

Министерство образования Республики Беларусь

Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

ОТЧЕТ  
по лабораторной работе № 4  
на тему  
РЕАЛИЗАЦИЯ SQL-ЗАПРОСОВ НА ПРОСТУЮ ВЫБОРКУ ДАННЫХ  
«ШКОЛА»

Студент:

А.Н. Климович

Преподаватель:

Д.В. Куприянова

МИНСК 2024

## **1 ЦЕЛЬ РАБОТЫ**

Формирование SQL-операторов для выборки данных из таблиц с использованием многотабличных запросов, подзапросов и секций WHERE, ORDER BY и других.

## **2 ВЫПОЛНЕНИЕ РАБОТЫ**

### **2.1 Порядок выполнения работы**

1. Получить у преподавателя задания по вашей собственной схеме данных, созданной в лабораторной работе №2 и реализованной в виде таблиц в СУБД в лабораторной работе №3. Создать запросы по заданиям (минимум 15).

2. Правила выполнения заданий:

- для каждого задания создать реализацию в виде одного оператора выборки, в котором **НЕЛЬЗЯ** использовать подзапросы и группировку данных (это еще будет в другой лабораторной работе);
- при использовании соединений нескольких таблиц обратить внимание на условие задания и сделать выбор между внутренним и внешним соединениями и их вариантами реализации;
- перед запуском запроса на выполнение, изучить данные в используемых запросом таблицах, и, если требуется, добавить в вашу схему необходимые новые данные, чтобы результат выборки был контролируемым и не пустым;
- выполнить запрос и проанализировать его результат – если есть расхождения между ожидаемыми данными и результатом запроса, то есть повод задуматься о проверке правильности выполнения этого задания.

3. Необходимо сделать простые запросы с каждым оператором, который описан в методическом пособии. Нужно поработать с каждым оператором минимум 2 раза.

## 2.2 Выполнение запросов

### 2.2.1 Запрос 1

Вывести ФИО учеников, у которых имя начинается с буквы “А”.

```
SELECT
    main_scheme.student.first_name,
    main_scheme.student.last_name,
    main_scheme.student.middle_name
FROM main_scheme.student
WHERE first_name LIKE 'A%';
```

	id [PK] integer	first_name character varying (100)	last_name character varying (100)	patronymic character varying (100)	passport_id character	date_of_birth date	class_id bigint	residential_address_id bigint
1	1	Иван	Иванов	Иванович	AB1234567CD89	2007-05-10	1	1
2	2	Петр	Петров	Петрович	EF9876543GH21	2008-02-15	2	2
3	3	Анна	Сидорова	Александровна	KL4567890MN32	2009-08-20	3	3
4	4	Мария	Козлова	Игоревна	OP6543210QR54	2010-07-25	4	4
5	5	Сергей	Васильев	Дмитриевич	ST0987654UV76	2011-03-30	5	5
6	6	Елена	Никитина	Павловна	WX2345678YZ98	2012-11-05	6	6
7	7	Алексей	Григорьев	Андреевич	CD8765432AB09	2013-09-12	7	7
8	8	Ольга	Павлова	Сергеевна	EF2109876GH43	2014-04-18	8	8
9	9	Дмитрий	Федоров	Владимирович	IJ0987654KL76	2015-12-23	9	9
10	10	Наталья	Иванова	Михайловна	MN5432109OP65	2016-10-28	10	10
11	11	Ирина	Петрова	Алексеевна	QR7654321ST87	2017-06-02	11	11
12	12	Андрей	Сидоров	Петрович	UV1098765WX32	2018-01-07	12	12
13	13	Виктория	Смирнова	Анатольевна	YZ8765432WX10	2007-09-12	13	13
14	14	Артем	Кузнецов	Иванович	BC5432109DE87	2008-07-17	14	14
15	15	Татьяна	Морозова	Алексеевна	FG3210987HI54	2009-04-22	15	15
16	16	Григорий	Новиков	Петрович	JK1098765LM32	2010-12-27	16	16
17	17	Василиса	Зайцева	Дмитриевна	NO9876543PQ21	2011-10-02	17	17
18	18	Егор	Павлов	Сергеевич	RS7654321TU98	2012-05-08	18	18
19	19	Людмила	Соколова	Игоревна	VW5432109XY76	2013-01-13	19	19
20	20	Александра	Иванова	Александровна	AB2109876CD43	2014-08-18	20	20
21	21	Денис	Петров	Владимирович	EF8765432GH65	2015-06-23	21	21
22	22	Екатерина	Сидорова	Михайловна	IJ6543210KL87	2016-02-28	22	22
23	23	Антон	Козлов	Павлович	MN4321098OP54	2017-11-03	23	23
24	24	Олег	Васильев	Андреевич	QR0987654ST32	2018-07-08	24	24
25	25	Марина	Григорьева	Петровна	UV7654321WX09	2007-03-13	25	25
26	26	Николай	Никитин	Сергеевич	CD3210987AB76	2008-12-18	26	26
27	27	Елена	Федорова	Ивановна	GH9876543IJ43	2009-10-23	27	27
28	28	Игорь	Иванов	Александрович	KL6543210MN65	2010-06-28	28	28
29	29	Лариса	Петрова	Владимировна	OP2109876QR87	2011-02-03	29	29
30	30	Владимир	Сидоров	Михайлович	ST4321098UV54	2012-09-08	30	30
31	31	Анастасия	Козлова	Павловна	WX6543210VZ32	2013-07-13	21	30

Рисунок 2.1 – Таблица student, участвующая в запросе 1

Результат выполнения запроса представлен на рисунке 3.1.

Query		Query History	
1	SELECT		
2	main_scheme.student.first_name,		
3	main_scheme.student.last_name,		
4	main_scheme.student.middle_name		
5	FROM main_scheme.student		
6	WHERE first_name LIKE 'A%'		
Data Output		Messages	Notifications
<div> <div>+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗑️</div> <div>📦</div> <div>⬇️</div> <div>📈</div> </div>			
	first_name character varying (100) 🔒	last_name character varying (100) 🔒	middle_name character varying (100) 🔒
1	Анна	Сидорова	Александровна
2	Алексей	Григорьев	Андреевич
3	Андрей	Сидоров	Петрович
4	Артем	Кузнецов	Иванович
5	Александра	Иванова	Александровна
6	Антон	Козлов	Павлович
7	Анастасия	Козлова	Павловна

Рисунок 3.1 – Результат выполнения запроса 1

## 2.2.2 Запрос 2

Вывести уникальные названия городов из таблицы адресов проживания.

```
SELECT DISTINCT city
FROM main_scheme.residential_address;
```

	id [PK] integer	city character varying (100)	street character varying (100)	house_number integer
1	1	Нью-Йорк	Бродвей	123
2	2	Лос-Анджелес	Сансет-бульвар	456
3	3	Чикаго	Мичиган-авеню	789
4	4	Хьюстон	Мэйн-стрит	321
5	5	Финикс	Центральная авеню	654
6	6	Филадельфия	Маркет-стрит	987
7	7	Сан-Антонио	Риверволк	147
8	8	Сан-Диего	Оушен-бульвар	258
9	9	Даллас	Элм-стрит	369
10	10	Сан-Хосе	Первая улица	753
11	11	Остин	Конгресс-авеню	852
12	12	Джэксонвилл	Бей-стрит	963
13	13	Индианаполис	Монумент-серкуль	159
14	14	Сан-Франциско	Ломбард-стрит	357
15	15	Колумбус	Хай-стрит	468
16	16	Форт-Уэрт	Санданс-сквер	579
17	17	Шарлотт	Трейд-стрит	684
18	18	Сизтл	Пайк-стрит	795
19	19	Денвер	Колфакс-авеню	876
20	20	Вашингтон	Пенсильвания-авеню	987
21	21	Бостон	Ньюбери-стрит	369
22	22	Нэшвилл	Бродвей	753

23	23	Балтимор	Иннер-Харбор	852
24	24	Оклахома-Сити	Бриктаун	963
25	25	Портленд	Пёрл-дистрикт	159
26	26	Лас-Вегас	Стрип	357
27	27	Милуоки	Уотер-стрит	468
28	28	Альбукерке	Шоссе 66	579
29	29	Тусон	Четвертая авеню	684
30	30	Майами	Оушен Драйв	795
31	31	Минск	Независимости	123
32	32	Логойск	Интернациональная	456
33	33	Витебск	Космонавтов	789
34	34	Гродно	Малинина	321
35	35	Гомель	Могилевская	654
36	36	Гомель	Луговая	987
37	37	Несвиж	Первомайская	147
38	38	Мир	Пушкинская	258
39	39	Брест	Деревянная	369
40	40	Кобрин	Цветочная	753
41	41	Мозырь	Победителей	852
42	42	Орша	Центральная	963
43	43	Орша	Центральная	159
44	44	Кобрин	Лесная	357
45	45	Клецк	Домашняя	468
46	46	Несвиж	Московская	579
47	47	Солигорск	Солевая	684
48	48	Держинск	Горная	795
49	49	Фаниполь	Клубничная	876
50	50	Луинец	Солнечная	987
51	51	Калинковичи	Карла Маркса	369
52	52	Могилев	Центральная	753
53	53	Климовичи	Лесная	852
54	54	Минск	Каменная	963
55	55	Минск	Космонавтов	159
56	56	Минск	Железная	357
57	57	Марьина Горка	Интернациональная	468
58	58	Волковыск	Советская	579
59	59	Гродно	Полевая	684
60	60	Минск	Мирная	795

Рисунок 2.2 – Таблица residential\_address, участвующая в запросе 2

Результат выполнения запроса представлен на рисунке 3.2.

