Министерство образования Республики Беларусь

Учреждение образования БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет информационный технологий и управления

Кафедра интеллектуальных информационных технологий

Дисциплина: «Проектировка баз знаний»

Лабораторная работа №1 по теме: «Математические основы реляционных языков»

Студент гр. 121702 Колтович Д.С. Проверила:

Липницкая Н. Г.

Тема

Математические основы реляционных языков

Цель

Освоение основ работы с реляционными базами данных

Задание

- 1. По словесному запросу к базе данных преподавателей, предметов и студенческих групп составить последовательность реляционных операций, необходимых для выполнения этого запроса (текст операций).
- 2. Привести результат выполнения каждой операции в виде отношения (таблицы)

Условие задания

1. Требуется создать базы данных с таблицами, представленными ниже (таблицы 1-4), а также написать запросы к ним

ПРЕПОДАВАТЕЛЬ

ТАБЛИЦА 1.1.

ЛичныйНоме	Фамили	Должност	Кафедра	Специальность	ТелефонДомаш		
р	Я	ь			ний		
221Л	Фролов	Доцент	ЭВМ	АСОИ, ЭВМ	487		
222Л	Костин	Доцент	ЭВМ	ЭВМ	543		
225Л	Бойко	Профессо	АСУ	АСОИ, ЭВМ	112		
		р					
430Л	Глазов	Ассистент	ΤФ	СД	421		
110Л	Петров	Ассистент	Экономи	Международная	324		
			ки	экономика			

ПРЕДМЕТ ТАБЛИЦА 1.2.

КодовыйНомер	НазваниеПредме	КоличествоЧасо	Специальность	Семестр
Предмета	та	В		
12Π	Мини ЭВМ	36	ЭВМ	1
14Π	ПЭВМ	72	ЭВМ	2
17Π	СУБД ПК	48	АСОИ	4
18Π	BKCC	52	АСОИ	6
34Π	Физика	30	СД	6
22Π	Аудит	24	Бухучета	3

СТУДЕНЧЕСКАЯ _ГРУППА

ТАБЛИЦА 1.3.

КодовыйНомер	Название	Количество	Специальность	Фамилия
Группы	Группы	Человек		Старосты
8Γ	Э-12	18	ЭВМ	Иванова
7Γ	Э-15	22	ЭВМ	Сеткин
4Γ	AC-9	24	АСОИ	Балабанов
3Г	AC-8	20	АСОИ	Чижов
17Γ	C-14	29	СД	Амросов
12Γ	M-6	16	Международная	Трубин
			экономика	
10Γ	Б-4	21	Бухучет	Зязюткин

ПРЕПОДАВАТЕЛЬ_ПРЕПОДАЕТ_ПРЕДМЕТЫ_В_ГРУППАХ ТАБЛИЦА 1.4.

КодовыйНом	КодовыйНом	ЛичныйНом	НомерАудитор
ер	ер	ер	ии
Группы	Предмета		
8Г	12Π	222Л	112
8Г	14Π	221Л	220
8Г	17Π	222Л	112
7Γ	14Π	221Л	220
7Γ	17Π	222Л	241
7Γ	18Π	225Л	210
4Γ	12Π	222Л	112
4Γ	18Π	225Л	210
3Г	12Π	222Л	112
3Г	17Π	221Л	241
3Г	18Π	225Л	210
17Γ	12Π	222Л	112
17Γ	22Π	110Л	220
17Γ	34П	430Л	118
12Γ	12Π	222Л	112
12Γ	22Π	110Л	210
10Γ	12Π	222Л	210
10Γ	22Π	110Л	210

2. Требуется создать базы данных с таблицами, представленными ниже (таблицы 1-4), а также написать запросы к ним

Поставщики S

Π#	Имя П	Статус	Город
П1	Петров	20	Москва
П2	Синицин	10	Таллинн
ПЗ	Федоров	30	Таллинн
П4	Чаянов	20	Минск
П5	Крюков	30	Киев

Детали Р

Д#	Имя Д	Цвет	Разме Р	Город
Д1	Болт	Красн ый	12	Москва
Д2	Гайка	Зелена я	17	Минск
ДЗ	Диск	Черны й	17	Вильнюс
Д4	Диск	Черны й	14	Москва
Д5	Корпу с	Красн ый	12	Минск
Д6	Крыш ки	Красн ый	19	Москва

