Министерство образования и науки РФ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Норильский государственный индустриальный институт»
Кафедра Информационных систем и технологий

Отчет: Лабораторная работа №4

Тема: Проектирование реляционный базы данных postgreSQL

По предмету: Базы данных

Выполнил:

Посметюк А.И.

Группы:

ИС – 21

Проверил:

Беляев И.С.

Норильск 2023г

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

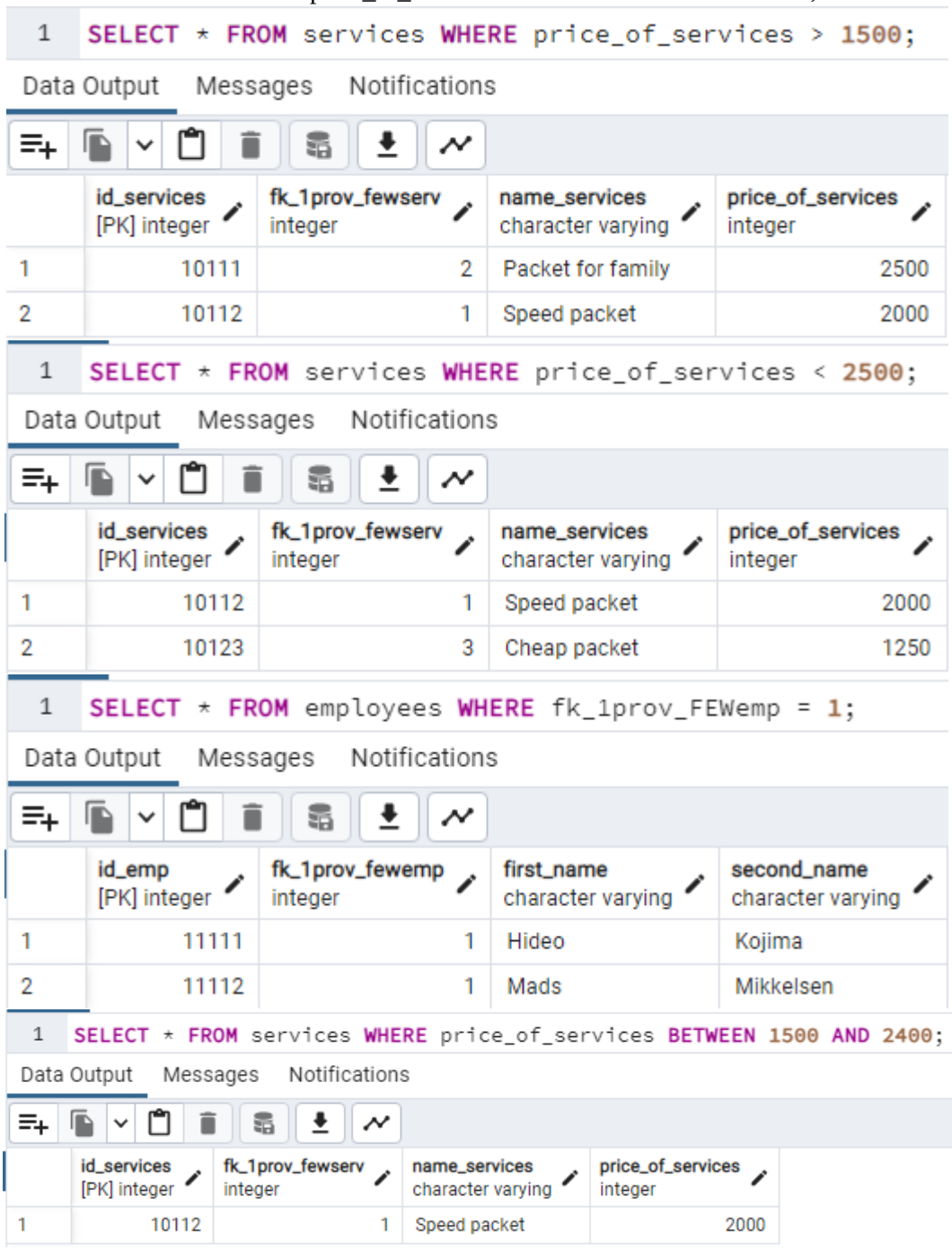
Задание №4

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.

