

**Федеральное государственное бюджетное образовательное учреждение
высшего образования**

«Заполярный государственный университет им. Н.М. Федоровского»

кафедра информационных систем и технологий

Отчёт по лабораторной работе № 4

по дисциплине

«Базы данных»

Студент: Цимбалюк А.С.

Группа: ИС-21

Преподаватель: Беляев И.С.

Норильск 2023

Лабораторная работа №4

Тема: SQL. Запросы. Цель работы: Получение практических навыков работы с СУБД и языком SQL (оператор SELECT)

Задание:

1. Разработать запросы к базе данных, созданной и заполненной на предыдущих лабораторных работах, следующих видов:
 - a. запрос с условием на числовые данные ($>$, $<$, $=$, between);
 - b. запрос с условием на текстовые данные (LIKE, IN);
 - c. запрос с вычисляемым полем;
 - d. запрос к нескольким таблицам (без явного указания JOIN);
 - e. запрос с агрегирующей функцией (AVG, SUM, COUNT, MIN, MAX);
 - f. запрос с группировкой (GROUP BY);
 - g. запрос с сортировкой (ORDER BY);
 - h. запрос с вложенным подзапросом (не менее 3 видов);
 - i. запрос с оператором UNION;
 - j. запрос с оператором INTERSECT;
 - k. запрос с оператором EXCEPT;
 - l. запрос с выражением CASE;
 - m. запрос с оператором JOIN (пять видов);
 - n. иерархический запрос.
2. Для каждого запроса подписать, что именно он возвращает с учетом предметной области (запросы со смыслом, а не только синтаксически правильные операторы).

1. а. Запрос с условием на числовые данные (>,<=, between):

```
select * from product
where price between 1000 and 10000
```

Скриншот №1 – «Вывод цены в промежутке от 1000 до 10000»

<pre>1 select * from product 2 where price between 1000 and 10000</pre>				
Data Output Messages Notifications				
	id_product [PK] integer	type_product character varying (64)	price integer	
1	1	Квасомет	9999	
2	2	Ланчбокс	1259	

- б. Запрос с условием на текстовые данные (LIKE, IN):

```
select * from sotr
where patronymic like '%ич'
```

Скриншот №2 – «Вывод сотрудников, чье отчество оканчивается на «ИЧ»»

Query Query History					
<pre>1 select * from sotr 2 where patronymic like '%ич'</pre>					
Data Output Messages Notifications					
	id_sotr [PK] integer	surname character varying (64)	forename character varying (64)	patronymic character varying (64)	position_sotr character varying (64)
1	1	Иванов	Павел	Иванович	Уборщик
2	3	Квасько	Василий	Артемович	Главный инженер

- с. Запрос с вычисляемым полем:

```
select sum(price * quantity) from post_met
```

Скриншот №3 – «Вывод общей суммы всех металлов у поставщика»

Query		Query History
1	<code>select sum(price * quantity) from post_met</code>	
Data Output		Messages Notifications
	sum bigint	
1	5944500	

d. Запрос к нескольким таблицам (без явного указания JOIN):

```
select * from client, sotr
к
```

Скриншот №4 – «Вывод сотрудников и покупателей вместе»

1

select * from client, sotr

Data Output

Messages

Notifications

	id_client integer	fk_product_id integer	profit integer	adress character varying (64)	id_sotr integer	surname character varying (64)	forename character varying (64)	patronymic character varying (64)	position_sotr character varying (64)
1	1	1	10199	ул. Бегичева д. 13	1	Иванов	Павел	Иванович	Уборщик
2	1	1	10199	ул. Бегичева д. 13	2	Шлик	Аргентина	Васильевна	Менеджер по продажам
3	1	1	10199	ул. Бегичева д. 13	3	Квасько	Василий	Артемович	Главный инженер
4	2	2	1559	ул. Киркорова д. 48	1	Иванов	Павел	Иванович	Уборщик
5	2	2	1559	ул. Киркорова д. 48	2	Шлик	Аргентина	Васильевна	Менеджер по продажам
6	2	2	1559	ул. Киркорова д. 48	3	Квасько	Василий	Артемович	Главный инженер
7	3	3	999	ул. Проспект Гомез д. 21	1	Иванов	Павел	Иванович	Уборщик
8	3	3	999	ул. Проспект Гомез д. 21	2	Шлик	Аргентина	Васильевна	Менеджер по продажам
9	3	3	999	ул. Проспект Гомез д. 21	3	Квасько	Василий	Артемович	Главный инженер

e. Запрос с агрегирующей функцией (AVG, SUM, COUNT, MIN, MAX):

```
select min(price) as "Минимальная цена продукта" from product
```