Проекты Ј

ПР#	Имя ПР	Город
ПР1	ИПР1	Минск
ПР2	ИПР2	Таллинн
ПР3	ИПР3	Псков
ПР4	ИПР4	Псков
ПР5	ИПР4	Москва
ПР6	ИПР6	Саратов
ПР7	ИПР7	Москва

Кол-во деталей поставляемых одним поставщиком для одного проекта

Π#	Д#	ПР#	S
П1	Д1	ПР1	200
П1	Д1	ПР2	700
П2	Д3	ПР1	400
П2	Д2	ПР2	200
П2	Д3	ПР3	200
П2	Д3	ПР4	500
П2	Д3	ПР5	600
П2	Д3	ПР6	400
П2	Д3	ПР7	800
П2	Д5	ПР2	100
П3	Д3	ПР1	200
П3	Д4	ПР2	500
Π4	Д6	ПР3	300
Π4	Д6	ПР7	300
П5	Д2	ПР2	200
П5	Д2	ПР4	100
П5	Д5	ПР5	500
П5	Д5	ПР7	100
П5	Д6	ПР2	200
П5	Д1	ПР2	100
П5	Д3	ПР4	200
П5	Д4	ПР4	800
П5	Д5	ПР4	400
П5	Д6	ПР4	500

Выполнение задания

Задание 1.

Запросы для создания таблиц

Таблица 1.1.

```
CREATE TABLE teachers (
   id INT NOT NULL AUTO_INCREMENT,
   personal_number VARCHAR(16) UNIQUE NOT NULL,
   lastname VARCHAR(64) NOT NULL,
   job_title VARCHAR(64) NOT NULL,
   department VARCHAR(16) NOT NULL,
   specialization VARCHAR(64) NOT NULL,
   home_phone VARCHAR(16) NOT NULL,
   PRIMARY KEY (id) );
```

Таблица 1.2.

```
CREATE TABLE subjects (

id INT NOT NULL AUTO_INCREMENT,

code_number VARCHAR(16) NOT NULL UNIQUE,

name VARCHAR(64) NOT NULL,

hours INT NOT NULL,

specialization VARCHAR(64) NOT NULL,

semester INT NOT NULL,

PRIMARY KEY (id)

);
```

Таблица 1.3.

```
CREATE TABLE student_groups (
   id INT NOT NULL AUTO_INCREMENT,
   code_number VARCHAR(16) NOT NULL UNIQUE,
   name VARCHAR(64) NOT NULL,
   number_of_students INT NOT NULL,
   specialization VARCHAR(64) NOT NULL,
   headman_lastname VARCHAR(64) NOT NULL,
   PRIMARY KEY (id)
);
```

Таблица 1.4.

```
CREATE TABLE teacher_teaches_subjects_in_groups (
   id INT NOT NULL AUTO_INCREMENT,
   group_code_number VARCHAR(16) NOT NULL,
   subject_code_number VARCHAR(16) NOT NULL,
   personal_number VARCHAR(64) NOT NULL,
   audit_number INT NOT NULL,
   PRIMARY KEY (id)
  );
```

Запросы для заполнения таблиц

Таблица 1.1.

```
INSERT INTO teachers (personal_number, lastname, job_title, department, specialization, home_phone)

VALUES ('221Л', 'Фролов', 'Доцент', 'ЭВМ', 'АСОИ, ЭМВ', '487'),

('222Л', 'КОСТИН', 'ДОЦЕНТ', 'ЭВМ', 'ЭМВ', '543'),

('225Л', 'БОЙКО', 'Профессор', 'АСУ', 'АСОИ ЭМВ', '112'),

('430Л', 'Глазов', 'АССИСТЕНТ', 'ТФ', 'СД', '421'),

('110Л', 'ПЕТРОВ', 'АССИСТЕНТ', 'ЭКОНОМИКИ', 'МЕЖДУНАРОДНАЯ ЭКОНОМИКА', '324');
```

Таблица 1.2.

```
INSERT INTO subjects ( code_number, name, hours, specialization, semester ) VALUES

('12П', 'Мини ЭВМ', 36, 'ЭВМ', 1),

('14П', 'ПЭВМ', 72, 'ЭВМ', 2),

('17П', 'СУБД ПК', 48, 'АСОИ', 4),

('18П', 'ВКСС', 52, 'АСОИ', 6),

('34П', 'ФИЗИКА', 30, 'СД', 6),

('22П', 'АУДИТ', 24, 'БУХУЧЕТА', 3);
```

Таблица 1.3.

