

**Министерство образования и науки РФ  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Заполярный государственный университет имени Н.М.  
Федоровского» Кафедра Информационных систем и  
технологий**

**Отчет: Лабораторная работа №1**

**Тема: Проектирование реляционной базы данных PostgreSQL**

**По предмету: Базы данных**

**Выполнил:**

Серебряков М. А.

**Группы:**

ИС – 21

**Проверил:**

Беляев И.С.

**Норильск 2023г**

**Цель работы:** Получение практических навыков проектирования реляционных баз данных и работы в CASE-средстве.

**Задание:** спроектировать согласно варианту реляционную базу данных, используя методологию IDEF1X, для этого необходимо:

- построить ER-модель (логическую и физическую) с помощью CASE-средства, например dbdiagram.io или visual-paradigm или Lucidchart (не менее 7 сущностей),
- описать ограничения целостности (CONSTRAINT, UNIQUE)
- вывести программный код создания БД на языке целевой СУБД (прямое проектирование),
- перенести код в СУБД (выполнить),

### **Описание**

В данном задании спроектирована реляционная база данных по предметной области «Туроператор» которая включает в себя следующие 7 сущностей:

Экскурсии;

Страны;

Туры;

Отели;

Работники;

Бронирование;

Клиенты;

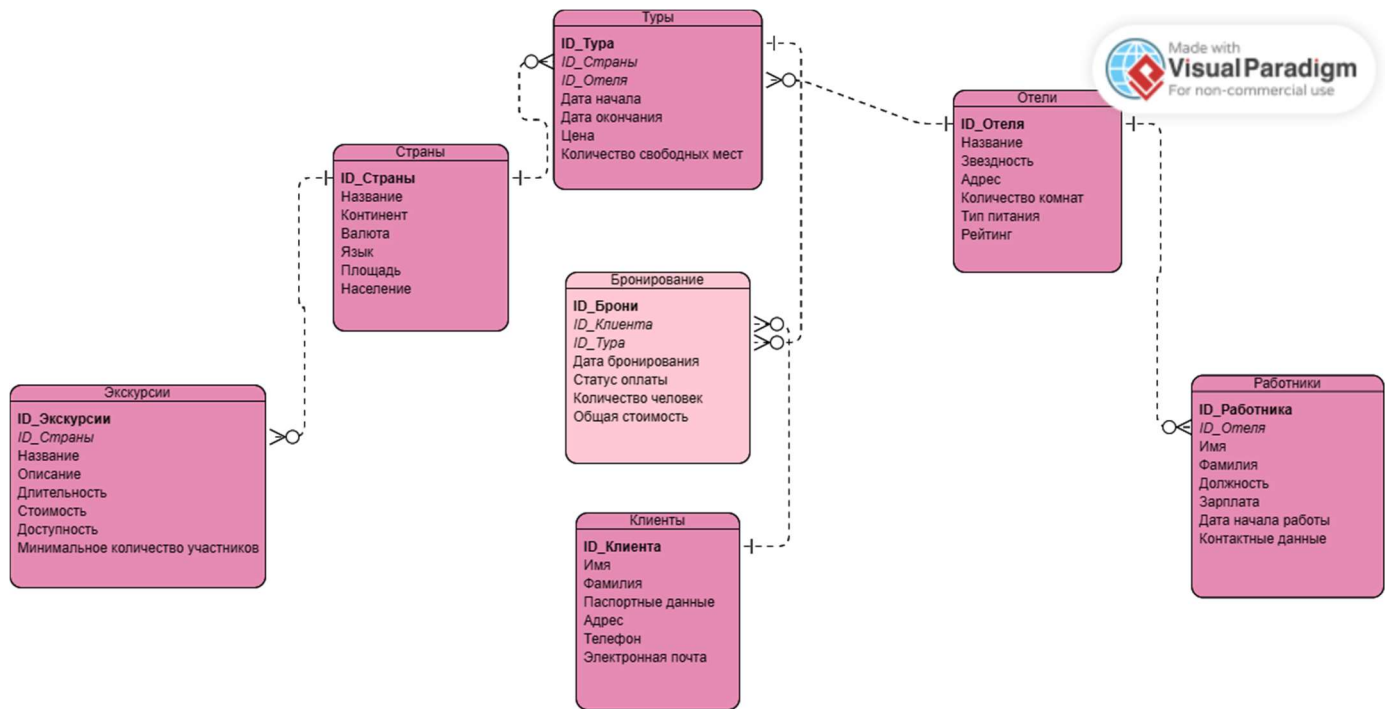


Рис 1. – Логическая модель базы данных в visual-paradigm

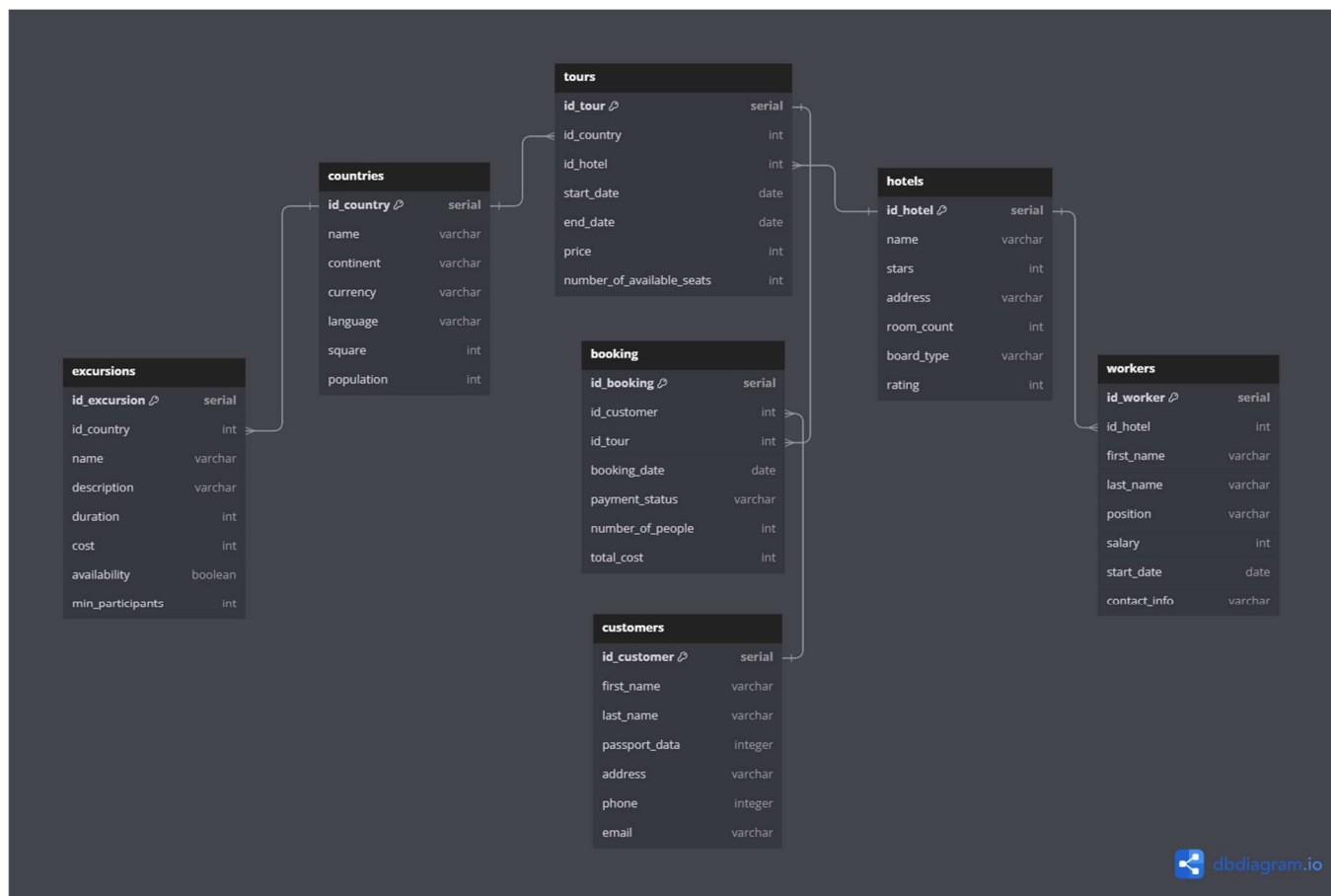


Рис 2. – Физическая модель базы данных в dbdiagram

```
CREATE TABLE "excursions" (  
  "id_excursion" serial PRIMARY KEY,  
  "id_country" int,  
  "name" varchar,  
  "description" varchar,  
  "duration" int,  
  "cost" int,  
  "availability" boolean,  
  "min_participants" int  
);
```

```
CREATE TABLE "countries" (  
  "id_country" serial PRIMARY KEY,  
  "name" varchar,  
  "continent" varchar,  
  "currency" varchar,  
  "language" varchar,  
  "square" int,  
  "population" int  
);
```

```
CREATE TABLE "tours" (  
  "id_tour" serial PRIMARY KEY,  
  "id_country" int,  
  "id_hotel" int,  
  "start_date" date,  
  "end_date" date,  
  "price" int,  
  "number_of_available_seats" int  
);
```

```
CREATE TABLE "hotels" (  
  "id_hotel" serial PRIMARY KEY,  
  "name" varchar,  
  "stars" int,  
  "address" varchar,  
  "room_count" int,  
  "board_type" varchar,  
  "rating" int  
);
```

```
CREATE TABLE "workers" (  
  "id_worker" serial PRIMARY KEY,  
  "id_hotel" int,  
  "first_name" varchar,  
  "last_name" varchar,  
  "position" varchar,
```

```
"salary" int,  
"start_date" date,  
