ОТЧЁТ

ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 7

по теме: Знакомство с MongoDB

по дисциплине: Основы проектирования баз данных

Санкт-Петербург 2020

Выполнил:

студент группы Y2436

\_\_\_\_\_\_\_\_Морус Е.Л.

Министерство образования и науки Российской Федерации

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

УНИВЕРСИТЕТ ИТМО»

Факультет среднего профессионального образования

Проверил:

\_\_\_\_\_\_\_\_ Говоров А. И.

Дата: «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2020г.

Оценка \_\_\_\_\_\_\_\_\_\_\_

Специальность 09.02.07 Информационные системы и программирование

Цель работы

Цель практической работы №7:овладеть практическими навыками и умениями реализации баз данных в MongoDB.

Задание

Для выполнения работы Вам потребуется:

1. Попытаться уменьшить размер модели реляционной базы данных, с которой Вы работали.
2. Реализовать Вашу модель в MongoDB.
3. Заполнить модель данными, настолько, чтобы все запросы, реализуемые Вами в пункте 2.d, выдавали непустые строки.
4. Реализовать минимум 40% Ваших запросов, сделанных в работе по реляционным базам данных в MongoDB.

выполнение

1) Схема физической модели БД, спроектированной с использованием CA Erwin Data Modeler представлена на рисунке №1.

Diagram

Description automatically generated

*Рисунок 1 Модель базы данных*

**2)** Логи реализации модели:

> use database;

switched to db dataabse

> db.createCollection('office');

{ "ok" : 1 }

> db.office.insert({name: 'ForexClub', country: 'USA'});

WriteResult({ "nInserted" : 1 })

> db.office.insert({name: 'FxPro', country: 'Sweden'});

WriteResult({ "nInserted" : 1 })

> db.office.insert({name: 'GlobalFX', country: 'Norway'});

WriteResult({ "nInserted" : 1 })

> db.office.insert({name: 'Fibo Group', country: 'Canada'});

WriteResult({ "nInserted" : 1 })

> db.office.insert({name: 'Finam', country: 'USA'});

WriteResult({ "nInserted" : 1 })

> db.office.insert({name: 'Nord FX', country: 'Sweden'});

WriteResult({ "nInserted" : 1 })

> db.office.insert({name: 'XM', country: 'USA'});

WriteResult({ "nInserted" : 1 })

> db.office.insert({name: 'FBS', country: 'China'});

WriteResult({ "nInserted" : 1 })

> db.createCollection('broker');

{ "ok" : 1 }

> db.broker.insert({name: 'Adam Stenton', phoneNumber: '+7(921)123-12-12', officeID: '5f9684ead98438480629bc2a'});

WriteResult({ "nInserted" : 1 })

> db.broker.insert({name: 'Anna Stenton', phoneNumber: '+7(921)456-34-23', officeID: '5f9684ead98438480629bc2c'});

WriteResult({ "nInserted" : 1 })

> db.broker.insert({name: 'Harry Potter', phoneNumber: '+7(921)789-56-34', officeID: '5f9684ead98438480629bc2b'});

WriteResult({ "nInserted" : 1 })

> db.broker.insert({name: 'Donald Trump', phoneNumber: '+7(921)012-78-45', officeID: '5f9684ead98438480629bc2d'});

WriteResult({ "nInserted" : 1 })

> db.broker.insert({name: 'Vladimir Putin', phoneNumber: '+7(921)345-90-56', officeID: '5f9684ead98438480629bc2e'});

WriteResult({ "nInserted" : 1 })

> db.broker.insert({name: 'Loxozavr', phoneNumber: '+7(921)678-12-67', officeID: '5f9684ead98438480629bc2b'});

WriteResult({ "nInserted" : 1 })

> db.broker.insert({name: 'Prosto zavr', phoneNumber: '+7(921)901-34-78', officeID: '5f9684ead98438480629bc2f'});

WriteResult({ "nInserted" : 1 })

> db.broker.insert({name: 'Pushok Tushokovich', phoneNumber: '+7(921)234-56-89', officeID: '5f9684ead98438480629bc30'});