Таблица 1.4.

```
INSERT INTO teacher_teaches_subjects_in_groups ( group_code_number, subject_code_number, personal_number, audit_number ) VALUES

('8\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112),
('8\Gat{Gat}', '17\Gat{Gat}', '22\Gat{Gat}', 22\Gat{Gat}', 24\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 21\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 21\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112\Gat},
('3\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112\Gat),
('3\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112\Gat}),
('17\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112\Gat}),
('17\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112\Gat}),
('17\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112\Gat}),
('12\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112\Gat}),
('12\Gat{Gat}', '12\Gat{Gat}', '22\Gat{Gat}', 112\Gat}),
('10\Gat}', '12\Gat{Gat}', '22\Gat{Gat}', '110\Gat}', 22\Gat{Gat}', '110\Gat}', 22\Gat{Gat}', '110\Gat}', 22\Gat{Gat}', '110\Gat}', '22\Gat}', '110\Gat}', '12\Gat}', '110\Gat}', '110\Gat}', '12\Gat}', '110\Gat}', '110\G
```

Запросы для работы с таблицами

1.1. Получить полную информацию обо всех преподавателях.

Запрос:

```
SELECT

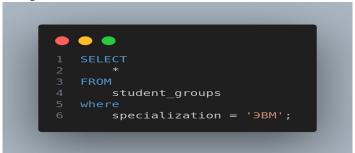
*
FROM
teachers;
```

Результат:

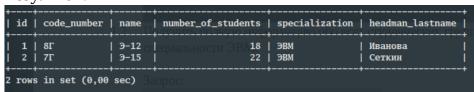
i personal_n	umber lastname	job_title	department	specialization	home_phone
 2 221Л	 Фролов	+ Доцент	 ЭВМ	 АСОИ, ЭВМ	487
3 222Л	Костин	Доцент	∣ ∋BM	ЭBM	543
1 225Л	Бойко	Профессор	ACY	ACOИ, ЭВМ	112
5 430Л	Глазов	Ассистент	Τ Φ	СД	421
5 110Л	Петров	Ассистент	Экономики	Международная экономика	324

1.2. Получить полную информацию обо всех студенческих группах на специальности ЭВМ.

Запрос:



Результат:



1.3. Получить личный номер преподавателя и номера аудиторий, в которых они преподают предмет с кодовым номером 18П.

```
1 SELECT
2    personal_number,
3    audit_number
4 FROM
5    teacher_teaches_subjects_in_groups
6 where
7    subject_code_number = '18N';
```

1.4. Получить номера предметов и названия предметов, которые ведет преподаватель Костин.

Запрос:

```
1 SELECT
2     t.subject_code_number,
3     s.name
4 FROM
5     teacher_teaches_subjects_in_groups t
6     JOIN subjects s ON t.subject_code_number = s.code_number
7     JOIN teachers teach ON teach.personal_number = t.personal_number
8 WHERE
9     teach.lastname = 'KOCTUH';
```

Результат:

1.5. Получить номер группы, в которой ведутся предметы преподавателем Фроловым.

```
1 SELECT
2 group_code_number
3 FROM
4 teacher_teaches_subjects_in_groups t
5 JOIN teachers ON t.personal_number = teachers.personal_number
6 WHERE
7 teachers.lastname = 'ΦροποΒ';
```

1.6. Получить информацию о предметах, которые ведутся на специальности АСОИ.

Запрос:

```
SELECT

*
3 FROM
4 subjects
5 WHERE
6 specialization LIKE 'ACON';
```

Результат:

1.7. Получить информацию о преподавателях, которые ведут предметы на специальности АСОИ.

Запрос:

```
SELECT
DISTINCT teachers.*
from
teacher_teaches_subjects_in_groups t
JOIN teachers ON teachers.personal_number = t.personal_number
JOIN subjects s ON s.code_number = t.subject_code_number
where
s.specialization = 'ACON';
```

Результат:

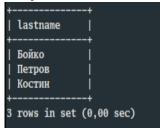
```
id | personal_number | lastname
                                       | job_title
                                                            | department |
                                                                           specialization
                                                                                             | home_phone
  3 | 222Л
                         Костин
                                                              ЭВМ
                                                                                               543
  4 |
      225Л
                         Бойко
                                                              АСУ
                                                                           АСОИ, ЭВМ
                                                                                               112
                                        Профессор
                                                                           АСОИ, ЭВМ
      221Л
                         Фролов
                                        Доцент
                                                              ЭВМ
                                                                                               487
3 rows in set (0,00 sec)
```

1.8. Получить фамилии преподавателей, которые ведут предметы в 210 аудитории.

