

Федеральное государственное автономное образовательное учреждение  
высшего образования  
"Санкт-Петербургский национальный исследовательский университет информационных  
технологий, механики и оптики"

Отчетный лист  
по предмету "Базы данных"

Вариант №7

Группа: К3241

Студент: Клишин Никита Дмитриевич

Номер лабораторной	Оценка	Дата сдачи	Подпись преподавателя
Лабораторная 1			
Лабораторная 2			
Лабораторная 3			
Лабораторная 4			
Лабораторная 5			
Лабораторная 6			
Лабораторная 7			

Санкт-Петербург, 2020.

## Описание:

Создать программную систему, ориентированную на администрацию птицефабрики и позволяющую работать с информацией о работниках фабрики и об имеющихся на ней курах.

О каждой курице должна храниться следующая информация: вес, возраст, порода, количество ежемесячно получаемых от курицы яиц, а также информация о местонахождении курицы.

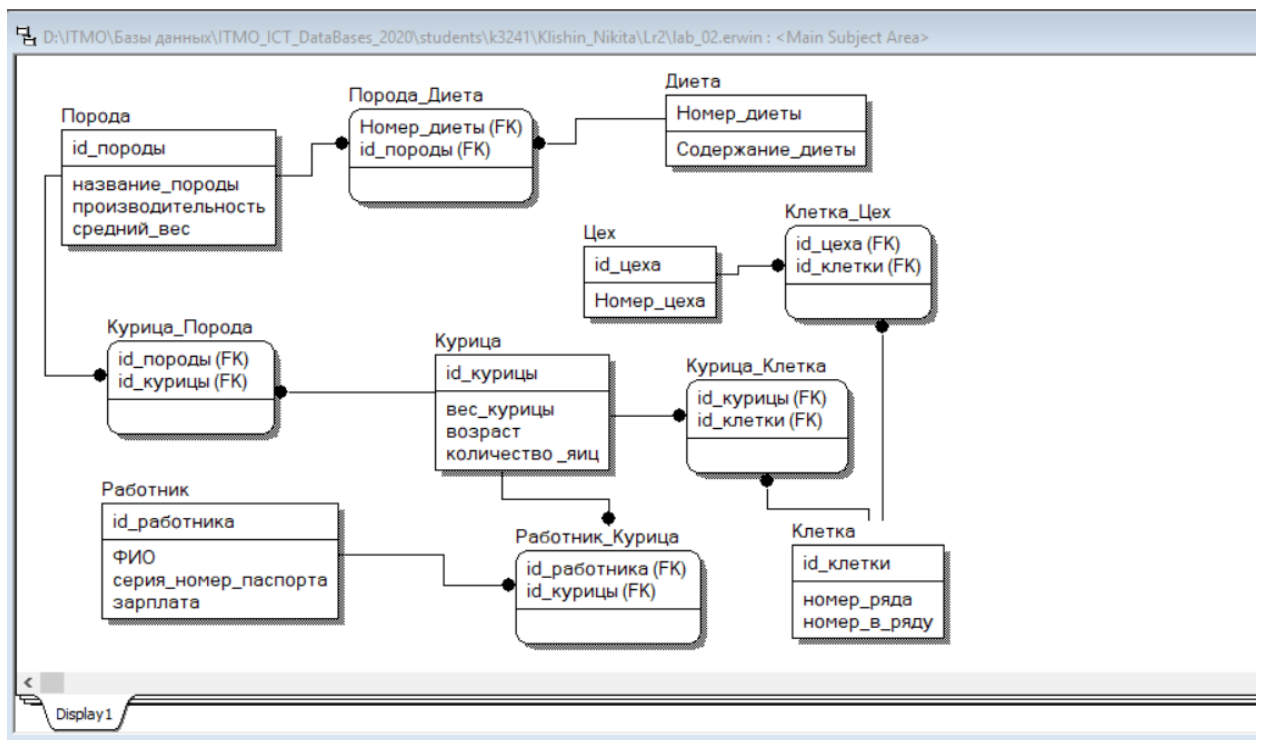
Сведения о породе включают в себя: название породы, среднее количество яиц в месяц (производительность) и средний вес, номер рекомендованной и содержание диеты.

Птицефабрика имеет несколько цехов. В каждой клетке курицей может находиться несколько кур. Код клетки, где находится курица, характеризуется номером цеха, номером ряда в цехе и номером клетки в ряду.