Скриншот №5 – «Вывод минимальной цены произведенного продукта»

Query Query History					
1	<code>select min(price) as "Минимальная цена продукта" from product</code>				
Data Output Messages Notifications					
<div><div>≡+</div><div>📄</div><div>▼</div><div>📋</div><div>🗑️</div><div>🗄️</div><div>⬇️</div><div>📈</div></div>					
	<table><thead><tr><th></th><th>Минимальная цена продукта integer</th></tr></thead><tbody><tr><td>1</td><td>899</td></tr></tbody></table>		Минимальная цена продукта integer	1	899
	Минимальная цена продукта integer				
1	899				

f. Запрос с группировкой (GROUP BY):

```
select surname, count(*) from sotr group by surname
```

Скриншот №6 – «Вывод фамилий сотрудников и кол-во человек с одинаковыми фамилиями»

1 `select` surname, `count`(*) `from` sotr `group by` surname

2

Data Output Messages Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈

	surname character varying (64) 🔒	count bigint 🔒
1	Квасько	1
2	Шпик	1
3	Иванов	2

g. Запрос с сортировкой (ORDER BY):

```
select surname, count(*) from sotr  
group by surname  
order by count asc
```

Скриншот №7 – «Вывод фамилий сотрудников и кол-во человек с одинаковыми фамилиями по возрастанию»

Query Query History

1

2

3

```
select surname, count(*) from sotr
group by surname
order by count asc
```

Data Output Messages Notifications

≡+

▼

	<div>surname</div> <div>character varying (64)</div> <div></div>	<div>count</div> <div>bigint</div> <div></div>
1	Квасько	1
2	Шпик	1
3	Иванов	2

h. Запрос с вложенным подзапросом (не менее 3 видов);

```
select * from product
where price > (select AVG(price) from product)
```

Скриншот №8 – «Вывод продуктов, у которых стоимость выше средней цены»

Query

Query History

1

select * from product

2

where price > (select AVG(price) from product)

Data Output

Messages

Notifications

id_product

[PK] integer

type_product

character varying (64)

price

integer

1

1

Квасомет

9999

```
select * from post_stan
where post_stan_id = (select post_stan_id from post_stan where
quantity = (select max(quantity) from post_stan))
```

Скриншот №9 – «Вывод компании, которая имеет наибольшее количество станков»

1	select * from post_stan
2	where post_stan_id = (select post_stan_id from post_stan where quantity = (select max(quantity) from post_stan))

Data Output						
post_stan_id	name_post	type_material	quantity	price	adress	
[PK] integer	character varying (64)	character varying (64)	integer	integer	character varying (64)	
1	3	Grande-butera	Плоскошлифовальный с прямоугольным столом стан...	25	1999	ул. Риотовская д. 30

```
select * from post_stan
where name_post = (select name_post from post_stan where price =
(select max(price) from post_stan))
```

Скриншот №10 – «Вывод компании, которая имеет самую цену за единицу товара»










1	select * from post_stan
2	where name_post = (select name_post from post_stan where price = (select max(price) from post_stan))

Data Output						
post_stan_id	name_post	type_material	quantity	price	adress	
[PK] integer	character varying (64)	character varying (64)	integer	integer	character varying (64)	
1	2	Зельдская	Алмазно-расточный станок	3	15666	ул. Хентайская д. 69

i. Запрос с оператором UNION:

```
select adress from post_met
union all select adress from client
```










Скриншот №11 – «Вывод адресов поставщиков металла и клиентов»

Query		Query History
1	<code>select address from post_met</code>	
2	<code>union all select address from client</code>	
Data Output		Messages Notifications
<div></div>		
	adress character varying (64) 	
1	ул. Федоровского д. 7А	
2	ул. Резидентов д. 444	
3	ул. Карского д. 12А	
4	ул. Бегичева д. 13	
5	ул. Киркорова д. 48	
6	ул. Проспект Гомез д. 21	

ж. запрос с оператором INTERSECT:

```
select surname from sotr
intersect select surname from client
```