WriteResult({ "nInserted" : 1 })

> db.broker.insert({name: 'Abba Cadabra', phoneNumber: '+7(921)567-78-90', officeID: '5f9684ebd98438480629bc31'});

WriteResult({ "nInserted" : 1 })

> db.createCollection('auction');

{ "ok" : 1 }

> db.auction.insert({date: new Date('2020-10-02'), managerResponse: true, sales: ['5f9692abd98438480629bc99', '5f9692abd98438480629bc9a']});

WriteResult({ "nInserted" : 1 })

> db.auction.insert({date: new Date('2020-07-12'), managerResponse: false, sales: ['5f9692abd98438480629bc9b', '5f9692abd98438480629bc9c']});

WriteResult({ "nInserted" : 1 })

> db.auction.insert({date: new Date('2021-11-02'), managerResponse: true, sales: ['5f9692abd98438480629bc9d', '5f9692abd98438480629bc9e', '5f9692abd98438480629bc9f']});

WriteResult({ "nInserted" : 1 })

> db.auction.insert({date: new Date('2019-05-01'), managerResponse: false, sales: ['5f9692aed98438480629bca0']});

WriteResult({ "nInserted" : 1 })

> db.auction.insert({date: new Date('2021-12-12'), managerResponse: true, sales: ['5f9692abd98438480629bc99', '5f9692abd98438480629bc9a']});

WriteResult({ "nInserted" : 1 })

> db.auction.insert({date: new Date('2020-11-10'), managerResponse: false, sales: ['5f9692abd98438480629bc9b']});

WriteResult({ "nInserted" : 1 })

> db.auction.insert({date: new Date('2020-02-25'), managerResponse: true, sales: ['5f9692abd98438480629bc9c', '5f9692abd98438480629bc9d', '5f9692abd98438480629bc9e']});

WriteResult({ "nInserted" : 1 })

> db.createCollection('consignment');

{ "ok" : 1 }

> db.consignment.insert({prepayment: true, status: 'In stock', brokerID: '5f9688e3d98438480629bc32', products: [{productID: '5f968972d98438480629bc54', amount: 25}, {productID: '5f968972d98438480629bc55', amount: 40}, {productID: '5f968972d98438480629bc56', amount: 40}]});

WriteResult({ "nInserted" : 1 })

> db.consignment.insert({prepayment: false, status: 'Sold', brokerID: '5f9688e3d98438480629bc33', products: [{productID: '5f968972d98438480629bc55', amount: 30}, {productID: '5f968972d98438480629bc56', amount: 150}]});

WriteResult({ "nInserted" : 1 })

> db.consignment.insert({prepayment: true, status: 'In stock', brokerID: '5f9688e3d98438480629bc34', products: [{productID: '5f968972d98438480629bc56', amount: 15}, {productID: '5f968972d98438480629bc57', amount: 50}, {productID: '5f968972d98438480629bc5a', amount: 10}]});

WriteResult({ "nInserted" : 1 })

> db.consignment.insert({prepayment: false, status: 'Sold', brokerID: '5f9688e3d98438480629bc35', products: [{productID: '5f968972d98438480629bc57', amount: 50}, {productID: '5f968972d98438480629bc58', amount: 10}]});

WriteResult({ "nInserted" : 1 })

> db.consignment.insert({prepayment: false, status: 'Sold', brokerID: '5f9688e3d98438480629bc36', products: [{productID: '5f968972d98438480629bc58', amount: 100}, {productID: '5f968972d98438480629bc59', amount: 60}]});

WriteResult({ "nInserted" : 1 })

> db.consignment.insert({prepayment: true, status: 'In stock', brokerID: '5f9688e3d98438480629bc37', products: [{productID: '5f968972d98438480629bc59', amount: 40}, {productID: '5f968972d98438480629bc5a', amount: 40}, {productID: '5f968972d98438480629bc56', amount: 150}]});

WriteResult({ "nInserted" : 1 })

> db.consignment.insert({prepayment: false, status: 'Sold', brokerID: '5f9688e3d98438480629bc38', products: [{productID: '5f968972d98438480629bc5a', amount: 70}, {productID: '5f968974d98438480629bc5b', amount: 90}]});

