

Федеральное государственное автономное образовательное учреждение
высшего образования
"Санкт-Петербургский национальный исследовательский университет информационных
технологий, механики и оптики"

Отчетный лист
по предмету "Базы данных"

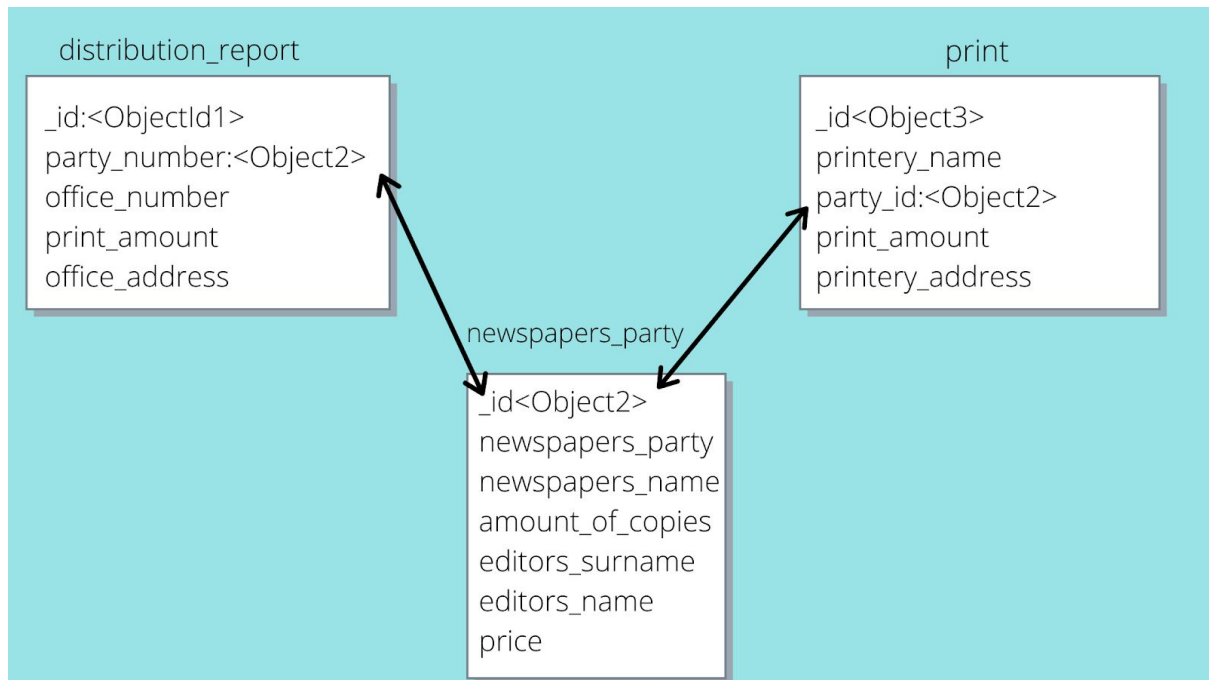
Лабораторная работа № 6
Знакомство с MongoDB

Выполнила: Кустова Екатерина, группа К3243

Проверил: Говоров А.И.

Модель базы данных

После уменьшения размера, модель реляционной базы данных выглядит следующим образом:



То есть представляет из себя нормализованную модель с использованием ссылок между документами.

Создаём базу данных newspapers, работаем с ней:

```
> use newspapers
switched to db newspapers
```