О работниках птицефабрики в БД должна храниться следующая информация: паспортные данные, зарплата, закрепленные за работником клетки. Директор птицефабрики может принять или уволить работника.

Не должно быть кур, не обслуживаемых ни одним работником. Количество кур может изменяться как в большую, так и в меньшую сторону, в отдельные моменты времени часть клеток может пустовать.

## Модель БД:



## Запросы:

1. Выбрать всех записи о chicken с атрибутами id\_chicken, chicken\_weight и отсортировать их по id\_chicken.  
Запрос: `select chicken.id_chicken, chicken.chicken_weight from chicken order by chicken.id_chicken`

### Пример выполнения:

The screenshot shows a database query editor interface. At the top, there are tabs for 'Query Editor', 'История запросов', and 'Scratch Pad'. The 'Query Editor' tab is active, displaying a SQL query: `select chicken.id_chicken, chicken.chicken_weight from chicken order by`. Below the query editor, there are tabs for 'Результат', 'План выполнения', 'Сообщения', and 'Notifications'. The 'Результат' tab is active, showing a table with two columns: 'id\_chicken' (integer, PK) and 'chicken\_weight' (integer). The table contains five rows of data:

	id_chicken [PK] integer	chicken_weight integer
1	1	5
2	2	3
3	3	4
4	4	6
5	5	5

2. Выбрать всех записи о chicken с атрибутами id\_chicken, chicken\_weight где chicken\_weight больше 3, но меньше 6.

Запрос: `select id_chicken, chicken_weight from chicken where (chicken_weight > 3) and (chicken_weight < 6)`

### Пример выполнения:

The screenshot shows a database query editor interface. At the top, there are tabs for 'Query Editor', 'История запросов', and 'Scratch Pad'. The 'Query Editor' tab is active, displaying a SQL query: `select id_chicken, chicken_weight from chicken where (chicken_weight > 3`. Below the query editor, there are tabs for 'Результат', 'План выполнения', 'Сообщения', and 'Notifications'. The 'Результат' tab is active, showing a table with two columns: 'id\_chicken' (integer, PK) and 'chicken\_weight' (integer). The table contains three rows of data:

	id_chicken [PK] integer	chicken_weight integer
1	1	5
2	3	4
3	5	5

3. Выбрать всех работников которые старше 20 лет.  
Запрос: `select id_worker from worker where worker.birth_date < CURRENT_DATE - INTERVAL '20 years'`

Пример выполнения:

The screenshot shows a database query editor interface. At the top, there is a toolbar with various icons for file operations, search, and execution. Below the toolbar, the interface is divided into two main sections: 'Query Editor' and 'Scratch Pad'. The 'Query Editor' contains a SQL query: `select id_worker from worker where worker.birth_date < CURRENT_DATE - I`. Below the query editor, there is a 'Result' tab showing the results of the query. The results are displayed in a table with two columns: 'id\_worker' and 'fio'. The table contains four rows of data.

id_worker	fio
1	Иванов Иван Иванович
2	Иванов Петр Иванович
3	Иванов Алексей Иванович
4	Иванов Сергей Иванович

4. Выбрать всех работников длина ФИО которых меньше 19 символов.  
Запрос: `select id_worker, fio from worker where char_length(worker.fio) < 21`

Пример выполнения:

The screenshot shows a database query editor interface. At the top, there is a toolbar with various icons for file operations, search, and execution. Below the toolbar, the interface is divided into two main sections: 'Query Editor' and 'Scratch Pad'. The 'Query Editor' contains a SQL query: `select id_worker, fio from worker where char_length(worker.fio) < 21`. Below the query editor, there is a 'Result' tab showing the results of the query. The results are displayed in a table with two columns: 'id\_worker' and 'fio'. The table contains two rows of data.

id_worker	fio
1	Иванов Иван Иванович
2	Иванов Петр Иванович

5. Выбрать имена всех работников.  
Запрос: `select id_worker, substring (fio from '%#' % '#'% for '#') from worker`  
Пример выполнения:

The screenshot shows a database query editor interface. At the top is a toolbar with various icons for file operations, search, and execution. Below the toolbar, there are tabs for 'Query Editor', 'История запросов' (Query History), and 'Scratch Pad'. The 'Query Editor' tab is active, displaying a SQL query:

```
1 select id_worker, substring(fio from '%#' % #'%' for '#') from worker
```

Below the query editor, there are tabs for 'Результат' (Result), 'План выполнения' (Execution Plan), 'Сообщения' (Messages), and 'Notifications'. The 'Результат' tab is active, showing a table with the following data:

	id_worker [PK] integer	substring text
1	1	Иван
2	2	Петр
3	3	Дмитрий
4	4	Александр
5	5	Павел

6. Выбрать всех кур, за которыми закреплены работники.