a)

```
SELECT * FROM services WHERE price_of_services > 1500;
SELECT * FROM services WHERE price_of_services < 2500;
SELECT * FROM employees WHERE fk_1prov_FEWemp = 1;
SELECT * FROM services WHERE price_of_services BETWEEN 1500 AND 2400;
```



```
1  SELECT * FROM services WHERE price_of_services > 1500;
```

Data Output | Messages | Notifications

	id_services [PK] integer	fk_1prov_fewserv integer	name_services character varying	price_of_services integer
1	10111	2	Packet for family	2500
2	10112	1	Speed packet	2000

```
1  SELECT * FROM services WHERE price_of_services < 2500;
```

Data Output | Messages | Notifications

	id_services [PK] integer	fk_1prov_fewserv integer	name_services character varying	price_of_services integer
1	10112	1	Speed packet	2000
2	10123	3	Cheap packet	1250

```
1  SELECT * FROM employees WHERE fk_1prov_FEWemp = 1;
```

Data Output | Messages | Notifications

	id_emp [PK] integer	fk_1prov_fewemp integer	first_name character varying	second_name character varying
1	11111	1	Hideo	Kojima
2	11112	1	Mads	Mikkelsen

```
1  SELECT * FROM services WHERE price_of_services BETWEEN 1500 AND 2400;
```

Data Output | Messages | Notifications

	id_services [PK] integer	fk_1prov_fewserv integer	name_services character varying	price_of_services integer
1	10112	1	Speed packet	2000

Скриншот №1 – Запросы с условием на числовые данные

b)

SELECT second_name FROM employees WHERE second_name LIKE ‘%n’

```
SELECT second_name FROM employees WHERE second_name LIKE '%n';
```

Data Output | Messages | Notifications

	second_name character varying
1	Mikkelsen
2	Joosten

Скриншот №2 – Запрос с условием на текстовые данные

c)

SELECT first_name, second_name || ' (' || id_user || ')' FROM users ORDER BY first_name, second_name;

```
SELECT first_name, second_name || ' (' || id_user || ') ' FROM users ORDER BY first_name, second_name;
```

Data Output | Messages | Notifications

	first_name character varying	?column? text
1	Ada	Wong (192825568)
2	Barak	Obeme (1928255255)
3	Dmitry	Puchkov (19282551)

Скриншот №3 – Запрос с вычисляемым полем

d)

SELECT * FROM services, servers WHERE id_services = fk_1serv_FEWservers

```
SELECT * FROM services, servers WHERE id_services = fk_1serv_FEWservers
```

