

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

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.

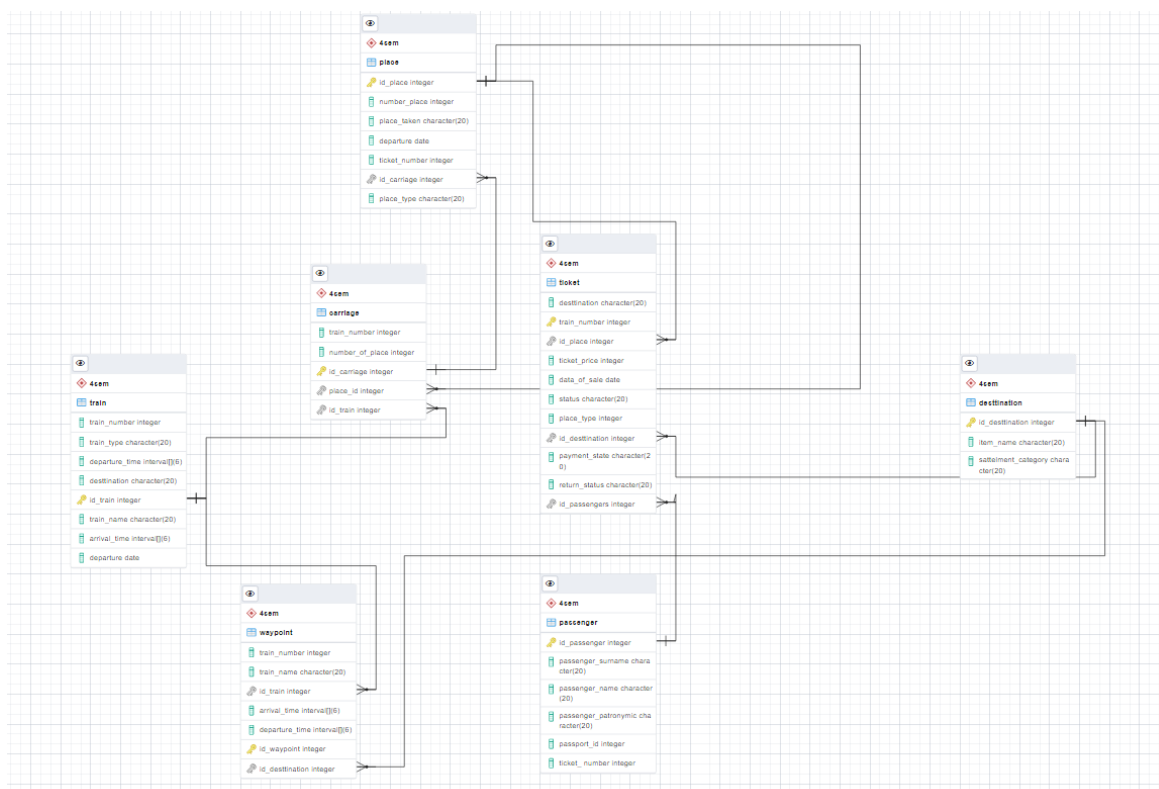
Технология выполнения работы:

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) Номера поездов, у которых остались билеты на сегодня

```
select distinct train.train_number, train.id_train 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.place_taken = 'Нет')
```

```
1 select distinct train.train_number, train.id_train from "4sem".train, "4sem".carriage, "4sem".place
2 where (train.departure = current_date ) and (train.id_train = carriage.id_train)
3 and (carriage.id_carriage = place.id_carriage) and (place.place_taken = 'Нет')
```

	train_number integer	id_train [PK] integer
1	1	1

```
select distinct train.train_number, train.id_train from "4sem".train, "4sem".carriage, "4sem".place
where (train.id_train = carriage.id_train) and (train.departure = current_date )
and (carriage.id_carriage = place.id_carriage) and (place.place_taken = 'Нет')
```

2) Список поездов со свободными местами до Москвы

```
select distinct train.train_number, train.departure from "4sem".train, "4sem".carriage, "4sem".place
where (train.id_train = carriage.id_train)
and (carriage.id_carriage = place.id_carriage) and (place.place_taken = 'Нет') and (train.destination = 'Москва')
```

	train_number integer	departure date
1	1	2022-10-03

3) Количество свободных мест эконом класса, в 1 вагоне поезда до Москвы сегодня

```
select count(id_place) from "4sem".place
where place.place_taken = 'Нет' and place_type = 'Эконом' and id_carriage = 1
```

	count bigint
1	3

```
select count(id_place) from "4sem".place
```

```
where place.place_taken = 'Нет' and place_type = 'Эконом' and id_carriage = 1
```

4) Количество вагонов в 1 поезде, в которых мест больше 35

```
select count(id_carriage) from "4sem".carriage
where id_train = 1 and number_of_place > 35
```

	count bigint	
1		5

```
select count(id_carriage) from "4sem".carriage
```

```
where id_train = 1 and number_of_place > 35
```

