

**Министерство образования и науки Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**“ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**  
**УНИВЕРСИТЕТ ИТМО”**

**ФАКУЛЬТЕТ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

**ОТЧЕТ**  
**ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5**  
**«Построение запросов»**

Специальность 09.02.07 «Информационные системы и программирование»

Дисциплина «Основы проектирования баз данных»

Преподаватель:

Говоров А.И. \_\_\_\_\_

«\_\_» \_\_\_\_\_ 2020 г.

Оценка \_\_\_\_\_

Выполнил:

студент группы Y2337

Примаченко А.А.

Санкт-Петербург  
2019/2020

Цель работы: овладеть практическими навыками создания и написания запросов к базам данных PostgreSQL 11, заполненных рабочими данными.

Индивидуальное задание: реализовать не менее 10 SQL-запросов на PostgreSQL, следуя индивидуальному заданию.

Название создаваемой БД – Birzha. Её схема представлена на рисунке 1.

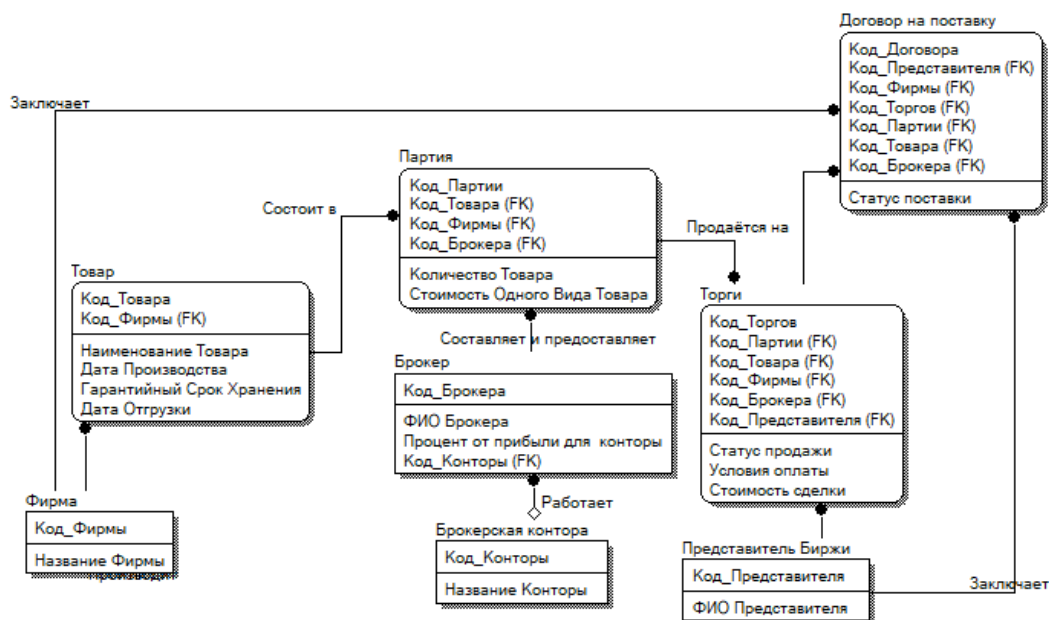


Рисунок 1 – Схема инфологической модели данных БД, выполненная в среде CA ERwin Data Modeler.

Ниже по тексту представлены запросы и соответствующие им рисунки 2-11, содержащие результат выполнения самого запроса.

Запрос 1. Вывести имена брокеров, их процент и бюро, в которых они состоят. Отсортировать результат по проценту в порядке возрастания.

```
select  "Broker".name,  "BrokersBureau".name,  "Broker".percent  from
public."Broker",      public."BrokersBureau"      where      "Broker".bureau_id      =
"BrokersBureau".id order by "Broker".percent asc
```

	<b>name</b> character varying (30)	<b>name</b> character varying (30)	<b>percent</b> double precision
1	Bobik	Cats & Dogs	0
2	Barsik	Cats & Dogs	0.04
3	Mannel Comstock	Cool Brokers	0.12
4	John Smith	International Coalition	0.12
5	Ivan Karavaev	Warsaw Agreement	0.5
6	Daughter Jenny	Trading Family	0.8
7	Son Jimmy	Trading Family	0.8
8	Father Nicolas	Trading Family	1

Рисунок 2 – Результат выполнения запроса 1

Запрос 2. Вывести названия всех продуктов, доставленных до истечения срока годности.

```
select "Product".name, "Product".production_date, "Product".best_before_date,
"Product".shipment_date from public."Product" where "Product".shipment_date is not
null and "Product".shipment_date < "Product".best_before_date
```

	<b>name</b> character varying (30)	<b>production_date</b> date	<b>best_before_date</b> date	<b>shipment_date</b> date
1	Cheesecake	2020-04-01	2020-04-02	2020-04-01
2	Croissant	2020-04-01	2020-04-03	2020-04-02
3	Pancakes	2020-04-01	2020-04-03	2020-04-02
4	Apples	2020-04-01	2020-10-01	2020-05-17