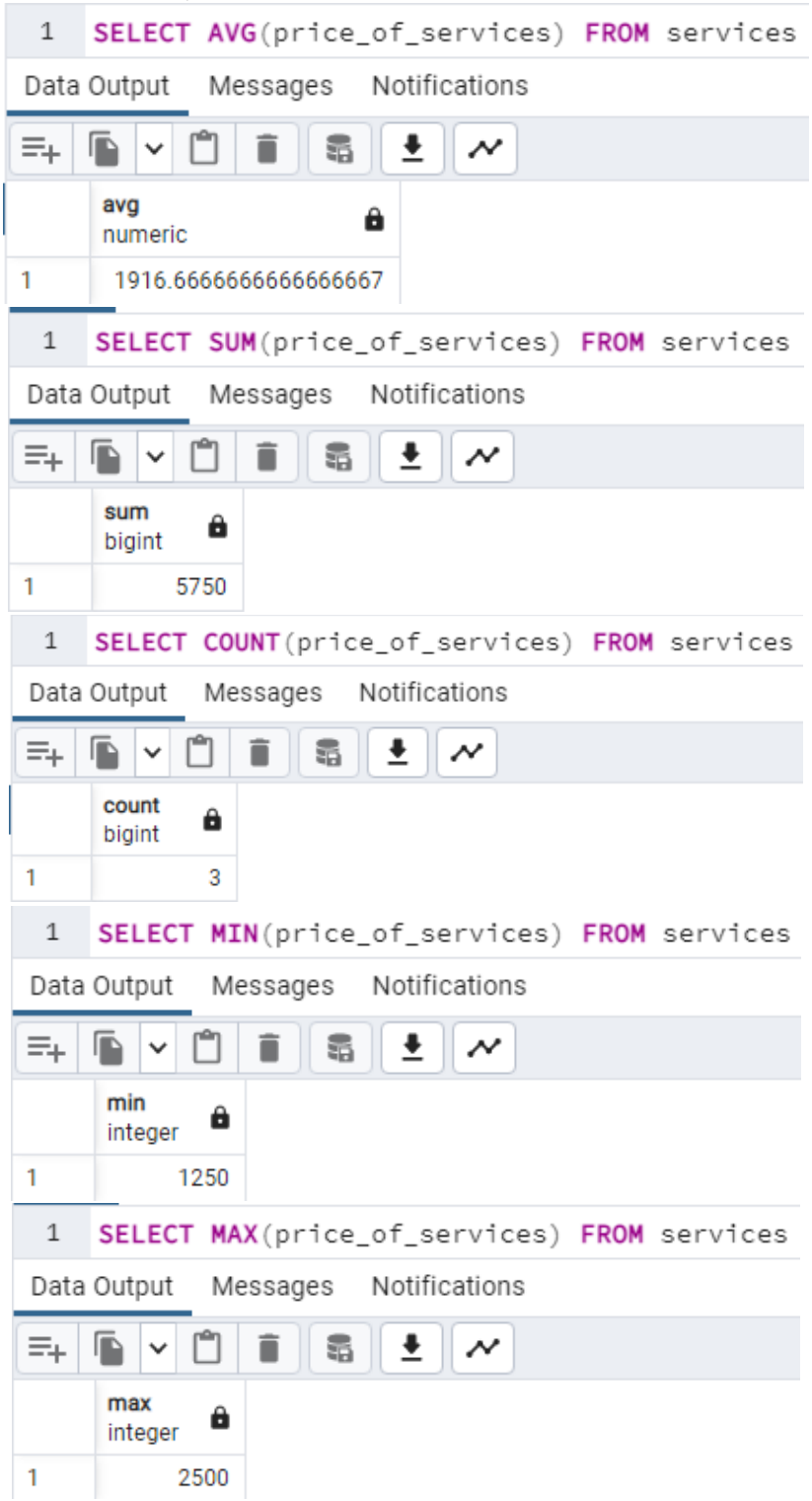
Data Output | Messages | Notifications

	id_services integer	fk_1prov_fewserv integer	name_services character varying	price_of_services integer	id_servers bigint	fk_1serv_fewservers integer	name_servers character varying
1	10123	3	Cheap packet	1250	1921682551	10123	SERVER1
2	10112	1	Speed packet	2000	19216825568	10112	SERVER68
3	10111	2	Packet for family	2500	192168255255	10111	SERVER255

Скриншот №4 – Запрос к нескольким таблицам

e)

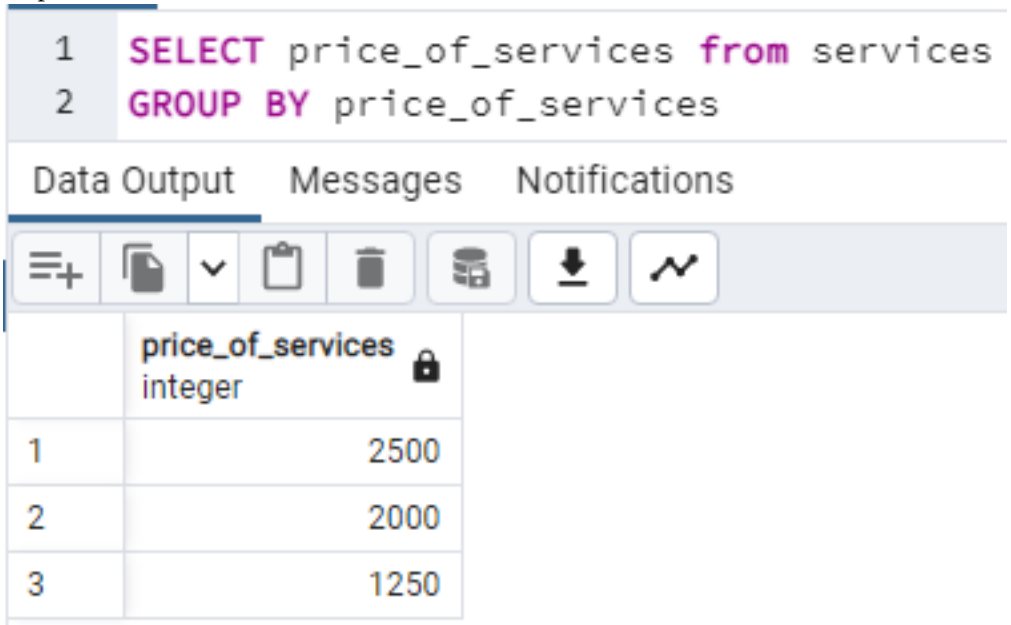
```
SELECT AVG(price_of_services) FROM services
SELECT SUM(price_of_services) FROM services
SELECT COUNT(price_of_services) FROM services
SELECT MIN(price_of_services) FROM services
SELECT MAX(price_of_services) FROM services
```



