**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №2**

**по дисциплине «Базы данных»**

Тема: Реализация базы данных в СУБД PostgreSQL.

|  |  |  |
| --- | --- | --- |
| Студент гр. 1303 |  | Кузнецов Н.А. |
| Преподаватель |  | Заславский М.М. |

Санкт-Петербург

2023

## Цель работы.

## Целью данной лабораторной работы является овладение навыками проектирования, создания и настройки базы данных в системе управления базами данных PostgreSQL. В рамках выполнения лабораторной работы необходимо создать структуру базы данных, определить таблицы, поля и связи между ними, а также настроить доступ к базе данных. Кроме того, необходимо изучить основные SQL-запросы для взаимодействия с базой данных и убедится, что она функционирует корректно.

## Задание.

Вариант 13.

Пусть требуется создать программную систему, предназначенную для работников справочной службы кинотеатров города. Такая система должна обеспечивать хранение сведений о кинотеатрах города, о фильмах, которые в них демонстрируются, о сеансах и билетах на эти сеансы. Сведения о кинотеатре — это его название, район города, где расположен кинотеатр, категория, вместимость. Сведения о фильме — это название фильма, режиссер, оператор, актеры, сыгравшие главные роли, жанр; производство, наличие призов кинофестивалей, продолжительность сеанса, кадр из фильма для рекламы. Кроме того, должна храниться информация о репертуаре кинотеатров на месяц, то есть о том какие фильмы, когда и где демонстрируются, о ценах на билеты и о количестве свободных мест на тот или иной сеанс. На разных сеансах в одном кинотеатре могут идти разные фильмы, а если в кинотеатре несколько залов, то и на одном. Кинотеатр может ввести новый фильм в репертуар или убрать фильм из репертуара. Работник справочной службы может корректировать перечень фильмов, находящихся в прокате – добавлять новые фильмы и снимать с проката, а также перечень кинотеатров, поскольку кинотеатры могут открываться или закрываться, причем иногда временно, например, на ремонт. Цена билета определяется прокатной стоимостью копии фильма, сеансом и категорией кинотеатра. Справочной службе могут потребоваться следующие сведения о текущем состоянии проката фильмов в городе:

Репертуар кинотеатра?

Адрес и район кинотеатра ?

Число свободных мест на данный сеанс в указанном кинотеатре?

Цена билетов на данный сеанс в указанном кинотеатре?

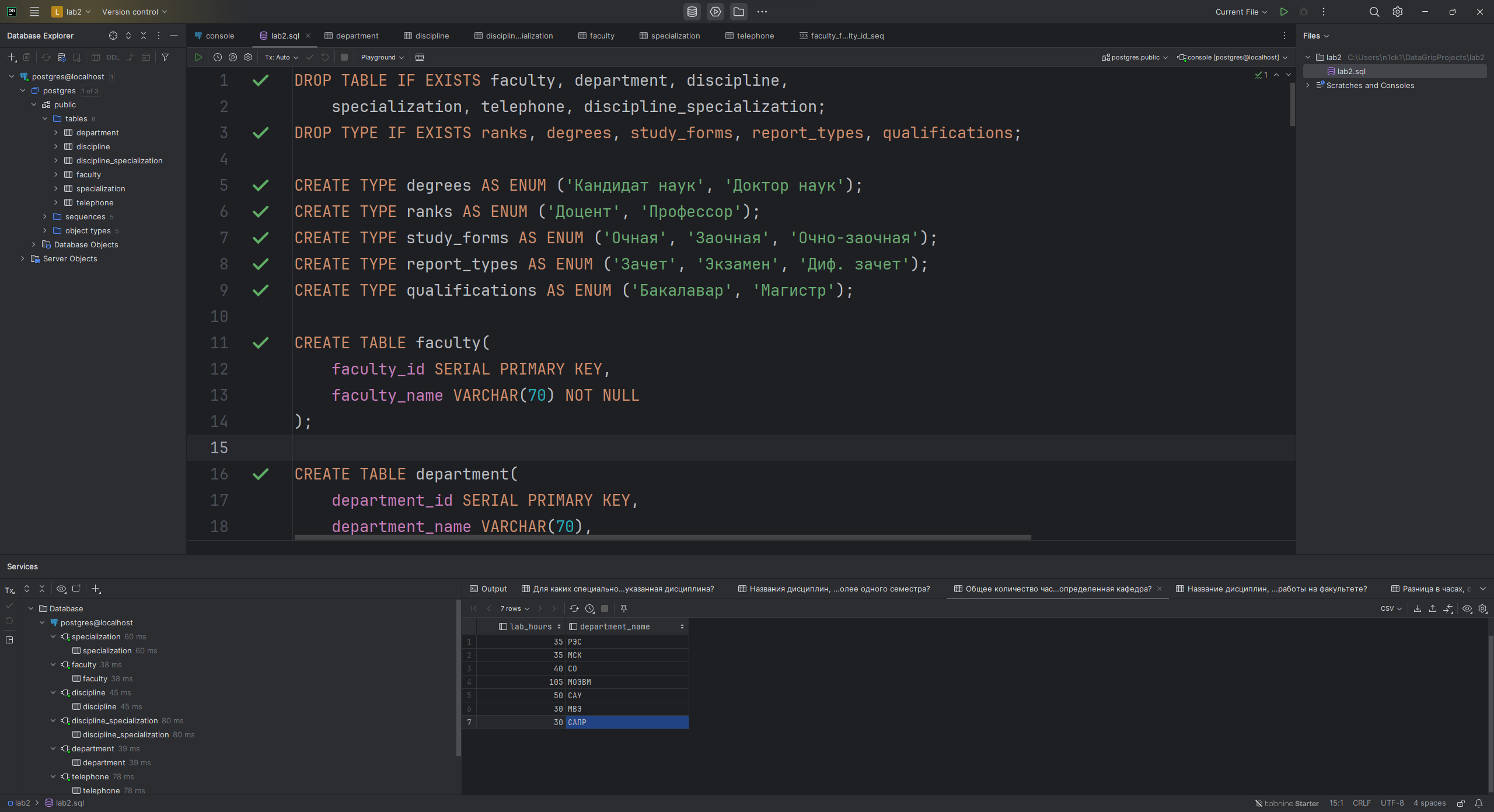
Жанр, производство и режиссер данного фильма ?

