МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА №  43

ОТЧЕТ

ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

### Старший преподаватель                                                               Путилова Н. В.

должность, уч. Степень, звание   подпись, дата                    инициалы, фамилия

ОТЧЁТ О ЛАБОРАТОРНОЙ РАБОТЕ №12

Манипулирование данными в документной базе данных.

по курсу: ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ

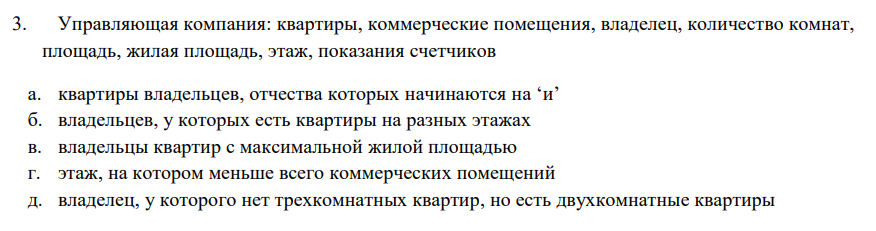
РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. 4136                                                                        Бобрович Н. С.

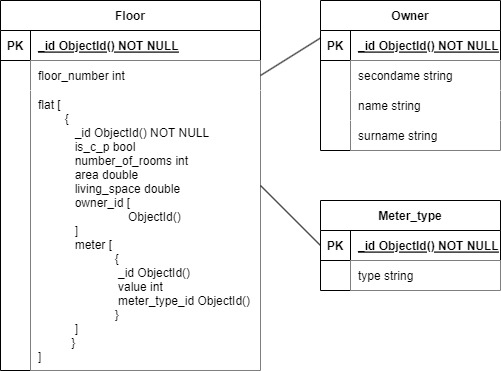
                                                                         подпись, дата                      инициалы, фамилия

Санкт-Петербург 2024

Вариант 3:



Физическая модель БД:



Обновление и удаление данных:

floor4\_id = ObjectId()

flat4\_id = ObjectId()

meter5\_id = ObjectId()

meter6\_id = ObjectId()

db.floor.insertOne(

{

"\_id": floor4\_id,

"floor\_number": 4,

"flats": [

{

"\_id": flat4\_id,

"number\_of\_rooms": 3,

"area": 150.2,

"living\_space": 144.8,

"is\_c\_p": false,

"owner\_id": [owner1\_id],

"meters": [

{

"\_id": meter5\_id,

"value": 220.2,

"meter\_type": meter\_type1\_id

},

{

"\_id": meter6\_id,

"value": 210.2,

"meter\_type": meter\_type2\_id

}

]

}

]

}

)

db.floor.updateOne({ floor\_number: 4 }, { $set: { floor\_number: 5 } })

db.floor.updateOne({ floor\_number: 5 }, { $set: { floor\_number: 4 } })

db.floor.updateOne({ floor\_number: 4 }, { $set: { "flats.0.is\_c\_p": true } })

db.floor.updateOne({ floor\_number: 4 }, { $unset: { flats: 0 } })

db.floor.deleteOne({ floor\_number: 4 })

Запросы на выборку из базы данных в соответствии с вариантом задания:

**а) текст запроса: квартиры владельцев, отчества которых начинаются на ‘и’**

db.floor.aggregate([

{ $unwind: "$flats" },

{ $lookup: { from: "owner", localField: "flats.owner\_id", foreignField: "\_id", as: "flats.owner\_id" } },

{ $unwind: "$flats.owner\_id" },

{ $match: { "flats.is\_c\_p": false, "flats.owner\_id.patronym": { $regex: '^И', $options: "i" } } },

{ $replaceWith: "$flats" }

])

**а) наборы данных:**

{

\_id: ObjectId('6644db146e9f43949ffd01bf'),

number\_of\_rooms: 3,

area: 150.2,

living\_space: 144.8,

is\_c\_p: false,

owner\_id: {

\_id: ObjectId('6644db146e9f43949ffd01ba'),

name: 'Иван',

surname: 'Иванов',

patronym: 'Иванович'

},

meters: [

{

\_id: ObjectId('6644db146e9f43949ffd01c2'),

value: 220.2,

meter\_type: ObjectId('6644db146e9f43949ffd01b8')

},

{

\_id: ObjectId('6644db146e9f43949ffd01c3'),

value: 210.2,

meter\_type: ObjectId('6644db146e9f43949ffd01b9')

}

]