Скриншот №5 – Запросы с агрегирующей функцией

f)

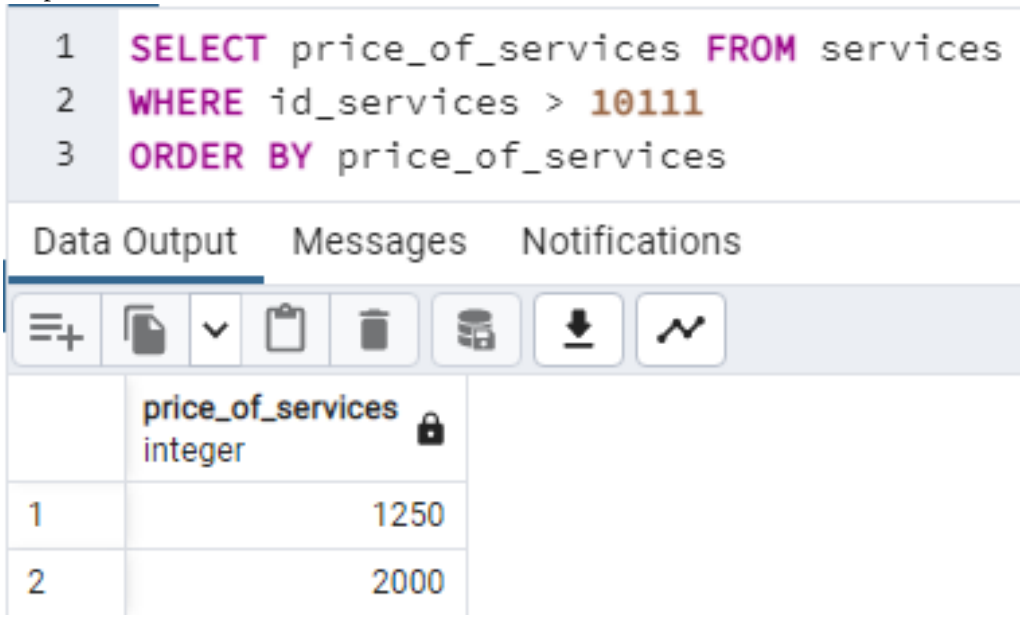
```
SELECT price_of_services from services
GROUP BY price_of_services
```



Скриншот №6 – запрос с группировкой

g)

```
SELECT price_of_services FROM services
WHERE id_services > 10111
ORDER BY prices_of_services
```



Скриншот №7 – Запрос с сортировкой

h)

```
SELECT *
FROM services
WHERE price_of_services > (SELECT AVG(price_of_services) FROM services);
```

```
1  SELECT *
2  FROM services
3  WHERE price_of_services > (SELECT AVG(price_of_services) FROM services);
```

Data Output | Messages | Notifications

	id_services [PK] integer	fk_1prov_fewserv integer	name_services character varying	price_of_services integer
1	10111	2	Packet for family	2500
2	10112	1	Speed packet	2000

Скриншот №8 – Запрос с вложенным подзапросом

i)

```
SELECT * FROM users
UNION
SELECT * FROM employees
```

Query | Query History

```
1  SELECT * FROM users
2  UNION
3  SELECT * FROM employees
```

