

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего
образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет

по лабораторной работе «Создание таблиц базы данных PostgreSQL.
Заполнение таблиц рабочими данными»

по дисциплине «**Базы данных**»

Автор: Акулов Алексей

Факультет: ФИКТ

Группа: K32391

Преподаватель: Говорова М.М.



Санкт-Петербург 2022

Цель работы: овладеть практическими навыками создания таблиц базы данных PostgreSQL 1X, заполнения их рабочими данными, резервного копирования и восстановления БД.

Оборудование: компьютерный класс.

Программное обеспечение: СУБД PostgreSQL 1X, pgAdmin 4.

Практическое задание:

1. Создать запросы и представления на выборку данных к базе данных PostgreSQL (согласно индивидуальному заданию, часть 2 и 3).
2. Составить 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов.
3. Изучить графическое представление запросов и просмотреть историю запросов.
4. Создать простой и составной индексы для двух произвольных запросов и сравнить время выполнения запросов без индексов и с индексами. Для получения плана запроса использовать команду EXPLAIN.

Вариант 6. БД «Пассажир»

Описание предметной области: Информационная система служит для продажи железнодорожных билетов. Билеты могут продаваться на текущие сутки или предварительно (не более чем за 45 суток). Цена билета при предварительной продаже снижается на 5%. Билет может быть приобретен в кассе или онлайн. Если билет приобретен в кассе, необходимо знать, в какой. Для каждой кассы известны номер и адрес. Кассы могут располагаться в различных населенных пунктах.

Поезда курсируют по расписанию, но могут назначаться дополнительные поезда на заданный период или определенные даты.

По всем промежуточным остановкам на маршруте известны название, тип населенного пункта, время прибытия, отправления, время стоянки.

БД должна содержать следующий минимальный набор сведений: Номер поезда. Название поезда. Тип поезда. Пункт назначения. Пункт назначения для проданного билета. Номер вагона. Тип вагона. Количество мест в вагоне. Цена билета. Дата отправления. Дата прибытия. Дата прибытия для пункта назначения проданного билета. Время отправления. Номер вагона в поезде. Номер билета. Место. Тип места. Фамилия пассажира. Имя пассажира. Отчество пассажира. Паспортные данные.

- I. Название модели – «База пассажиров»
- II. Состав реквизитов сущностей
 - a. Пассажир (Паспортные данные, Фамилия, Имя, Отчество, Телефон, Электронная почта)

- b. Билет (Номер, Пассажир, Пункт назначения, Пункт отправления, Место, Цена, Статус оплаты, Статус возврата, В кассе или онлайн)
- c. Место (Номер места, Вагон, Статус занятости)
- d. Вагон (Номер вагона, Номер в поезде, Поезд, Количество мест, Тип)
- e. Поезд (Номер поезда, Расписание, Дата отправления, Дата прибытия, Выполнение, Название)
- f. Расписание поездов (Номер маршрута, Время отправления, Время прибытия, Тип, Пункт отправления, Пункт прибытия, Периодичность)
- g. Остановка (Номер, Название, Тип, Маршрут)
- h. Остановка поезда (Время прибытия, Время стоянки, Время отправления)
- i. Касса (Номер кассы, Адрес, Населенный пункт)

Задание 2. Создать запросы:

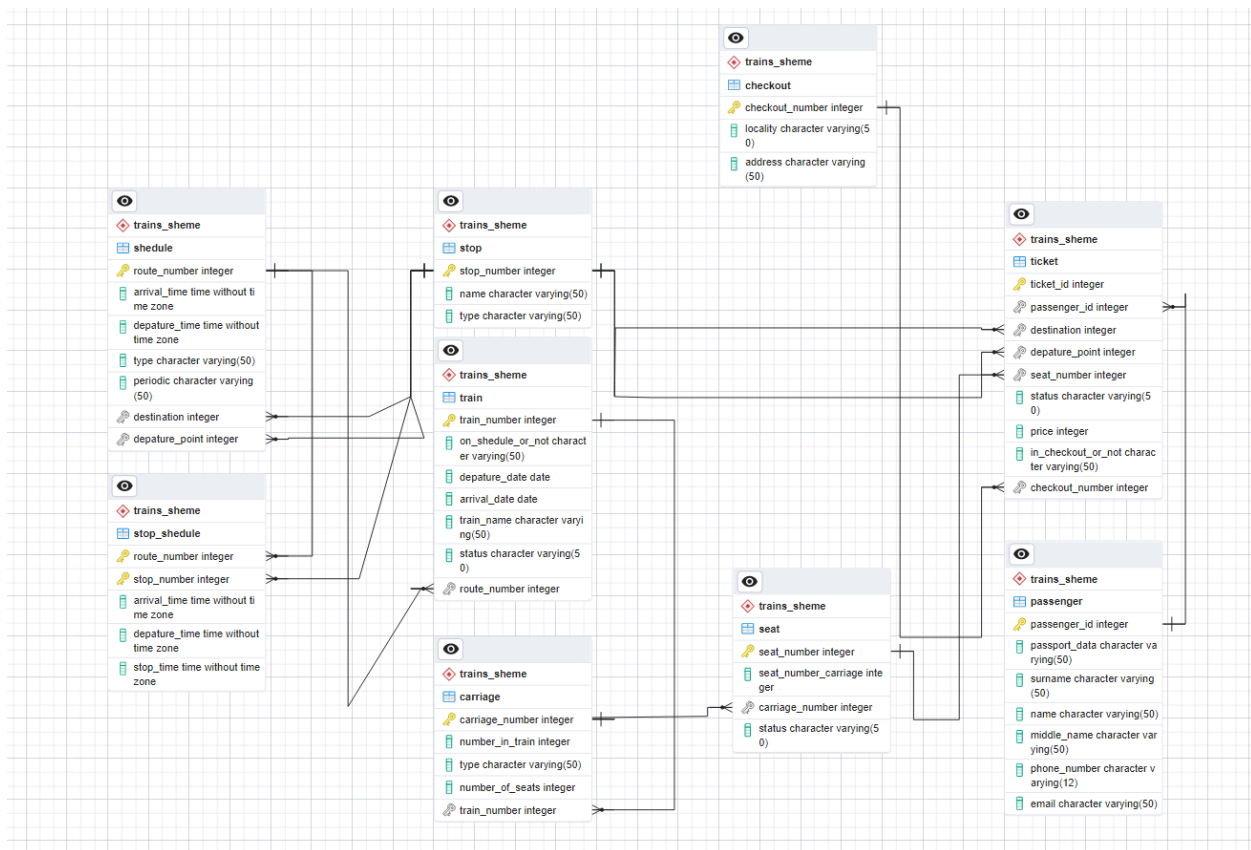
- Свободные места на все поезда, отправляющиеся с вокзала в течение следующих суток.
- Список пассажиров, отправившихся в Москву всеми рейсами за прошедшие сутки.
- Номера поездов, на которые проданы все билеты на следующие сутки.
- Свободные места в купейные вагоны всех рейсов до Москвы на текущие сутки.
- Выручка от продажи билетов на все поезда за прошедшие сутки.
- Общее количество билетов, проданных по всем направлениям в вагоны типа “СВ”.
- Номера и названия поездов, все вагоны которых были заполнены менее чем наполовину за прошедшие сутки.