Добавляем данные в таблицу newspapers_party

```
> db.newspapers_party.insert({newspapers_party: 1, newspapers_name: 'News', amount_of_copies: 1000, editors_surname: 'Eddison', editors_name: 'Mark', price: 4});
WriteResult({ "nInserted" : 1 })
> db.newspapers_party.insert({newspapers_party: 2, newspapers_name: 'Herald', amount_of_copies: 3000, editors_surname: 'Huddson', editors_name: 'Finn', price: 3});
WriteResult({ "nInserted" : 1 })
> db.newspapers_party.insert({newspapers_party: 3, newspapers_name: 'Whistler', amount_of_copies: 1000, editors_surname: 'Hummel', editors_name: 'Kurt', price: 4});
WriteResult({ "nInserted" : 1 })
> db.newspapers_party.insert({newspapers_party: 4, newspapers_name: 'MusicNews', amount_of_copies: 2000, editors_surname: 'Roberts', editors_name: 'Agata', price: 5});
WriteResult({ "nInserted" : 1 })
> db.newspapers_party.insert({newspapers_party: 5, newspapers_name: 'ScienceNews', amount_of_copies: 2000, editors_surname: 'Mangold', editors_name: 'Maya', price: 5});
WriteResult({ "nInserted" : 1 })
```

И в таблицы distribution_report и print аналогичным образом.

Проверяем добавленные данные:

```
print(reson((inserted - 1)))
> db.distribution_report.find()
{"_id": ObjectId("5ef4f0ca87a2b2f7fa3474f"), "party_number": ObjectId("5ef4d315d87a2b2f7fa34745"), "office_number": 1, "print_amount": 200, "office_address": "HallamStreet1"}
{"_id": ObjectId("5ef50021d87a2b2f7fa34750"), "party_number": ObjectId("5ef4d315d87a2b2f7fa34745"), "office_number": 2, "print_amount": 800, "office_address": "RoseberyAvenue2"}
{"_id": ObjectId("5ef500ade87a2b2f7fa34751"), "party_number": ObjectId("5ef4d40cd87a2b2f7fa34746"), "office_number": 1, "print_amount": 500, "office_address": "HallamStreet1"}
{"_id": ObjectId("5ef50104d87a2b2f7fa34752"), "party_number": ObjectId("5ef4d40cd87a2b2f7fa34746"), "office_number": 2, "print_amount": 500, "office_address": "RoseberyAvenue2"}
{"_id": ObjectId("5ef50130d87a2b2f7fa34753"), "party_number": ObjectId("5ef4d40cd87a2b2f7fa34746"), "office_number": 3, "print_amount": 500, "office_address": "DevonshireStreet3"}
{"_id": ObjectId("5ef501d5d87a2b2f7fa34754"), "party_number": ObjectId("5ef4d40cd87a2b2f7fa34746"), "office_number": 4, "print_amount": 500, "office_address": "Tobbeystreet4"}
{"_id": ObjectId("5ef50211d87a2b2f7fa34755"), "party_number": ObjectId("5ef4d40cd87a2b2f7fa34746"), "office_number": 5, "print_amount": 1000, "office_address": "NottinghamStreet5"}
{"_id": ObjectId("5ef5031cd87a2b2f7fa34756"), "party_number": ObjectId("5ef4d4d6d87a2b2f7fa34747"), "office_number": 1, "print_amount": 500, "office_address": "HallamStreet1"}