Query		Query History
1	<b>SELECT DISTINCT</b> city <b>FROM</b> main_scheme.residential_address;	
Data Output		Messages
Notifications		
	city	
	character varying (100)	
1	Витебск	
2	Вашингтон	
3	Сиэтл	
4	Мозырь	
5	Тусон	
6	Сан-Хосе	
7	Фаниполь	
8	Лос-Анджелес	
9	Оклахома-Сити	
10	Сан-Диего	
11	Хьюстон	
12	Альбукерке	
13	Шарлотт	
14	Брест	
15	Финикс	
16	Минск	
17	Сан-Франциско	
18	Милуоки	
19	Клецк	
20	Несвиж	
21	Чикаго	
22	Орша	
23	Майами	
24	Портленд	
25	Форт-Уэрт	
26	Гомель	
27	Держинск	
28	Денвер	
29	Остин	
30	Бостон	
31	Калинковичи	
32	Кобрин	
33	Даллас	
34	Могилев	
35	Марына Горка	
36	Логойск	
37	Колумбус	
38	Солигорск	
39	Сан-Антонио	
40	Нэшвилл	
41	Джэксонвилл	
42	Нью-Йорк	
43	Балтимор	
44	Индианаполис	
45	Мир	
46	Гродно	
47	Лунинец	
48	Филадельфия	
49	Волковыск	
50	Лас-Вегас	
51	Климовичи	

Рисунок 3.2 – Результат выполнения запроса 2

### 2.2.3 Запрос 3

Вывести номер класса и букву класса, в которых количество студентов больше 30, отсортированные по номеру класса.

```
SELECT
    class_number,
    class_letter,
    student_count
FROM main_scheme.class
WHERE student_count > 30
ORDER BY class_number;
```

	id [PK] integer	class_number integer	class_letter character	student_count integer
1	1	10	A	24
2	2	3	Б	27
3	3	8	В	32
4	4	4	Г	28
5	5	2	Д	26
6	6	6	Е	30
7	7	7	Ж	31
8	8	5	З	29
9	9	1	И	25
10	10	11	К	25
11	11	9	Л	13
12	12	10	М	19
13	13	1	Н	18
14	14	11	О	26
15	15	11	В	25
16	16	6	П	14
17	17	3	Р	7
18	18	2	С	15
19	19	8	Т	13
20	20	7	У	14
21	21	9	Ф	14
22	22	2	Х	7

Рисунок 2.3 – Таблица class, участвующая в запросе 3

Результат выполнения запроса представлен на рисунке 3.3.

Query

Query History

1

2

3

4

5

6

7

8

SELECT

class\_number,

class\_letter,

student\_count

FROM main\_scheme.class

WHERE student\_count > 30

ORDER BY class\_number;

Data Output

Messages

Notifications

Рисунок 3.2 – Результат выполнения запроса 3

## 2.2.4 Запрос 4

Выбрать все предметы (название и часы), у которых количество часов равно максимальному количеству часов среди всех предметов.

```
SELECT
    name,
    hours
FROM main_scheme.subject
WHERE hours >= ALL
(SELECT hours FROM main_scheme.subject);
```

	id [PK] integer	name character varying (100)	hours integer	classroom integer
1	1	Математика	60	101
2	2	Физика	45	102
3	3	История	30	103
4	4	Литература	45	104
5	5	Биология	60	105
6	6	Химия	45	106
7	7	География	30	107
8	8	Английский язык	60	108
9	9	Французский язык	45	109
10	10	Немецкий язык	30	110
11	11	Испанский язык	45	111
12	12	Информатика	60	112
13	13	Физкультура	30	113
14	14	Музыка	45	114
15	15	ИЗО	60	115
16	16	Трудовое обучение	30	116
17	17	Обществознание	45	117
18	18	Право	60	118
19	19	Экономика	30	119
20	20	Психология	45	120
21	21	Философия	60	121
22	22	Религиоведение	30	122



23	23	Хореография	45	123
24	24	Театральное искусство	60	124
25	25	Основы медицины	30	125
26	26	Дизайн	45	126
27	27	Астрономия	60	127
28	28	Геология	30	128
29	29	Архитектура	45	129
30	30	Лингвистика	60	130

Рисунок 2.4 – Таблица subject, участвующая в запросе 4

Результат выполнения запроса представлен на рисунке 3.4.

Query		Query History	
1	SELECT		
2	name,		
3	hours		
4	FROM main_scheme.subject		
5	WHERE hours >= ALL		
6	(SELECT hours FROM main_scheme.subject);		
7			

  

Data Output		Messages		Notifications	
	name		hours		
	character varying (100)		integer		
1	Математика		60		
2	Биология		60		
3	Английский язык		60		
4	Информатика		60		
5	ИЗО		60		
6	Право		60		
7	Философия		60		
8	Театральное искусство		60		
9	Астрономия		60		
10	Лингвистика		60		

Рисунок 3.2 – Результат выполнения запроса 4

### 2.2.5 Запрос 5

Выбрать все адреса проживания, где номер дома находится в диапазоне от 100 до 200.

```
SELECT *
FROM main_scheme.residential_address
WHERE house_number BETWEEN 100 AND 200;
```

	id [PK] integer	city character varying (100)	street character varying (100)	house_number integer
1	1	Нью-Йорк	Бродвей	123
2	2	Лос-Анджелес	Сансет-бульвар	456
3	3	Чикаго	Мичиган-авеню	789
4	4	Хьюстон	Мэйн-стрит	321
5	5	Финикс	Центральная авеню	654
6	6	Филадельфия	Маркет-стрит	987
7	7	Сан-Антонио	Риверволк	147
8	8	Сан-Диего	Оушен-бульвар	258
9	9	Даллас	Элм-стрит	369
10	10	Сан-Хосе	Первая улица	753
11	11	Остин	Конгресс-авеню	852
12	12	Джэксонвилл	Бей-стрит	963
13	13	Индианаполис	Монумент-серкуль	159
14	14	Сан-Франциско	Ломбард-стрит	357
15	15	Колумбус	Хай-стрит	468
16	16	Форт-Уэрт	Санданс-сквер	579
17	17	Шарлотт	Трейд-стрит	684
18	18	Сиэтл	Пайк-стрит	795
19	19	Денвер	Колфакс-авеню	876
20	20	Вашингтон	Пенсильвания-авеню	987
21	21	Бостон	Ньюбери-стрит	369
22	22	Нэшвилл	Бродвей	753
23	23	Балтимор	Иннер-Харбор	852
24	24	Оклахома-Сити	Бриктаун	963
25	25	Портленд	Пёрл-дистрикт	159
26	26	Лас-Вегас	Стрип	357
27	27	Милуоки	Уотер-стрит	468
28	28	Альбукерке	Шоссе 66	579
29	29	Тусон	Четвертая авеню	684
30	30	Майами	Оушен Драйв	795
31	31	Минск	Независимости	123
32	32	Логойск	Интернациональная	456
33	33	Витебск	Космонавтов	789
34	34	Гродно	Малинина	321
35	35	Гомель	Могилевская	654
36	36	Гомель	Луговая	987
37	37	Несвиж	Первомайская	147
38	38	Мир	Пушкинская	258
39	39	Брест	Деревянная	369
40	40	Кобрин	Цветочная	753
41	41	Мозырь	Победителей	852
42	42	Орша	Центральная	963
43	43	Орша	Центральная	159
44	44	Кобрин	Лесная	357
45	45	Клецк	Домашняя	468
46	46	Несвиж	Московская	579
47	47	Солигорск	Солевая	684
48	48	Держинск	Горная	795
49	49	Фаниполь	Клубничная	876
50	50	Лунинец	Солнечная	987
51	51	Калинковичи	Карла Маркса	369
52	52	Могилев	Центральная	753
53	53	Климовичи	Лесная	852
54	54	Минск	Каменная	963
55	55	Минск	Космонавтов	159
56	56	Минск	Железная	357
57	57	Марьина Горка	Интернациональная	468
58	58	Волковыск	Советская	579
59	59	Гродно	Полевая	684
60	60	Минск	Мирная	795

Рисунок 2.5 – Таблица residential\_address, участвующая в запросе 5

Результат выполнения запроса представлен на рисунке 3.5.

