Белорусский Государственный Университет
Информатики и Радиоэлектроники
Кафедра ЭВМ

Отчет по лабораторной работе № 5

Тема: «Реализация SQL-запросов на выборку данных с использованием подзапросов, агрегатных функций, группировки и операций над множествами»
ВАРИАНТ № 32 – БАССЕЙН

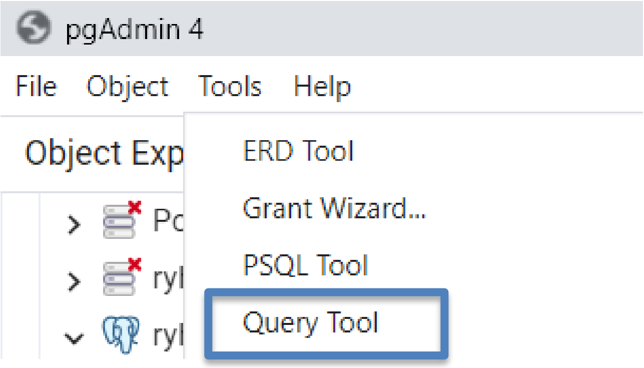
Студент: Д.В. Деруго

Проверила: Д.В. Куприянова

Минск 2023

1. Запросы на выборку данных

Для создания запросов в pgAdmin4 необходимо выбрать инструмент Query Tool:

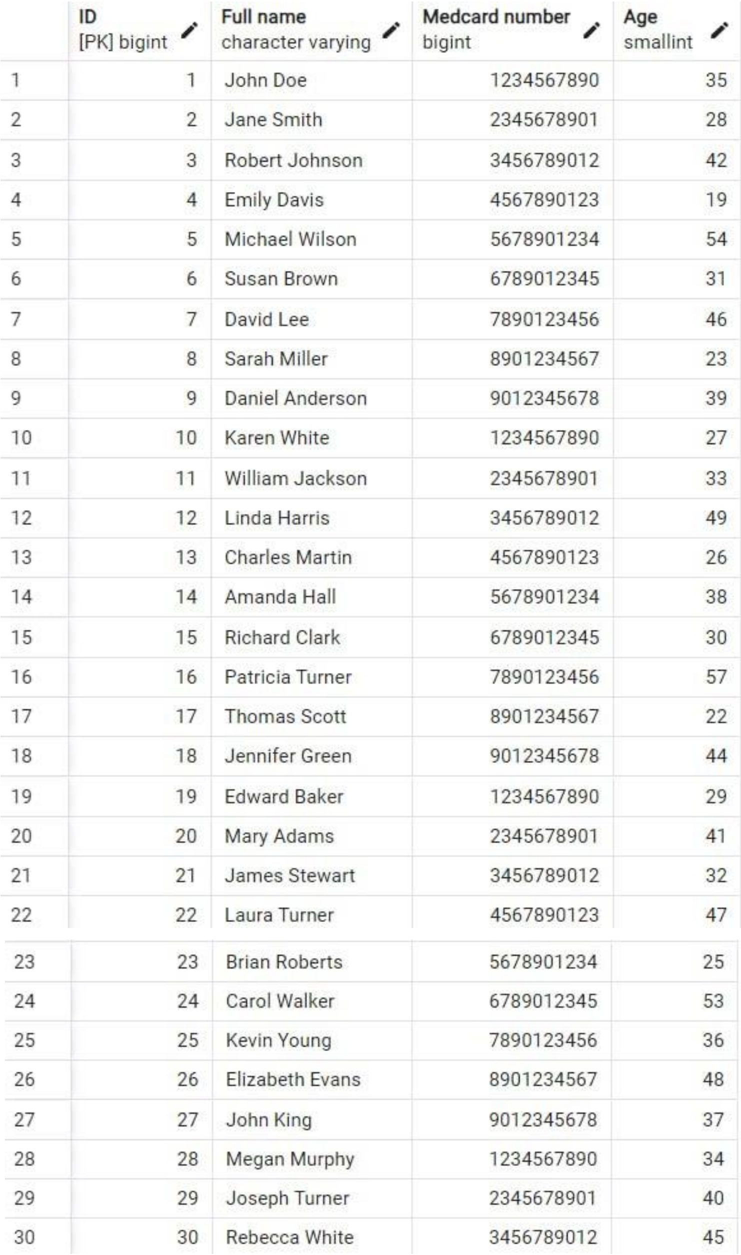


Для каждой из таблиц сделаем по три запроса.

1.1 Таблица “Visitor”

Первоначальный вид таблицы получим с помощью запроса:

```
SELECT * FROM “Visitor”
```

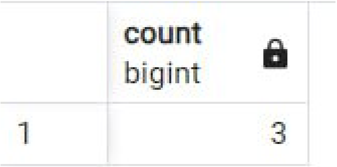


	ID [PK] bigint	Full name character varying	Medcard number bigint	Age smallint
1	1	John Doe	1234567890	35
2	2	Jane Smith	2345678901	28
3	3	Robert Johnson	3456789012	42
4	4	Emily Davis	4567890123	19
5	5	Michael Wilson	5678901234	54
6	6	Susan Brown	6789012345	31
7	7	David Lee	7890123456	46
8	8	Sarah Miller	8901234567	23
9	9	Daniel Anderson	9012345678	39
10	10	Karen White	1234567890	27
11	11	William Jackson	2345678901	33
12	12	Linda Harris	3456789012	49
13	13	Charles Martin	4567890123	26
14	14	Amanda Hall	5678901234	38
15	15	Richard Clark	6789012345	30
16	16	Patricia Turner	7890123456	57
17	17	Thomas Scott	8901234567	22
18	18	Jennifer Green	9012345678	44
19	19	Edward Baker	1234567890	29
20	20	Mary Adams	2345678901	41
21	21	James Stewart	3456789012	32
22	22	Laura Turner	4567890123	47
23	23	Brian Roberts	5678901234	25
24	24	Carol Walker	6789012345	53
25	25	Kevin Young	7890123456	36
26	26	Elizabeth Evans	8901234567	48
27	27	John King	9012345678	37
28	28	Megan Murphy	1234567890	34
29	29	Joseph Turner	2345678901	40
30	30	Rebecca White	3456789012	45

1. Вывести количество людей с возрастом от 20 до 25

```
SELECT COUNT(*) FROM “Visitor” WHERE “Age” IN
(20,21,22,23,24,25)
```

Результат выполнения запроса:

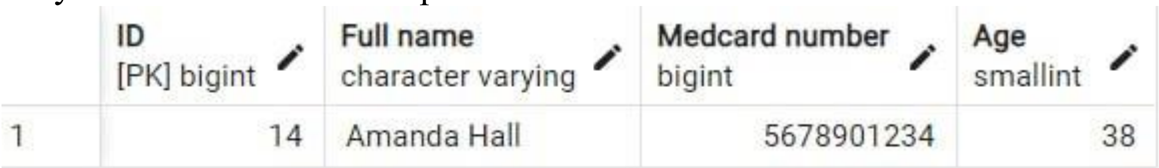


	count bigint
1	3

2. Вывести человека с первым по алфавиту именем:

```
SELECT * FROM “Visitor” WHERE “Full name” =
(SELECT MIN(“Full name”) FROM “Visitor”)
```

Результат выполнения запроса:

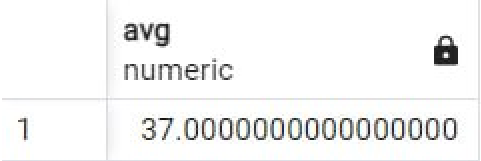


	ID [PK] bigint	Full name character varying	Medcard number bigint	Age smallint
1	14	Amanda Hall	5678901234	38

3. Посчитать средний возраст посетителей

```
SELECT AVG(“Age”) from “Visitor”
```

Результат выполнения запроса:

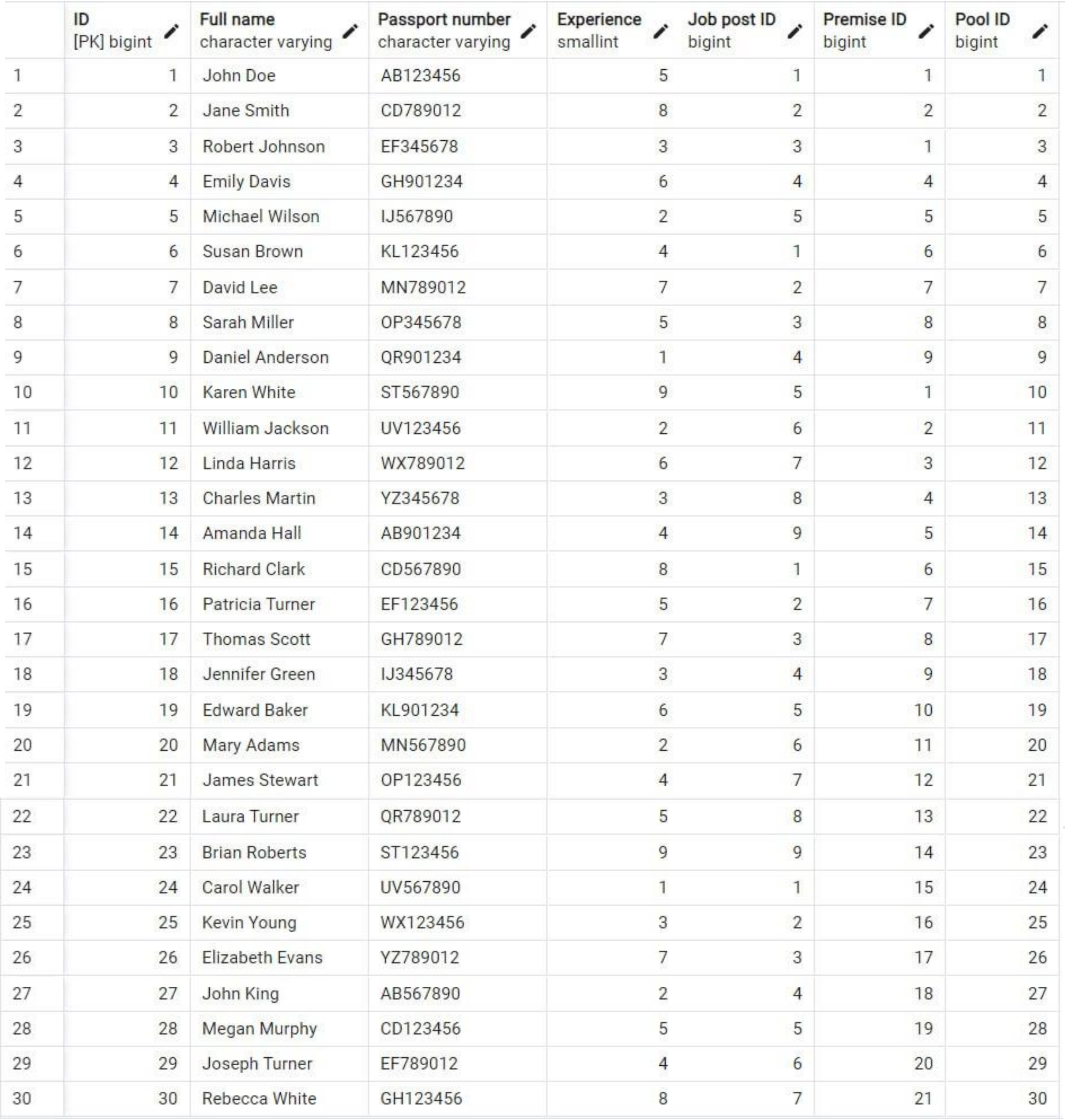


	avg numeric
1	37.0000000000000000

1.2 Таблица “Employee”

Первоначальный вид таблицы получим с помощью запроса:

```
SELECT * FROM “Employee”
```

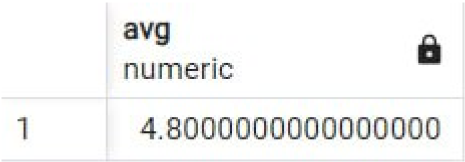


	ID [PK] bigint	Full name character varying	Passport number character varying	Experience smallint	Job post ID bigint	Premise ID bigint	Pool ID bigint
1	1	John Doe	AB123456	5	1	1	1
2	2	Jane Smith	CD789012	8	2	2	2
3	3	Robert Johnson	EF345678	3	3	1	3
4	4	Emily Davis	GH901234	6	4	4	4
5	5	Michael Wilson	IJ567890	2	5	5	5
6	6	Susan Brown	KL123456	4	1	6	6
7	7	David Lee	MN789012	7	2	7	7
8	8	Sarah Miller	OP345678	5	3	8	8
9	9	Daniel Anderson	QR901234	1	4	9	9
10	10	Karen White	ST567890	9	5	1	10
11	11	William Jackson	UV123456	2	6	2	11
12	12	Linda Harris	WX789012	6	7	3	12
13	13	Charles Martin	YZ345678	3	8	4	13
14	14	Amanda Hall	AB901234	4	9	5	14
15	15	Richard Clark	CD567890	8	1	6	15
16	16	Patricia Turner	EF123456	5	2	7	16
17	17	Thomas Scott	GH789012	7	3	8	17
18	18	Jennifer Green	IJ345678	3	4	9	18
19	19	Edward Baker	KL901234	6	5	10	19
20	20	Mary Adams	MN567890	2	6	11	20
21	21	James Stewart	OP123456	4	7	12	21
22	22	Laura Turner	QR789012	5	8	13	22
23	23	Brian Roberts	ST123456	9	9	14	23
24	24	Carol Walker	UV567890	1	1	15	24
25	25	Kevin Young	WX123456	3	2	16	25
26	26	Elizabeth Evans	YZ789012	7	3	17	26
27	27	John King	AB567890	2	4	18	27
28	28	Megan Murphy	CD123456	5	5	19	28
29	29	Joseph Turner	EF789012	4	6	20	29
30	30	Rebecca White	GH123456	8	7	21	30

