Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет инфокоммуникационных технологий

ЛАБОРАТОРНАЯ РАБОТА №2

Запросы на выборку и модификацию данных, представления и индексы в PostgreSQL

**Выполнил**: студент 3 курса ИКТ

группы К33401 Ф.И.О.: **Мамин И. И.**

**Проверила**: **Говорова Марина Михайловна**

Санкт-Петербург 2022

**Цель работы:** овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

**Оборудование:** компьютерный класс.

**Программное обеспечение:** СУБД PostgreSQL, 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.

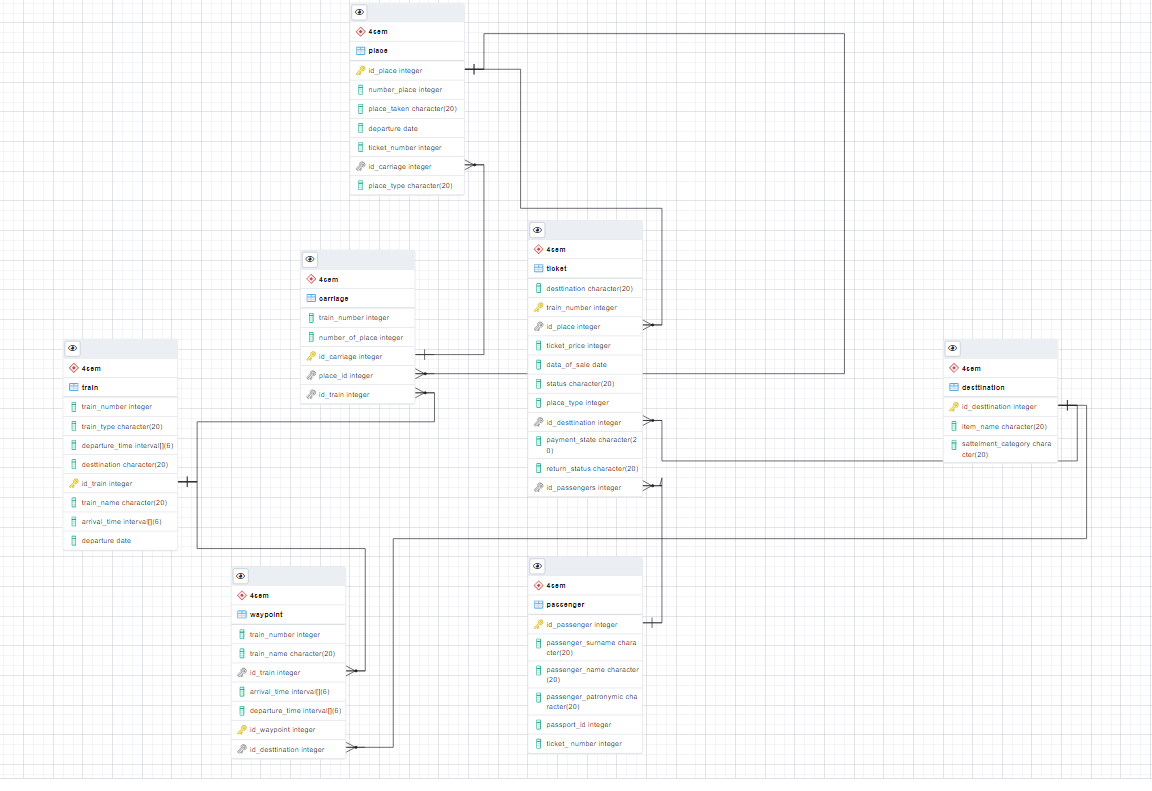
**Технология выполнения работы:**

**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 %.

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.