Какие фильмы имеют награды, когда и в каких кинотеатрах они демонстрируются?

В каких кинотеатрах в указанный день на указанных сеансах демонстрируется комедия?

## Выполнение работы.

Данная база данных будет создана с помощью DataGrip.

Рис. 1 – DataGrip

Далее продемонстрировано то, какие таблицы были созданы и как они заполнены.

1. Таблица faculty

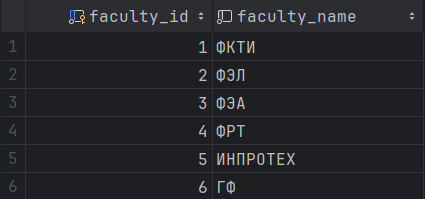


Рис. 1 – Таблица Cinema

1. Таблица department

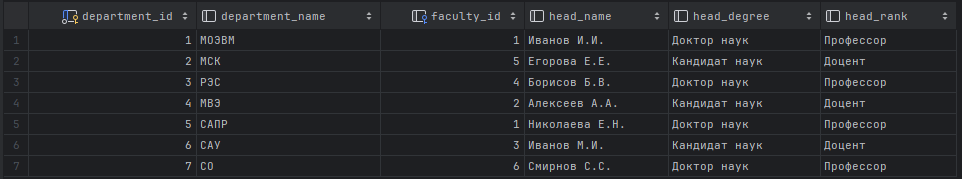


Рис. 2 – Таблица Cinema

1. Таблица telephone

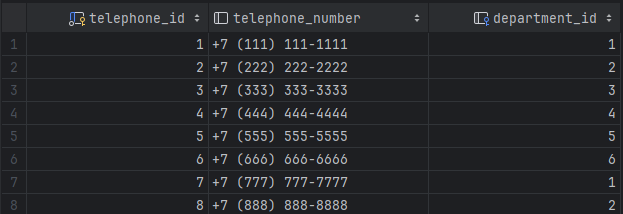


Рис. 3 – Таблица telephone

1. Таблица specialization

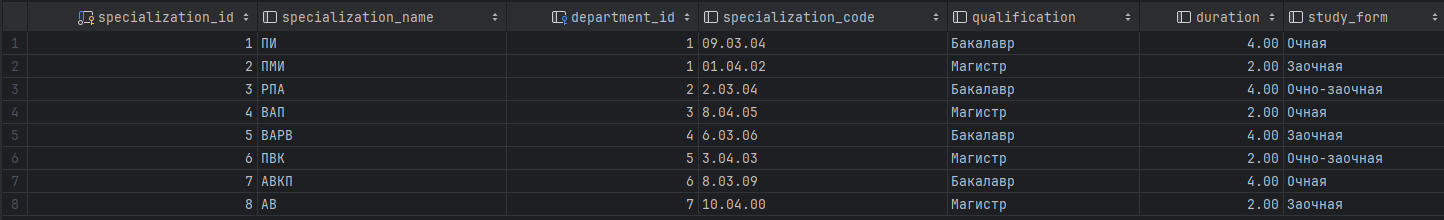


