

КАФЕДРА №

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

должность, уч. степень, звание

подпись, дата

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №12

Манипулирование данными в документной базе данных.

по дисциплине: Проектирование баз данных

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР.

подпись, дата

инициалы, фамилия

Санкт-Петербург
2024

Цель работы

1)Привести пример обновления и удаления данных из документной базы данных, разработанной в работе 11. (по 2 примера: один из вложенного документа, другой из документа коллекции верхнего уровня)

2)Выполнить запросы на выборку из базы данных в соответствии с вариантом задания.

(Запрос на поиск подстроки.

Запрос на использование одного документа несколько раз.

Запрос с условием на агрегатную функцию.

Запрос получением агрегатной функции от агрегатной функции.

Запрос на разность 2 запросов)

Задание (13 вариант):

Калькулятор бюджета физического лица: пользователь калькулятора (разные уровни доступа), категория дохода (продажа, зарплата), категория расхода (еда, счета за КУ, здоровье...), статьи дохода и расхода, дата расхода/дохода

а. статьи категорий, которые относятся к спорту (содержат слово спорт)

б. месяц, в котором были разные статьи дохода

в. категория, по которой наибольшие расходы в текущем году

г. категория, по которой не было расходов в феврале

д. пользователь, добавивший наименьшее количество статей

Физическая модель:

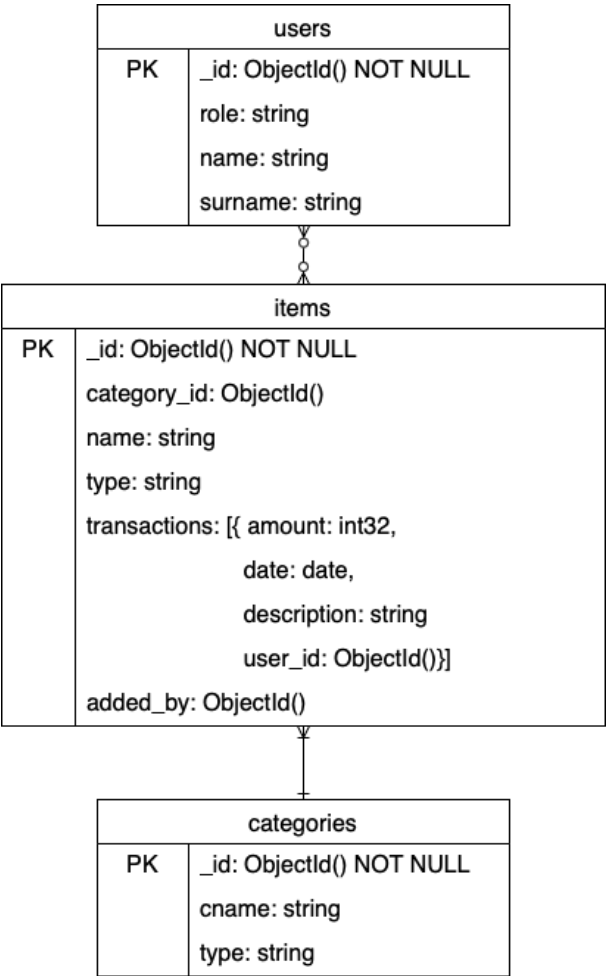


Таблица тестовых данных

Запрос	Данные, удовлетворяющие условиям	Данные, не удовлетворяющие условиям
а. статьи категорий, которые относятся к спорту (содержат слово спорт)	Спортзал, спортивный инвентарь	Мясо, напитки, зарплата за январь, премия за январь, подарок за январь, подарок за февраль
б. месяц, в котором были разные статьи дохода	1 (январь)	2 (февраль)
в. категория, по которой наибольшие расходы в текущем году	Спорт	Продукты, зарплата, подарки
г. категория, по которой не было расходов в феврале	Спорт	Продукты

д. пользователь, добавивший наименьшее количество статей	Иван Иванов	Петр Петров

Запросы и возвращаемые данные:

а. статьи категорий, которые относятся к спорту (содержат слово спорт)

```
db.categories.aggregate([
  { $match: { name: /спорт/i } },
  { $lookup: {
    from: "items",
    localField: "_id",
    foreignField: "category_id",
    as: "items"
  }},
  { $unwind: "$items" },
  { $project: { "items.name": 1, _id: 0 } }
])
```

Вывод:

```
< {
  items: {
    name: 'Спортзал'
  }
}
{
  items: {
    name: 'Спортивный инвентарь'
  }
}
```

б. месяц, в котором были разные статьи дохода

```
db.items.aggregate([
  { $match: { type: "доход" } },
  { $unwind: "$transactions" },
  { $group: {
    _id: { month: { $month: "$transactions.date" }, year: { $year: "$transactions.date" } },
    uniqueIncomeItems: { $addToSet: "$name" }
  }},
  { $project: {
    month: "$_id.month",
    year: "$_id.year",
    numberOfUniqueIncomeItems: { $size: "$uniqueIncomeItems" },
    _id: 0
  }},
  { $match: { numberOfUniqueIncomeItems: { $gt: 1 } } }
]);
```

