

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ**

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет «Инфокоммуникационных технологий»

Направление подготовки «45.03.04 Интеллектуальные системы в гуманитарной сфере»

О Т Ч Е Т

Тема задания: Лабораторная работа №4. SQL-запросы.

Выполнил:

Студент Садовщиков И.И. К3243

(Ф. И.О.)

Номер группы

Проверил:

Преподаватель Говоров А. И.

(Ф. И.О.)

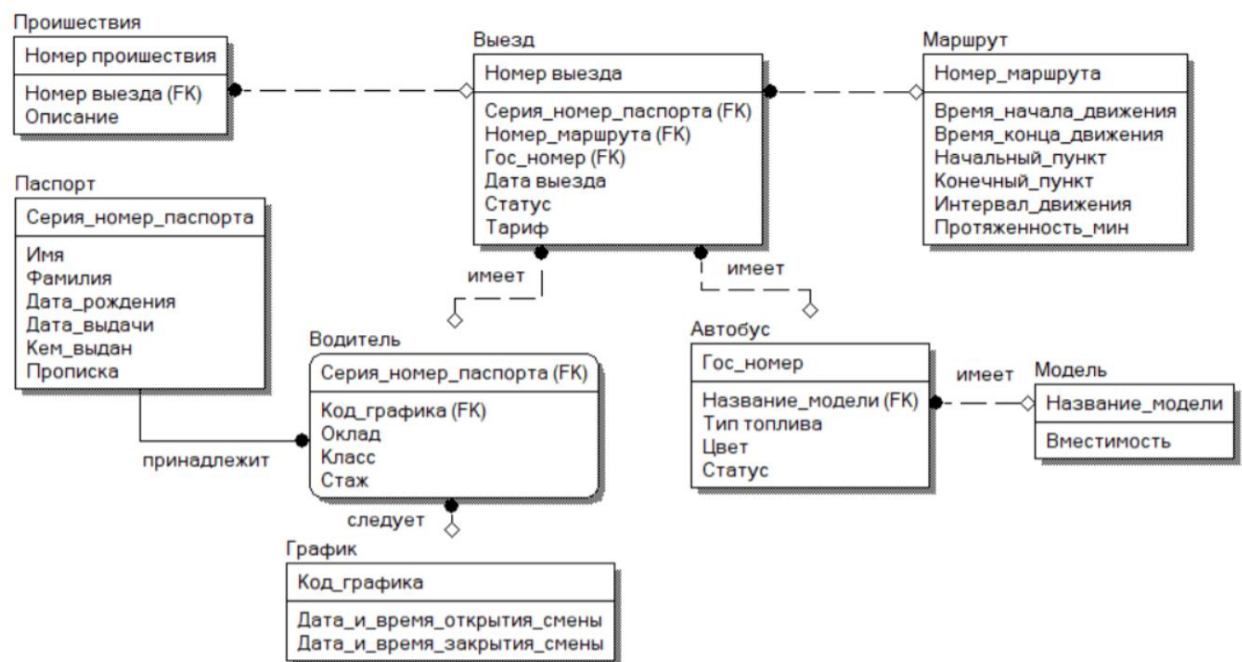
Санкт-Петербург

2020

Задание для второй лабораторной работы (вариант 9)

- Создать программную систему, предназначенную для диспетчера автобусного парка частной транспортной фирмы.

Модель данных в нотации IDEF1X



Запросы к базе данных

- 1) Получаем имя, фамилию и дату рождения водителей, отсортированных по стажу вождения

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT first_name, last_name, experience, birthday
2 FROM driver
3 INNER JOIN passport ON passport.serial_num = driver.serial_num
4 ORDER BY experience
5
```

Data Output Explain Messages Notifications

| | first_name character varying (32) | last_name character varying (32) | experience integer | birthday date |
|---|--------------------------------------|-------------------------------------|-----------------------|------------------|
| 1 | Анатолий | Старых | 2 | 1985-05-15 |
| 2 | Владимир | Новых | 3 | 1982-03-10 |
| 3 | Дмитрий | Пучков | 6 | 1965-06-02 |
| 4 | Василий | Пупкин | 11 | 1978-01-22 |
| 5 | Ашот | Арутюнян | 20 | 1975-08-11 |

- 2) Получаем информацию обо всех автобусах, которые не находятся в ремонте и используют дизельное топливо

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT *
2 FROM bus
3 WHERE status='в эксплуатации' AND fuel_type='дизель'
4
```