1. Вывести средний опыт сотрудников

```
SELECT AVG(“Experience”) FROM “Employee”
```

Результат выполнения запроса:

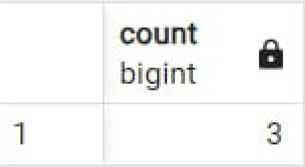


	avg numeric
1	4.8000000000000000

2. Вывести количество имен сотрудников, начинающихся с буквы R

```
SELECT COUNT(*) FROM "Employee" WHERE "Full name"
LIKE 'R%'
```

Результат выполнения запроса

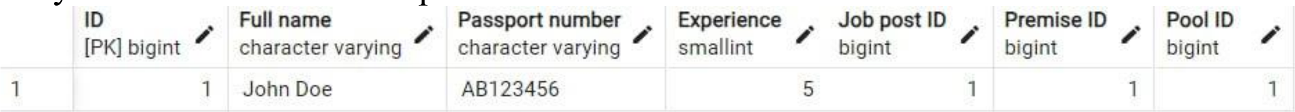


	count bigint
1	3

3. Вывести работника с опытом 5 лет и минимальным номером паспорта

```
SELECT * FROM "Employee" WHERE ("Experience" = 5 AND
"Passport number" = (SELECT MIN("Passport number")
FROM "Employee")
```

Результат выполнения запроса:

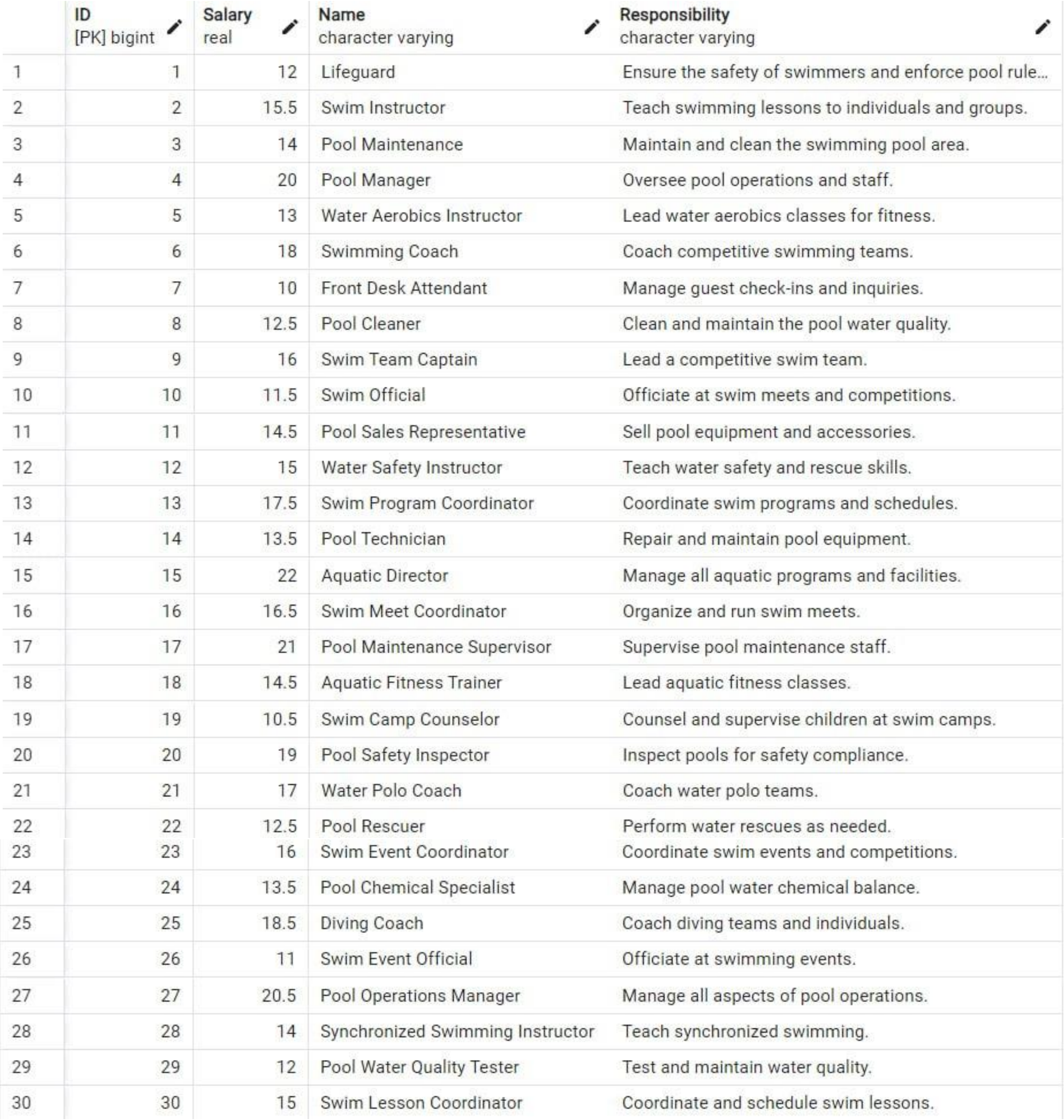


	ID [PK] bigint	Full name character varying	Passport number character varying	Experience smallint	Job post ID bigint	Premise ID bigint	Pool ID bigint
1	1	John Doe	AB123456	5	1	1	1

1.3 Таблица “Job post”

Первоначальный вид таблицы получим с помощью запроса:

```
SELECT * FROM “Job post”
```