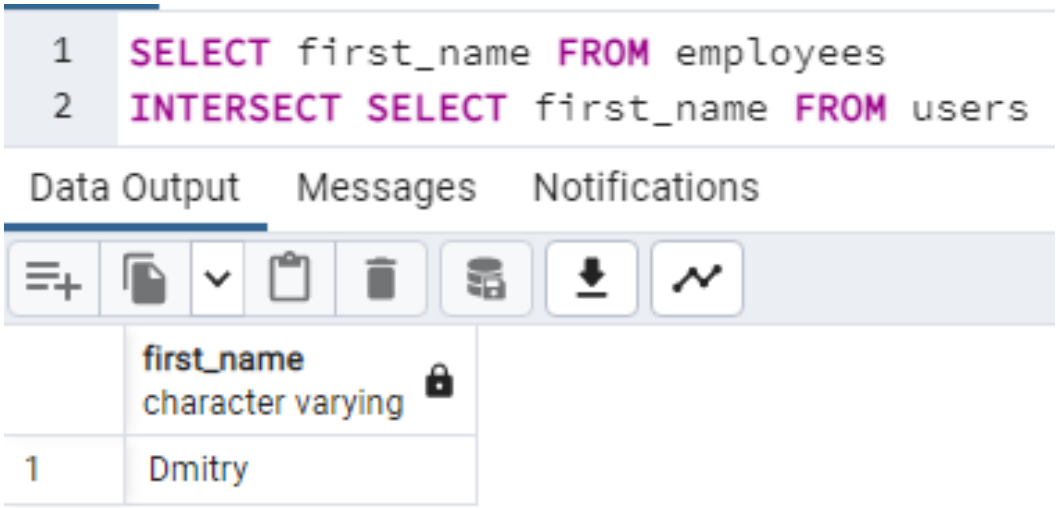
Data Output | Messages | Notifications

	id_user bigint	fk_1site_fewusers bigint	first_name character varying	second_name character varying
1	11112	1	Mads	Mikkelsen
2	11131	2	Stefanie	Joosten
3	192825568	1921122551	Ada	Wong
4	1928255255	1922042551	Barak	Obeme
5	11123	3	Norman	Ridus
6	19282551	192425550	Dmitry	Puchkov
7	11111	1	Hideo	Kojima

Скриншот №9 – запрос с оператором UNION

j)

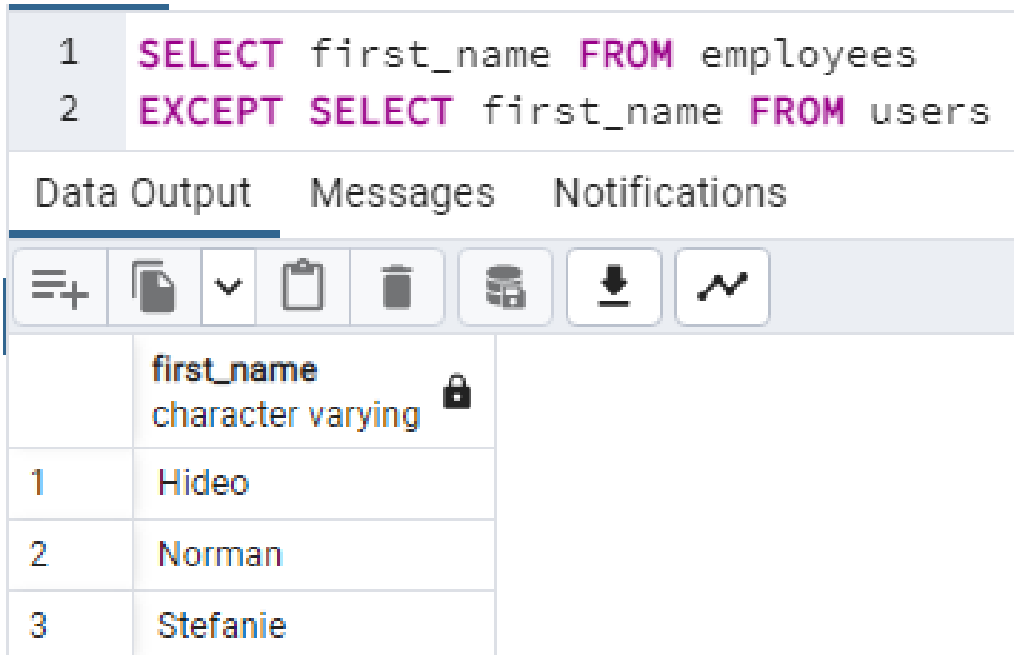
```
SELECT first_name FROM employees
INTERSECT SELECT first_name FROM users
```