{"_id": ObjectId("5ef5035cd87a2b2f7fa34757"), "party_number": ObjectId("5ef4d4d6d87a2b2f7fa34747"), "office_number": 2, "print_amount": 500, "office_address": "RoseberyAvenue2"}
{"_id": ObjectId("5ef5036fd87a2b2f7fa34758"), "party_number": ObjectId("5ef4d5b2d87a2b2f7fa34748"), "office_number": 3, "print_amount": 1000, "office_address": "DevonshireStreet3"}
{"_id": ObjectId("5ef50363d87a2b2f7fa34759"), "party_number": ObjectId("5ef4d5b2d87a2b2f7fa34748"), "office_number": 4, "print_amount": 1000, "office_address": "Tobbeystreet4"}
{"_id": ObjectId("5ef50739d87a2b2f7fa3475a"), "party_number": ObjectId("5ef4d702d87a2b2f7fa34749"), "office_number": 4, "print_amount": 1000, "office_address": "Tobbeystreet4"}
{"_id": ObjectId("5ef5075bd87a2b2f7fa3475b"), "party_number": ObjectId("5ef4d702d87a2b2f7fa34749"), "office_number": 5, "print_amount": 1000, "office_address": "NottinghamStreet5"}
> db.print.find()
{"_id": ObjectId("5ef4e160d87a2b2f7fa3474a"), "print_name": "Jamesons", "party_id": ObjectId("5ef4d315d87a2b2f7fa34745"), "print_amount": 500, "printery_address": "OxfordStreet1"}
{"_id": ObjectId("5ef4e59e87a2b2f7fa3474b"), "print_name": "Story", "party_id": ObjectId("5ef4d40cd87a2b2f7fa34746"), "print_amount": 1000, "printery_address": "CalmStreet3"}
{"_id": ObjectId("5ef4f64d87a2b2f7fa3474c"), "print_name": "Hicksons", "party_id": ObjectId("5ef4d4d6d87a2b2f7fa34747"), "print_amount": 500, "printery_address": "NottinghamStreet5"}
{"_id": ObjectId("5ef4f64d87a2b2f7fa3474d"), "print_name": "Story", "party_id": ObjectId("5ef4d5b2d87a2b2f7fa34748"), "print_amount": 2000, "printery_address": "FallsStreet2"}
{"_id": ObjectId("5ef4f866d87a2b2f7fa3474e"), "print_name": "Jamesons", "party_id": ObjectId("5ef4d702d87a2b2f7fa34749"), "print_amount": 2000, "printery_address": "OxfordStreet1"}
> db.newspapers_party.find()
{"_id": ObjectId("5ef4d315d87a2b2f7fa34745"), "newspapers_party": 1, "newspapers_name": "News", "amount_of_copies": 1000, "editors_surname": "Eddison", "editors_name": "Mark", "price": 4}
{"_id": ObjectId("5ef4d40cd87a2b2f7fa34746"), "newspapers_party": 2, "newspapers_name": "Herald", "amount_of_copies": 3000, "editors_surname": "Huddson", "editors_name": "Finn", "price": 3}
{"_id": ObjectId("5ef4d4d6d87a2b2f7fa34747"), "newspapers_party": 3, "newspapers_name": "Whistler", "amount_of_copies": 1000, "editors_surname": "Hummel", "editors_name": "Kurt", "price": 4}
{"_id": ObjectId("5ef4d5b2d87a2b2f7fa34748"), "newspapers_party": 4, "newspapers_name": "MusicNews", "amount_of_copies": 2000, "editors_surname": "Roberts", "editors_name": "Agata", "price": 5}
{"_id": ObjectId("5ef4d702d87a2b2f7fa34749"), "newspapers_party": 5, "newspapers_name": "ScienceNews", "amount_of_copies": 2000, "editors_surname": "Mangold", "editors_name": "Maya", "price": 5}
```