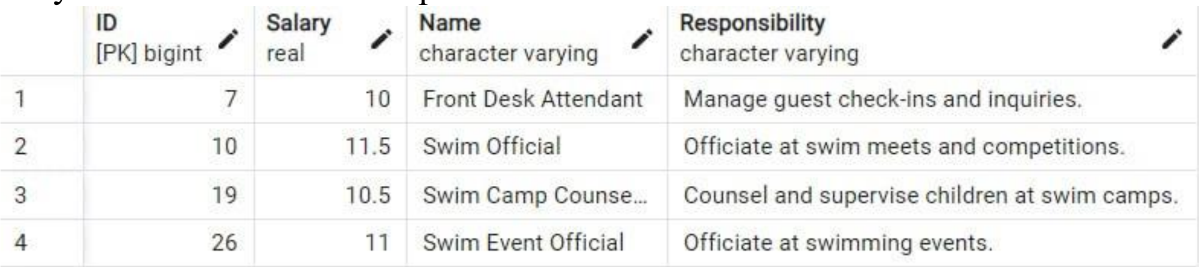


	ID [PK] bigint	Salary real	Name character varying	Responsibility character varying
1	1	12	Lifeguard	Ensure the safety of swimmers and enforce pool rule…
2	2	15.5	Swim Instructor	Teach swimming lessons to individuals and groups.
3	3	14	Pool Maintenance	Maintain and clean the swimming pool area.
4	4	20	Pool Manager	Oversee pool operations and staff.
5	5	13	Water Aerobics Instructor	Lead water aerobics classes for fitness.
6	6	18	Swimming Coach	Coach competitive swimming teams.
7	7	10	Front Desk Attendant	Manage guest check-ins and inquiries.
8	8	12.5	Pool Cleaner	Clean and maintain the pool water quality.
9	9	16	Swim Team Captain	Lead a competitive swim team.
10	10	11.5	Swim Official	Officiate at swim meets and competitions.
11	11	14.5	Pool Sales Representative	Sell pool equipment and accessories.
12	12	15	Water Safety Instructor	Teach water safety and rescue skills.
13	13	17.5	Swim Program Coordinator	Coordinate swim programs and schedules.
14	14	13.5	Pool Technician	Repair and maintain pool equipment.
15	15	22	Aquatic Director	Manage all aquatic programs and facilities.
16	16	16.5	Swim Meet Coordinator	Organize and run swim meets.
17	17	21	Pool Maintenance Supervisor	Supervise pool maintenance staff.
18	18	14.5	Aquatic Fitness Trainer	Lead aquatic fitness classes.
19	19	10.5	Swim Camp Counselor	Counsel and supervise children at swim camps.
20	20	19	Pool Safety Inspector	Inspect pools for safety compliance.
21	21	17	Water Polo Coach	Coach water polo teams.
22	22	12.5	Pool Rescuer	Perform water rescues as needed.
23	23	16	Swim Event Coordinator	Coordinate swim events and competitions.
24	24	13.5	Pool Chemical Specialist	Manage pool water chemical balance.
25	25	18.5	Diving Coach	Coach diving teams and individuals.
26	26	11	Swim Event Official	Officiate at swimming events.
27	27	20.5	Pool Operations Manager	Manage all aspects of pool operations.
28	28	14	Synchronized Swimming Instructor	Teach synchronized swimming.
29	29	12	Pool Water Quality Tester	Test and maintain water quality.
30	30	15	Swim Lesson Coordinator	Coordinate and schedule swim lessons.

1. Вывести работников с заработной платой меньше 12

```
SELECT * FROM "Job post" WHERE "Salary" < 12
```

Результат выполнения запроса:

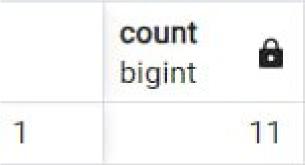


	ID [PK] bigint	Salary real	Name character varying	Responsibility character varying
1	7	10	Front Desk Attendant	Manage guest check-ins and inquiries.
2	10	11.5	Swim Official	Officiate at swim meets and competitions.
3	19	10.5	Swim Camp Counse...	Counsel and supervise children at swim camps.
4	26	11	Swim Event Official	Officiate at swimming events.

2. Вывести количество работ, где поле ответственности содержит слово «pool»

```
SELECT COUNT(*) FROM "Job post" WHERE
"Responsibility" LIKE '%pool%'
```

Результат выполнения запроса:

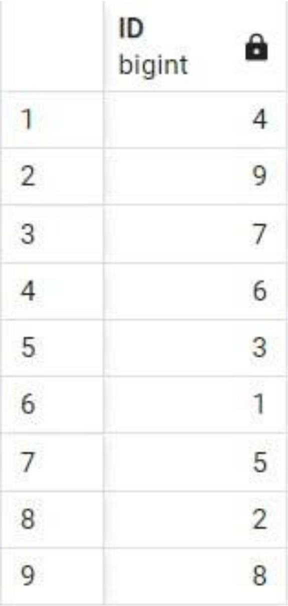


	count bigint
1	11

3. Вывести ID незанятых вакансий

```
SELECT "ID" FROM "Job post" INTERSECT SELECT "Job
post ID" FROM "Employee"
```

Результат выполнения запроса:

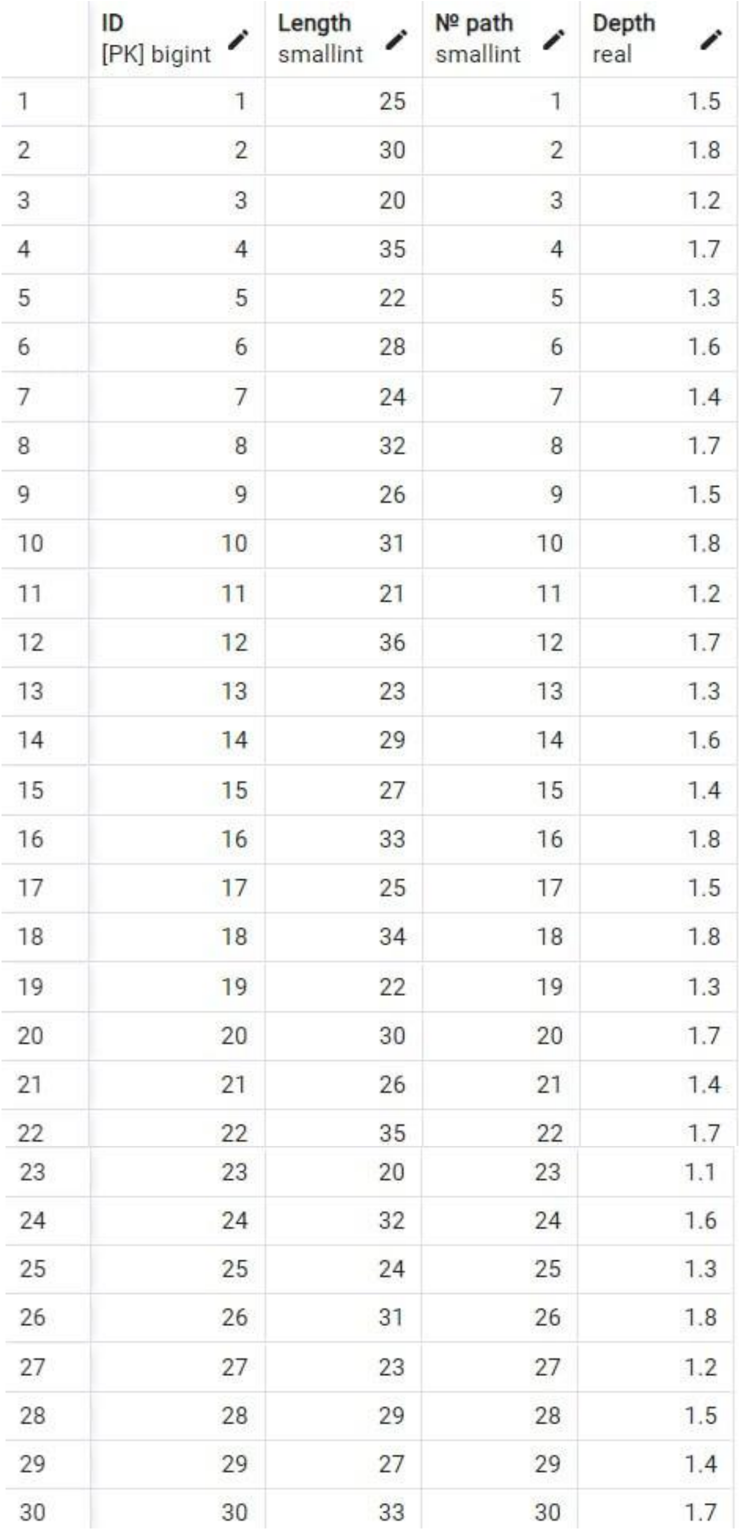


	ID bigint
1	4
2	9
3	7
4	6
5	3
6	1
7	5
8	2
9	8

1.4 Таблица “Swimming pool”

Первоначальный вид таблицы получим с помощью запроса:

```
SELECT * FROM “Swimming pool”
```

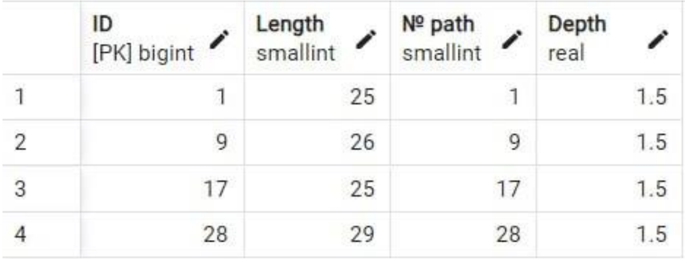


	ID [PK] bigint	Length smallint	№ path smallint	Depth real
1	1	25	1	1.5
2	2	30	2	1.8
3	3	20	3	1.2
4	4	35	4	1.7
5	5	22	5	1.3
6	6	28	6	1.6
7	7	24	7	1.4
8	8	32	8	1.7
9	9	26	9	1.5
10	10	31	10	1.8
11	11	21	11	1.2
12	12	36	12	1.7
13	13	23	13	1.3
14	14	29	14	1.6
15	15	27	15	1.4
16	16	33	16	1.8
17	17	25	17	1.5
18	18	34	18	1.8
19	19	22	19	1.3
20	20	30	20	1.7
21	21	26	21	1.4
22	22	35	22	1.7
23	23	20	23	1.1
24	24	32	24	1.6
25	25	24	25	1.3
26	26	31	26	1.8
27	27	23	27	1.2
28	28	29	28	1.5
29	29	27	29	1.4
30	30	33	30	1.7

1. Создать временную таблицу, в которую занести все бассейны с глубиной 1.5

```
WITH "Pools" AS (SELECT * FROM "Swimming pool" WHERE
"Depth" = '1.5') SELECT * FROM "Pools"
```

Результат выполнения запроса:

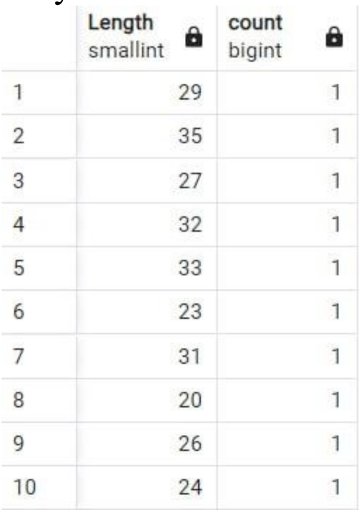


	ID [PK] bigint	Length smallint	№ path smallint	Depth real
1	1	25	1	1.5
2	9	26	9	1.5
3	17	25	17	1.5
4	28	29	28	1.5

2. Вывести длину и количество бассейнов, количество дорожек которых больше 20

```
SELECT "Length", COUNT(*) FROM "Swimming pool" WHERE
"№ path" > 20 GROUP BY "Length"
```

Результат выполнения запроса:

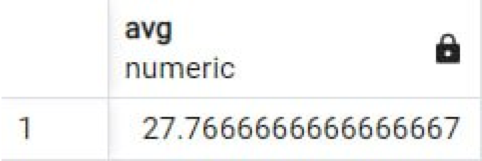


	Length smallint	count bigint
1	29	1
2	35	1
3	27	1
4	32	1
5	33	1
6	23	1
7	31	1
8	20	1
9	26	1
10	24	1

3. Вывести среднюю длину дорожек

```
SELECT AVG("Length") FROM "Swimming pool"
```