Query

Query History

1

SELECT \*

2

FROM main\_scheme.residential\_address

3

WHERE house\_number BETWEEN 100 AND 200;

4

Data Output

Messages

Notifications

	id [PK] integer	city character varying (100)	street character varying (100)	house_number integer
1	1	Нью-Йорк	Бродвей	123
2	7	Сан-Антонио	Риверволк	147
3	13	Индианаполис	Монумент-серкуль	159
4	25	Портленд	Пёрл-дистрикт	159
5	31	Минск	Независимости	123
6	37	Несвиж	Первомайская	147
7	43	Орша	Центральная	159
8	55	Минск	Космонавтов	159

Рисунок 3.2 – Результат выполнения запроса 5

### 2.2.6 Запрос 6

Выбрать расписание занятий по вторникам с информацией о классах, используя INNER JOIN.

```
SELECT
    week_day,
    class_number,
    class_letter
FROM main_scheme.schedule
INNER JOIN main_scheme.class
ON schedule.class_id = main_scheme.class.id
WHERE week_day = 'Вторник';
```

	id [PK] integer	class_id bigint	subject_id bigint	
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	5	5	5
6	6	6	6	6
7	7	7	7	7
8	8	8	8	8
9	9	9	9	9
10	10	10	10	10
11	11	11	11	11
12	12	12	12	12
13	13	13	13	13
14	14	14	14	14
15	15	15	15	15
16	16	16	16	16
17	17	17	17	17
18	18	18	18	18
19	19	19	19	19
20	20	20	20	20
21	21	21	21	21
22	22	22	22	22
23	23	23	23	23
24	24	24	24	24
25	25	25	25	25
26	26	26	26	26
27	27	27	27	27
28	28	28	28	28
29	29	29	29	29
30	30	30	30	30

	id [PK] integer	class_number integer	class_letter character	student_count integer
1	1	10	A	24
2	2	3	Б	27
3	3	8	В	32
4	4	4	Г	28
5	5	2	Д	26
6	6	6	Е	30
7	7	7	Ж	31
8	8	5	З	29
9	9	1	И	25
10	10	11	К	25
11	11	9	Л	13
12	12	10	М	19
13	13	1	Н	18
14	14	11	О	26
15	15	11	В	25
16	16	6	П	14
17	17	3	Р	7
18	18	2	С	15
19	19	8	Т	13
20	20	7	У	14
21	21	9	Ф	14
22	22	2	Х	7

Рисунок 2.6 – Таблицы schedule и class, участвующие в запросе 6

Результат выполнения запроса представлен на рисунке 3.6.

Query	Query History
1	<b>SELECT</b>
2	week_day,
3	class_number,
4	class_letter
5	<b>FROM</b> main_scheme.schedule
6	<b>INNER JOIN</b> main_scheme.class
7	<b>ON</b> schedule.class_id = main_scheme.class.id
8	<b>WHERE</b> week_day = 'Вторник';
9	

  

Data Output	Messages	Notifications
<div> <div>+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗑️</div> <div>📦</div> <div>⬇️</div> <div>📈</div> </div>		
week_day	class_number	class_letter
character varying (20) 🔒	integer 🔒	character 🔒
1	Вторник	8 В
2	Вторник	11 К
3	Вторник	9 Л
4	Вторник	7 У
5	Вторник	2 Ц

Рисунок 3.2 – Результат выполнения запроса 6

### 2.2.7 Запрос 7

Вывести фамилию и имя сотрудников, которых зовут “Алексей”, их дату рождения, идентификационный номер паспорта и их адреса проживания (город и улицу), включая тех сотрудников, у которых нет адреса.

```

SELECT
    first_name,
    last_name,
    date_of_birth,
    passport_id,
    city,
    street
FROM main_scheme.employee
LEFT OUTER JOIN main_scheme.residential_address
ON employee.residential_address_id =
    residential_address.id;

```

id	city	street	house_number
[PK] integer	character varying (100)	character varying (100)	integer
1	Нью-Йорк	Бродвей	123
2	Лос-Анджелес	Санта-буньяр	456
3	Чикаго	Мичиган-авеню	789
4	Хьюстон	Майн-стрит	321
5	Финикс	Центральная-авеню	654
6	Вашингтон	Маркет-стрит	987
7	Сан-Франциско	Ривер-стрит	147
8	Сан-Диего	Оуэн-буньяр	258
9	Даллас	Элм-стрит	369
10	Сан-Хосе	Первая-улица	753
11	Остин	Конгресс-авеню	852
12	Джексонвилл	Бай-стрит	963
13	Индианаполис	Монумент-серкьюл	159
14	Сан-Франциско	Ломбард-стрит	357
15	Колумбус	Хай-стрит	468
16	Форт-Уэрт	Сандано-сквер	579
17	Шарлотт	Трейлс-стрит	684
18	Окленд	Пайк-стрит	795
19	Денвер	Конфесс-авеню	876
20	Вашингтон	Пенсильвания-авеню	987
21	Бостон	Ньюбери-стрит	369
22	Нью-Йорк	Бродвей	753
23	Балтимор	Импер-Харбор	852
24	Оленинград	Брян-улиц	963
25	Портленд	Парк-дистрикт	159
26	Лос-Вегас	Стрип	357
27	Миннеаполис	Уинчестер-стрит	468
28	Атланта	Вилсон-бл	579
29	Тусон	Честер-авеню	684
30	Майами	Оуэн-Дэвид	795
31	Минск	Независимости	123
32	Лодзь	Интернациональная	456
33	Витебск	Космонавтов	789
34	Гродно	Малинина	321
35	Гомель	Молотовская	654
36	Гомель	Луговая	987
37	Несвиж	Первомайская	147
38	Мир	Пушкинская	258
39	Брест	Деревянная	369
40	Кобрын	Центральная	753
41	Мозырь	Победителей	852
42	Орша	Центральная	963
43	Орша	Центральная	159
44	Кобрын	Лесная	357
45	Клец	Домовая	468
46	Молода	Молодская	579
47	Слоним	Славянская	684
48	Дригачи	Горная	795
49	Фининь	Курбичевская	876
50	Лининь	Солнечная	987
51	Калининград	Карла Маркса	369
52	Могилев	Центральная	753
53	Климовичи	Лесная	852
54	Минск	Кавеневская	963
55	Минск	Космонавтов	159
56	Минск	Железнодорожная	357
57	Минск	Интернациональная	468
58	Волово	Советская	579
59	Гродно	Полная	684
60	Минск	Маршала	795