Запрос:

```
SELECT
DISTINCT t.lastname
FROM
teachers t
JOIN teacher_teaches_subjects_in_groups ttsg ON ttsg.personal_number = t.personal_number
WHERE
ttsg.audit_number = 210;
```

Результат:



1.9. Получить названия предметов и названия групп, которые ведут занятия в аудиториях с 100 по 200.

Запрос:

```
SELECT
s.name AS subject_name,
g.name AS group_name

FROM
subjects s
JOIN teacher_teaches_subjects_in_groups ttsg ON s.code_number = ttsg.subject_code_number
JOIN student_groups g ON ttsg.group_code_number = g.code_number
WHERE
stsg.audit_number BETWEEN 100
AND 200;
```

Результат:

```
subject_name
                group_name
Мини ЭВМ
                3-12
СУБД ПК
                3-12
Мини ЭВМ
                | AC-9
                | AC-8
Мини ЭВМ
Мини ЭВМ
               | C-14
               | C-14
Физика
Мини ЭВМ
                | M-6
```

1.10. Получить пары номеров групп с одной специальности.

```
SELECT
sl.code_number AS group1,
s2.code_number AS group2
FROM
student_groups s1
JOIN student_groups s2 ON sl.specialization = s2.specialization
WHERE
NOT sl.code_number = s2.code_number;
```

1.11. Получить общее количество студентов, обучающихся на специальности ЭВМ.

Запрос:

```
SELECT
SUM(number_of_students) AS num_of_students
FROM
student_groups
WHERE
specialization = '3BM';
```

Результат:

```
| num_of_students |
| num_of_students |
| ------
| 40 |
| ------
| row in set (0,01 sec)
```

1.12. Получить номера преподавателей, обучающих студентов по специальности ЭВМ.

```
SELECT
t.personal_number
from
teachers t
JOIN teacher_teaches_subjects_in_groups ttsg ON t.personal_number = ttsg.personal_number
JOIN student_groups g ON g.code_number = ttsg.group_code_number
WHERE
g.specialization = 'JBM';
```

1.13. Получить номера предметов, изучаемых всеми студенческими группами.

Запрос:

Результат:

```
Empty set (0,00 sec)
```

1.14. Получить фамилии преподавателей, преподающих те же предметы, что и преподаватель преподающий предмет с номером 14П.

```
SELECT DISTINCT t.lastname FROM teachers t

JOIN teacher_teaches_subjects_in_groups ttsg ON t.personal_number = ttsg.personal_number

WHERE

ttsg.subject_code_number IN (

SELECT
subject_code_number

FROM
teacher_teaches_subjects_in_groups

WHERE
personal_number = (

SELECT

DISTINCT personal_number

FROM
teacher_teaches_subjects_in_groups

WHERE
subject_code_number = '14N'

WHERE
subject_code_number = '14N'

)

18 );
```

1.15. Получить информацию о предметах, которые не ведет преподаватель с личным номером 221Л.

Запрос:

```
SELECT
s.*
from
subjects s
JOIN teacher_teaches_subjects_in_groups t on s.code_number = t.subject_code_number
WHERE
t.personal_number != '221Л';
```

Результат:

```
id | code_number | name
                                      | hours | specialization
                                                                 | semester |
                   | Мини ЭВМ
                                          36 | ЭBM
                                                                          1 |
  3 | 17П
                                          48 | ACON
                   | СУБД ПК
                                                                          4 |
  4 | 18Π
                   | BKCC
                                          52 | АСОИ
                                                                          6
   6
      22Π
                                          24 | Бухучета
                                                                          3 |
                   | Аудит
                                          30 | СД
   5
      34∏
                   | Физика
                                                                          6 |
5 rows in set (0,01 sec)
```

1.16. Получить информацию о предметах, которые не изучаются в группе M-6.

```
id |
     code_number | name
                                    | hours | specialization | semester |
                                         72 | ЭBM
     14П
                   ПЭВМ
                                                                         2 |
                                         48 | АСОИ
   | 17II
                   | СУБД ПК
                                                                         4 |
                                         52 | ACOИ
30 | СД
 4 | 18Π
                    BKCC
                                                                         6 |
   | 34II
                                                                         6 |
rows in set (0,00 sec)
```

1.17. Получить информацию о доцентах, преподающих в группах ЗГ и 8Г.

