "+79056789900", "email" : "gerasimov@mail.ru" }
```

3. Таблица «Заявка»

```
> db.application.insert({id_service: ObjectId("5efe5a0dde22f61be7d55543"), id_client: ObjectId("5efe5adde22f61be7d55549"), date_application: '2020/02/20', date_payment: '2020/02/20', status: 'transferred', name_worker: 'Julia Zelenina'});
WriteResult({ "nInserted" : 1 })
> db.application.insert({id_service: ObjectId("5efe5a0dde22f61be7d55547"), id_client: ObjectId("5efe5adde22f61be7d5554b"), date_application: '2019/07/17', date_payment: '2019/07/24', status: 'completed', name_worker: 'Konstantin Sviridov'});
WriteResult({ "nInserted" : 1 })
> db.application.insert({id_service: ObjectId("5efe5a0dde22f61be7d55546"), id_client: ObjectId("5efe5adde22f61be7d5554d"), date_application: '2020/03/06', date_payment: '2020/03/14', status: 'processed', name_worker: 'Veronika Smirnova'});
WriteResult({ "nInserted" : 1 })
> db.application.insert({id_service: ObjectId("5efe5a0dde22f61be7d55545"), id_client: ObjectId("5efe5adde22f61be7d5554a"), date_application: '2020/04/05', date_payment: '2020/04/06', status: 'completed', name_worker: 'Maria Petrova'});
WriteResult({ "nInserted" : 1 })
> db.application.insert({id_service: ObjectId("5efe5a0dde22f61be7d55544"), id_client: ObjectId("5efe5adde22f61be7d5554c"), date_application: '2019/08/07', date_payment: '2019/08/09', status: 'completed', name_worker: 'Victor Korablev'});
WriteResult({ "nInserted" : 1 })
> db.application.find()
{ "_id" : ObjectId("5efe5c5dde22f61be7d5554e"), "id_service" : ObjectId("5efe5a0dde22f61be7d55543"), "id_client" : ObjectId("5efe5adde22f61be7d55549"), "date_application" : "2020/02/20", "date_payment" : "2020/02/20", "status" : "transferred", "name_worker" : "Julia Zelenina" }
{ "_id" : ObjectId("5efe5c5dde22f61be7d5554f"), "id_service" : ObjectId("5efe5a0dde22f61be7d55547"), "id_client" : ObjectId("5efe5adde22f61be7d5554b"), "date_application" : "2019/07/17", "date_payment" : "2019/07/24", "status" : "completed", "name_worker" : "Konstantin Sviridov" }
{ "_id" : ObjectId("5efe5c5dde22f61be7d55550"), "id_service" : ObjectId("5efe5a0dde22f61be7d55546"), "id_client" : ObjectId("5efe5adde22f61be7d5554d"), "date_application" : "2020/03/06", "date_payment" : "2020/03/14", "status" : "processed", "name_worker" : "Veronika Smirnova" }
{ "_id" : ObjectId("5efe5c5dde22f61be7d55551"), "id_service" : ObjectId("5efe5a0dde22f61be7d55545"), "id_client" : ObjectId("5efe5adde22f61be7d5554a"), "date_application" : "2020/04/05", "date_payment" : "2020/04/06", "status" : "completed", "name_worker" : "Maria Petrova" }
{ "_id" : ObjectId("5efe5c5dde22f61be7d55552"), "id_service" : ObjectId("5efe5a0dde22f61be7d55544"), "id_client" : ObjectId("5efe5adde22f61be7d5554c"), "date_application" : "2019/08/07", "date_payment" : "2019/08/09", "status" : "completed", "name_worker" : "Victor Korablev" }
```

4) Запросы

1. Показать информацию о заявках, оставленных 6 марта 2020 года (для удобства восприятия исключаем id заявки)

```
db.application.find({date_application: '2020/03/06'}, {_id: 0});
```

```
> db.application.find({date_application: '2020/03/06'}, {_id: 0});
{ "id_service" : ObjectId("5efe5a0dde22f61be7d55546"), "id_client" : ObjectId("5efe5adde22f61be7d5554d"), "date_application" : "2020/03/06", "date_payment" : "2020/03/14", "status" : "processed", "name_worker" : "Veronika Smirnova" }
>
```

2. Показать стоимость самой бюджетной услуги агентства

```
db.price.aggregate(
[
{
$group:
{
_id: {},
minPrice: { $min: "$price" }
}
}
]);
```

```
> db.price.aggregate(
... [
... {
... $group:
... {
... _id: {},
... minPrice: { $min: "$price" }
... }
... }
... ]
... );
{ "_id" : { }, "minPrice" : 200 }
>
```

3. Вывести информацию об услугах, стоимость которых находится больше 2 тысяч рублей за единицу продукции (для удобства восприятия исключаем id услуги)

```
db.price.find({price: {$gt: 2000}}, {_id: 0});
```

```
> db.price.find({price: {$gt: 2000}}, {_id: 0});
{ "price" : 2500, "type_service" : "metro advertisement" }
{ "price" : 3000, "type_service" : "widescreen banner" }
>
```

4. Вывести ФИ сотрудников, уже выполнивших свою заявку (для удобства восприятия исключаем id заявки)

```
db.application.find({status: 'completed'}, {name_worker: 1, _id: 0})
```

```
> db.application.find({status: 'completed'}, {name_worker: 1, _id: 0})
{ "name_worker" : "Konstantin Sviridov" }
{ "name_worker" : "Maria Petrova" }
{ "name_worker" : "Victor Korablev" }
>
```

5. Количество заявок по категориям: выполнено, передано исполнителю, в процессе

```
function map(){
    emit(this.status, 1);
}
function reduce(key, values) {
    var sum = 0;
    for(var i in values) {
        sum += values[i];
    }
    return sum;
}
```

```
db.application.mapReduce(map, reduce, {out: "result"})
db.result.find()
```

```

> function map(){
...   emit(this.status, 1);
... }
> function reduce(key, values) {
...   var sum = 0;
...   for(var i in values) {
...     sum += values[i];
...   }
...   return sum;
... }
> db.application.mapReduce(map, reduce,{out:"result"})
{
  "result" : "result",
  "timeMillis" : 56,
  "counts" : {
    "input" : 5,
    "emit" : 5,
    "reduce" : 1,
    "output" : 3
  },
  "ok" : 1
}
> db.result.find()
{ "_id" : "completed", "value" : 3 }
{ "_id" : "processed", "value" : 1 }
{ "_id" : "transferred", "value" : 1 }

```

6. Вывести ФИ клиентов, длина электронных адресов которых больше 15

```

db.client.find({"email": {"$exists": true}, "$expr": {"$gt": [{"$strLenCP": "$email"}, 15]}}, {"name_client": 1,
_id: 0})

```

```

> db.client.find({
...   "email": { "$exists": true },
...   "$expr": { "$gt": [ { "$strLenCP": "$email" }, 15 ] }
... }, {"name_client": 1, _id: 0})
{ "name_client" : "Ivan Ivanov" }
{ "name_client" : "Leonid Gerasimov" }
>

```

5) Вывод

В результате выполнения работы были получены навыки работы с MongoDB.