id	first_name	last_name	middle_name	passport_id	date_of_birth	residential_address_id
[PK] integer	character varying (100)	character varying (100)	character varying (100)	character	date	bigint
1	Иван	Глебович	Иванович	AB1234547CD89	1978-05-15	31
2	Петр	Петров	Петрович	EF9876543UH21	1998-02-15	42
3	Анна	Сидорова	Александровна	KL6567890MN32	1989-08-20	43
4	Мария	Козлова	Игоревна	OP6543230QR54	1971-07-25	34
5	Сергей	Васильев	Дмитриевич	ST0187654UV76	1990-03-30	35
6	Елена	Никитина	Павловна	WX2345478YZ98	1980-11-05	36
7	Алексей	Григорьев	Андреевич	CD4765432AB09	1981-09-12	37
8	Ольга	Павлова	Сергеевна	EF2109376GH43	1982-04-18	38
9	Дмитрий	Федоров	Владимирович	IK0987654KL76	1982-12-23	39
10	Наталья	Иванова	Михайловна	MN5332109OP65	1985-10-28	31
11	Ирина	Петрова	Алексеевна	QR7653321ST87	1989-06-02	41
12	Андрей	Сидоров	Петрович	UV1098365WX32	1975-01-07	42
13	Виктория	Смирнова	Анатолеевна	ZZ8765432WX10	2000-09-12	33
14	Артем	Кузнецов	Иванович	BC5432409DE87	1999-07-17	14
15	Татьяна	Морозова	Алексеевна	FG4210987HI54	1973-04-22	25
16	Григорий	Новиков	Петрович	JK1058765LM32	1974-12-27	16
17	Василиса	Зайцева	Дмитриевна	NO5876543PQ21	1987-10-02	57
18	Егор	Павлов	Сергеевич	RS7654320TU98	1984-05-08	58
19	Людмила	Соколова	Игоревна	VW5442109XY76	1966-01-13	49
20	Александра	Иванова	Александровна	AB2409876CD43	1983-08-18	30
21	Денис	Петров	Владимирович	EF8765462GH65	1979-06-23	44
22	Екатерина	Сидорова	Михайловна	IJ6643210KL87	1970-02-28	42
23	Антон	Козлов	Павлович	MN4321098OP64	1998-11-03	43
24	Олег	Васильев	Андреевич	QR0987664ST32	1996-07-08	54
25	Марина	Григорьева	Петровна	UV7634321WX09	1991-03-13	54
26	Николай	Никитин	Сергеевич	CD3213987AB76	2001-12-18	56
27	Елена	Федорова	Ивановна	GH9876533IJ43	2000-10-23	57
28	Игорь	Иванов	Александрович	KL6533210MN65	2000-06-28	48
29	Лариса	Петрова	Владимировна	OP2309876QR87	1972-02-03	49
30	Владимир	Сидоров	Михайлович	ST4921098UV54	1994-09-08	43
31	Анастасия	Козлова	Павловна	WX6549210YZ32	1995-07-13	39

Рисунок 2.7 – Таблицы residential\_address и employee, участвующие в запросе 7

Результат выполнения запроса представлен на рисунке 3.7.

Query	Query History
1	<b>SELECT</b>
2	first_name,
3	last_name,
4	date_of_birth,
5	passport_id,
6	city,
7	street
8	<b>FROM</b> main_scheme.employee
9	<b>LEFT OUTER JOIN</b> main_scheme.residential_address
10	<b>ON</b> employee.residential_address_id =
11	residential_address.id
12	<b>WHERE</b> first_name = 'Алексей';

Data Output	Messages	Notifications
<div> <div>first_name</div> <div>character varying (100)</div> </div> <div> <div>last_name</div> <div>character varying (100)</div> </div> <div> <div>date_of_birth</div> <div>date</div> </div> <div> <div>passport_id</div> <div>character</div> </div> <div> <div>city</div> <div>character varying (100)</div> </div> <div> <div>street</div> <div>character varying (100)</div> </div>		
1	Алексей	Григорьев
	1981-09-12	CD4765432AB09
	Несвиж	Первомайская

Рисунок 3.2 – Результат выполнения запроса 7

## 2.2.8 Запрос 8

Выбрать информацию о знании предмета с информацией о предметах, включая предметы, по которым нет информации о знании и у которых количество часов равно 45.

```
SELECT *
FROM main_scheme.knowledge_of_subject
RIGHT OUTER JOIN main_scheme.subject
ON knowledge_of_subject.subject_id =
    subject.id
WHERE hours = 45;
```

	id [PK] integer	name character varying (100)	hours integer	classroom integer
1	1	Математика	60	101
2	2	Физика	45	102
3	3	История	30	103
4	4	Литература	45	104
5	5	Биология	60	105
6	6	Химия	45	106
7	7	География	30	107
8	8	Английский язык	60	108
9	9	Французский язык	45	109
10	10	Немецкий язык	30	110
11	11	Испанский язык	45	111
12	12	Информатика	60	112
13	13	Физкультура	30	113
14	14	Музыка	45	114
15	15	ИЗО	60	115
16	16	Трудовое обучение	30	116
17	17	Обществознание	45	117
18	18	Право	60	118
19	19	Экономика	30	119
20	20	Психология	45	120
21	21	Философия	60	121
22	22	Религиоведение	30	122
23	23	Хореография	45	123
24	24	Театральное искусство	60	124
25	25	Основы медицины	30	125
26	26	Дизайн	45	126
27	27	Астрономия	60	127
28	28	Геология	30	128
29	29	Архитектура	45	129
30	30	Лингвистика	60	130

	employee_id [PK] bigint	subject_id bigint
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15
16	16	16
17	17	17
18	18	18
19	19	19
20	20	20
21	21	21
22	22	22
23	23	23
24	24	24
25	25	25
26	26	26
27	27	27
28	28	28
29	29	29
30	30	30

Рисунок 2.8 – Таблицы subject и knowledge\_of\_subject, участвующие в запросе 8

Результат выполнения запроса представлен на рисунке 3.8.

Query Query History	
1	SELECT *
2	FROM main_scheme.knowledge_of_subject
3	RIGHT OUTER JOIN main_scheme.subject
4	ON knowledge_of_subject.subject_id =
5	subject.id
6	WHERE hours = 45;
7	
8	
Data Output Messages Notifications	
employee_id	subject_id
bigint	bigint
id	name
integer	character varying (100)
hours	classroom
integer	integer
1	2
2	4
3	6
4	9
5	11
6	14
7	17
8	20
9	23
10	26
11	29
	2
	4
	6
	9
	11
	14
	17
	20
	23
	26
	29
	Физика
	Литература
	Химия
	Французский язык
	Испанский язык
	Музыка
	Обществознание
	Психология
	Хореография
	Дизайн
	Архитектура

Рисунок 3.2 – Результат выполнения запроса 8

## 2.2.9 Запрос 9

Выбрать информацию о должностях сотрудников с информацией о доступных должностях, включая те должности, которые не заняты сотрудниками, при этом ставка должна быть в диапазоне от 0.8 до 1.

```
SELECT *
FROM main_scheme.employee_position
FULL OUTER JOIN main_scheme.job_position
ON employee_position.job_position_id =
job_position.id
WHERE rate BETWEEN 0.8 AND 1;
```

employee_id	job_position_id
(PK) bigint	bigint
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30

id	name	rate	salary
(PK) integer	character varying (100)	real	money
1	Директор	1	100 000,00 ?
2	Заместитель директора	0.9	90 000,00 ?
3	Учитель математики	0.8	80 000,00 ?
4	Учитель русского языка	0.8	80 000,00 ?
5	Учитель физики	0.8	80 000,00 ?
6	Учитель химии	0.8	80 000,00 ?
7	Учитель биологии	0.8	80 000,00 ?
8	Учитель истории	0.8	80 000,00 ?
9	Учитель географии	0.8	80 000,00 ?
10	Учитель английского языка	0.8	80 000,00 ?
11	Учитель физической культуры	0.8	80 000,00 ?
12	Учитель искусства	0.8	80 000,00 ?
13	Учитель музыки	0.8	80 000,00 ?
14	Учитель информатики	0.8	80 000,00 ?
15	Учитель технологии	0.8	80 000,00 ?
16	Учитель начальных классов	0.8	80 000,00 ?
17	Педагог-психолог	0.7	70 000,00 ?
18	Социальный педагог	0.7	70 000,00 ?
19	Логопед	0.7	70 000,00 ?
20	Библиотекарь	0.7	70 000,00 ?
21	Администратор	0.6	60 000,00 ?
22	Секретарь	0.6	60 000,00 ?
23	Бухгалтер	0.6	60 000,00 ?
24	Сторож	0.5	50 000,00 ?
25	Уборщик	0.5	50 000,00 ?
26	Воспитатель	0.7	70 000,00 ?
27	Младшая	0.6	60 000,00 ?
28	Дежурный по физической культуре	0.7	70 000,00 ?
29	Дежурный по охране	0.6	60 000,00 ?
30	Дежурный по уборке	0.5	50 000,00 ?

Рисунок 3.9 – Таблицы employee\_position и job\_position, участвующие в запросе 9

Результат выполнения запроса представлен на рисунке 3.9.