**Схема логической модели БД в нотации IDEF1X**

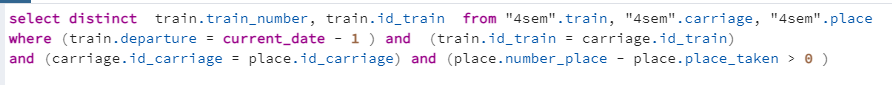


1) Свободные места на все поезда, отправляющиеся с вокзала в течение следующих суток.

select distinct train.train\_number, train.id\_train from "4sem".train, "4sem".carriage, "4sem".place

where (train.departure = current\_date - 1 ) and (train.id\_train = carriage.id\_train)

and (carriage.id\_carriage = place.id\_carriage) and (place.number\_place - place.place\_taken > 0 )



Изображение выглядит как стол

Автоматически созданное описание

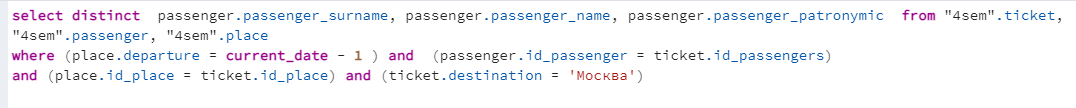
2) Список пассажиров, отправившихся в Москву всеми рейсами за прошедшие сутки.

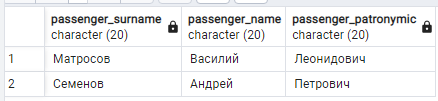
select distinct passenger.passenger\_surname, passenger.passenger\_name, passenger.passenger\_patronymic from "4sem".ticket,

"4sem".passenger, "4sem".place

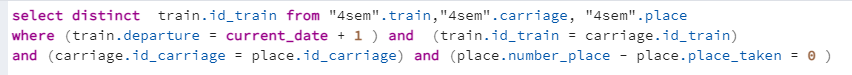
where (place.departure = current\_date - 1 ) and (passenger.id\_passenger = ticket.id\_passengers)

and (place.id\_place = ticket.id\_place) and (ticket.destination = 'Москва')





3) Номера поездов, на которые проданы все билеты на следующие сутки



Изображение выглядит как стол

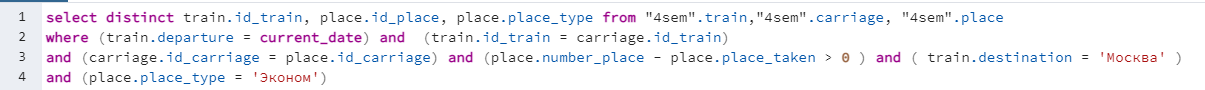
Автоматически созданное описание

select distinct train.id\_train from "4sem".train,"4sem".carriage, "4sem".place

where (train.departure = current\_date + 1 ) and (train.id\_train = carriage.id\_train)

and (carriage.id\_carriage = place.id\_carriage) and (place.number\_place - place.place\_taken = 0 )

4) Свободные места в купейные вагоны всех рейсов до Москвы на текущие сутки.



Изображение выглядит как стол

Автоматически созданное описание

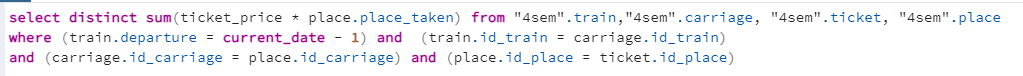
select train.id\_train, place.id\_place, place.place\_type from "4sem".train,"4sem".carriage, "4sem".place

where (train.departure = current\_date) and (train.id\_train = carriage.id\_train)

and (carriage.id\_carriage = place.id\_carriage) and (place.number\_place - place.place\_taken > 0 ) and ( train.destination = 'Москва' )

and (place.place\_type = 'Эконом')

5) Выручка от продажи билетов на все поезда за прошедшие сутки.



Изображение выглядит как стол

Автоматически созданное описание

select distinct sum(ticket\_price \* place.place\_taken) from "4sem".train,"4sem".carriage, "4sem".ticket, "4sem".place

where (train.departure = current\_date - 1) and (train.id\_train = carriage.id\_train)

and (carriage.id\_carriage = place.id\_carriage) and (place.id\_place = ticket.id\_place)

6) Общее количество билетов, проданных по всем направлениям в вагоны типа “ Скоростной”.



