

**Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ”
ФАКУЛЬТЕТ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

**ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5
«Выполнение запросов в PostgreSQL»**

Специальность 09.02.07 «Информационные системы и программирование»

Дисциплина «Основы проектирования баз данных»

Преподаватель:

Говоров А.И. _____

«__» _____ 2020 г.

Оценка _____

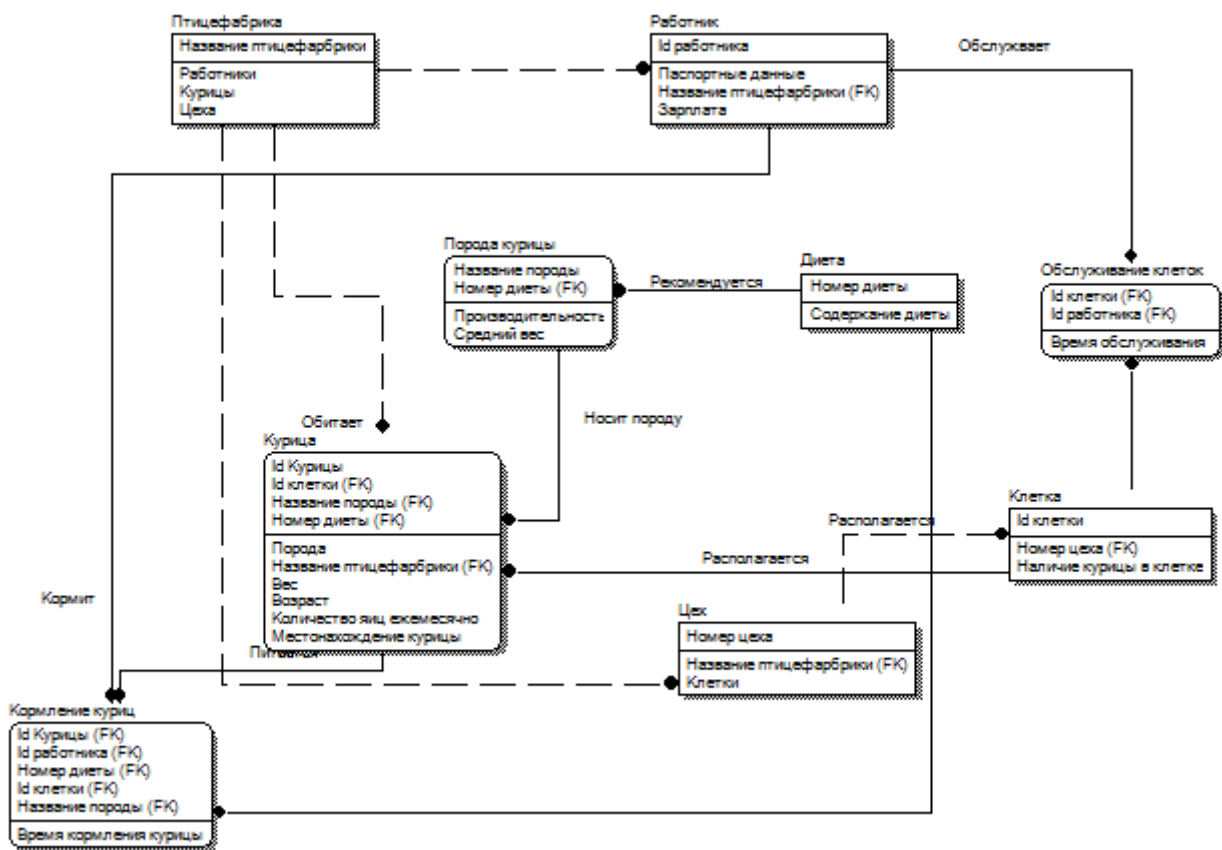
Выполнил:

студент группы Y2337

Князев А. А.

Санкт-Петербург
2019/2020

Индивидуальное задание: 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, и содержание диеты. Птицефабрика имеет несколько цехов. В каждой клетке может находиться несколько кур.



1. Выбор значений, заданных атрибутов из более, чем двух таблиц, с сортировкой – от 1 балла;

```
select "Id_породы","Id_курица","Содержание_диеты" from public."Курица",public."Диета" where "Курица"."Номер_диеты" = "Диета"."Номер_диеты" group by "Диета"."Номер_диеты", "Курица"."Id_курица";
```

```
1 select "Id_породы","Id_курица","Содержание_диеты" from public."Курица",public."Диета" where "Курица"."Номер_диеты" = "Диета"."Номер_диеты" group by "Диета"."Номер_диеты", "Курица"."Id_курица";
```

Результат	План выполнения	Сообщения	Notifications
Id_породы integer	Id_курица integer	Содержание_диеты character (20)	
1	2	2	Eda
2	1	1	Вода

