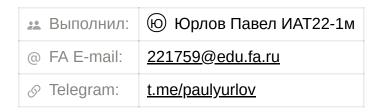
Lab #3

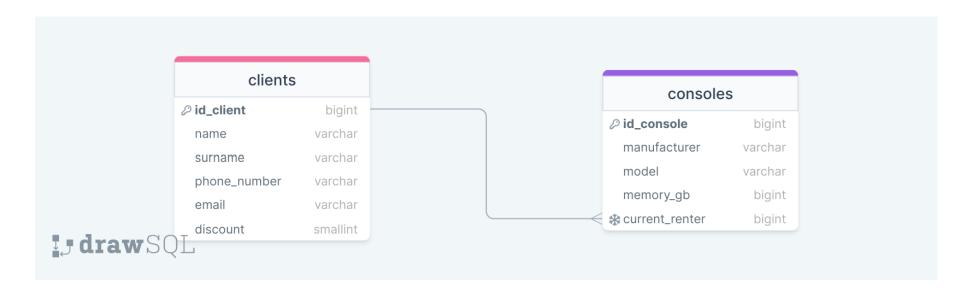


Задачи лабораторной:

- 1. Создать схему РБД в соответсвии с вашим вариантом
- 2. Создать БД по созданной вами схеме в любой СУБД
- 3. Заполнить базу данных произвольными данными
- 4. Представить созданную на предыдущем шаге БД в нереляционном виде, записанную в JSON

Ход выполнения работы:

1. Составить схему РБД в соответсвии с вашим вариантом



2. Создать БД по созданной схеме в любой СУБД

```
postgres@kali:/home/kali$ createdb RentConsole
could not change directory to "/home/kali": Permission denied
postgres@kali:/home/kali$ psql RentConsole
could not change directory to "/home/kali": Permission denied
psql (15.1 (Debian 15.1-1))
Type "help" for help.

RentConsole=# []
```

```
Q : 0 ×
 \blacksquare
                                          kali@kali: ~
RentConsole=# CREATE TABLE "clients"(
    "id_client" BIGINT NOT NULL,
    "name" VARCHAR(255) NOT NULL,
    "surname" VARCHAR(255) NOT NULL,
    "phone_number" VARCHAR(255) NOT NULL,
    "email" VARCHAR(255) NOT NULL,
    "discount" SMALLINT NOT NULL
CREATE TABLE
RentConsole=# ALTER TABLE
    "clients" ADD PRIMARY KEY("id_client");
ALTER TABLE
RentConsole=# CREATE TABLE "consoles"(
    "id_console" BIGINT NOT NULL,
    "manufacturer" VARCHAR(255) NOT NULL,
    "model" VARCHAR(255) NOT NULL,
    "memory_gb" BIGINT NOT NULL,
    "current_renter" BIGINT
CREATE TABLE
RentConsole=# ALTER TABLE
    "consoles" ADD PRIMARY KEY("id_console");
ALTER TABLE
RentConsole=# ALTER TABLE
   "consoles" ADD CONSTRAINT "consoles_current_renter_unique" UNIQUE("current_renter");
ALTER TABLE
RentConsole=# ALTER TABLE
    "consoles" ADD CONSTRAINT "consoles_current_renter_foreign" FOREIGN KEY("current_renter")
 REFERENCES "clients"("id_client");
ALTER TABLE
RentConsole=#
```

3. Заполнить базу данных произвольными данными

Lab #3 2

```
: 008
  \blacksquare
                                                 kali@kali: ~
                                                                                         Q
RentConsole=# INSERT INTO clients VALUES (1, 'Ivan', 'Ivanov', '+7123456789', 'ivan@ivanov.ru', 0);
INSERT 0 1
RentConsole=# INSERT INTO clients VALUES (2, 'Peter', 'Petrov', '+7987654321', 'pete@petrov.ru', 20);
RentConsole=# INSERT INTO consoles VALUES (1, 'Microsoft', 'Xbox X', 256, 1);
INSERT 0 1
RentConsole=# INSERT INTO consoles VALUES (2, 'Sony', 'PS4', 512, NULL);
INSERT 0 1
RentConsole=#
```

4. Представить созданную на предыдущем шаге БД в нереляционном виде, записанную в JSON

```
// clients //
[
_id: ObjectId ("1"),
"name": "Ivan",
"surname": "Ivanov",
"phone_number": "+7123456789",
"email": "ivan@ivanov.ru",
"discount": 0
},
_id: ObjectId ("2"),
"name": "Peter",
"surname": "Petrov",
"phone_number": "+7987654321",
"email": "pete@petrov.ru",
"discount": 20
}
]
// consoles //
[
_id: ObjectId ("1"),
"manufacturer": "Microsoft",
"model": "Xbox X",
"memory_gb": 256,
"current_renter": ObjectId("1")
},
{
_id: ObjectId ("2"),
"manufacturer": "Sony",
"model": "PS4",
"memory_gb": 512,
"current_renter": null
}
]
```

Lab #3 3

Теоретические вопросы

1. Дайте определения терминам JSON и BSON.

JSON - текстовый формат обмена данными, основанный на JavaScript.

BSON - бинарная версия JSON

- 2. На каких структурах строится JSON?
- Набор пар ключ/значение
- пронумерованный набор значений
- 3. Какие формы представления данных используются в JSON?

Объект - неупорядоченное множество пар имя/значение, заключенные в скобки {}

Массив - это множество значений, имеющих порядковые номера

Строка - это упорядоченное множество из нуля и более символов Unicode

4. Возможно ли ссылаться из одних документов MongoDB на другие? Если возможно, то какие механизмы используются для этого?

Возможно, путем сохранения в докуцменте указания на ObjectId

5. Возможно ли в MongoDB использовать массив документов?

Да

6. Существует ли возможность в MongoDB использовать вложенные документы?

Да