WriteResult({ "nInserted" : 1 })

> db.consignment.insert({prepayment: true, status: 'Sold', brokerID: '5f9688e3d98438480629bc39', products: [{productID: '5f968974d98438480629bc5b', amount: 65}, {productID: '5f968972d98438480629bc54', amount: 100}, {productID: '5f968972d98438480629bc55', amount: 50}]});

WriteResult({ "nInserted" : 1 })

> db.createCollection('company');

{ "ok" : 1 }

> db.company.insert({name: 'Samsung', country: 'China'});

WriteResult({ "nInserted" : 1 })

> db.company.insert({name: 'Apple', country: 'USA'});

WriteResult({ "nInserted" : 1 })

> db.company.insert({name: 'Huawei', country: 'China'});

WriteResult({ "nInserted" : 1 })

> db.company.insert({name: 'Yandex', country: 'Russia'});

WriteResult({ "nInserted" : 1 })

> db.company.insert({name: 'Google', country: 'USA'});

WriteResult({ "nInserted" : 1 })

> db.company.insert({name: 'Microsoft', country: 'USA'});

WriteResult({ "nInserted" : 1 })

> db.company.insert({name: 'Sony', country: 'Japan'});

WriteResult({ "nInserted" : 1 })

> db.company.insert({name: 'Amazon', country: 'USA'});

WriteResult({ "nInserted" : 1 })

> db.createCollection('product');

{ "ok" : 1 }

> db.product.insert({name: 'Share', productionDate: new Date('2020-10-02'), expirationDate: new Date('2020-12-02'), measure: 'Dollar', companyID: '5f96890ad98438480629bc42', price: 100});

WriteResult({ "nInserted" : 1 })

> db.product.insert({name: 'Future', productionDate: new Date('2020-05-11'), expirationDate: new Date('2020-11-11'), measure: 'Dollar', companyID: '5f96890ad98438480629bc43', price: 115});

WriteResult({ "nInserted" : 1 })

> db.product.insert({name: 'Bond', productionDate: new Date('2020-07-02'), expirationDate: new Date('2020-10-02'), measure: 'Rouble', companyID: '5f96890ad98438480629bc44', price: 95});

WriteResult({ "nInserted" : 1 })

> db.product.insert({name: 'Share', productionDate: new Date('2020-01-30'), expirationDate: new Date('2020-09-10'), measure: 'Dollar', companyID: '5f96890ad98438480629bc45', price: 150});

WriteResult({ "nInserted" : 1 })

> db.product.insert({name: 'Future', productionDate: new Date('2020-05-06'), expirationDate: new Date('2020-11-19'), measure: 'Rouble', companyID: '5f96890ad98438480629bc46', price: 200});

WriteResult({ "nInserted" : 1 })

> db.product.insert({name: 'Bond', productionDate: new Date('2020-09-01'), expirationDate: new Date('2020-10-27'), measure: 'Euro', companyID: '5f96890ad98438480629bc47', price: 210});

WriteResult({ "nInserted" : 1 })

> db.product.insert({name: 'Share', productionDate: new Date('2020-10-26'), expirationDate: new Date('2020-12-01'), measure: 'Dollar', companyID: '5f96890ad98438480629bc48', price: 180});

WriteResult({ "nInserted" : 1 })

> db.product.insert({name: 'Share', productionDate: new Date('2020-07-30'), expirationDate: new Date('2020-09-05'), measure: 'Euro', companyID: '5f96890ad98438480629bc47', price: 160});

WriteResult({ "nInserted" : 1 })

> db.createCollection('work');