Запрос:

```
SELECT
DISTINCT t.*
FROM
teachers t
JOIN teacher_teaches_subjects_in_groups ttsg ON ttsg.personal_number = t.personal_number
WHERE
t.job_title = 'Доцент'
AND (
ttsg.group_code_number = '3F'
OR ttsg.group_code_number = '8F'
);
```

Результат:

1.18. Получить номера предметов, номера преподавателей, номера групп, в которых ведут занятия преподаватели с кафедры ЭВМ, имеющих специальность АСОИ.

```
1 SELECT
2   ttsg.subject_code_number,
3   ttsg.personal_number,
4   ttsg.group_code_number
5 FROM
6   teacher_teaches_subjects_in_groups ttsg
7   JOIN teachers t ON t.personal_number = ttsg.personal_number
8 WHERE
9   t.specialization LIKE 'ACOU%'
10   AND t.department = 'ЭВМ';
```

1.19. Получить номера групп с такой же специальностью, что и специальность преподавателей.

Запрос:

```
SELECT
g.code_number
FROM
teacher_teaches_subjects_in_groups ttsg
JOIN teachers t ON t.personal_number = ttsg.personal_number
JOIN student_groups g on g.code_number = ttsg.group_code_number
WHERE
LOCATE(g.specialization, t.specialization);
```

Результат:

1.20. Получить номера преподавателей с кафедры ЭВМ, преподающих предметы по специальности, совпадающей со специальностью студенческой группы.

```
SELECT
DISTINCT ttsg.personal_number
FROM
teacher_teaches_subjects_in_groups ttsg
JOIN subjects s ON s.code_number = ttsg.subject_code_number
JOIN teachers t on t.personal_number = ttsg.personal_number
JOIN student_groups g ON g.code_number = ttsg.group_code_number
WHERE
LOCATE('JBM', t.specialization)
AND s.specialization = g.specialization;
```

1.21. Получить специальности студенческой группы, на которых работают преподаватели кафедры АСУ.

Запрос:

```
SELECT
DISTINCT g.specialization
FROM
student_groups g
JOIN teacher_teaches_subjects_in_groups ttsg on ttsg.group_code_number = g.code_number
JOIN teachers t on t.personal_number = ttsg.personal_number
WHERE
t.department = 'ACY';
```

Результат:

```
+----+
| specialization |
+-----+
| ЭВМ |
| АСОИ |
+-----+
2 rows in set (0,00 sec)
```

1.22. Получить номера предметов, изучаемых группой АС-8.

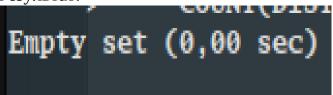
```
1 SELECT
2   ttsg.subject_code_number
3  FROM
4   teacher_teaches_subjects_in_groups ttsg
5   JOIN student_groups g ON ttsg.group_code_number = g.code_number
6  WHERE
7   g.name = 'AC-8';
```

1.23. Получить номера студенческих групп, которые изучают те же предметы, что и студенческая группа АС-8.

Запрос:

```
SELECT
DISTINCT ttsg.group_code_number
FROM
teacher_teaches_subjects_in_groups ttsg
JOIN student_groups sg ON ttsg.group_code_number = sg.code_number
LEFT JOIN teacher_teaches_subjects_in_groups ttsg_8 ON sg.name = 'AC-8'
AND ttsg_8.group_code_number = sg.code_number
WHERE
ttsg.subject_code_number IN (ttsg_8.subject_code_number)
AND NOT sg.name = 'AC-8'
GROUP BY
ttsg.group_code_number
HAVING
COUNT(DISTINCT ttsg.subject_code_number) = COUNT(DISTINCT ttsg_8.subject_code_number);
```

Результат:



1.24. Получить номера студенческих групп, которые не изучают предметы, преподаваемых в студенческой группе АС-8.

```
• • •
   SELECT
       DISTINCT sg.code_number
   FROM
       student_groups sg
       sg.code_number NOT IN (
           SELECT
                DISTINCT ttsg.group_code_number
                teacher teaches subjects in groups ttsg
                ttsg.subject_code_number IN (
                    SELECT
                        DISTINCT ttsg.subject_code_number
                    FROM
                        teacher teaches subjects in groups ttsg
                        ttsg.group_code_number IN (
                            SELECT
                                DISTINCT sg2.code_number
                            FROM
                                student_groups sg2
                                sg2.name = 'AC-8'
```

```
Empty set (0,00 sec)
```

1.25. Получить номера студенческих групп, которые не изучают предметы, преподаваемых преподавателем 430Л.

```
SELECT
        DISTINCT sg.code_number
   FROM
       student_groups sg
   WHERE
       sg.code_number NOT IN (
           SELECT
                DISTINCT ttsg.group code number
            FROM
                teacher teaches subjects in groups ttsg
           WHERE
                ttsg.personal number = '430Л'
12
13
        );
```

1.26. Получить номера преподавателей, работающих с группой Э-15, но не преподающих предмет 12П.

Запрос:

```
SELECT
DISTINCT ttsg.personal_number
FROM
teacher_teaches_subjects_in_groups ttsg
JOIN student_groups sg ON ttsg.group_code_number = sg.code_number
WHERE
sg.name = '9-15'
AND ttsg.personal_number NOT IN (
SELECT
ttsg_inner.personal_number
FROM
teacher_teaches_subjects_in_groups ttsg_inner
WHERE
ttsg_inner.subject_code_number = '12\Pi'
);
```

Результат:

```
| personal_number |
| personal_number |
| 221Л |
| 225Л |
| 2 rows in set (0,00 sec)
```

Задание 2

Запросы для создания таблиц

Таблица «Поставщики S»

```
CREATE TABLE suppliers (
id INT NOT NULL AUTO_INCREMENT,
name VARCHAR(16) NOT NULL,
status INT NOT NULL,
city VARCHAR(64),
PRIMARY KEY (id) );
```

Таблица «Детали Р»

```
CREATE TABLE details (
id INT NOT NULL AUTO_INCREMENT,
name VARCHAR(64) NOT NULL,
color VARCHAR(64) NOT NULL,
size INT NOT NULL,
city VARCHAR(64) NOT NULL,
PRIMARY KEY (id)
);
```

Таблица «Проекты J»

```
CREATE TABLE projects (

id INT NOT NULL AUTO_INCREMENT,

name VARCHAR(64) NOT NULL,

city VARCHAR(64) NOT NULL,

PRIMARY KEY (id)

);
```

Таблица «Количество деталей, поставляемых одним поставщиком для одного проекта»

```
CREATE TABLE num_of_ditails_per_project (

id INT NOT NULL AUTO_INCREMENT,

supplier_id INT,

detail_id INT,

project_id INT,

s INT,

PRIMARY KEY (id),

FOREIGN KEY (supplier_id) REFERENCES suppliers (id) ON DELETE CASCADE,

FOREIGN KEY (detail_id) REFERENCES details (id) ON DELETE CASCADE,

FOREIGN KEY (project_id) REFERENCES projects (id) ON DELETE CASCADE

);
```

Запросы для работы с таблицами

23. Получить номера поставщиков, поставляющих по крайней мере одну деталь, поставляемую по крайней мере одним поставщиком, который поставляет по крайней мере одну красную деталь.

Запрос:

```
1 SELECT DISTINCT supplier_id
2 FROM num_of_ditails_per_project
3 WHERE detail_id IN (
4 SELECT DISTINCT detail_id
5 FROM num_of_ditails_per_project
6 WHERE supplier_id IN (
7 SELECT DISTINCT supplier_id
8 FROM num_of_ditails_per_project
9 WHERE detail_id IN (
10 SELECT DISTINCT detail_id
11 FROM details
12 WHERE color = 'Красный'
13 )
14 )
15 );
```

Результат:

20. Получить цвета деталей, поставляемых поставщиком П1.

Запрос:

28. Получить номера проектов, для которых не поставляются красные детали поставщиками из Лондона.