Рис. 4 – Таблица specialization

1. Таблица discipline

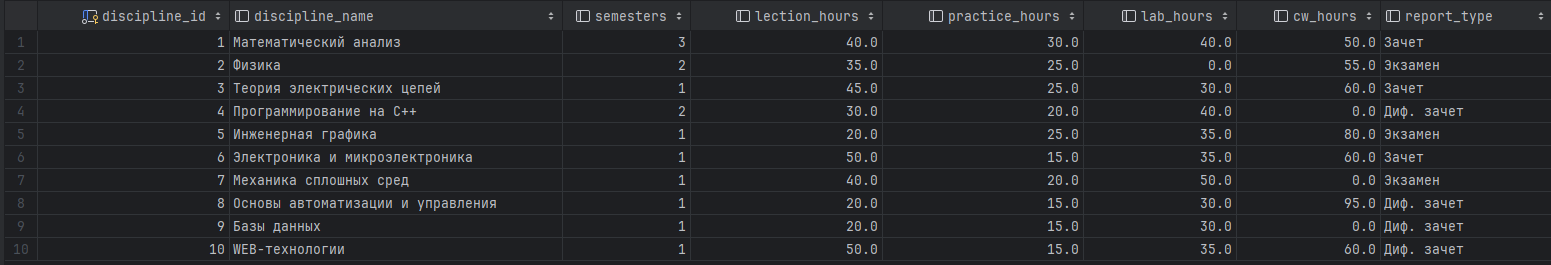


Рис. 5 – Таблица discipline

1. Таблица discipline\_specialization

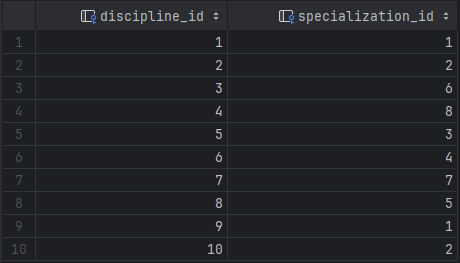
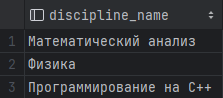


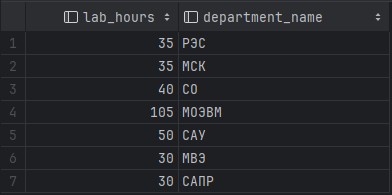
Рис. 6 – Таблица discipline\_specialization

Результаты запросов:

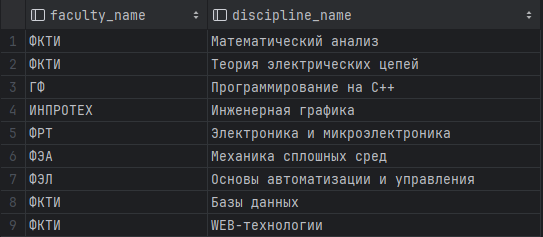
Названия дисциплин, которые читаются более одного семестра?



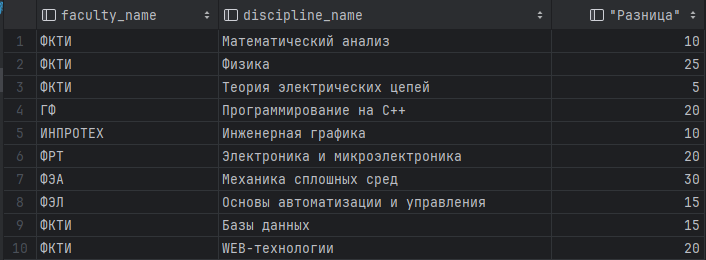
Общее количество часов, отводимых на лабораторные работы в одном из семестров, проведение которых обеспечивает определенная кафедра?