Изображение выглядит как стол

Автоматически созданное описание

select sum(place.place\_taken) from "4sem".train,"4sem".carriage, "4sem".place

where (train.id\_train = carriage.id\_train) and (carriage.id\_carriage = place.id\_carriage) and (train\_type = 'Скоростной')

7) Номера и названия поездов, все вагоны которых были заполнены менее чем наполовину за прошедшие сутки

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание

select train.train\_number, train.train\_name from "4sem".train,"4sem".carriage, "4sem".place

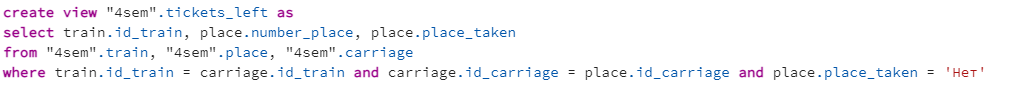
where (train.id\_train = carriage.id\_train) and (carriage.id\_carriage = place.id\_carriage)

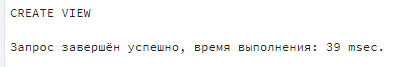
and (train.departure = current\_date - 1)

group by train.train\_number, train.train\_name

having (sum(place.place\_taken) < sum(place.number\_place / 2))

**Представления**





create view "4sem".tickets\_left as

select train.id\_train, place.number\_place, place.place\_taken

from "4sem".train, "4sem".place, "4sem".carriage

where train.id\_train = carriage.id\_train and carriage.id\_carriage = place.id\_carriage and place.place\_taken = 'Нет'

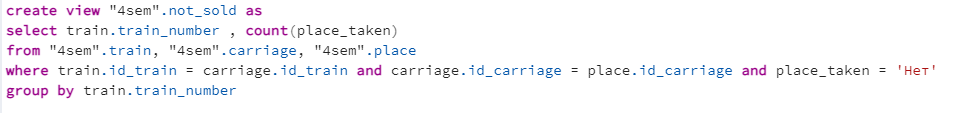


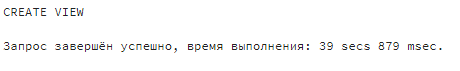
Изображение выглядит как стол

Автоматически созданное описание

select \* from "4sem".tickets\_left

**2)Количество непроданных билетов на все поезда, формирующиеся за прошедшие сутки (номер поезда, тип вагона, количество)**

****

****

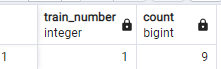
**create view "4sem".not\_sold as**

**select train.train\_number , count(place\_taken)**

**from "4sem".train, "4sem".carriage, "4sem".place**

**where train.id\_train = carriage.id\_train and carriage.id\_carriage = place.id\_carriage and place\_taken = 'Нет'**

**group by train.train\_number**

****

**Индексы**

1) Простой индекс

Изображение выглядит как стол

Автоматически созданное описание

select \* from “4sem”.place where place\_type = ‘Эконом’

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание

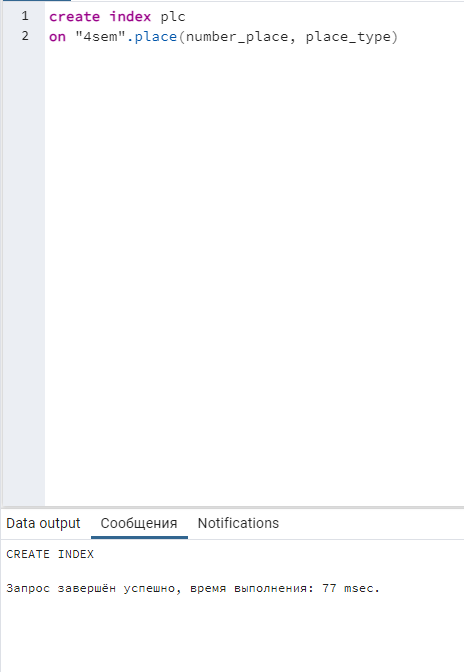
2) Сложный индекс

Изображение выглядит как стол

Автоматически созданное описание

select \* from "4sem".place

where place.place\_type = 'Эконом' and place.number\_place > 10



Изображение выглядит как стол

Автоматически созданное описание

**Запросы на модификацию данных**

1. insert

Изображение выглядит как текст

Автоматически созданное описание

1. update

Изображение выглядит как текст

Автоматически созданное описание

1. delete

Изображение выглядит как текст

Автоматически созданное описание

Вывод: **я** овладел практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.