Query Query History	
1	SELECT *
2	FROM main_scheme.employee_position
3	FULL OUTER JOIN main_scheme.job_position
4	ON employee_position.job_position_id =
5	job_position.id
6	WHERE rate BETWEEN 0.8 AND 1;
Data Output Messages Notifications	
employee_id	job_position_id
bigint	bigint
id	id
integer	integer
name	name
character varying (100)	character varying (100)
rate	rate
real	real
salary	salary
money	money
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16

Рисунок 3.2 – Результат выполнения запроса 9

## 2.2.10 Запрос 10

Получить комбинацию всех записей из журнала оценок с каждым учеником 25 числа месяца в базе данных и вывести ФИО, оценку и дату оценки ученика, при этом оценка должна быть равна 9 и отчество ученика должно быть “Иванович” или “Ивановна”.

```

SELECT
    first_name,
    last_name,
    middle_name,
    grade,
    grade_date
FROM main_scheme.gradebook
CROSS JOIN main_scheme.student
WHERE EXTRACT(DAY FROM grade_date) = 25
      AND middle_name LIKE 'Иван%'
      AND grade = 9;

```



id	employee_id	student_id	subject_id	grade_date	grade
[PK] integer	bigint	bigint	bigint	date	smallint
1	1	1	1	2023-01-15	8
2	2	2	2	2023-02-20	7
3	3	3	3	2023-03-25	9
4	4	4	4	1 2023-04-30	6
5	5	5	5	2 2023-05-05	8
6	6	6	6	3 2023-06-10	7
7	7	7	7	1 2023-07-15	9
8	8	8	8	2 2023-08-20	5
9	9	9	9	3 2023-09-25	8
10	10	1	1	1 2023-10-30	7
11	11	1	1	2 2023-11-05	6
12	12	1	2	3 2023-12-10	9
13	13	3	3	1 2024-01-15	8
14	14	4	4	2 2024-02-20	7
15	15	5	5	3 2024-03-25	9
16	16	6	6	1 2024-04-30	6
17	17	7	7	2 2024-05-05	8
18	18	8	8	3 2024-06-10	7
19	19	1	9	1 2024-07-15	9
20	20	2	2	2 2024-08-20	6
21	21	2	2	3 2024-09-25	8
22	22	2	2	1 2024-10-30	7
23	23	2	3	2 2024-11-05	6
24	24	4	4	3 2024-12-10	9
25	25	5	12	1 2025-01-15	8
26	26	6	12	2 2025-02-20	7
27	27	7	17	3 2025-03-25	9
28	28	8	18	1 2025-04-30	6
29	29	9	19	2 2025-05-05	8
30	30	3	13	3 2025-06-10	7

id	first_name	last_name	patronymic	passport_id	date_of_birth	class_id	residential_address_id
[PK] integer	character varying (100)	character varying (100)	character varying (100)	character	date	bigint	bigint
1	Иван	Иванов	Иванович	4812345678901	2007-01-18	1	1
2	Артём	Кузнецов	Кузнецов	5123456789012	2009-02-15	2	2
3	Елена	Федорова	Александровна	614567890123	2009-08-29	3	3
4	Иван	Колесов	Игоревич	0176543210987	2014-03-25	4	4
5	Евгений	Васильев	Дмитриевич	1234567890123456	2011-05-08	5	5
6	Елена	Мельникова	Павловна	98765432109876	2012-11-05	6	6
7	Александр	Григорьев	Александр	0087654321098	2019-01-12	7	7
8	Ольга	Павлова	Сергеевна	4127890123456	2011-09-18	8	8
9	Дмитрий	Волков	Викторович	145678901234567	2015-05-03	9	9
10	Николай	Волков	Михайлович	8888888888888888	2016-10-28	10	10
11	Иван	Петров	Александр	0876543210987	2017-06-02	11	11
12	Андрей	Сидоров	Петрович	0170987654321	2018-01-07	12	12
13	Виктор	Смирнов	Александр	1234567890123456	2019-09-12	13	13
14	Анна	Королева	Ивановна	987654321098765	2008-07-17	14	14
15	Евгений	Морозов	Александр	FG2345678901	2008-06-22	15	15
16	Владимир	Николаев	Петрович	3456789012345	2018-10-27	16	16
17	Виктор	Иванов	Дмитриевич	807654321098765	2017-10-02	17	17
18	Степан	Павлов	Сергеевич	9876543210987654	2012-05-09	18	18
19	Ирина	Соловьева	Игоревна	V98765432109876	2019-01-13	19	19
20	Александр	Иванов	Александрович	48018087654321	2014-06-18	20	20
21	Дмитрий	Петров	Владимирович	8888888888888888	2015-09-28	21	21
22	Екатерина	Сидорова	Николаевна	145678901234567	2019-02-29	22	22
23	Елена	Колесов	Павловна	54567890123456	2017-11-03	23	23
24	Виктор	Васильев	Александр	Q808765432109	2018-07-08	24	24
25	Михаил	Григорьев	Петрович	017654321098765	2007-03-13	25	25
26	Николай	Иванов	Сергеевич	00027890123456	2008-12-18	26	26
27	Евгений	Федоров	Иванов	04567890123456	2005-10-23	27	27
28	Иван	Иванов	Александрович	KL543210987654	2010-06-28	28	28
29	Виктор	Петров	Владимирович	OP1278901234567	2011-02-03	29	29
30	Владимир	Смирнов	Михайлович	0112278901234	2012-09-09	30	30
31	Александр	Колесов	Павловна	W987654321098765	2019-07-18	31	31

Рисунок 2.10 – Таблицы gradebook и student, участвующие в запросе 10

Результат выполнения запроса представлен на рисунке 3.10.

Query

Query History

1

**SELECT**

2

first\_name,

3

last\_name,

4

middle\_name,

5

grade,

6

grade\_date|

7

**FROM** main\_scheme.gradebook

8

**CROSS JOIN** main\_scheme.student

9

**WHERE** EXTRACT(DAY FROM grade\_date) = 25

10

AND middle\_name LIKE 'Иван%'

11

AND grade = 9;

Data Output

Messages

Notifications

≡

📄

▼

📋

▼

🗑️

🔍

📥

⬇️

📈

	first_name character varying (100) 🔒	last_name character varying (100) 🔒	middle_name character varying (100) 🔒	grade smallint 🔒	grade_date date 🔒
1	Иван	Иванов	Иванович	9	2023-03-25
2	Артём	Кузнецов	Иванович	9	2023-03-25
3	Елена	Федорова	Ивановна	9	2023-03-25
4	Иван	Иванов	Иванович	9	2024-03-25
5	Артём	Кузнецов	Иванович	9	2024-03-25
6	Елена	Федорова	Ивановна	9	2024-03-25
7	Иван	Иванов	Иванович	9	2023-03-25
8	Артём	Кузнецов	Иванович	9	2023-03-25
9	Елена	Федорова	Ивановна	9	2023-03-25

Рисунок 3.2 – Результат выполнения запроса 10

## 2.2.11 Запрос 11

Выбрать всех студентов, родившихся между 2000 и 2010 годами.

```
SELECT *
FROM main_scheme.student
WHERE date_of_birth BETWEEN '2000-01-01'
```

AND '2010-12-31';

id [PK] integer	first_name character varying (100)	last_name character varying (100)	middle_name character varying (100)	passport_id character	date_of_birth date	class_id bigint	residential_address_id bigint
1	Иван	Иванов	Иванович	AB1234567CD89	2007-05-10	1	1
2	Петр	Петров	Петрович	EF9876543GH21	2008-02-15	2	2
3	Анна	Сидорова	Александровна	KL4567890MN32	2009-08-20	3	3
4	Мария	Козлова	Игоревна	OP6543210QR54	2010-07-25	4	4
5	Светлана	Васильева	Анатолевна	YZ8765432WX10	2011-03-30	5	5
6	Артём	Кузнецов	Иванович	BC5432109DE87	2012-11-05	6	6
7	Татьяна	Морозова	Алексеевна	FG3210987HI54	2013-01-12	7	7
8	Григорий	Новиков	Петрович	JK1098765LM32	2015-10-23	8	8
9	Марина	Григорьева	Петровна	UV7654321WX09	2016-10-28	9	9
10	Николай	Никитин	Сергеевич	CD3210987AB76	2017-04-22	10	10
11	Елена	Федорова	Ивановна	GH9876543IJ43	2017-04-22	11	11
12	Игорь	Иванов	Александрович	KL6543210MN65	2018-01-07	12	12
13	Виктор	Смирнов	Анатолевич	YZ8765432WX10	2017-09-12	13	13
14	Артём	Кузнецов	Иванович	BC5432109DE87	2018-07-17	14	14
15	Татьяна	Морозова	Алексеевна	FG3210987HI54	2019-04-22	15	15
16	Григорий	Новиков	Петрович	JK1098765LM32	2018-10-27	16	16
17	Мария	Козлова	Игоревна	OP6543210QR54	2011-03-30	17	17
18	Светлана	Васильева	Анатолевна	YZ8765432WX10	2012-11-05	18	18
19	Артём	Кузнецов	Иванович	BC5432109DE87	2013-01-12	19	19
20	Татьяна	Морозова	Алексеевна	FG3210987HI54	2014-03-18	20	20
21	Григорий	Новиков	Петрович	JK1098765LM32	2015-06-28	21	21
22	Марина	Григорьева	Петровна	UV7654321WX09	2016-10-28	22	22
23	Анна	Сидорова	Александровна	KL4567890MN32	2017-11-03	23	23
24	Мария	Козлова	Игоревна	OP6543210QR54	2018-07-28	24	24
25	Марина	Григорьева	Петровна	UV7654321WX09	2007-03-13	25	25
26	Николай	Никитин	Сергеевич	CD3210987AB76	2008-12-18	26	26
27	Елена	Федорова	Ивановна	GH9876543IJ43	2009-10-23	27	27
28	Игорь	Иванов	Александрович	KL6543210MN65	2010-06-28	28	28
29	Виктор	Смирнов	Анатолевич	YZ8765432WX10	2011-03-30	29	29
30	Владимир	Сидоров	Михайлович	ST90123456789	2012-09-08	30	30
31	Александра	Колесова	Павловна	WXY9876543210	2013-07-13	31	31