Скриншот №10 – запрос с оператором INTERSECT

k)

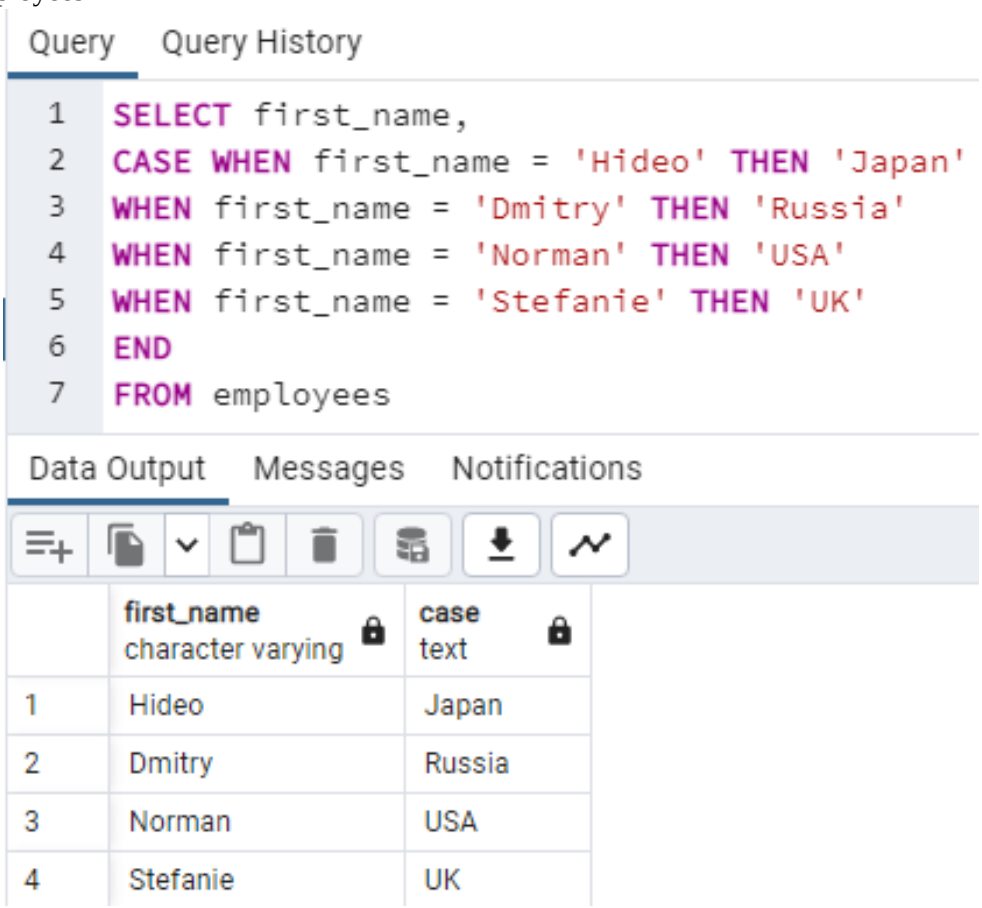
```
SELECT first_name FROM employees
EXCEPT SELECT first_name FROM users
```



Скриншот №11 – запрос с оператором EXCEPT

l)

```
SELECT first_name,
CASE WHEN first_name = 'Hideo' THEN 'Japan'
WHEN first_name = 'Dmitry' THEN 'Russia'
WHEN first_name = 'Norman' THEN 'USA'
WHEN first_name = 'Stefanie' THEN 'UK'
END
FROM employees
```



Скриншот №12 – запрос с оператором CASE

m)

SELECT * FROM servers
JOIN routing ON servers.id_servers = fk_1server_FEWroutes
SELECT * FROM servers
INNER JOIN services ON id_services = fk_1serv_FEWroutes
SELECT first_name, second_name, name_company FROM employees
NATURAL JOIN providers
SELECT * FROM servers
LEFT JOIN routing ON routing.id_route = servers.id_servers
SELECT * FROM servers
RIGHT JOIN routing ON routing.id_route = servers.id_servers

```
1  SELECT * FROM servers
2  JOIN routing ON servers.id_servers = fk_1server_FEWroutes
```