Вывод:

```
< {
  month: 1,
  year: 2024,
  numberOfUniqueIncomeItems: 3
}
```

в. категория, по которой наибольшие расходы в текущем году

```
var currentDate = new Date();
var currentYear = currentDate.getFullYear();

db.items.aggregate([
  { $match: {
    type: "расход",
    "transactions.date": {
      $gte: new Date(currentYear, 0, 1),
      $lt: currentDate
    }
  } },
  { $unwind: "$transactions" },
  { $group: { _id: "$category_id", totalSpent: { $sum: "$transactions.amount" } } },
  { $sort: { totalSpent: -1 } },
  { $group: { _id: null, maxSpent: { $max: "$totalSpent" }, categories: { $push: { category_id:
"$_id", totalSpent: "$totalSpent" } } } },
  { $unwind: "$categories" },
  { $match: { $expr: { $eq: ["$categories.totalSpent", "$maxSpent"] } } }, // Исправление здесь
  { $lookup: {
    from: "categories",
    localField: "categories.category_id",
    foreignField: "_id",
    as: "categoryInfo"
  } },
  { $unwind: "$categoryInfo" },
  { $project: { _id: 0, totalSpent: "$categories.totalSpent", "categoryInfo.name": 1,
"categoryInfo.type": 1 } }
]);
```

Вывод:

```
< {
  totalSpent: 18000,
  category: {
    name: 'Спорт',
    type: 'расход'
  }
}
```

г. категория, по которой не было расходов в феврале

```
db.categories.aggregate([
  {
    $lookup: {
      from: "items",
      localField: "_id",
      foreignField: "category_id",
      as: "items"
    }
  },
  {
    $project: {
      name: 1,
      type: 1,
      hasFebTransactions: {
        $anyElementTrue: {
          $map: {
            input: "$items",
            as: "item",
            in: {
              $and: [
                { $eq: ["$$item.type", "расход"] },
                { $eq: [{ $size: { $filter: { input: "$$item.transactions", as: "transaction", cond: { $eq:
[ { $month: "$$transaction.date" }, 2] } } } }, 0] }
              ]
            }
          }
        }
      }
    }
  },
  {
    $group: {
      _id: "$name",
      totalSpent: { $sum: "$hasFebTransactions" }
    }
  },
  {
    $sort: { totalSpent: -1 }
  },
  {
    $limit: 1
  }
])
```

```
{
  $project: { name: 1, _id: 0 }
}
])
```

Вывод:

```
< {
  name: 'Спорт'
}
```

д. пользователь, добавивший наименьшее количество статей

```
db.items.aggregate([
  { $group: { _id: "$added_by", itemCount: { $sum: 1 } } },
  { $group: { _id: "$itemCount", users: { $push: "$_id" } } },
  { $sort: { "_id": 1 } },
  { $limit: 1 },
  { $unwind: "$users" },
  { $lookup: {
    from: "users",
    localField: "users",
    foreignField: "_id",
    as: "user"
  }},
  { $unwind: "$user" },
  { $project: { _id: 0, name: "$user.name", surname: "$user.surname", itemCount: "$_id" } }
])
```

Вывод:


```
< {
  user: {
    role: 'admin',
    name: 'Иван',
    surname: 'Иванов'
  }
}
```

Обновление данных:

Обновление вложенного документа (изменение описания транзакции):

До:

```
{
  _id: ObjectId('65dd25de325f1b9354de3d2a')
  category_id: ObjectId('65dd1be8325f1b9354de3d1d')
  name: "Мясо"
  type: "расход"
  transactions: Array (1)
    0: Object
      amount: 700
      date: 2024-02-01T00:00:00.000+00:00
      description: "Покупка курицы"
      user_id: ObjectId('65dd0e91325f1b9354de3d1a')
      added_by: ObjectId('65dd0e6a325f1b9354de3d17')
```

```
db.items.updateOne(
  { "transactions.description": "Покупка курицы" },
  { $set: { "transactions.$.description": "Покупка куриного филе" } }
)
```

Вывод:

```
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

После:

```
_id: ObjectId('65dd25de325f1b9354de3d2a')
category_id: ObjectId('65dd1be8325f1b9354de3d1d')
name: "Мясо"
type: "расход"
▼ transactions: Array (1)
  ▼ 0: Object
    amount: 700
    date: 2024-02-01T00:00:00.000+00:00
    description: "Покупка куриного филе"
    user_id: ObjectId('65dd0e91325f1b9354de3d1a')
    added_by: ObjectId('65dd0e6a325f1b9354de3d17')
```

Возвращаем данные:

```
db.items.updateOne(
  { "transactions.description": "Покупка куриного филе" },
  { $set: { "transactions.$.description": "Покупка курицы" } }
)
```

Обновление документа верхнего уровня (изменение имени пользователя):

До:

```
_id: ObjectId('65dd0e91325f1b9354de3d18')
role: "user"
name: "Алексей"
surname: "Алексеев"
```

```
db.users.updateOne(
  { name: "Алексей", surname: "Алексеев" },
  { $set: { name: "Александр" } }
)
```

Вывод:

```
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

После:

```
_id: ObjectId('65dd0e91325f1b9354de3d18')
role : "user"
name : "Александр"
surname : "Алексеев"
```

Возвращаем данные:

```
db.users.updateOne(
  { name: "Александр", surname: "Алексеев" },
  { $set: { name: "Алексей" } }
)
```

Удаление данных:

Удаление вложенного документа (удаление конкретной транзакции):

Сначала добавим тестовый документ:

До:

```
_id: ObjectId('65dd25de325f1b9354de3d2b')
category_id : ObjectId('65dd1be8325f1b9354de3d1d')
name : "Напитки"
type : "расход"
▼ transactions : Array (1)
  ▼ 0: Object
    amount : 300
    date : 2024-03-01T00:00:00.000+00:00
    description : "Покупка сока"
    user_id : ObjectId('65dd0e91325f1b9354de3d1a')
    added_by : ObjectId('65dd0e6a325f1b9354de3d17')
```

```
db.items.updateOne(
  { name: "Напитки" },
  { $push: { transactions: { amount: 100, date: new Date(), description: "Тестовая покупка",
user_id: ObjectId() } } }
)
```

Вывод:

```
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

После:

```
_id: ObjectId('65dd25de325f1b9354de3d2b')
category_id: ObjectId('65dd1be8325f1b9354de3d1d')
name: "Напитки"
type: "расход"
▼ transactions: Array (2)
  ▼ 0: Object
    amount: 300
    date: 2024-03-01T00:00:00.000+00:00
    description: "Покупка сока"
    user_id: ObjectId('65dd0e91325f1b9354de3d1a')
  ▼ 1: Object
    amount: 100
    date: 2024-03-03T13:42:00.993+00:00
    description: "Тестовая покупка"
    user_id: ObjectId('65e47e287940d9593b23e0a3')
added_by: ObjectId('65dd0e6a325f1b9354de3d17')
```

Удаление:

```
db.items.updateOne(
  { name: "Напитки" },
  { $pull: { transactions: { description: "Тестовая покупка" } } }
)
```

Вывод:

```
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

После:

```
_id: ObjectId('65dd25de325f1b9354de3d2b')
category_id: ObjectId('65dd1be8325f1b9354de3d1d')
name: "Напитки"
type: "расход"
▼ transactions: Array (1)
  ▼ 0: Object
    amount: 300
    date: 2024-03-01T00:00:00.000+00:00
    description: "Покупка сока"
    user_id: ObjectId('65dd0e91325f1b9354de3d1a')
  added_by: ObjectId('65dd0e6a325f1b9354de3d17')
```

Удаление документа из коллекции верхнего уровня (удаление категории):

Сначала добавим тестовый документ:

До:

```
_id: ObjectId('65dd1be8325f1b9354de3d1c')
name: "Спорт"
type: "расход"
```

```
_id: ObjectId('65dd1be8325f1b9354de3d1d')
name: "Продукты"
type: "расход"
```

```
_id: ObjectId('65dd1be8325f1b9354de3d1e')
name: "Зарплата"
type: "доход"
```

```
_id: ObjectId('65dd1be8325f1b9354de3d1f')
name: "Подарки"
type: "доход"
```

```
db.categories.insertOne({ name: "Тестовая категория", type: "расход" })
```

Вывод:

```
< {
  acknowledged: true,
  insertedId: ObjectId('65e47f037940d9593b23e0a4')
}
```

После:

<pre>_id: ObjectId('65dd1be8325f1b9354de3d1c') name: "Спорт" type: "расход"</pre>
<pre>_id: ObjectId('65dd1be8325f1b9354de3d1d') name: "Продукты" type: "расход"</pre>
<pre>_id: ObjectId('65dd1be8325f1b9354de3d1e') name: "Зарплата" type: "доход"</pre>
<pre>_id: ObjectId('65dd1be8325f1b9354de3d1f') name: "Подарки" type: "доход"</pre>
<pre>_id: ObjectId('65e47f037940d9593b23e0a4') name: "Тестовая категория" type: "расход"</pre>

Удаление:

```
db.categories.deleteOne({ name: "Тестовая категория" })
```

Вывод:

```
< {
  acknowledged: true,
  deletedCount: 1
}
```

После:

```
_id: ObjectId('65dd1be8325f1b9354de3d1c')  
name : "Спорт"  
type : "расход"
```

```
_id: ObjectId('65dd1be8325f1b9354de3d1d')  
name : "Продукты"  
type : "расход"
```

```
_id: ObjectId('65dd1be8325f1b9354de3d1e')  
name : "Зарплата"  
type : "доход"
```

```
_id: ObjectId('65dd1be8325f1b9354de3d1f')  
name : "Подарки"  
type : "доход"
```