Запрос: `select id_chicken from chicken where exists (select id_chicken from chicken_worker where chicken_worker.id_chicken=chicken.id_chicken)`

Пример выполнения:

The screenshot shows a database query editor interface. At the top is a toolbar with various icons for file operations, search, and execution. Below the toolbar, there are tabs for 'Query Editor', 'История запросов' (Query History), and 'Scratch Pad'. The 'Query Editor' tab is active, displaying a SQL query:

```
1 from chicken_worker where chicken_worker.id_chicken=chicken.id_chicken)
```

Below the query editor, there are tabs for 'Результат' (Result), 'План выполнения' (Execution Plan), 'Сообщения' (Messages), and 'Notifications'. The 'Результат' tab is active, showing a table with the following data:

	id_chicken [PK] integer
1	1
2	2
3	3
4	4

7. Выводит количество кур.

Запрос: `select count(id_chicken) from chicken`

Пример выполнения:

Query Editor История запросов Scratch Pad

```
1 select count(id_chicken) from chicken
```

Результат План выполнения Сообщения Notifications

	count	bigint
1	5	

8. Выводит среднюю производительность кур в первом цехе.

Запрос: `select avg(number_of_eggs) from chicken where cell in (select id_cell from cell where shop in (select id_shop from shop where shop.number_of_shop = 1))`

Пример выполнения:

lab\_03/postgres@PostgreSQL 12

Query Editor История запросов Scratch Pad

```
1 select avg(number_of_eggs) from chicken where cell in (select id_cell from cell where
2 (select id_shop from shop where shop.number_of_shop = 1))
```

Результат План выполнения Сообщения Notifications

	avg	numeric
1	28.5000000000000000	

9. Выбрать всех куриц, сгруппировать их по весу и подсчитать суммарную производительность групп куриц каждого веса.

Запрос: `select chicken_weight, sum (number_of_eggs) from chicken group by chicken_weight having (chicken_weight > 3)`

Пример выполнения:

The screenshot shows the PostgreSQL Query Editor interface. The query editor contains the following SQL query:

```
1 select chicken_weight, sum (number_of_eggs) from chicken group by chicken_weight having (chicken_weight > 3)
```

The results tab is active, displaying the following table:

	chicken_weight integer	sum bigint
1	6	28
2	5	50
3	4	25

10. Выбрать всех работников, за каждым из которых закреплена хотя бы одна курица.

Запрос: `select id_worker, fio from worker where (id_worker = any (select id_worker from chicken_worker))`

Пример выполнения:

The screenshot shows the PostgreSQL Query Editor interface. The query editor contains the following SQL query:

```
1 er, fio from worker where (id_worker = any (select id_worker from chicken_worker))
```

The results tab is active, displaying the following table:

	id_worker [PK] integer	fio character varying (100)
1	1	Иванов Иван Иванович
2	2	Иванов Петр Иванович
3	5	Петров Павел Геннадьевич

11. Выбрать всех работников за каждым из которых закреплена хотя бы одна курица весом более 5 кг.

Запрос: `select id_worker, fio from worker where exists (select id_chicken from chicken where (chicken_weight > 5) and id_chicken in (select id_chicken from chicken_worker where id_worker = worker.id_worker))`

Пример выполнения:

The screenshot shows the PostgreSQL Query Editor interface. The query editor contains the following SQL query:

```
1 : id_chicken from chicken where (chicken_weight > 5) and id_chicken in (select id_
```

The result pane shows a table with the following data:

	id_worker [PK] integer	fio character varying (100)
1	5	Петров Павел Геннадьевич

12. Выбрать веса кур, которые присутствуют и в 1 и в 3 цехах.