Рисунок 3 – Результат выполнения запроса 2

Запрос 3. Вывести все просроченные товары, которые не были доставлены, вместе с тем временем, что они пролежали на складе на момент 17 мая 2020 года.

```
select "Product".name, "Product".production_date, "Product".best_before_date,
age("Product".best_before_date) from public."Product" where
"Product".best_before_date is not null and "Product".shipment_date is null
```

	<b>name</b> character varying (30)	<b>production_date</b> date	<b>best_before_date</b> date	<b>age</b> interval
1	Sushi	2020-04-01	2020-04-03	1 mon 14 days
2	Cutlet	2020-04-01	2020-04-03	1 mon 14 days

Рисунок 4 – Результат выполнения запроса 3

Запрос 4. Вывести имена всех представителей биржи в верхнем регистре.

```
SELECT upper("ExchangesAgent".name) as "Name" FROM
public."ExchangesAgent"
```

	<b>Name</b> text
1	JOHN SMITH
2	IVAN KARAVAEV
3	JACK LOAF
4	KIRA LIGHT
5	NICK HASKELL

Рисунок 5 – Результат выполнения запроса 4

Запрос 5. Вывести брокеров чей процент, что они отдают бюро, ниже среднего.

```
SELECT "Broker".name as "Name", "Broker".percent as "%" FROM
public."Broker" WHERE "Broker".percent < (SELECT AVG("Broker".percent)
FROM public."Broker" );
```

	Name character varying (30)	% double precision
1	Bobik	0
2	Barsik	0.04
3	Mannel Comstock	0.12
4	John Smith	0.12

Рисунок 6 – Результат выполнения запроса 5

Запрос 6. Вывести наименование продукта, который стоит больше всего единиц за штуку, среди всех сделок. Также вывести эту цену.

```
SELECT "Product".name as "Name", (SELECT max("Product-
Consignment".price_for_unit) FROM public."Product-Consignment") as "Max Price"
FROM public."Product-Consignment", public."Product" WHERE "Product-
Consignment".price_for_unit = (SELECT max("Product-
Consignment".price_for_unit) FROM public."Product-Consignment") and "Product-
Consignment".product_id = "Product".id
```

	Name character varying (30)	Max Price double precision
1	Personal Computer	1920

Рисунок 7 – Результат выполнения запроса 6

Запрос 7. Отобразить ID сделки и стоимость самого дорогого товара оттуда, но выводить нужно только те позиции, где самый дорогой товар стоит меньше 500.

```
SELECT "Product-Consignment".consignment_id, max("Product-
Consignment".price_for_unit) FROM public."Product-Consignment" GROUP BY
"Product-Consignment".consignment_id HAVING max("Product-
Consignment".price_for_unit) < 500
```

	consignment_id integer	max double precision
1	1	128
2	4	450
3	2	333.33
4	3	111

Рисунок 8 – Результат выполнения запроса 7

Запрос 8. Вывести Представителей Биржи, у которых полностью совпадает имя с каким-либо из Брокеров, и, соответственно, наоборот.

```
SELECT      "ExchangesAgent".name      as      "Name"      FROM
public."ExchangesAgent"  WHERE  "ExchangesAgent".name  =  ANY(SELECT
"Broker".name FROM public."Broker")
```

	Name character varying (30)
1	John Smith
2	Ivan Karavaev

Рисунок 9 – Результат выполнения запроса 8

Запрос 9. Вывести всех людей занесённых в базу данных «Биржа». Представленные в базе люди – Представители Биржи и Брокеры. Стоит заметить, что в предыдущем запросе было найдено, что есть люди являющиеся и брокерами, и представителями биржи.

```
SELECT      "ExchangesAgent".name      as      "Name"      FROM
public."ExchangesAgent"  WHERE  "ExchangesAgent".name  not  in  (SELECT
"Broker".name FROM public."Broker") UNION SELECT "Broker".name as "Name"
FROM public."Broker"
```

	Name character varying (30)
1	John Smith
2	Bobik
3	Barsik
4	Son Jimmy
5	Daughter Jenny
6	Mannel Comstock
7	Kira Light
8	Ivan Karavaev
9	Nick Haskell
10	Jack Loaf
11	Father Nicolas

Рисунок 10 – Результат выполнения запроса 9

Запрос 10. Вывести наименование самого дорогого товара из каждой сделки. Также вывести ID сделки.

```
SELECT "Product".name, "Product-Consignment".consignment_id, "Product-
Consignment".price_for_unit FROM public."Product-Consignment" LEFT JOIN
"Product" ON "Product-Consignment".product_id = "Product".id WHERE "Product-
Consignment".price_for_unit      in      (SELECT      MAX("Product-
```

Consignment".price\_for\_unit) FROM public."Product-Consignment" GROUP BY "Product-Consignment".consignment\_id)

	<b>name</b> character varying (30)	<b>consignment_id</b> integer	<b>price_for_unit</b> double precision
1	Stool	1	128
2	Katana	2	333.33
3	Katana	3	111
4	Cheesecake	4	450
5	Personal Computer	5	1920

*Рисунок 11 – Результат выполнения запроса 10*

Вывод: в ходе работы было составлено десять запросов, в следствии чего были получены практические навыки составления и реализации запросов к базам данных PostgreSQL 11.