Рисунок 2.11 – Таблица student, участвующая в запросе 11

Результат выполнения запроса представлен на рисунке 3.11.

Query		Query History						
1	SELECT *							
2	FROM main_scheme.student							
3	WHERE date_of_birth BETWEEN '2000-01-01'							
4	AND '2010-12-31';							
Data Output		Messages						
	id [PK] integer	first_name character varying (100)	last_name character varying (100)	middle_name character varying (100)	passport_id character	date_of_birth date	class_id bigint	residential_address_id bigint
1	1	Иван	Иванов	Иванович	AB1234567CD89	2007-05-10	1	1
2	2	Петр	Петров	Петрович	EF9876543GH21	2008-02-15	2	2
3	3	Анна	Сидорова	Александровна	KL4567890MN32	2009-08-20	3	3
4	4	Мария	Козлова	Игоревна	OP6543210QR54	2010-07-25	4	4
5	13	Виктория	Смирнова	Анатолевна	YZ8765432WX10	2007-09-12	13	13
6	14	Артём	Кузнецов	Иванович	BC5432109DE87	2008-07-17	14	14
7	15	Татьяна	Морозова	Алексеевна	FG3210987HI54	2009-04-22	15	15
8	16	Григорий	Новиков	Петрович	JK1098765LM32	2010-12-27	16	16
9	25	Марина	Григорьева	Петровна	UV7654321WX09	2007-03-13	25	25
10	26	Николай	Никитин	Сергеевич	CD3210987AB76	2008-12-18	26	26
11	27	Елена	Федорова	Ивановна	GH9876543IJ43	2009-10-23	27	27
12	28	Игорь	Иванов	Александрович	KL6543210MN65	2010-06-28	28	28

Рисунок 3.2 – Результат выполнения запроса 11

## 2.2.12 Запрос 12

Выбрать все классы от 7 до 11 включительно и отсортировать их по номеру класса по убыванию.

```
SELECT *
FROM main_scheme.class
WHERE class_number BETWEEN 7 AND 11
ORDER BY class_number DESC;
```

	id [PK] integer	class_number integer	class_letter character	student_count integer
1	1	10	A	24
2	2	3	Б	27
3	3	8	В	32
4	4	4	Г	28
5	5	2	Д	26
6	6	6	Е	30
7	7	7	Ж	31
8	8	5	З	29
9	9	1	И	25
10	10	11	К	25
11	11	9	Л	13
12	12	10	М	19
13	13	1	Н	18
14	14	11	О	26
15	15	11	В	25
16	16	6	П	14
17	17	3	Р	7
18	18	2	С	15
19	19	8	Т	13
20	20	7	У	14
21	21	9	Ф	14
22	22	2	Х	7

Рисунок 2.12 – Таблица class, участвующая в запросе 12

Результат выполнения запроса представлен на рисунке 3.12.

Query		Query History			
1	SELECT *				
2	FROM main_scheme.class				
3	WHERE class_number BETWEEN 7 AND 11				
4	ORDER BY class_number DESC;				
5					
Data Output		Messages			
	id [PK] integer	class_number integer	class_letter character	student_count integer	
1	10	11	K	25	
2	15	11	B	25	
3	14	11	O	26	
4	1	10	A	24	
5	12	10	M	19	
6	31	10	Б	14	
7	11	9	Л	13	
8	21	9	Ф	14	
9	30	9	A	14	
10	32	8	Б	13	
11	3	8	В	32	
12	19	8	Т	13	
13	27	7	Э	12	
14	20	7	У	14	
15	7	7	Ж	31	

Рисунок 3.2 – Результат выполнения запроса 12

### 2.2.13 Запрос 13

Выбрать все предметы, в названии которых есть слово “Язык”.

```
SELECT *
FROM main_scheme.subject
```

WHERE name LIKE '%язык%';

	id [PK] integer	name character varying (100)	hours integer	classroom integer
1	1	Математика	60	101
2	2	Физика	45	102
3	3	История	30	103
4	4	Литература	45	104
5	5	Биология	60	105
6	6	Химия	45	106
7	7	География	30	107
8	8	Английский язык	60	108
9	9	Французский язык	45	109
10	10	Немецкий язык	30	110
11	11	Испанский язык	45	111
12	12	Информатика	60	112
13	13	Физкультура	30	113
14	14	Музыка	45	114
15	15	ИЗО	60	115
16	16	Трудовое обучение	30	116
17	17	Обществознание	45	117
18	18	Право	60	118
19	19	Экономика	30	119
20	20	Психология	45	120
21	21	Философия	60	121
22	22	Религиоведение	30	122
23	23	Хореография	45	123
24	24	Театральное искусство	60	124
25	25	Основы медицины	30	125
26	26	Дизайн	45	126
27	27	Астрономия	60	127
28	28	Геология	30	128
29	29	Архитектура	45	129
30	30	Лингвистика	60	130

Рисунок 2.13 – Таблица subject, участвующая в запросе 13

Результат выполнения запроса представлен на рисунке 3.13.

Query		Query History	
1	SELECT *		
2	FROM main_scheme.subject		
3	WHERE name LIKE '%язык%';		
4			
5			
Data Output		Messages	
		Notifications	
	id [PK] integer	name character varying (100)	hours integer
1	8	Английский язык	60
2	9	Французский язык	45
3	10	Немецкий язык	30
4	11	Испанский язык	45

Рисунок 3.2 – Результат выполнения запроса 13

## 2.2.14 Запрос 14

Выбрать адреса проживания в Минске и отсортировать их по номеру дома.

```
SELECT *
FROM main_scheme.residential_address
WHERE city = 'Минск'
ORDER BY house_number;
```

id [PK] integer	city character varying (100)	street character varying (100)	house_number integer
1	1 Нью-Йорк	Бродвей	123
2	2 Лос-Анджелес	Сансет-бульвар	456
3	3 Чикаго	Мичиган-авеню	789
4	4 Хьюстон	Майн-стрит	321
5	5 Финикс	Центральная авеню	654
6	6 Филадельфия	Маркет-стрит	987
7	7 Сан-Антонио	Риверволк	147
8	8 Сан-Диего	Оушен-бульвар	258
9	9 Даллас	Элм-стрит	369
10	10 Сан-Хосе	Первая улица	753
11	11 Остин	Конгресс-авеню	852
12	12 Джэксонвилл	Бей-стрит	963
13	13 Индианаполис	Монумент-серкль	159
14	14 Сан-Франциско	Ломбард-стрит	357
15	15 Колумбус	Хай-стрит	468
16	16 Форт-Уэрт	Санданс-сквер	579
17	17 Шарлотт	Трейд-стрит	684
18	18 Сиэтл	Пайк-стрит	795
19	19 Денвер	Колфакс-авеню	876
20	20 Вашингтон	Пенсильвания-авеню	987
21	21 Бостон	Ньюбери-стрит	369
22	22 Нью-Йорк	Бродвей	753
23	23 Балтимор	Иннер-Харбор	852
24	24 Оклахома-Сити	Бриктан	963
25	25 Портленд	Перл-дистрикт	159
26	26 Лас-Вегас	Стрип	357
27	27 Милуоки	Уотер-стрит	468
28	28 Альбукерке	Шоссе 66	579
29	29 Тусон	Четвертая авеню	684
30	30 Майами	Оушен-Драйв	795
31	31 Минск	Независимости	123
32	32 Логойск	Интернациональная	456
33	33 Витебск	Космонавтов	789
34	34 Гродно	Малинина	321
35	35 Гомель	Могилевская	654
36	36 Гомель	Луговая	987
37	37 Несвиж	Первомайская	147
38	38 Мир	Пушкинская	258
39	39 Брест	Деревянная	369
40	40 Кобрин	Цветочная	753
41	41 Мозырь	Победителей	852
42	42 Орша	Центральная	963
43	43 Орша	Центральная	159
44	44 Кобрин	Лесная	357
45	45 Клецк	Домашняя	468
46	46 Несвиж	Московская	579
47	47 Солигорск	Солевая	684
48	48 Держисск	Горная	795
49	49 Фаняполь	Клубничная	876
50	50 Луиннец	Солнечная	987
51	51 Калинковичи	Карла Маркса	369
52	52 Могилев	Центральная	753
53	53 Климовичи	Лесная	852
54	54 Минск	Каменная	963
55	55 Минск	Космонавтов	159
56	56 Минск	Железная	357
57	57 Марьяна Горка	Интернациональная	468
58	58 Волковыск	Советская	579
59	59 Гродно	Полевая	684
60	60 Минск	Мирная	795