5) Номера билетов, купленных в бизнес классе в Санкт-Петербург

```
select place.ticket_number from "4sem".place, "4sem".ticket, "4sem".destination
where ticket.id_place = place.id_place and ticket.place_type = 'Бизнес' and ticket.id_destination = destination.id_destination
and destination.item_name = 'Санкт-Петербург'
```

	ticket_number integer	
1		8

```
select place.ticket_number from "4sem".place, "4sem".ticket, "4sem".destination
```

```
where ticket.id_place = place.id_place and ticket.place_type = 'Бизнес' and ticket.id_destination =
destination.id_destination
```

```
and destination.item_name = 'Санкт-Петербург'
```

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'
```

```
CREATE VIEW
```

```
Запрос завершён успешно, время выполнения: 39 мсек.
```

```
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'
```

```
select * from "4sem".tickets_left
```

	id_train integer	number_place integer	place_taken character (20)
1	1	19	Нет
2	1	12	Нет
3	1	15	Нет
4	1	15	Нет
5	1	10	Нет
6	1	2	Нет

```
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
```

CREATE VIEW

Запрос завершён успешно, время выполнения: 39 secs 879 msec.

```
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
```

	train_number integer	count bigint
1	1	9

Индексы

1) Простой индекс

1

2

3

```
select * from "4sem".place
where place_type = 'Эконом'
```

Data outputСообщенияNotifications

	id_place [PK] integer	number_place integer	departure date	ticket_number integer	id_carriage integer	place_type character (20)	place_taken integer
1	2	15	2022-10-03	2	1	Эконом	14
2	3	15	2022-10-03	3	1	Эконом	3
3	5	15	2022-10-03	5	1	Эконом	2
4	6	15	2022-10-03	6	1	Эконом	11
5	7	15	2022-10-03	7	1	Эконом	15
6	8	15	2022-10-03	8	1	Эконом	13

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

```
1 create index economy
2 on "4sem".place(place_type)
3 where place_type = 'Эконом'
```

Data output Сообщения Notifications

CREATE INDEX

Запрос завершён успешно, время выполнения: 2 min 17 secs.

```

1 select * from "4sem".place
2 where place_type = 'Эконом'

```

Data output Сообщения Notifications

	id_place [PK] integer	number_place integer	departure date	ticket_number integer	id_carriage integer	place_type character (20)	place_taken integer
1	2	15	2022-10-03	2	1	Эконом	14
2	3	15	2022-10-03	3	1	Эконом	3
3	5	15	2022-10-03	5	1	Эконом	2
4	6	15	2022-10-03	6	1	Эконом	11
5	7	15	2022-10-03	7	1	Эконом	15
6	8	15	2022-10-03	8	1	Эконом	13

2) Сложный индекс


```
1 select * from "4sem".place
2 where place.place_type = 'Эконом' and place.number_place > 10
```

Data output Сообщения Notifications

	id_place [PK] integer	number_place integer	departure date	ticket_number integer	id_carriage integer	place_type character (20)	place_taken integer
1	2	15	2022-10-03	2	1	Эконом	14
2	3	15	2022-10-03	3	1	Эконом	3
3	5	15	2022-10-03	5	1	Эконом	2
4	6	15	2022-10-03	6	1	Эконом	11
5	7	15	2022-10-03	7	1	Эконом	15
6	8	15	2022-10-03	8	1	Эконом	13

select * from "4sem".place

where place.place_type = 'Эконом' and place.number_place > 10

```
1 create index plc
2 on "4sem".place(number_place, place_type)
```

Data output Сообщения Notifications

CREATE INDEX

Запрос завершён успешно, время выполнения: 77 мсек.

```

1 select * from "4sem".place
2 where place.place_type = 'Эконом' and place.number_place > 10
3

```

Data output Сообщения Notifications

	id_place [PK] integer	number_place integer	departure date	ticket_number integer	id_carriage integer	place_type character (20)	place_taken integer
1	2	15	2022-10-03	2	1	Эконом	14
2	3	15	2022-10-03	3	1	Эконом	3
3	5	15	2022-10-03	5	1	Эконом	2
4	6	15	2022-10-03	6	1	Эконом	11
5	7	15	2022-10-03	7	1	Эконом	15
6	8	15	2022-10-03	8	1	Эконом	13

Запросы на модификацию данных

1) insert

```

1 insert into "4sem".destination
2 (id_destination,item_name,sattelment_category)
3 values (111,'Кемерово', (select sattelment_category from "4sem".destination where item_name = 'Екатеринбург'))

```

Data output **Сообщения** Notifications

INSERT 0 1

Запрос завершён успешно, время выполнения: 62 мсек.

2) update

```
1 update "4sem".destination set item_name = 'Псков'
2 where id_destination in
3 (select id_destination from "4sem".destination where item_name = 'Кемерово')
4
```

Data output Сообщения Notifications

UPDATE 1

Запрос завершён успешно, время выполнения: 1 min 6 secs.

3) delete

```
1 delete from "4sem".destination where id_destination in
2 (select id_destination from "4sem".destination where item_name = 'Псков')
```

Data output Сообщения Notifications

DELETE 1

Запрос завершён успешно, время выполнения: 29 secs 976 msec.

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