"contact_info" varchar  
);
```

```
CREATE TABLE "booking" (  
  "id_booking" serial PRIMARY KEY,  
  "id_customer" int,  
  "id_tour" int,  
  "booking_date" date,  
  "payment_status" varchar,  
  "number_of_people" int,  
  "total_cost" int  
);
```

```
CREATE TABLE "customers" (  
  "id_customer" serial PRIMARY KEY,  
  "first_name" varchar,  
  "last_name" varchar,  
  "passport_data" integer,  
  "address" varchar,  
  "phone" varchar UNIQUE,  
  "email" varchar UNIQUE  
);
```

```
ALTER TABLE "excursions" ADD FOREIGN KEY ("id_country")  
REFERENCES "countries" ("id_country");
```

```
ALTER TABLE "tours" ADD FOREIGN KEY ("id_country")  
REFERENCES "countries" ("id_country");
```

```
ALTER TABLE "tours" ADD FOREIGN KEY ("id_hotel")  
REFERENCES "hotels" ("id_hotel");
```

```
ALTER TABLE "booking" ADD FOREIGN KEY ("id_tour")  
REFERENCES "tours" ("id_tour");
```

```
ALTER TABLE "booking" ADD FOREIGN KEY ("id_customer")  
REFERENCES "customers" ("id_customer");
```

```
ALTER TABLE "workers" ADD FOREIGN KEY ("id_hotel")  
REFERENCES "hotels" ("id_hotel");
```

Листинг программы №1 – на PostgreSQL

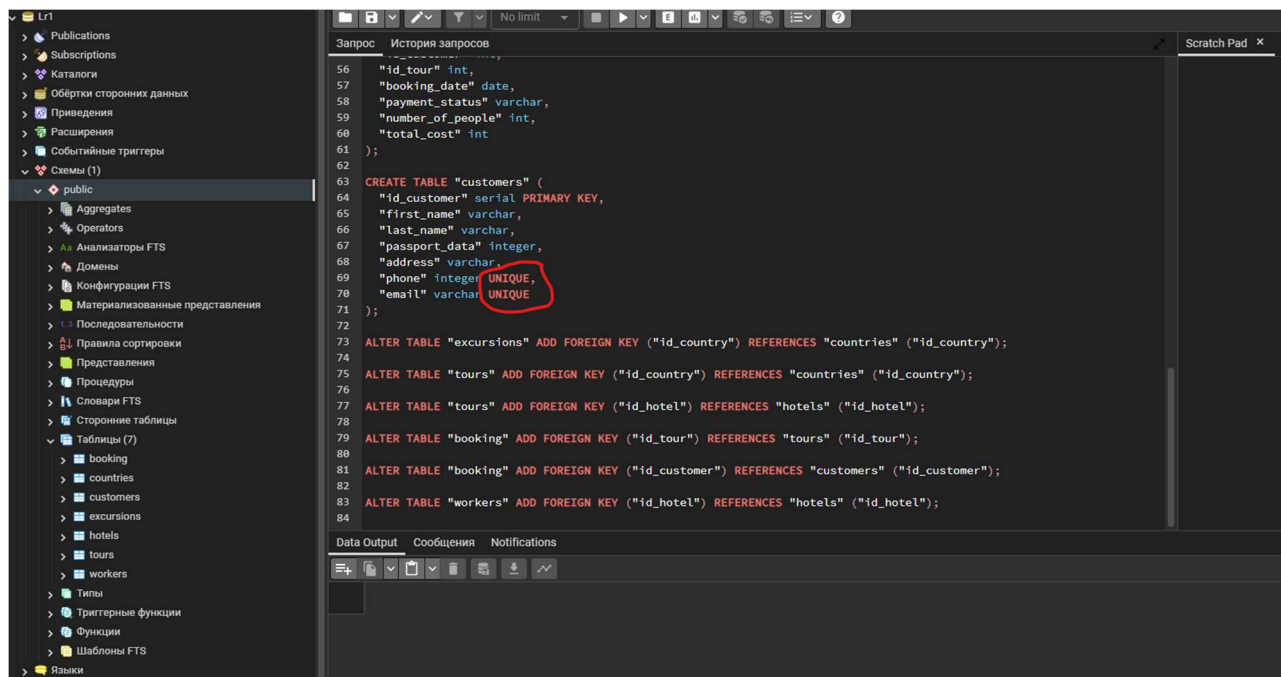


Рис 3. – Снимок экрана экспортированной базы в СУБД и использование ограничения UNIQUE