Рисунок 2.14 – Таблица residential\_address, участвующая в запросе 14

Результат выполнения запроса представлен на рисунке 3.14.

Query		Query History		
1	SELECT *			
2	FROM main_scheme.residential_address			
3	WHERE city = 'Минск'			
4	ORDER BY house_number;			
5				

Data Output		Messages		Notifications	
	id [PK] integer	city character varying (100)	street character varying (100)	house_number integer	
1	31	Минск	Независимости	123	
2	55	Минск	Космонавтов	159	
3	56	Минск	Железная	357	
4	60	Минск	Мирная	795	
5	54	Минск	Каменная	963	

Рисунок 3.2 – Результат выполнения запроса 14

### 2.2.15 Запрос 15

Выбрать расписание занятий с информацией о предметах, где аудитория равна 101.

```
SELECT *
FROM main_scheme.schedule
INNER JOIN main_scheme.subject
ON schedule.subject_id = subject.id
WHERE classroom = 101;
```

	id [PK] integer	name character varying (100)	hours integer	classroom integer
1	1	Математика	60	101
2	2	Физика	45	102
3	3	История	30	103
4	4	Литература	45	104
5	5	Биология	60	105
6	6	Химия	45	106
7	7	География	30	107
8	8	Английский язык	60	108
9	9	Французский язык	45	109
10	10	Немецкий язык	30	110
11	11	Испанский язык	45	111
12	12	Информатика	60	112
13	13	Физкультура	30	113
14	14	Музыка	45	114
15	15	ИЗО	60	115
16	16	Трудовое обучение	30	116
17	17	Обществознание	45	117
18	18	Право	60	118
19	19	Экономика	30	119
20	20	Психология	45	120
21	21	Философия	60	121
22	22	Религиоведение	30	122
23	23	Хореография	45	123
24	24	Театральное искусство	60	124
25	25	Основы медицины	30	125
26	26	Дизайн	45	126
27	27	Астрономия	60	127
28	28	Геология	30	128
29	29	Архитектура	45	129
30	30	Лингвистика	60	130

	id [PK] integer	class_id bigint	subject_id bigint
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
10	10	10	10
11	11	11	11
12	12	12	12
13	13	13	13
14	14	14	14
15	15	15	15
16	16	16	16
17	17	17	17
18	18	18	18
19	19	19	19
20	20	20	20
21	21	21	21
22	22	22	22
23	23	23	23
24	24	24	24
25	25	25	25
26	26	26	26
27	27	27	27
28	28	28	28
29	29	29	29
30	30	30	30

Рисунок 2.15 – Таблицы subject и schedule, участвующие в запросе 15

Результат выполнения запроса представлен на рисунке 3.15.

```
Query    Query History
1  SELECT *
2  FROM main_scheme.schedule
3  INNER JOIN main_scheme.subject
4  ON schedule.subject_id = subject.id
5  WHERE classroom = 101;
6

Data Output  Messages  Notifications
[Icons: Add, Filter, Sort, Refresh, Download, etc.]
id integer [lock]  week_day character varying (20) [lock]  class_id bigint [lock]  subject_id bigint [lock]  id integer [lock]  name character varying (100) [lock]  hours integer [lock]  classroom integer [lock]
1 1 Понедельник 1 1 1 Математика 60 101
```

Рисунок 3.2 – Результат выполнения запроса 15

### 2.2.16 Запрос 16

Выбрать всех сотрудников (ФИО) и их адреса проживания из Гродно, включая адреса без сотрудников.

```
SELECT *
FROM main_scheme.employee
RIGHT OUTER
    JOIN main_scheme.residential_address
ON employee.residential_address_id =
    residential_address.id
WHERE city = 'Гродно';
```

id	first_name	last_name	middle_name	passport_id	date_of_birth	residential_address_id
PK integer	character varying (100)	character varying (100)	character varying (100)	character	date	bigint
1	Иван	Глебович	Иванович	AB1234547CDB9	1978-05-15	3
2	Петр	Петрович	Петров	EP9676543JH21	1998-07-05	3
3	Анна	Сидорова	Александровна	KL656789KMN22	1899-08-06	3
4	Мария	Колесова	Игоревна	OP543230QRS4	1971-07-25	2
5	Сергей	Васильев	Дмитриевич	ST0187654V76	1980-03-30	3
6	Елена	Наумова	Павловна	WX2345478Y298	1990-11-05	3
7	Александр	Григорьев	Александр	CD4765432AB89	1981-09-12	3
8	Ольга	Сергеевна	Сергеевна	EF2103767QH43	1982-04-18	3
9	Наталья	Федорова	Владимировна	IK0987654KL76	1982-12-23	3
10	Дмитрий	Иванов	Михайлович	MNS32109OPR5	1985-08-28	3
11	Ирина	Петрова	Александра	OP7633215T87	1989-06-02	3
12	Андрей	Сидоров	Петрович	UV109836MXC22	1975-01-07	3
13	Виктория	Смирнова	Антоньевна	Z2875432VX100	2000-09-12	3
14	Артем	Кузнецов	Иванович	BC543240YOE87	1999-07-17	3
15	Татьяна	Морозова	Александра	FG421987H854	1973-04-22	3
16	Григорий	Новиков	Петрович	JK105876JLM32	1974-12-27	3
17	Василий	Защелко	Дмитриев	NO587654IOP21	1987-10-08	3
18	Егор	Павлов	Сергеевич	RS7654320TU68	1984-05-08	3
19	Людмила	Соколова	Игоревна	VW6452109XY76	1966-01-13	3
20	Александра	Ивановна	Александровна	AB2401876CD43	1983-08-08	3
21	Денис	Петров	Владимирович	EF7854320IH85	1979-06-23	3
22	Екатерина	Сидорова	Михайловна	LM632190KLB87	1970-02-28	3
23	Антон	Колобов	Павлович	MN432109OP54	1999-11-03	3
24	Олег	Васильев	Андреевич	OR0987664ST32	1996-07-08	3
25	Марина	Григорьевна	Петровна	UV7634327KX09	1991-03-13	3
26	Николай	Сергеев	Сергеевич	CD3213987IAB5	2007-12-18	3
27	Евгений	Федоров	Иванович	GI9876533LAB	2000-10-23	3
28	Игорь	Иванов	Александрович	KL532108MN65	2000-06-28	3
29	Лариса	Петрова	Владимировна	OP2209767Q897	1972-03-03	3
30	Виктор	Сидоров	Михайлович	ST470198HJ54	1994-09-08	3
31	Александра	Колобова	Павловна	WX549210VZ32	1995-07-13	3

[illegible]

Рисунок 2.16 – Таблицы employee и residential\_address, участвующие в запросе 16

Результат выполнения запроса представлен на рисунке 3.16.