Задание 3. Создать представление:

- для пассажиров о наличии свободных мест на заданный рейс;
- количество непроданных билетов на все поезда, формирующиеся за прошедшие сутки (номер поезда, тип вагона, количество).

Выполнение:

1. После перенесения всех таблиц из прошлого задания с помощью скрипта в SQL получается схема базы данных

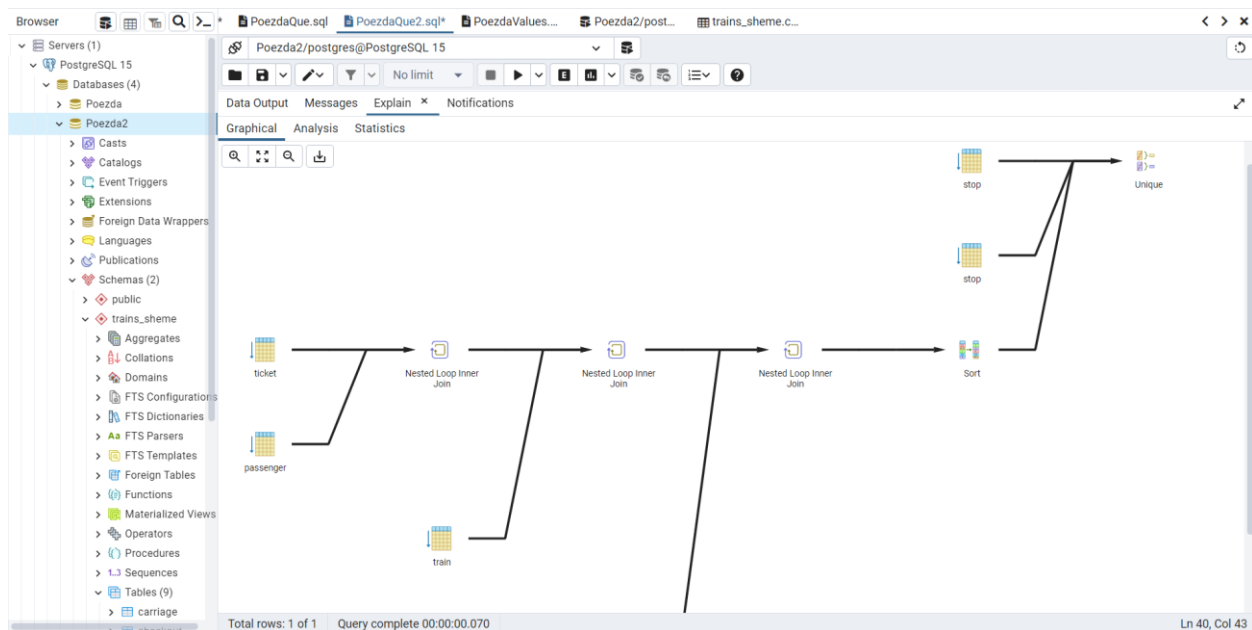


2. Далее делаем задания 2 и 3. Скрипт для запросов будет приложен отдельно. Здесь прикрепляю примеры ответов на последние два запроса

total_sv_tickets_sold		
bigint		
1	1	

	train_number [PK] integer	train_name character varying (50)
1	7	Train 7
2	9	Train 2

3. Также делаем три запроса с подзапросами, их скрипт также будет приложен к отчету
 4. Изучаем графическое представление запроса и смотрим историю



5. После этого проводим измерения с индексом и без. Файлы с результатами также прикрепляю отдельным файлом

Вывод:

PGAdmin позволяет хорошо работать с запросами всех типов. Также при формировании индекса ускоряется время работы