Data Output Explain Messages Notifications

| | gos_nomer [PK] character varying (16) | model_name character varying (16) | fuel_type character varying (16) | color character varying (16) | status character varying (16) |
|---|--|--------------------------------------|-------------------------------------|---------------------------------|----------------------------------|
| 1 | M482HB78 | ПА3-3205 | дизель | белый | в эксплуатации |

- 3) Получаем имя, фамилию, и опыт водителей, родившихся после 1980-го года, отсортировав по убыванию возраста

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT first_name, last_name, experience, birthday FROM driver
2 INNER JOIN passport ON passport.serial_num = driver.serial_num
3 WHERE birthday > '01.01.1980'
4 ORDER BY birthday
5
```

Data Output Explain Messages Notifications

| | first_name character varying (32) | last_name character varying (32) | experience integer | birthday date |
|---|--------------------------------------|-------------------------------------|-----------------------|------------------|
| 1 | Владимир | Новых | 3 | 1982-03-10 |
| 2 | Анатолий | Старых | 2 | 1985-05-15 |

- 4) Для водителей, имя которых заканчивается на '-ий' получаем гос. номера автобусов, на которых они осуществляли выезд, дату выездов и график работы

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT first_name, last_name, gos_nomer, departure_date
2 FROM driver
3 INNER JOIN passport ON passport.serial_num = driver.serial_num
4 INNER JOIN departure ON departure.serial_num = driver.serial_num
5 INNER JOIN schedule ON schedule.schedule_num = driver.schedule_num
6 WHERE first_name LIKE '%ий'
```

Data Output Explain Messages Notifications

| | first_name character varying (32) | last_name character varying (32) | gos_nomer character varying (16) | departure_date date |
|---|--------------------------------------|-------------------------------------|-------------------------------------|------------------------|
| 1 | Василий | Пупкин | A329BT78 | 2020-02-12 |
| 2 | Анатолий | Старых | A228YE78 | 2020-05-07 |
| 3 | Василий | Пупкин | A329BT78 | 2020-06-11 |

- 5) Получаем серию и номер паспорта вместе с объединенными именем и фамилией водителей

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT serial_num,
2 CONCAT_WS(' ',
3         first_name,
4         ' ',
5         last_name) AS name_surmane
6 FROM passport
7
```

Data Output Explain Messages Notifications

| | serial_num [PK] character varying (10) | name_surmane text |
|---|---|----------------------|
| 1 | 0901482765 | Василий Пупкин |
| 2 | 0505865371 | Анатолий Старых |
| 3 | 0209756142 | Владимир Новых |
| 4 | 0503875463 | Дмитрий Пучков |
| 5 | 0703564365 | Ашот Арутюнян |

- 6) Информация об органе, выдавшем паспорт записана заглавными буквами, что может вызвать неудобства. Выведем информацию, озаглавив только первую букву

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT serial_num, first_name, last_name,
2 CONCAT_WS(
3     ' ',
4     UPPER(LEFT(issued_by, 1)),
5     LOWER(SUBSTRING(issued_by, 2, length(issued_by))))
6 FROM passport
```

Data Output Explain Messages Notifications

| | serial_num [PK] character varying (10) | first_name character varying (32) | last_name character varying (32) | concat_ws text |
|---|---|--------------------------------------|-------------------------------------|---|
| 1 | 0901482765 | Василий | Пупкин | Тп №1 отдела уфмс россии по Санкт-Петербургу и Ленинградской обл. в а... |
| 2 | 0505865371 | Анатолий | Старых | Тп №69 отдела уфмс россии по Санкт-Петербургу и Ленинградской обл. в ... |
| 3 | 0209756142 | Владимир | Новых | Отдел уфмс россии по Санкт-Петербургу и Ленинградской области в петро... |
| 4 | 0503875463 | Дмитрий | Пучков | Отдел уфмс россии по Санкт-Петербургу и Ленинградской области в невоск... |
| 5 | 0703564365 | Ашот | Арутюнян | Отдел уфмс россии по Санкт-Петербургу и Ленинградской области в петро... |

- 7) С помощью более лаконичного выражения, можно озаглавить первую букву каждого слова