Data Output | Messages | Notifications

	id_servers bigint	fk_1serv_fewservers integer	name_servers character varying	id_route integer	fk_1server_fewroutes bigint	port_name character varying
1	1921682551	10123	SERVER1	66	1921682551	anycast
2	19216825568	10112	SERVER68	24	19216825568	unicast
3	192168255255	10111	SERVER255	1444	192168255255	broadcast

```
1  SELECT * FROM servers
2  INNER JOIN services ON id_services = fk_1serv_FEWservers
```

Data Output | Messages | Notifications

	id_servers bigint	fk_1serv_fewservers integer	name_servers character varying	id_services integer	fk_1prov_fewserv integer	name_services character varying	price_of_services integer
1	1921682551	10123	SERVER1	10123	3	Cheap packet	1250
2	19216825568	10112	SERVER68	10112	1	Speed packet	2000
3	192168255255	10111	SERVER255	10111	2	Packet for family	2500

```
1  SELECT first_name, second_name,name_company FROM employees
2  NATURAL JOIN providers
```

Data Output | Messages | Notifications

	first_name character varying	second_name character varying	name_company character varying
1	Hideo	Kojima	Rostelekom
2	Dmitry	Puchkov	Rostelekom
3	Norman	Ridus	Rostelekom
4	Stefanie	Joosten	Rostelekom
5	Hideo	Kojima	MTS
6	Dmitry	Puchkov	MTS
7	Norman	Ridus	MTS
8	Stefanie	Joosten	MTS
9	Hideo	Kojima	NORKOM
10	Dmitry	Puchkov	NORKOM
11	Norman	Ridus	NORKOM
12	Stefanie	Joosten	NORKOM

```
1  SELECT * FROM servers
2  LEFT JOIN routing ON routing.id_route = servers.id_servers
```

Data Output | Messages | Notifications

	id_servers bigint	fk_1serv_fewservers integer	name_servers character varying	id_route integer	fk_1server_fewroutes bigint	port_name character varying
1	1921682551	10123	SERVER1	[null]	[null]	[null]
2	19216825568	10112	SERVER68	[null]	[null]	[null]
3	192168255255	10111	SERVER255	[null]	[null]	[null]

```
1  SELECT * FROM servers
2  RIGHT JOIN routing ON routing.id_route = servers.id_servers
```

Data Output | Messages | Notifications

	id_servers bigint	fk_1serv_fewservers integer	name_servers character varying	id_route integer	fk_1server_fewroutes bigint	port_name character varying
1	1921682551	10123	SERVER1	[null]	[null]	[null]
2	19216825568	10112	SERVER68	[null]	[null]	[null]
3	192168255255	10111	SERVER255	[null]	[null]	[null]

Скриншот №13 – запрос с оператором JOIN

n)

with recursive rec (id_employees, post, parent_id, path) as (
select t1.id_employees, t1.post, t1.parent_id, cast (t1.post as varchar(50)) as path
from employees t1 where t1.post = 'Junior'
union
select t2.id_employees, t2.post, t2.parent_id, cast (rec.path || ' >>> ' || t2.post as varchar(50))
from employees t2 join rec on (rec.parent_id = t2.id_employees))
select id_employees, post, path from rec

```
1  with recursive rec (id_employees, post, parent_id, path) as (
2  select t1.id_employees, t1.post, t1.parent_id, cast (t1.post as varchar(50)) as path
3  from employees t1 where t1.post = 'Junior'
4  union
5  select t2.id_employees, t2.post, t2.parent_id, cast (rec.path || ' >>> ' || t2.post as varchar(50))
6  from employees t2 join rec on (rec.parent_id = t2.id_employees))
7  select id_employees, post, path from rec
```

Data Output | Messages | Notifications

	id_employees integer	post character varying	path character varying (50)
1	4	Junior	Junior
2	3	Middle	Junior >>> Middle
3	2	Senior	Junior >>> Middle >>> Senior
4	1	Director	Junior >>> Middle >>> Senior >>> Director

Скриншот №14 – иерархический запрос