{ "ok" : 1 }

> db.work.insert({data: new Date('2020-10-27'), amount: 140, officeID: '5f9684ead98438480629bc2a'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-27'), amount: 100, officeID: '5f9684ead98438480629bc2b'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-27'), amount: 200, officeID: '5f9684ead98438480629bc2c'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-27'), amount: 230, officeID: '5f9684ead98438480629bc2d'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-27'), amount: 200, officeID: '5f9684ead98438480629bc2e'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-27'), amount: 160, officeID: '5f9684ead98438480629bc2f'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-27'), amount: 155, officeID: '5f9684ead98438480629bc30'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-27'), amount: 160, officeID: '5f9684ebd98438480629bc31'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-28'), amount: 170, officeID: '5f9684ead98438480629bc2a'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-28'), amount: 190, officeID: '5f9684ead98438480629bc2b'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-28'), amount: 250, officeID: '5f9684ead98438480629bc2c'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-28'), amount: 230, officeID: '5f9684ead98438480629bc2d'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-28'), amount: 290, officeID: '5f9684ead98438480629bc2e'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-28'), amount: 140, officeID: '5f9684ead98438480629bc2f'});

WriteResult({ "nInserted" : 1 })

> db.work.insert({data: new Date('2020-10-28'), amount: 180, officeID: '5f9684ead98438480629bc30'});

WriteResult({ "nInserted" : 1 })

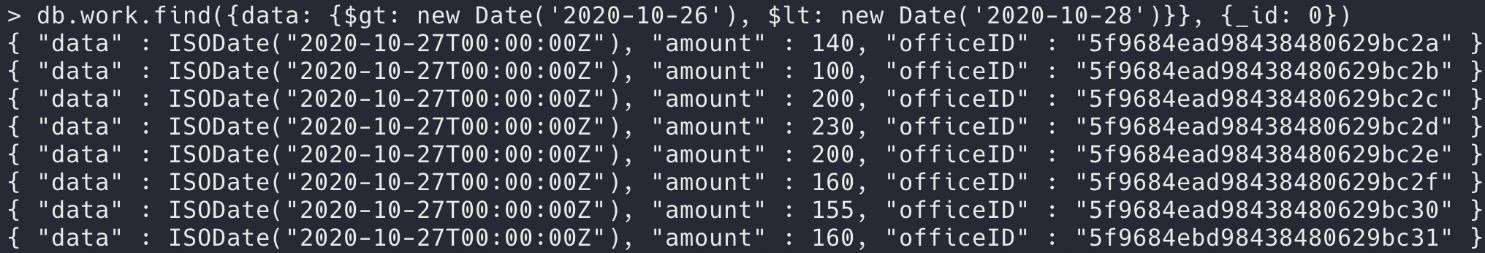
> db.work.insert({data: new Date('2020-10-28'), amount: 250, officeID: '5f9684ebd98438480629bc31'});

WriteResult({ "nInserted" : 1 })

**3) Запросы:**

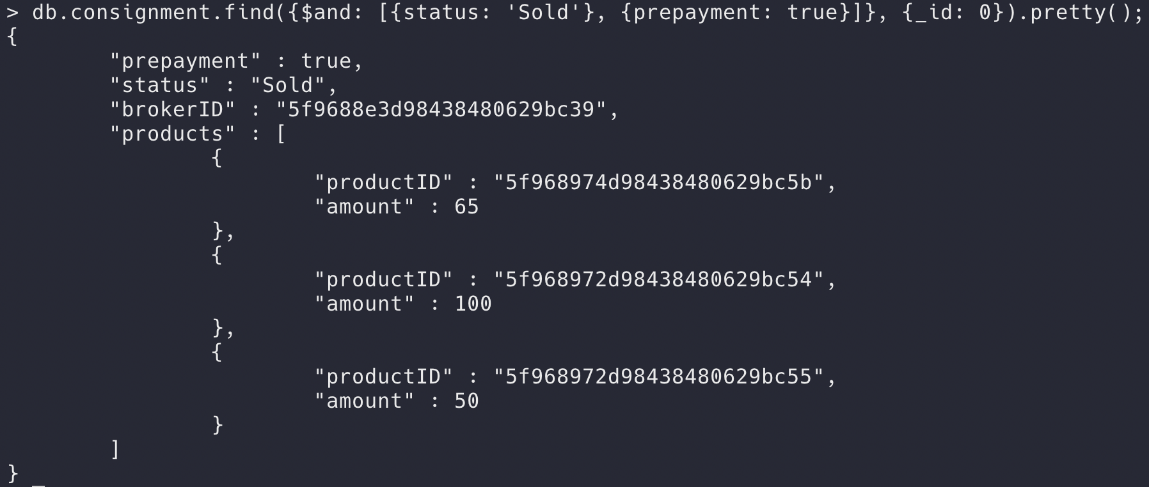
**1. Показать всю работу, которая проводилась в определенный промежуток времени**