Выполнение запросов:

- 1) Вывести все адреса, где печатаются газеты партиями 500 экземпляров:

```
db.print.find({print_amount: 500}, {printery_address: 1});
```

```
> db.print.find({print_amount: 500}, {printery_address: 1});
{"_id": ObjectId("5ef4e160d87a2b2f7fa3474a"), "printery_address": "OxfordStreet1"}
{"_id": ObjectId("5ef4f5d0d87a2b2f7fa3474c"), "printery_address": "NottinghamStreet5"}
>
```

- 2) Вся информация о партиях газет, сортировка по названиям газет:

```
db.newspapers_party.find().sort({newspapers_name: 1});
```

```
2000-06-26 00:26:26.730+0300 E QUERY [js] uncaught exception: ReferenceError: printery_address is not defined
@shell:1:37
> db.print.find({print_amount: 500}, {printery_address: 1});
{"_id": ObjectId("5ef4e160d87a2b2f7fa3474a"), "printery_address": "OxfordStreet1"}
{"_id": ObjectId("5ef4f5d0d87a2b2f7fa3474c"), "printery_address": "NottinghamStreet5"}
> db.newspapers_party.find().sort({newspapers_name: 1});
{"_id": ObjectId("5ef4d40cd87a2b2f7fa34746"), "newspapers_party": 2, "newspapers_name": "Herald", "amount_of_copies": 3000, "editors_surname": "Huddson", "editors_name": "Finn", "price": 3}
{"_id": ObjectId("5ef4d5b2d87a2b2f7fa34748"), "newspapers_party": 4, "newspapers_name": "MusicNews", "amount_of_copies": 2000, "editors_surname": "Roberts", "editors_name": "Agata", "price": 5}
{"_id": ObjectId("5ef4d315d87a2b2f7fa34745"), "newspapers_party": 1, "newspapers_name": "News", "amount_of_copies": 1000, "editors_surname": "Eddison", "editors_name": "Mark", "price": 4}
{"_id": ObjectId("5ef4d702d87a2b2f7fa34749"), "newspapers_party": 5, "newspapers_name": "ScienceNews", "amount_of_copies": 2000, "editors_surname": "Mangold", "editors_name": "Maya", "price": 5}
{"_id": ObjectId("5ef4d4d6d87a2b2f7fa34747"), "newspapers_party": 3, "newspapers_name": "Whistler", "amount_of_copies": 1000, "editors_surname": "Hummel", "editors_name": "Kurt", "price": 4}
```

- 3) Узнаем названия газет и цены на них, стоимость которых превышает 3:

```
db.newspapers_party.find({price: {$gt: 3}}, {newspapers_name: 1, price: 1});
```

```
{ "_id": ObjectId("5ef4d702d87a2b2f7fa34749"), "newspapers_party": 5, "newspapers_name": "ScienceNews", "price": 5 }
> db.newspapers_party.find({price: {$gt: 3}}, {newspapers_name: 1, price: 1});
{"_id": ObjectId("5ef4d315d87a2b2f7fa34745"), "newspapers_name": "News", "price": 4}
{"_id": ObjectId("5ef4d4d6d87a2b2f7fa34747"), "newspapers_name": "Whistler", "price": 4}
{"_id": ObjectId("5ef4d5b2d87a2b2f7fa34748"), "newspapers_name": "MusicNews", "price": 5}
{"_id": ObjectId("5ef4d702d87a2b2f7fa34749"), "newspapers_name": "ScienceNews", "price": 5}
>
```

4) Сколько партий газет было доставлено в каждое отделение:

```
db.distribution_report.aggregate([{$unwind : "$office_number"}, {$group : {_id : "$office_number", count : { $sum : 1 }}}]);
```

```
@shari1:1:105
> db.distribution_report.aggregate([{$unwind : "$office_number"}, {$group : {_id : "$office_number", count : { $sum : 1 }}}]);
{ "_id" : 2, "count" : 3 }
{ "_id" : 1, "count" : 3 }
{ "_id" : 5, "count" : 2 }
{ "_id" : 4, "count" : 3 }
{ "_id" : 3, "count" : 2 }
>
```

5) Какова средняя цена газеты в каждой партии:

```
const map = function(){emit(this.newspapers_name, this.price)}
const reduce = function(key, values) {return(Array.sum(values) / values.length);}
db.newspapers_party.mapReduce(map, reduce, "result")
db.result.find()
```

```
> const map = function(){emit(this.newspapers_name, this.price)}
> const reduce = function(key, values) {return(Array.sum(values) / values.length);}
```

```
@shari1:1:105
> db.newspapers_party.mapReduce(map, reduce, "result")
{
  "result" : "result",
  "timeMillis" : 674,
  "counts" : {
    "input" : 5,
    "emit" : 5,
    "reduce" : 0,
    "output" : 5
  },
  "ok" : 1
}
> db.result.find()
{ "_id" : "Herald", "value" : 3 }
{ "_id" : "MusicNews", "value" : 5 }
{ "_id" : "News", "value" : 4 }
{ "_id" : "ScienceNews", "value" : 5 }
{ "_id" : "Whistler", "value" : 4 }
>
```

Вывод:

В ходе выполнения работы были освоены основы работы с MongoDB, а именно - проектирование нереляционных баз данных, их заполнение и выполнение запросов.