2. Использование условий WHERE, состоящих из более, чем одного условия – от 1 балла;

```
select "Номер_диеты","Id_Птицефабрики" from public."Диета",public."Птицефабрика" where ("Номер_диеты" = 2 or "Номер_диеты" = 3) and "Id_Птицефабрики" = 1;
```

```
1 select "Номер_диеты","Id_Птицефабрики" from public."Диета",public."Птицефабрика" where ("Номер_диеты" = 2 or "Номер_диеты" = 3) and "Id_Птицефабрики" = 1;
```

Результат	План выполнения	Сообщения	Notifications
Номер_диеты integer	Id_Птицефабрики integer		
1	2		1
2	3		1

3. Использование функций для работы с датами – от 2 баллов;

```
select current_date - "Дата_кормления" as "Разница в днях",  
current_date,"Дата_кормления" from "Кормление куриц";
```

Query Editor История запросов

```
1 select current_date - "Дата_кормления" as "Разница в днях", current_date,"Дата_кормления" from "Кормление куриц";
```

Результат План выполнения Сообщения Notifications

	Разница в днях integer	current_date date	Дата_кормления date
1	2916	2020-04-05	2012-04-11
2	2915	2020-04-05	2012-04-12

4. Использование строковых функций – от 3 баллов;

```
select "Номер_Кормления" || '___' || "Номер_диеты","Номер_Кормле-  
ния","Номер_диеты" from "Кормление куриц";
```

BASE/postgres@PostgreSQL 12

Query Editor История запросов

```
1 select "Номер_Кормления" || '___' || "Номер_диеты","Номер_Кормления","Номер_диеты" from "Кормление куриц";
```

Результат План выполнения Сообщения Notifications

	?column? text	Номер_Кормления [PK] integer	Номер_диеты integer
1	1__2	1	2
2	2__1	2	1

5. запрос с использованием подзапросов – от 2 баллов (многострочный подзапрос - от 3 баллов);

```
select distinct "Номер_диеты","Содержание_диеты" from public."Диета"  
where "Номер_диеты" in (select "Номер_диеты" from public."Диета" where  
"Номер_диеты" > 2);
```

Query Editor

История запросов

1

select distinct "Номер_диеты", "Содержание_диеты" from public."Диета" where "Номер_диеты" in (select "Номер_диеты" from public."Диета" where "Номер_диеты" > 2

Результат

План выполнения

Сообщения

Notifications

Номер_диеты [PK] integer	Содержание_диеты character (20)
1	3 Вода и еда
2	4 тыблоко

6. вычисление групповой (агрегатной) функции – от 1 балла (с несколькими таблицами – от 3 баллов);




```
Select distinct max("Номер_диеты") from public."Диета";
```

1 Select distinct max("Номер_диеты") from public."Диета";			
Результат			
max integer			
1	4		

7. вычисление групповой (агрегатной) функции с условием HAVING – от 2 баллов;

```
select max("Номер_диеты"), "Содержание_диеты" from public."Диета"
group by "Номер_диеты" having max("Номер_диеты") > 2;
```

```
1 select max("Номер_диеты"), "Содержание_диеты" from public."Диета" group by "Номер_диеты" having max("Номер_диеты") > 2;
```

Результат		План выполнения	Сообщения	Notifications
 max integer	 Содержание_диеты character (20) 			
1	4	тыблоко		
2	3	Вода и еда		

8. использование предикатов EXISTS, ALL, SOME и ANY - от 4 баллов;

```
select distinct "Номер_диеты","Содержание_диеты" from public."Диета"
where "Номер_диеты" = any (select "Номер_диеты" from public."Диета"
where "Номер_диеты" != 1 and "Номер_диеты" != 4);
```

Query Editor

История запросов

S

1

```
select distinct "Номер_диеты", "Содержание_диеты" from public."Диета" where "Номер_диеты" = any (select "Номер_диеты" from public."Диета" where "Номер_диеты"
```

Результат

План выполнения

Сообщения

Notifications

	Номер_диеты [PK] integer	Содержание_диеты character (20)
1	2	Вода
2	3	Вода и еда

9. использование запросов с операциями реляционной алгебры (объединение, пересечение и т.д.) - от 3 баллов;

`select "Зарплата" from public."Работники" where "Зарплата" >= 2500;`

Query Editor История запросов

1 `select "Зарплата" from public."Работники" where "Зарплата" >= 2500;`

Результат План выполнения Сообщения Notifications

	Зарплата integer	
1	2500	
2	3000	

10.использование объединений запросов (inner join и т.д.) - от 3 баллов.

Select "Номер_диеты" from public."Диета" union select "Зарплата" from public."Работники".

Query Editor

История запросов

1

Select "Номер_диеты" from public."Диета" union select "Зарплата" from public."Работники";

Результат

План выполнения

Сообщения

Notifications

	Номер_диеты integer	
1	2	
2	3	
3	3000	
4	4	
5	1	
6	2500	
7	2000	