Запрос: (select chicken\_weight from chicken where chicken.cell in (select id\_cell from cell where cell.shop = 1)) intersect (select chicken\_weight from chicken where chicken.cell in (select id\_cell from cell where cell.shop = 3))

Пример выполнения:

The screenshot shows the PostgreSQL Query Editor interface. The query editor contains the following SQL query:

```
1 :om chicken where chicken.cell in (select id_cell from cell where cell.shop = 3))
```

The result pane shows a table with the following data:

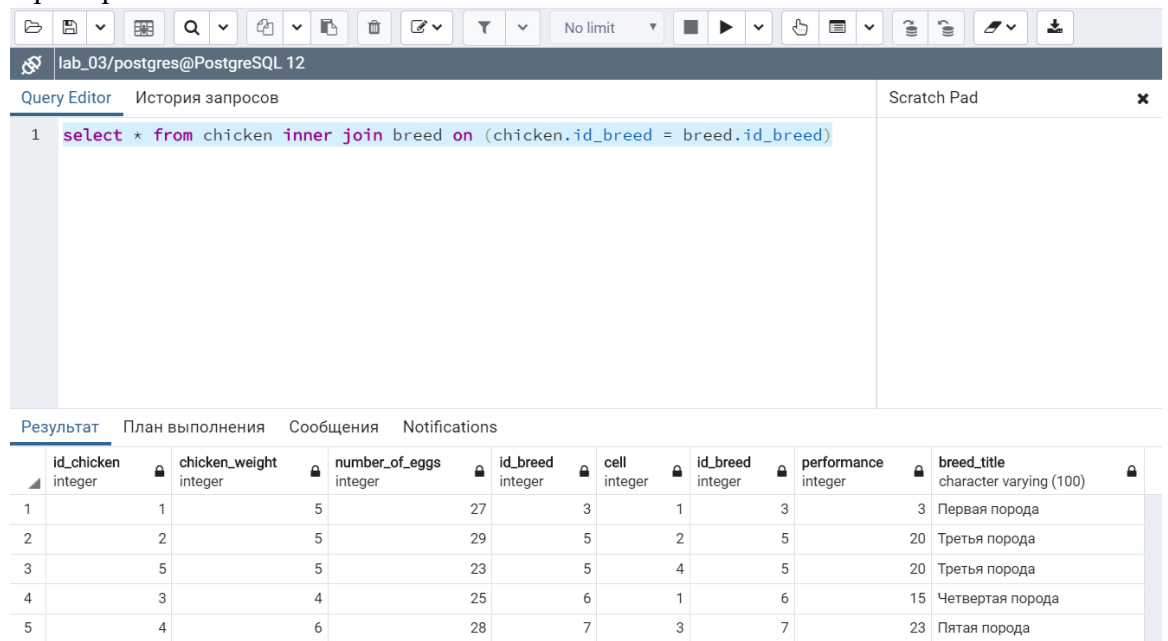
	chicken_weight integer
1	5



13. Вывести всех кур, с информацией о их породе.

Запрос: `select * from chicken inner join breed on (chicken.id_breed = breed.id_breed)`

Пример выполнения:



The screenshot shows a PostgreSQL query editor interface. The query editor contains the following SQL query:

```
1 select * from chicken inner join breed on (chicken.id_breed = breed.id_breed)
```

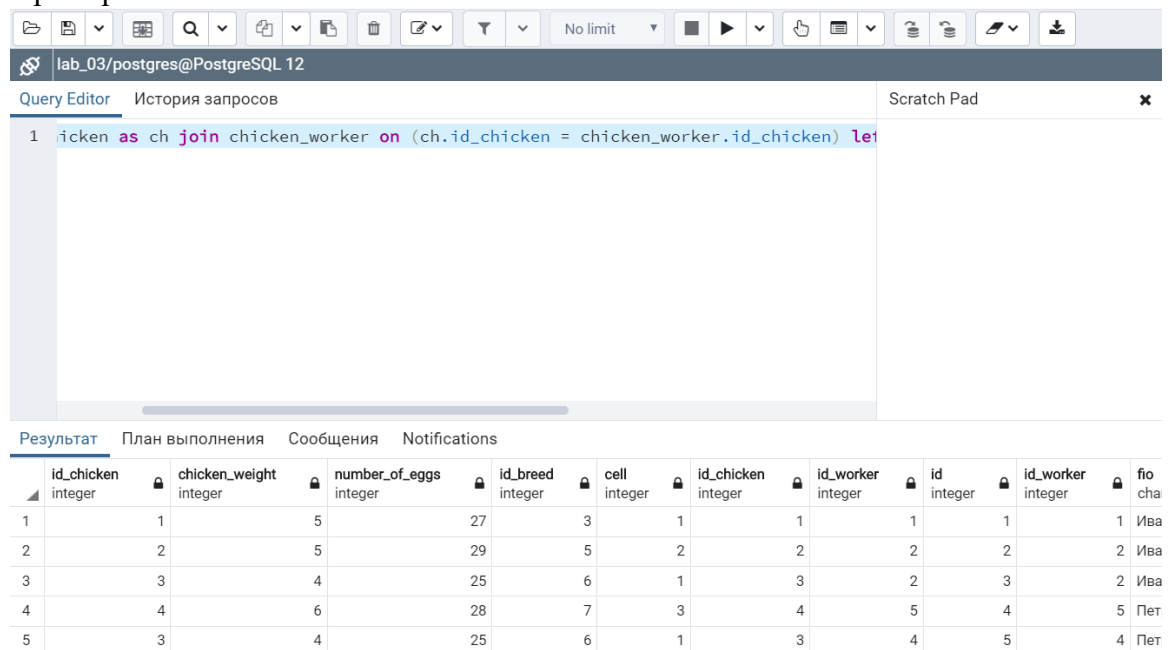
The results tab displays the following data:

	id_chicken integer	chicken_weight integer	number_of_eggs integer	id_breed integer	cell integer	id_breed integer	performance integer	breed_title character varying (100)
1	1	5	27	3	1	3	3	Первая порода
2	2	5	29	5	2	5	20	Третья порода
3	5	5	23	5	4	5	20	Третья порода
4	3	4	25	6	1	6	15	Четвертая порода
5	4	6	28	7	3	7	23	Пятая порода

14. Выводит информацию о курах и закрепленных за ними работников.

Запрос: `select * from chicken as ch join chicken_worker on (ch.id_chicken = chicken_worker.id_chicken) left join worker as w on (chicken_worker.id_worker=w.id_worker)`

Пример выполнения:



The screenshot shows a PostgreSQL query editor interface. The query editor contains the following SQL query:

```
1 chicken as ch join chicken_worker on (ch.id_chicken = chicken_worker.id_chicken) left join worker as w on (chicken_worker.id_worker=w.id_worker)
```

The results tab displays the following data:

	id_chicken integer	chicken_weight integer	number_of_eggs integer	id_breed integer	cell integer	id_chicken integer	id_worker integer	id_worker integer	id_worker integer	first_name character varying (100)
1	1	5	27	3	1	1	1	1	1	Ива
2	2	5	29	5	2	2	2	2	2	Ива
3	3	4	25	6	1	3	2	3	2	Ива
4	4	6	28	7	3	4	5	4	5	Пет
5	3	4	25	6	1	3	4	5	4	Пет

15. Выводит все возможные связи кур и работников.

Запрос: `select * from worker cross join chicken`

Пример выполнения:

lab\_03/postgres@PostgreSQL 12

Query Editor

История запросов

Scratch Pad

1

```
select * from worker cross join chicken
```

Результат

План выполнения

Сообщения

Notifications

	id_worker integer	fio character varying (100)	passport character varying (100)	salary integer	birth_date date	id_chicken integer	chicken_weight integer	number_ integer
11		3 Петров Дмитрий Иванович	1234 56/892	16000	2000-02-01	1	5	
12		3 Петров Дмитрий Иванович	1234 567892	16000	2000-02-01	3	4	
13		3 Петров Дмитрий Иванович	1234 567892	16000	2000-02-01	4	6	
14		3 Петров Дмитрий Иванович	1234 567892	16000	2000-02-01	5	5	
15		3 Петров Дмитрий Иванович	1234 567892	16000	2000-02-01	2	5	
16		4 Петров Александр Юрьевич	1234 567893	30000	2000-02-02	1	5	
17		4 Петров Александр Юрьевич	1234 567893	30000	2000-02-02	3	4	
18		4 Петров Александр Юрьевич	1234 567893	30000	2000-02-02	4	6	
19		4 Петров Александр Юрьевич	1234 567893	30000	2000-02-02	5	5	
20		4 Петров Александр Юрьевич	1234 567893	30000	2000-02-02	2	5	
21		5 Петров Павел Геннадьевич	1234 567894	25500	2001-03-01	1	5	
22		5 Петров Павел Геннадьевич	1234 567894	25500	2001-03-01	3	4	
23		5 Петров Павел Геннадьевич	1234 567894	25500	2001-03-01	4	6	
24		5 Петров Павел Геннадьевич	1234 567894	25500	2001-03-01	5	5	
25		5 Петров Павел Геннадьевич	1234 567894	25500	2001-03-01	2	5	