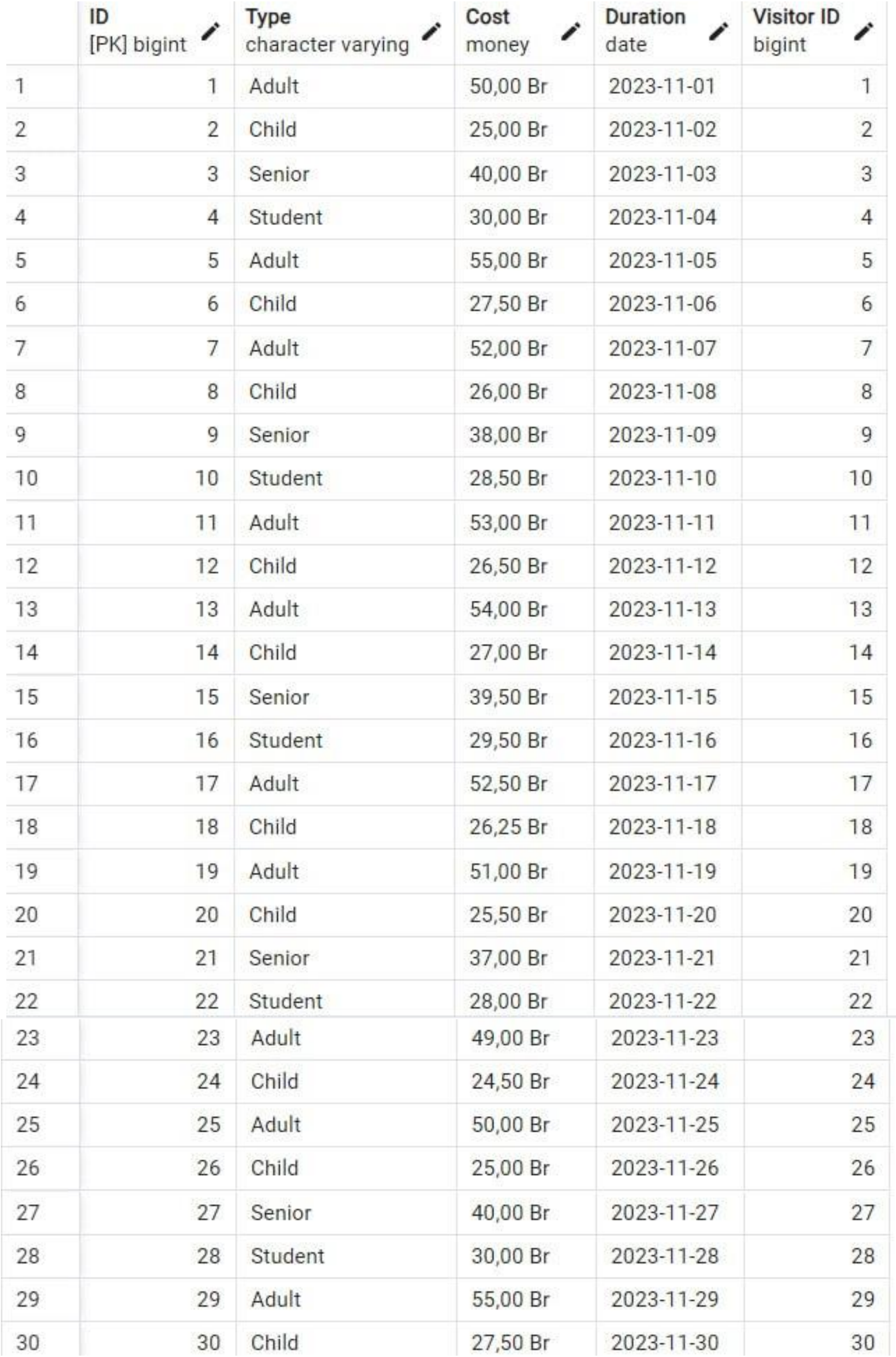
Результат выполнения запроса:



	avg numeric
1	27.7666666666666667

1.5 Таблица “Ticket”

Первоначальный вид таблицы получим с помощью запроса:
`SELECT * FROM “Ticket”`

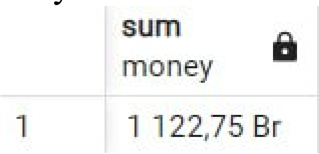


	ID [PK] bigint	Type character varying	Cost money	Duration date	Visitor ID bigint
1	1	Adult	50,00 Br	2023-11-01	1
2	2	Child	25,00 Br	2023-11-02	2
3	3	Senior	40,00 Br	2023-11-03	3
4	4	Student	30,00 Br	2023-11-04	4
5	5	Adult	55,00 Br	2023-11-05	5
6	6	Child	27,50 Br	2023-11-06	6
7	7	Adult	52,00 Br	2023-11-07	7
8	8	Child	26,00 Br	2023-11-08	8
9	9	Senior	38,00 Br	2023-11-09	9
10	10	Student	28,50 Br	2023-11-10	10
11	11	Adult	53,00 Br	2023-11-11	11
12	12	Child	26,50 Br	2023-11-12	12
13	13	Adult	54,00 Br	2023-11-13	13
14	14	Child	27,00 Br	2023-11-14	14
15	15	Senior	39,50 Br	2023-11-15	15
16	16	Student	29,50 Br	2023-11-16	16
17	17	Adult	52,50 Br	2023-11-17	17
18	18	Child	26,25 Br	2023-11-18	18
19	19	Adult	51,00 Br	2023-11-19	19
20	20	Child	25,50 Br	2023-11-20	20
21	21	Senior	37,00 Br	2023-11-21	21
22	22	Student	28,00 Br	2023-11-22	22
23	23	Adult	49,00 Br	2023-11-23	23
24	24	Child	24,50 Br	2023-11-24	24
25	25	Adult	50,00 Br	2023-11-25	25
26	26	Child	25,00 Br	2023-11-26	26
27	27	Senior	40,00 Br	2023-11-27	27
28	28	Student	30,00 Br	2023-11-28	28
29	29	Adult	55,00 Br	2023-11-29	29
30	30	Child	27,50 Br	2023-11-30	30

1. Вывести суммарную стоимость всех абонементов

```
SELECT SUM("Cost") FROM "Ticket"
```

Результат выполнения запроса:

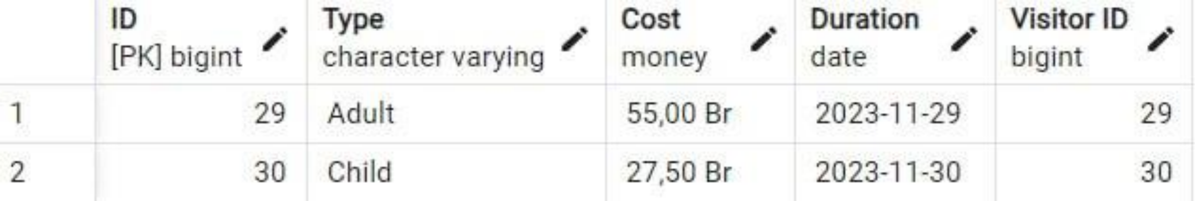


	sum money
1	1 122,75 Br

2. Вывести абонементы, чей срок далее 28.11.2023

```
SELECT * FROM "Ticket"
WHERE "Duration" > '%2023-11-28%'
```

Результат выполнения запроса:

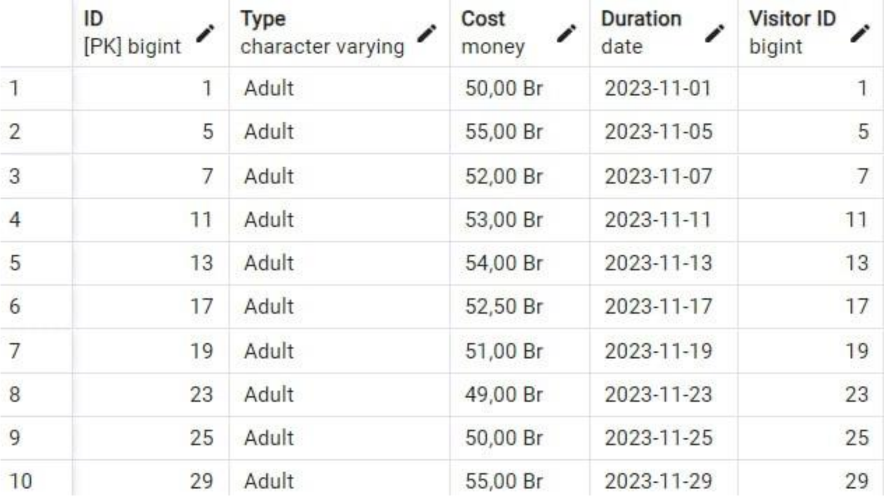


	ID [PK] bigint	Type character varying	Cost money	Duration date	Visitor ID bigint
1	29	Adult	55,00 Br	2023-11-29	29
2	30	Child	27,50 Br	2023-11-30	30

3. Вывести взрослые абонементы, чьи цены больше самого дорогого детского

```
SELECT * FROM "Ticket" WHERE "Type" = 'Adult' AND
"Cost" > (SELECT MAX("Cost") FROM "Ticket" WHERE
"Type" = 'Child')
```

Результат выполнения запроса:

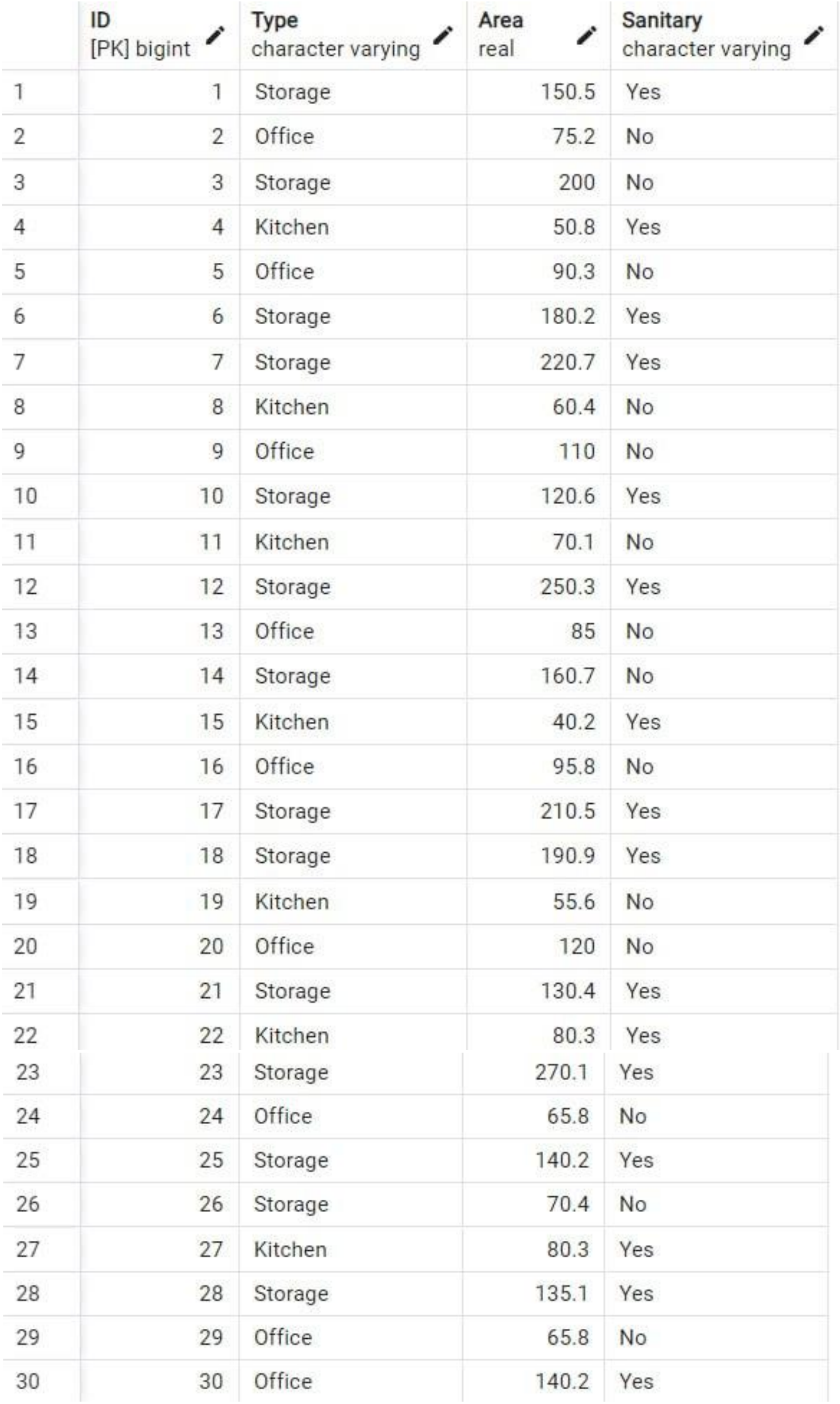


	ID [PK] bigint	Type character varying	Cost money	Duration date	Visitor ID bigint
1	1	Adult	50,00 Br	2023-11-01	1
2	5	Adult	55,00 Br	2023-11-05	5
3	7	Adult	52,00 Br	2023-11-07	7
4	11	Adult	53,00 Br	2023-11-11	11
5	13	Adult	54,00 Br	2023-11-13	13
6	17	Adult	52,50 Br	2023-11-17	17
7	19	Adult	51,00 Br	2023-11-19	19
8	23	Adult	49,00 Br	2023-11-23	23
9	25	Adult	50,00 Br	2023-11-25	25
10	29	Adult	55,00 Br	2023-11-29	29

1.6 Таблица “Auxiliary premises”

Первоначальный вид таблицы получим с помощью запроса:

```
SELECT * FROM “Auxiliary premises”
```