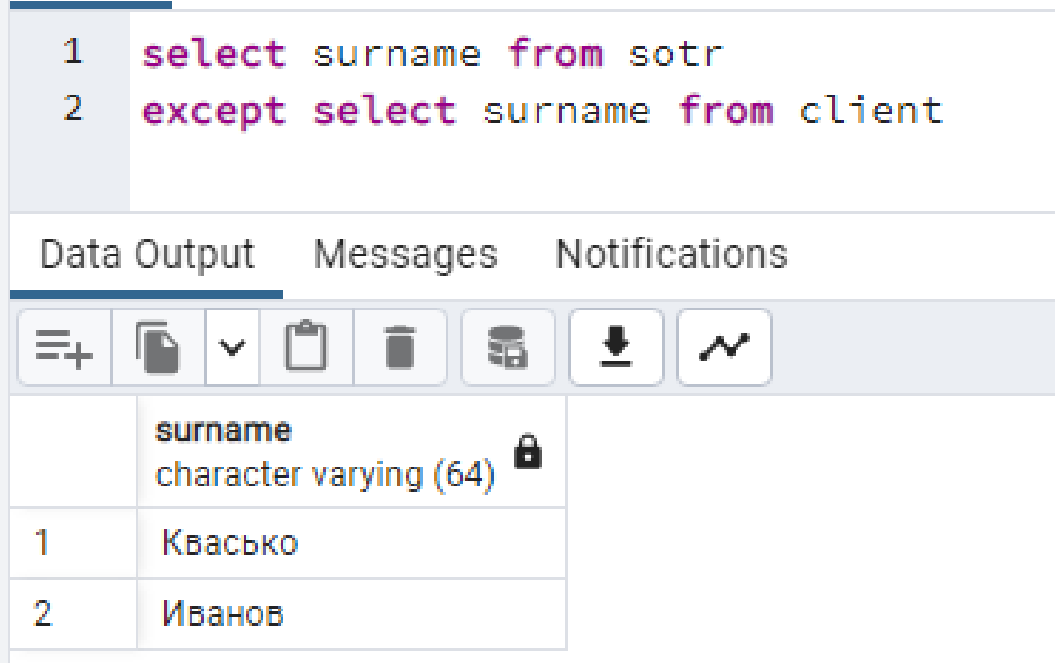
Скриншот №12 – «Вывод сотрудников, которые приобрели товар»

Query		Query History
1	<code>select surname from sotr</code>	
2	<code>intersect select surname from client</code>	
Data Output		Messages Notifications
<div></div>		
	surname character varying (64) 	
1	Шпик	

k. Запрос с оператором EXCEPT:

```
select surname from sotr  
except select surname from client
```


Скриншот №13 – «Вывод фамилий клиентов без учета клиентов-сотрудников»



The screenshot shows a SQL IDE interface. The top panel contains a query with two lines:

```
1 select surname from sotr  
2 except select surname from client
```

Below the query editor, there are tabs for "Data Output", "Messages", and "Notifications". The "Data Output" tab is active, displaying a table with the results of the query. The table has two columns: "surname" (character varying (64)) and a lock icon. The results are as follows:

	surname character varying (64) 
1	Квасько
2	Иванов

l. Запрос с выражением CASE:

```
select surname,  
case when surname = 'Иванов' then 'Брат того Вани'  
when surname = 'Шпик' then 'Шпиц'  
when surname = 'Квасько' then 'Квас'  
end nickname  
from sotr
```

Скриншот №14 – «Вывод кличек сотрудников»

```
1 select surname,
2 case when surname = 'Иванов' then 'Брат того Вани'
3 when surname = 'Шпик' then 'Шпиц'
4 when surname = 'Квасько' then 'Квас'
5 end nickname
6 from sotr
```

Data Output Messages Notifications

	surname character varying (64)	nickname text
1	Иванов	Брат того Вани
2	Шпик	Шпиц
3	Квасько	Квас
4	Иванов	Брат того Вани

m. Запрос с оператором JOIN (пять видов):

```
select id_company, position_sotr from company
join sotr on id_sotr = fk_sotr_id
```

Скриншот №15 – «Вывод объединенной таблицы компании и существующих должностей в ней»

```
1 select id_company, position_sotr from company
2 join sotr on id_sotr = fk_sotr_id
```