**db.work.find({data: {$gt: new Date('2020-10-26'), $lt: new Date('2020-10-28')}}, {\_id: 0})**



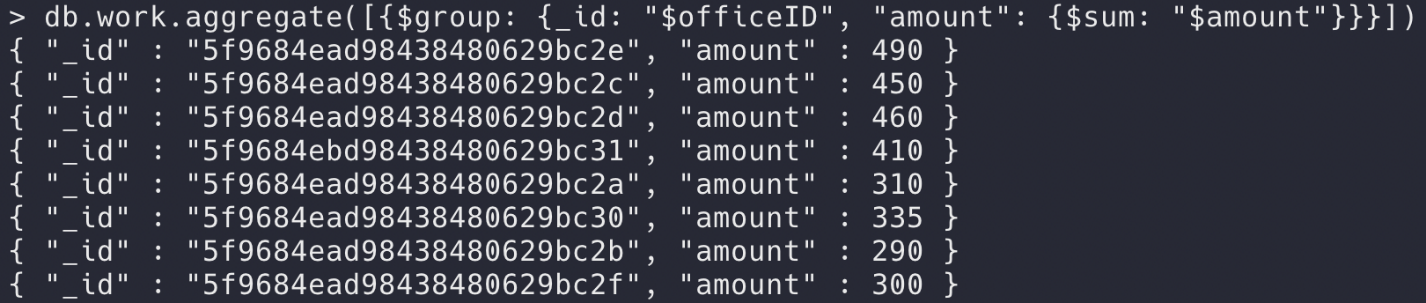
**2. Вывести все партии, у которых статус "Проданы" и была предоплата**

**db.consignment.find({$and: [{status: 'Sold'}, {prepayment: true}]}, {\_id: 0});**



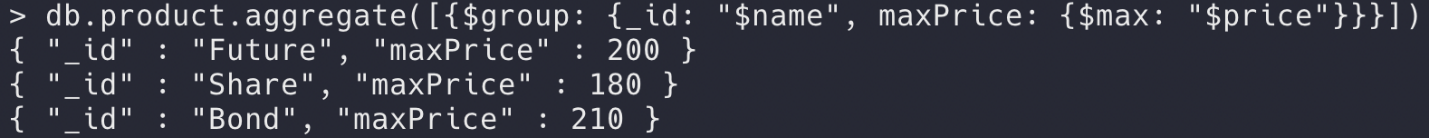
**3. Для каждого офисса вывести сколько было произведено работы**

**db.work.aggregate([{$group: {\_id: "$officeID", "amount": {$sum: "$amount"}}}])**



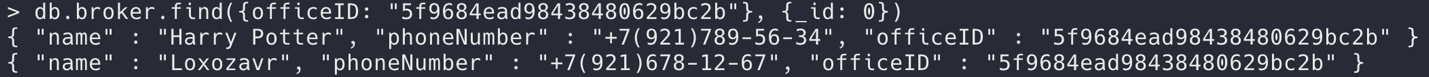
**4. Для каждого типа товара вывести самую высокую цену**

**db.product.aggregate([{$group: {\_id: "$name", maxPrice: {$max: "$price"}}}])**



**5. Вывести всех брокеров заданной фирмы**

**db.broker.find({officeID: "5f9684ead98438480629bc2b"}, {\_id: 0})**



**6. Вывести партии, в которых есть хоть один продукт с количество больше 30**

**db.consignment.find({products: {$elemMatch: {amount: {$gte: 30}}}})**



**ВЫВОД**

**В лабораторной работе №7 были получены практические навыки создания базы данных в MongoDB, заполнение их рабочими данными и выполнение запросов.**