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT serial_num, first_name, last_name, INITCAP(issued_by)
2 FROM passport
```

Data Output Explain Messages Notifications

| | serial_num [PK] character varying (10) | first_name character varying (32) | last_name character varying (32) | initcap text |
|---|---|--------------------------------------|-------------------------------------|--|
| 1 | 0901482765 | Василий | Пупкин | Тп №1 Отдела Уфмс России По Санкт-Петербургу И Ленинградской Обл. ... |
| 2 | 0505865371 | Анатолий | Старых | Тп №69 Отдела Уфмс России По Санкт-Петербургу И Ленинградской Обл. ... |
| 3 | 0209756142 | Владимир | Новых | Отдел Уфмс России По Санкт-Петербургу И Ленинградской Области В Пе... |
| 4 | 0503875463 | Дмитрий | Пучков | Отдел Уфмс России По Санкт-Петербургу И Ленинградской Области В Не... |
| 5 | 0703564365 | Ашот | Арутюнян | Отдел Уфмс России По Санкт-Петербургу И Ленинградской Области В Пе... |

- 8) Вычисляем водителей, попадавших в неприятности, с помощью подзапросов

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT serial_num, first_name, last_name
2 FROM passport
3 WHERE serial_num
4 IN (SELECT serial_num
5     FROM departure
6     WHERE departure_num
7     IN (SELECT departure_num
8         FROM incident))
9
```

Data Output Explain Messages Notifications

| | serial_num [PK] character varying (10) | first_name character varying (32) | last_name character varying (32) |
|---|---|--------------------------------------|-------------------------------------|
| 1 | 0505865371 | Анатолий | Старых |

- 9) Вычисляем среднюю длительность всех маршрутов, начальными пунктами которых являются станции метро

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT ROUND(AVG(route_length)) AS avg_length
2 FROM route
3 INNER JOIN departure ON departure.route_num = route.route_num
4 WHERE start_point LIKE ANY (ARRAY['м.%', 'ст. м.%', 'метро%'])
5
```

Data Output Explain Messages Notifications

| | avg_length numeric | |
|---|-----------------------|--|
| 1 | 70 | |

- 10) Получаем информацию о количестве водителей, получающих определенную зарплату, при условии, что таких водителей больше двух

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT salary, COUNT(serial_num) AS driver_amount
2 FROM driver
3 GROUP BY salary
4 HAVING COUNT(serial_num) > 2
5
```

Data Output Explain Messages Notifications

| | salary integer | driver_amount bigint | |
|---|-------------------|-------------------------|--|
| 1 | 10000 | 3 | |

- 11) Из всех автобусов, которые совершали выезд, получаем информацию о тех, которые не попадали в неприятности

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT bus.*
2 FROM departure AS dep
3 INNER JOIN bus ON bus.gos_nomer = dep.gos_nomer
4 WHERE dep.status = 'завершен'
5 AND NOT EXISTS (SELECT departure_num FROM incident WHERE departure_num=dep.departure_num)
6
```

Data Output Explain Messages Notifications

| | gos_nomer [PK] character varying (16) | model_name character varying (16) | fuel_type character varying (16) | color character varying (16) | status character varying (16) |
|---|--|--------------------------------------|-------------------------------------|---------------------------------|----------------------------------|
| 1 | A329BT78 | ПА3-3204 | бензин | белый | в эксплуатации |
| 2 | M482HB78 | ПА3-3205 | дизель | белый | в эксплуатации |

- 12) Какие из автобусов, имеющих в автопарке, хотя бы раз выезжали по маршруту (пересечение)?

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT gos_nomer FROM bus
2 INTERSECT
3 SELECT gos_nomer FROM departure
4
```

Data Output Explain Messages Notifications

| | gos_nomer character varying (16) |
|---|-------------------------------------|
| 1 | A329BT78 |
| 2 | A228YE78 |
| 3 | M482HB78 |
| 4 | A777AA78 |

13) Какие не выезжали ни разу (исключение)?

bus_park/postgres@PostgreSQL 12

Query Editor Query History

```
1 SELECT gos_nomer FROM bus
2 EXCEPT
3 SELECT gos_nomer FROM departure
4
```

Data Output Explain Messages Notifications

| | gos_nomer character varying (16) | |
|---|-------------------------------------|--|
| 1 | A983YB78 | |