Data Output Messages Notifications

	id_company integer	position_sotr character varying (64)
1	1	Уборщик
2	2	Менеджер по продажам
3	3	Главный инженер

```
select type_product, price, adress, surname from product
INNER JOIN client on id_product = fk_product_id
```

Скриншот №16 – «Вывод объединенной таблицы продуктов и клиентов, которые приобрели продукт»

1	select	type_product	,	price	,	adress	,	surname	from	product
2	INNER JOIN	client	on	id_product	=	fk_product_id				

Data Output
Messages
Notifications

	type_product character varying (64)	price integer	adress character varying (64)	surname character varying (64)
1	Квасомет	9999	ул. Бегичева д. 13	Шпик
2	Ланчбокс	1259	ул. Киркорова д. 48	Ковалев
3	Коса	899	ул. Проспект Гомез д. 21	Клеверов

```
select id_materials, name_post, type_material
from post_met
NATURAL JOIN materials
```

Скриншот №17 – «Вывод объединенной таблицы всех материалов и поставщиков металла»

1

select id_materials, name_post, type_material

2

from post_met

3

NATURAL JOIN materials

Data Output

Messages

Notifications

≡+

	<div>id_materials</div> <div>integer</div> <div></div>	<div>name_post</div> <div>character varying (64)</div> <div></div>	<div>type_material</div> <div>character varying (64)</div> <div></div>
1	1	Норникель	Никель
2	1	СельхозПлат	Серебро
3	1	СтройМет	Золото
4	2	Норникель	Никель
5	2	СельхозПлат	Серебро
6	2	СтройМет	Золото
7	3	Норникель	Никель
8	3	СельхозПлат	Серебро
9	3	СтройМет	Золото

```
select id_client, profit, sum(profit - extra_charge) as
extra_charge from client
group by id_client
```

Скриншот №18 – «Вывод наценки на товары «extra_charge»»

```

1 select id_client, profit, sum(profit - extra_charge) as extra_charge from client
2 group by id_client

```

Data Output

Messages

Notifications

≡+

📄

▼

📋

🗑️

🗑️

📥

⬇️

📈

	id_client [PK] integer	profit integer	extra_charge bigint
1	2	1559	300
2	1	10199	200
3	3	999	100

```
select product.type_product, product.price, profit from product
left join client on id_client = fk_product_id
```

Скриншот №19 – «Вывод объединенной таблицы продукта и суммы, которую получили с учетом наценки»

<pre>1 select product.type_product, product.price, profit from product 2 left join client on id_client = fk_product_id</pre>			
Data Output Messages Notifications			
	type_product character varying (64)	price integer	profit integer
1	Квасомет	9999	10199
2	Квасомет	9999	1559
3	Квасомет	9999	999
4	Ланчбокс	1259	10199
5	Ланчбокс	1259	1559
6	Ланчбокс	1259	999
7	Коса	899	10199
8	Коса	899	1559
9	Коса	899	999

n. Иерархический запрос:

```
with recursive temp1 (id, parent_id, position_sotr, path) as (
select t1.id, t1.parent_id, t1.position_sotr, cast (t1.position_sotr
as varchar(67)) as path
from sotr t1 where t1.position_sotr = 'Уборщик'
union
select t2.id, t2.parent_id, t2.position_sotr, cast (temp1.path || '->' || t2.position_sotr as varchar(67))
from sotr t2 join temp1 on (temp1.parent_id = t2.id))
select * from temp1
```

Скриншот №19 – «Вывод иерархии должностей»

<pre>1 with recursive temp1 (id, parent_id, position_sotr, path) as (2 select t1.id, t1.parent_id, t1.position_sotr, cast (t1.position_sotr as varchar(67)) as path 3 from sotr t1 where t1.position_sotr = 'Уборщик' 4 union 5 select t2.id, t2.parent_id, t2.position_sotr, cast (temp1.path '->' t2.position_sotr as varchar(67)) 6 from sotr t2 join temp1 on (temp1.parent_id = t2.id)) 7 select * from temp1</pre>					
Data Output Messages Notifications					
	id integer	parent_id integer	position_sotr character varying (64)	path character varying (67)	
1	4	3	Уборщик	Уборщик	
2	3	2	Менеджер по продажам	Уборщик->Менеджер по продажам	
3	2	1	Главный инженер	Уборщик->Менеджер по продажам->Главный инженер	
4	1	[null]	Директор персонала	Уборщик->Менеджер по продажам->Главный инженер->Директор персонала	