

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

«Национальный исследовательский университет ИТМО»

Факультет информационных технологий и программирования

Лабораторная работа №6

по предмету "Проектирование баз данных"

Вариант 1

Выполнил студент группы М34051

Кузнецов Илья

Преподаватель Демина Л.С.

Санкт-Петербург

2022

Вариант 1. Заказ продуктов, товаров в магазине

1. Создание коллекций и заполнение данными

```
db.user.insertMany([
  {
    _id: 'kuzznya',
    password: 'password',
    name: 'Ilya',
    surname: 'Kuznetsov',
    phone_number: '+353873432703',
    verified: true,
    address: '90 Greenville Place'
  },
  {
    _id: 'afterbvrner',
    password: 'password',
    name: 'Max',
    surname: 'Golish',
    phone_number: '88005553535',
    verified: true,
    address: 'Somewhere'
  },
  {
    _id: 'I-SER-I',
    password: 'password',
    name: 'Sergey',
    surname: 'Papikyan',
    phone_number: '88005553535',
    verified: true,
    address: 'Somewhere'
  }
]);

db.store.insertMany([
  {
    _id: 'Peter the Great delivery market',
    location: 'St. Petersburg'
  },
  {
    _id: 'Mausoleum market',
    location: 'Moscow'
  },
  {
    _id: 'Guinness delivery',
    location: 'Dublin'
  }
]);

const guinness = db.product.insertOne({
  name: 'Guinness',
  description: 'Guinness is good for you',
  price: 10000,
  category: 'Alcohol',
  params: {
    "ABV": 4.2,
```

```
        "Country": "Ireland",
        "Perfect": true
    }
}).insertedId;

const pinotGrigio = db.product.insertOne({
    name: 'Pinot Grigio',
    description: 'Italian white wine',
    price: 100000,
    category: 'Alcohol',
    params: {
        "ABV": 12,
        "Country": "Italy"
    }
}).insertedId;

const bordeaux = db.product.insertOne({
    name: 'Bordeaux',
    description: 'French red wine',
    price: 50000,
    category: 'Alcohol',
    params: {
        "ABV": 13,
        "Country": "France"
    }
}).insertedId;

const campoViejo = db.product.insertOne({
    name: 'Campo Viejo Rioja',
    description: 'Spanish red wine',
    price: 120000,
    category: 'Alcohol',
    params: {
        "ABV": 13.5,
        "Country": "Spain"
    }
}).insertedId;

const apples = db.product.insertOne({
    name: 'Apples',
    description: 'Local apples',
    price: 5000,
    category: 'Fruits',
    params: {
        "Country": "Russia"
    }
}).insertedId;

const peaches = db.product.insertOne({
    name: 'Peaches',
    description: 'Local peaches',
    price: 6000,
    category: 'Fruits',
    params: {
        "Country": "Russia"
    }
}).insertedId;
```

```
const carrot = db.product.insertOne({
  name: 'Carrot',
  description: 'Local carrot',
  price: 4500,
  category: 'Vegetables',
  params: {
    "Country": "Russia"
  }
}).insertedId;

const potato = db.product.insertOne({
  name: 'Potato',
  description: 'Local potato',
  price: 4000,
  category: 'Vegetables',
  params: {
    "Country": "Russia"
  }
}).insertedId;

const yogurt = db.product.insertOne({
  name: 'Yogurt',
  description: 'Turkish yogurt',
  price: 10000,
  category: 'Dairy',
  params: {
    "Fat %": 5,
    "Lactose": true
  }
}).insertedId;

const milk1 = db.product.insertOne({
  name: 'Milk "Happy farmer"',
  price: 10000,
  category: 'Dairy',
  params: {
    "Fat %": 2.5,
    "Lactose": true
  }
}).insertedId;

const milk2 = db.product.insertOne({
  name: 'Milk "Prostokvashino"',
  price: 11000,
  category: 'Dairy',
  params: {
    "Fat %": 3,
    "Lactose": true
  }
}).insertedId;

const spaghetti = db.product.insertOne({
  name: 'Spaghetti "Barilla"',
  price: 15000,
  category: 'Grocery'
}).insertedId;
```

```

db.order.insertMany([
  {
    user: 'kuzznya',
    store: 'Peter the Great delivery market',
    state: 'PAID',
    items: [
      { product: guinness, amount: 1000 },
      { product: pinotGrigio, amount: 2 },
      { product: potato, amount: 5 }
    ]
  },
  {
    user: 'afterbvrner',
    store: 'Peter the Great delivery market',
    state: 'DELIVERED',
    items: [
      { product: guinness, amount: 6 },
      { product: pinotGrigio, amount: 1 },
      { product: carrot, amount: 2 },
      { product: spaghetti, amount: 1 }
    ]
  }
]);

```

2. Создание индексов

```

db.product.createIndex(
  {
    name: 'text',
    description: 'text'
  },
  { name: 'productTextSearch', weights: { name: 2, description: 1 } }
);

```

```

db.product.explain().find({
  $text: { $search: 'Pinot' }
});

```

queryPlanner:

```

{
  "namespace": "test.product",
  "indexFilterSet": false,
  "parsedQuery": {
    "$text": {
      "$search": "Pinot",
      "$language": "english",
      "$caseSensitive": false,
      "$diacriticSensitive": false
    }
  },
  "queryHash": "8BF6BFCC",

```

```

"planCacheKey": "E2440218",
"maxIndexedOrSolutionsReached": false,
"maxIndexedAndSolutionsReached": false,
"maxScansToExplodeReached": false,
"winningPlan": {
  "stage": "TEXT_MATCH",
  "indexPrefix": {},
  "indexName": "productTextSearch",
  "parsedTextQuery": {
    "terms": [
      "pinot"
    ],
    "negatedTerms": [],
    "phrases": [],
    "negatedPhrases": []
  },
  "textIndexVersion": "3",
  "inputStage": {
    "stage": "FETCH",
    "inputStage": {
      "stage": "IXSCAN",
      "keyPattern": {
        "_fts": "text",
        "_ftsx": "1"
      },
      "indexName": "productTextSearch",
      "isMultiKey": true,
      "isUnique": false,
      "isSparse": false,
      "isPartial": false,
      "indexVersion": "2",
      "direction": "backward",
      "indexBounds": {}
    }
  }
},
"rejectedPlans": []
}

```

```

db.order.createIndex(
  {
    user: 1,
    store: 1
  },
  { name: 'userStoreOrderSearch' }
);

db.order.explain().find({
  user: 'kuzznya',
  store: 'Peter the Great delivery market'
});

```

3. Создание представления

```

db.createView(
  'orderPrice',
  'order',
  [
    {
      $unwind: '$items'
    },
    {
      $lookup: {
        from: 'product',
        localField: 'items.product',
        foreignField: '_id',
        as: 'product'
      }
    },
    {
      $unwind: '$product'
    },
    {
      $set: {
        total: { $multiply: ['$product.price', '$items.amount'] }
      }
    },
    {
      $group: {
        _id: "$_id",
        user: { $first: '$user' },
        items: { $push: '$items' },
        total: { $sum: '$total' },
        state: { $first: '$state' }
      }
    }
  ]
);

db.orderPrice.find({});

```