Запрос:

```
SELECT DISTINCT p.id
FROM projects p
WHERE NOT EXISTS (
SELECT *
FROM num_of_ditails_per_project nd
JOIN suppliers s ON nd.supplier_id = s.id
JOIN details d ON nd.detail_id = d.id
WHERE nd.project_id = p.id AND d.color = 'Красный' AND s.city = 'Лондон'
);
```

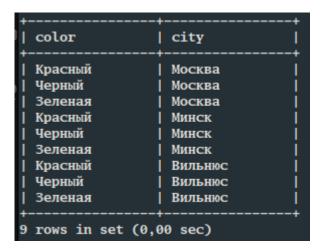
Результат:

```
+---+
| id |
| +---+
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
| 7 |
+---+
7 rows in set (0,00 sec)
```

5. Получить все сочетания "цвета деталей-города деталей".

Запрос:

```
1 SELECT DISTINCT d1.color, d2.city FROM details d1 CROSS JOIN details d2;
```



6. Получить все такие тройки "номера поставщиков-номера деталей-номера проектов", для которых выводимые поставщик, деталь и проект размещены в одном городе.

Запрос:

```
SELECT s.id AS supplier_id, d.id AS detail_id, p.id AS project_id
FROM num_of_ditails_per_project AS ndpp
JOIN suppliers s ON ndpp.supplier_id = s.id
JOIN details d ON ndpp.detail_id = d.id
JOIN projects p ON ndpp.project_id = p.id
WHERE s.city = d.city AND s.city = p.city;
```

Результат:

```
Empty set (0,00 sec)
```

10. Получить номера деталей, поставляемых поставщиком в Лондоне для проекта в Лондоне.

Запрос:

```
SELECT ndpp.detail_id AS detail_id
FROM num_of_ditails_per_project AS ndpp
JOIN suppliers AS s ON ndpp.supplier_id = s.id
JOIN projects AS p ON ndpp.project_id = p.id
WHERE s.city = 'Лондон' AND p.city = 'Лондон';
```

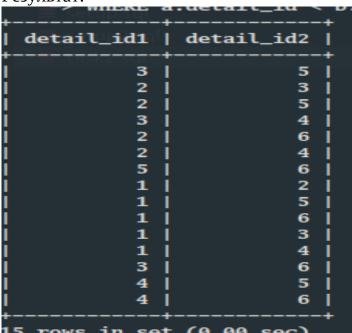
Empty set (0,00 sec)

14. Получить все такие пары номеров деталей, которые обе поставляются одновременно одним поставщиком.

Запрос:

```
SELECT DISTINCT a.detail_id AS detail_id1, b.detail_id AS detail_id2
FROM num_of_ditails_per_project AS a
JOIN num_of_ditails_per_project AS b ON a.supplier_id = b.supplier_id
WHERE a.detail_id < b.detail_id;</pre>
```

Результат:



34. Получить номера деталей, поставляемых либо лондонским поставщиком, либо для лондонского проекта.

```
SELECT DISTINCT d.id
FROM details d
JOIN num_of_ditails_per_project np ON d.id = np.detail_id
JOIN suppliers s ON np.supplier_id = s.id
JOIN projects p ON np.project_id = p.id
WHERE s.city = 'Лондон' OR p.city = 'Лондон'
```

```
Empty set (0,00 sec)
```

19. Получить имена проектов, обеспечиваемых поставщиком П1.

Запрос:

```
SELECT DISTINCT projects.name
FROM num_of_ditails_per_project
JOIN projects ON num_of_ditails_per_project.project_id = projects.id
WHERE num_of_ditails_per_project.supplier_id = 1;
```

Результат:

35. Получить пары "номер поставщика-номер детали", такие, что данный поставщик не поставляет данную деталь.

```
SELECT s.id AS supplier_id, d.id AS detail_id
FROM suppliers s
CROSS JOIN details d
WHERE (s.id, d.id) NOT IN (
SELECT supplier_id, detail_id
FROM num_of_ditails_per_project
)
8
9
```



Вывод

В лабораторной работе были изучены основы работы с реляционными СУБД и языком SQL. Были выполнены практические задания по созданию базы данных и выполнению запросов. Для работы использовались СУБД PostgreSQL и SQLite. В качестве программ для администрирования были использованы PgAdmin 4 и SQLite Studio. Стоит отметить, что дальнейший выбор СУБД зависит от масштабов проекта, в котором она используется — PostgreSQL подходит для проектов с огромными базами данных, SQLite — для маленьких приложений. Однако на данном этапе в силу небольших баз данных для обоих заданий, отличительной разницы между обоими СУБД обнаружено не было.