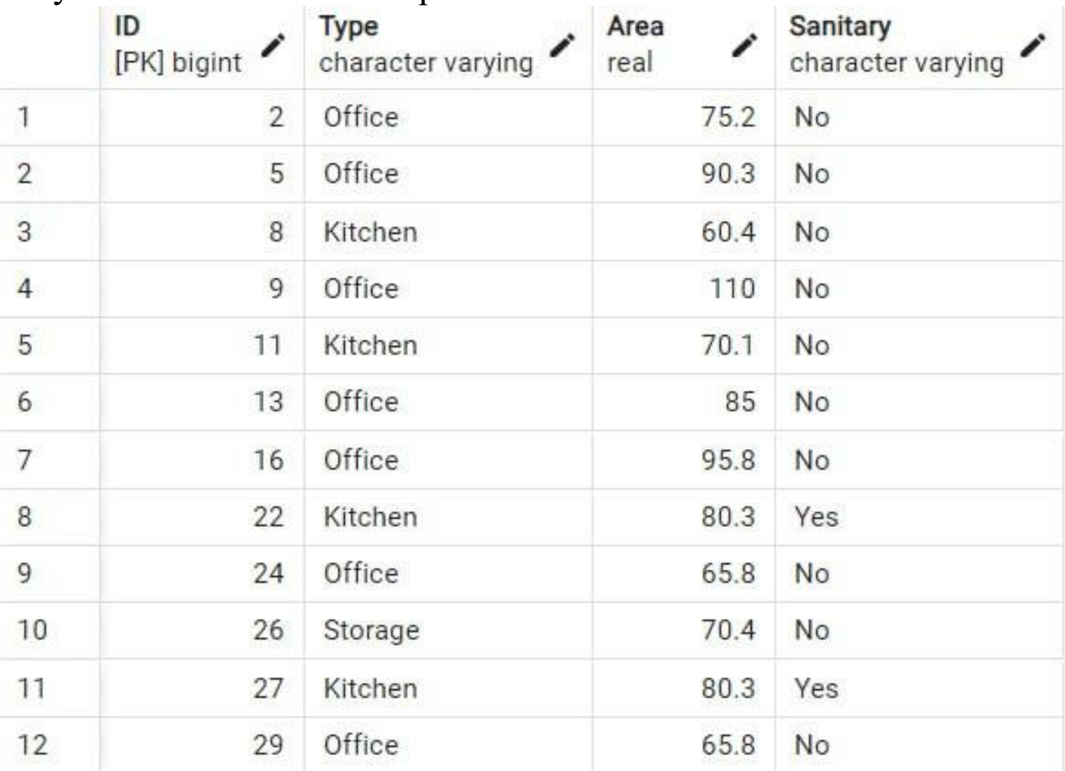


	ID [PK] bigint	Type character varying	Area real	Sanitary character varying
1	1	Storage	150.5	Yes
2	2	Office	75.2	No
3	3	Storage	200	No
4	4	Kitchen	50.8	Yes
5	5	Office	90.3	No
6	6	Storage	180.2	Yes
7	7	Storage	220.7	Yes
8	8	Kitchen	60.4	No
9	9	Office	110	No
10	10	Storage	120.6	Yes
11	11	Kitchen	70.1	No
12	12	Storage	250.3	Yes
13	13	Office	85	No
14	14	Storage	160.7	No
15	15	Kitchen	40.2	Yes
16	16	Office	95.8	No
17	17	Storage	210.5	Yes
18	18	Storage	190.9	Yes
19	19	Kitchen	55.6	No
20	20	Office	120	No
21	21	Storage	130.4	Yes
22	22	Kitchen	80.3	Yes
23	23	Storage	270.1	Yes
24	24	Office	65.8	No
25	25	Storage	140.2	Yes
26	26	Storage	70.4	No
27	27	Kitchen	80.3	Yes
28	28	Storage	135.1	Yes
29	29	Office	65.8	No
30	30	Office	140.2	Yes

1. Вывести помещения, чья площадь между 60 и 110

```
SELECT * FROM "Auxiliary premises" WHERE "Area"
BETWEEN 60 AND 110
```

Результат выполнения запроса:

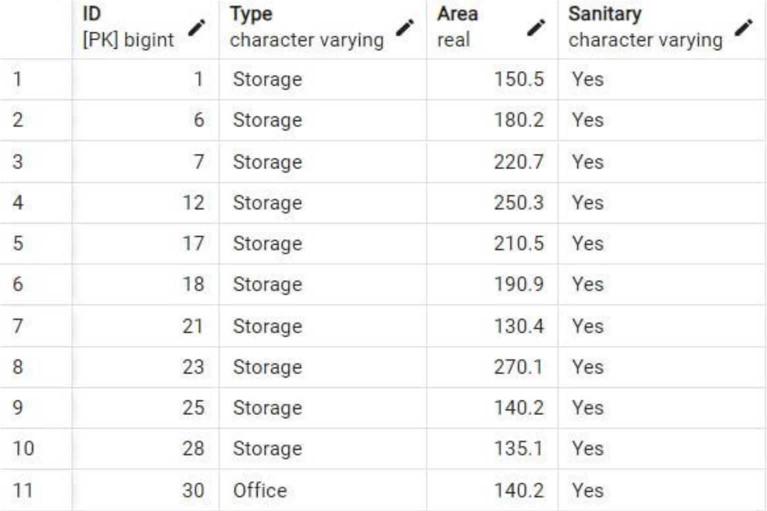


	ID [PK] bigint	Type character varying	Area real	Sanitary character varying
1	2	Office	75.2	No
2	5	Office	90.3	No
3	8	Kitchen	60.4	No
4	9	Office	110	No
5	11	Kitchen	70.1	No
6	13	Office	85	No
7	16	Office	95.8	No
8	22	Kitchen	80.3	Yes
9	24	Office	65.8	No
10	26	Storage	70.4	No
11	27	Kitchen	80.3	Yes
12	29	Office	65.8	No

2. Помещения, чьи площади больше средней и есть санитарные требования

```
SELECT * FROM "Auxiliary premises" WHERE "Area" >
((SELECT AVG("Area") FROM "Auxiliary premises")) AND
"Sanitary" = 'Yes'
```

Результат выполнения запроса:

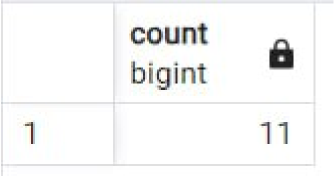


	ID [PK] bigint	Type character varying	Area real	Sanitary character varying
1	1	Storage	150.5	Yes
2	6	Storage	180.2	Yes
3	7	Storage	220.7	Yes
4	12	Storage	250.3	Yes
5	17	Storage	210.5	Yes
6	18	Storage	190.9	Yes
7	21	Storage	130.4	Yes
8	23	Storage	270.1	Yes
9	25	Storage	140.2	Yes
10	28	Storage	135.1	Yes
11	30	Office	140.2	Yes

3. Посчитать количество помещений типа склад с санитарными условиями

```
SELECT COUNT(*) FROM "Auxiliary premises" WHERE
"Type" LIKE 'Storage' AND "Sanitary" = 'Yes'
```

Результат выполнения запроса:



	count bigint 🔒
1	11

ВЫВОД

В ходе выполнения лабораторной работы были изучены и использованы на практике различные операторы для взаимодействия с базой данных такие как AVG, COUNT, MAX, MIN, SUM, GROUP BY оператора SELECT, HAVING в предложении GROUP BY, Подзапросы в операторе SELECT, IN, NOT IN, ANY, ALL, INTERSECT, UNION, WITH.