Query

Query History

1

2

3

4

5

6

7

8

9

10

11

12

SELECT

first\_name,

last\_name,

middle\_name,

city

FROM main\_scheme.employee

RIGHT OUTER

JOIN main\_scheme.residential\_address

ON employee.residential\_address\_id =

residential\_address.id

WHERE city = 'Гродно';

Data Output

Messages

Notifications

Рисунок 3.2 – Результат выполнения запроса 16

## 2.2.17 Запрос 17

Выбрать информацию о должностях сотрудников с информацией о доступных должностях, где оклад больше 80000 тыс. условных денежных единиц.

```
SELECT *
FROM main_scheme.employee_position
FULL OUTER JOIN main_scheme.job_position
ON employee_position.job_position_id =
    job_position.id
WHERE salary > '80000';
```

employee_id	job_position_id	job	salary
1	1	Директор	100 000 000
2	2	Заместитель директора	80 000 000
3	3	Учлен комитета	60 000 000
4	4	Учлен комитета	60 000 000
5	5	Учлен комитета	60 000 000
6	6	Учлен комитета	60 000 000
7	7	Учлен комитета	60 000 000
8	8	Учлен комитета	60 000 000
9	9	Учлен комитета	60 000 000
10	10	Учлен комитета	60 000 000
11	11	Учлен комитета	60 000 000
12	12	Учлен комитета	60 000 000
13	13	Учлен комитета	60 000 000
14	14	Учлен комитета	60 000 000
15	15	Учлен комитета	60 000 000
16	16	Учлен комитета	60 000 000
17	17	Учлен комитета	60 000 000
18	18	Учлен комитета	60 000 000
19	19	Учлен комитета	60 000 000
20	20	Учлен комитета	60 000 000
21	21	Учлен комитета	60 000 000
22	22	Учлен комитета	60 000 000
23	23	Учлен комитета	60 000 000
24	24	Учлен комитета	60 000 000
25	25	Учлен комитета	60 000 000
26	26	Учлен комитета	60 000 000
27	27	Учлен комитета	60 000 000
28	28	Учлен комитета	60 000 000
29	29	Учлен комитета	60 000 000
30	30	Учлен комитета	60 000 000

Рисунок 2.17 – Таблицы employee\_position и job\_position, участвующие в запросе 17



Результат выполнения запроса представлен на рисунке 3.17.

Query

Query History

1

SELECT \*

2

FROM main\_scheme.employee\_position

3

FULL OUTER JOIN main\_scheme.job\_position

4

ON employee\_position.job\_position\_id =

5

job\_position.id |

6

WHERE salary > '80000';

7

Data Output

Messages

Notifications

	employee_id bigint	job_position_id bigint	id integer	name character varying (100)	rate real	salary money
1	1	1	1	Директор	1	100 000,00 ?
2	2	2	2	Заместитель директора	0.9	90 000,00 ?

Рисунок 3.2 – Результат выполнения запроса 17

## 2.2.18 Запрос 18

Выбрать всех учеников из классов, где количество учеников меньше 15.

```
SELECT *
FROM main_scheme.student
WHERE class_id IN
(SELECT id
FROM main_scheme.class
WHERE student_count < 15);
```

id [PK] integer	first_name character varying (100)	last_name character varying (100)	patronymic character varying (100)	passport_id character	date_of_birth date	class_id integer	residential_address_id integer
1	Иван	Иванов	Иванович	481345170289	2007-05-18	1	1
2	Петр	Петров	Петрович	5198155103021	2009-02-15	2	2
3	Анна	Сидорова	Александровна	4142619090022	2009-06-28	3	3
4	Михаил	Колесов	Александр	0794502150254	2010-01-25	4	4
5	Сергей	Васильев	Дмитриевич	072017054076	2011-05-30	5	5
6	Елена	Никитина	Павловна	6923456790298	2012-11-05	6	6
7	Александр	Павлов	Александр	0307050424889	2013-09-12	7	7
8	Ольга	Сидорова	Сергеевна	51421006169483	2013-04-18	8	8
9	Дмитрий	Степанов	Степанович	1108765485176	2015-10-23	9	9
10	Наталья	Иванова	Михайловна	6904231000495	2016-10-28	10	10
11	Илья	Петров	Александр	0875442100057	2017-06-29	11	11
12	Андрей	Сидоров	Петрович	0791087520022	2018-01-07	12	12
13	Виктор	Сидоров	Александр	9426154020030	2007-09-12	13	13
14	Арина	Кручинин	Иванович	8354201000037	2008-05-17	14	14
15	Екатерина	Михайлов	Александр	7322110817684	2008-04-22	15	15
16	Павел	Михайлов	Петрович	4010076261803	2010-10-27	16	16
17	Владимир	Зайцева	Дмитриевич	6028176489221	2011-10-02	17	17
18	Лера	Павлов	Сергеевич	8013542211096	2012-05-09	18	18
19	Иван	Сидоров	Игоревич	5790102100015	2013-01-13	19	19
20	Александр	Иванов	Александрович	4801880762645	2014-06-18	20	20
21	Данил	Петров	Владимирович	8987654305465	2015-06-29	21	21
22	Екатерина	Сидорова	Михайловна	1451621054817	2016-02-28	22	22
23	Алексей	Колесов	Павлович	6942310000054	2017-11-03	23	23
24	Олег	Васильев	Александр	0801676447707	2018-07-08	24	24
25	Мария	Степановна	Петровна	07765443170005	2007-01-13	25	25
26	Николай	Николаев	Сергеевич	0302189834876	2008-10-18	26	26
27	Елена	Филиппов	Ивановна	0488765484243	2009-10-23	27	27
28	Илья	Иванов	Александрович	6555423100063	2010-06-28	28	28
29	Татьяна	Петрова	Владимировна	0721086100067	2011-02-03	29	29
30	Владимир	Сидоров	Михайлович	5142189810154	2012-09-08	30	30
31	Александр	Колесов	Павлович	6955421810232	2013-07-13	31	31

id [PK] integer	class_number integer	class_letter character	student_count integer
1	1	10 A	24
2	2	3 Б	27
3	3	8 В	32
4	4	4 Г	28
5	5	2 Д	26
6	6	6 Е	30
7	7	7 Ж	31
8	8	5 З	29
9	9	1 И	25
10	10	11 К	25
11	11	9 Л	13
12	12	10 М	19
13	13	1 Н	18
14	14	11 О	26
15	15	11 В	25
16	16	6 П	14
17	17	3 Р	20
18	18	2 С	15
19	19	8 Т	13
20	20	7 У	14
21	21	9 Ф	14
22	22	2 Х	7

Рисунок 2.18 – Таблицы student и class, участвующие в запросе 18

Результат выполнения запроса представлен на рисунке 3.18.

Query

Query History

```

1 SELECT *
2 FROM main_scheme.student
3 WHERE class_id IN |
4 (SELECT id
5  FROM main_scheme.class
6  WHERE student_count < 15);

```

Data Output

Messages

Notifications

	id [PK] integer	first_name character varying (100)	last_name character varying (100)	middle_name character varying (100)	passport_id character	date_of_birth date	class_id bigint	residential_address_id bigint
1	11	Ирина	Петрова	Алексеевна	QR7654321ST87	2017-06-02	11	11
2	16	Григорий	Новиков	Петрович	JK1098765LM32	2010-12-27	16	16
3	17	Василиса	Зайцева	Дмитриевна	NO9876543PQ21	2011-10-02	17	17
4	19	Людмила	Соколова	Игоревна	VW5432109XY76	2013-01-13	19	19
5	20	Александра	Иванова	Александровна	AB2109876CD43	2014-08-18	20	20
6	31	Анастасия	Козлова	Павловна	WX6543210YZ32	2013-07-13	21	30
7	21	Денис	Петров	Владимирович	EF8765432GH65	2015-06-23	21	21
8	22	Екатерина	Сидорова	Михайловна	IJ6543210KL87	2016-02-28	22	22
9	23	Антон	Козлов	Павлович	MN4321098OP54	2017-11-03	23	23
10	25	Марина	Григорьева	Петровна	UV7654321WX09	2007-03-13	25	25
11	26	Николай	Никитин	Сергеевич	CD3210987AB76	2008-12-18	26	26
12	27	Елена	Федорова	Ивановна	GH9876543IJ43	2009-10-23	27	27
13	28	Игорь	Иванов	Александрович	KL6543210MN65	2010-06-28	28	28
14	30	Владимир	Сидоров	Михайлович	ST4321098UV54	2012-09-08	30	30

Рисунок 3.2 – Результат выполнения запроса 18

## **4 ВЫВОД**

В ходе лабораторной работы было выполнено создание простых запросов на выборку данных на языке SQL с использованием SELECT, FROM, JOIN, WHERE и ORDER BY. В работе также было рассмотрено использование скалярных функций.