}

{

\_id: ObjectId('6644db146e9f43949ffd01bf'),

number\_of\_rooms: 3,

area: 150.2,

living\_space: 144.8,

is\_c\_p: false,

owner\_id: {

\_id: ObjectId('6644db146e9f43949ffd01ba'),

name: 'Иван',

surname: 'Иванов',

patronym: 'Иванович'

},

meters: [

{

\_id: ObjectId('6644db146e9f43949ffd01c2'),

value: 220.2,

meter\_type: ObjectId('6644db146e9f43949ffd01b8')

},

{

\_id: ObjectId('6644db146e9f43949ffd01c3'),

value: 210.2,

meter\_type: ObjectId('6644db146e9f43949ffd01b9')

}

]

}

**б) текст запроса: владельцев, у которых есть квартиры на разных этажах**

db.floor.aggregate([

{ $unwind: "$flats" },

{ $unwind: "$flats.owner\_id" },

{

$group: {

\_id: "$flats.owner\_id",

floors: { $addToSet: "$floor\_number" }

}

},

{ $match: { floors: { $not: { $size: 1 } } } },

{

$lookup: {

from: "owner",

localField: "\_id",

foreignField: "\_id",

as: "owner"

}

},

{ $unwind: "$owner" },

{

$project: {

\_id: 0,

fullName: {

$concat: [

"$owner.surname",

" ",

"$owner.name",

" ",

"$owner.patronym"

]

},

floors: 1

}

}

])

**б) наборы данных:**

{

floors: [

2,

1

],

fullName: 'Иванов Иван Иванович'

}

**в) текст запроса: владельцы квартир с максимальной жилой площадью**

db.floor.aggregate([

{ $unwind: "$flats" },

{ $match: { "flats.owner\_id": { $exists: true } } },

{ $sort: { "flats.living\_space": -1 } },

{

$group: {

\_id: null,

flats: { $push: "$flats" },

maxLivingSpace: { $max: "$flats.living\_space" }

}

},

{

$project: {

\_id: 0,

flats: {

$filter: {

input: "$flats",

cond: { $eq: ["$$this.living\_space", "$maxLivingSpace"] }

}

}

}

},

{

$unwind: "$flats"

},

{

$lookup: {

from: "owner",

localField: "flats.owner\_id",

foreignField: "\_id",

as: "owner\_data"

}

},

{ $unwind: "$owner\_data" },

{

$project: {

\_id: 0,

owner\_name: {

$concat: [

"$owner\_data.name",

" ",

"$owner\_data.surname",

" ",

"$owner\_data.patronym"

]

},

living\_space: "$flats.living\_space"

}

}

])

**в) наборы данных:**

{

owner\_name: 'Иван Иванов Иванович',

living\_space: 144.8

}

**г) текст запроса: этаж, на котором меньше всего коммерческих помещений**

db.floor.aggregate([

{

$project: {

floor\_number: 1,

commPremisesCount: {

$sum: {

$cond: [

{ $isArray: "$flats" },

{

$sum: {

$map: {

input: "$flats",

as: "flat",

in: { $cond: [ "$$flat.is\_c\_p", 1, 0 ] }

}

}

},

0

]

}

}

}

},

{ $sort: { commPremisesCount: 1 } },

{ $group: {

\_id: "$commPremisesCount",

floor\_numbers: { $push: "$floor\_number" }

}},

{ $sort: { \_id: 1 } },

{ $limit: 1 },

{ $project: {

\_id: 0,

floor\_numbers: 1

}}

])

**г) наборы данных:**

{

floor\_numbers: [

1,

3

]

}

**д) текст запроса: владелец, у которого нет трёхкомнатных квартир, но есть двухкомнатные квартиры**

db.floor.aggregate([

{ $unwind: "$flats" },

{ $unwind: "$flats.owner\_id" },

{

$group: {

\_id: "$flats.owner\_id",

hasThreeRooms: {

$max: {

$cond: [

{ $eq: ["$flats.number\_of\_rooms", 3] },

1,

0

]

}

},

hasTwoRooms: {

$max: {

$cond: [

{ $eq: ["$flats.number\_of\_rooms", 2] },

1,

0

]

}

}

}

},

{ $match: { hasThreeRooms: 0, hasTwoRooms: 1 } },

{

$lookup: {

from: "owner",

localField: "\_id",

foreignField: "\_id",

as: "owner\_data"

}

},

{ $unwind: "$owner\_data" },

{

$project: {

\_id: 0,

owner\_name: {

$concat: [

"$owner\_data.name",

" ",

"$owner\_data.surname",

" ",

"$owner\_data.patronym"

]

}

}

}

])

**д) наборы данных:**

{

owner\_name: 'Анна Иванова Ивановна'

}