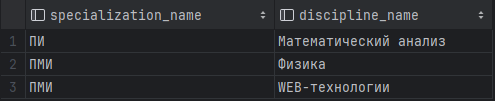
Название дисциплин, по которым проводятся лабораторные работы на факультете?



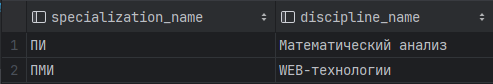
Разница в часах, отведенных по каждой дисциплине на лабораторные и практические занятия в одном из семестров на заданном факультете?



Дисциплины, по которым выполняют курсовые работы студенты указанной специальности?



Для каких специальностей читается указанная дисциплина?



Какое количество дисциплин входит в учебный план подготовки студентов по указанной специальности, и сколько лет осуществляется подготовка

## Выводы.

В ходе лабораторной работы была изучена работа с СУБД PostgreSQL с помощью программы DataGrip; создана база данных для кинотеатра с запросами на получение определенных данных. Также приобретены навыки проектирования, создания и настройки базы данных в системе управления базами данных PostgreSQL. В рамках выполнения лабораторной работы создана структура базы данных, определены таблицы, поля и связи между ними, а также настроен доступ к базе данных. Кроме того, изучены основные SQL-запросы для взаимодействия с базой данных.

**ПРИЛОЖЕНИЕ А**

Ссылка на исходный код: <https://www.db-fiddle.com/f/i8Yf7fbfeEkcY2Uf8HMYvU/0>

Ссылка на PR: <https://github.com/moevm/sql-2023-1303/pull/30>

Файл lab\_2

DROP TABLE IF EXISTS faculty, department, discipline,  
 specialization, telephone, discipline\_specialization;  
DROP TYPE IF EXISTS ranks, degrees, study\_forms, report\_types, qualifications;  
  
CREATE TYPE degrees AS ENUM ('Кандидат наук', 'Доктор наук');  
CREATE TYPE ranks AS ENUM ('Доцент', 'Профессор');  
CREATE TYPE study\_forms AS ENUM ('Очная', 'Заочная', 'Очно-заочная');  
CREATE TYPE report\_types AS ENUM ('Зачет', 'Экзамен', 'Диф. зачет');  
CREATE TYPE qualifications AS ENUM ('Бакалавар', 'Магистр');  
  
CREATE TABLE faculty(  
 faculty\_id SERIAL PRIMARY KEY,  
 faculty\_name VARCHAR(70) NOT NULL  
);  
  
CREATE TABLE department(  
 department\_id SERIAL PRIMARY KEY,  
 department\_name VARCHAR(70),  
 faculty\_id INTEGER REFERENCES faculty(faculty\_id),  
 head\_name VARCHAR(70),  
 head\_degree degrees,  
 head\_rank ranks  
);  
  
CREATE TABLE telephone(  
 telephone\_id SERIAL PRIMARY KEY,  
 telephone\_number VARCHAR(20),  
 department\_id INTEGER REFERENCES department(department\_id)  
);  
  
CREATE TABLE specialization(  
 specialization\_id SERIAL PRIMARY KEY,  
 specialization\_name VARCHAR(70),  
 department\_id INTEGER REFERENCES department(department\_id),  
 specialization\_code VARCHAR(20),  
 qualification VARCHAR(70),  
 duration DECIMAL(3, 2),  
 study\_form study\_forms  
);  
  
CREATE TABLE discipline(  
 discipline\_id SERIAL PRIMARY KEY,  
 discipline\_name VARCHAR(70),  
 semesters INTEGER,  
 lection\_hours DECIMAL(3,1),  
 practice\_hours DECIMAL(3,1),  
 lab\_hours DECIMAL(3,1),  
 cw\_hours DECIMAL(3,1),  
 report\_type report\_types  
);  
  
CREATE TABLE discipline\_specialization(  
 discipline\_id INTEGER REFERENCES discipline(discipline\_id),  
 specialization\_id INTEGER REFERENCES specialization(specialization\_id)  
);  
  
INSERT INTO faculty (faculty\_name)  
 VALUES  
 ('ФКТИ'),  
 ('ФЭЛ'),  
 ('ФЭА'),  
 ('ФРТ'),  
 ('ИНПРОТЕХ'),  
 ('ГФ');  
  
INSERT INTO department (department\_name, faculty\_id, head\_name, head\_degree, head\_rank)  
 VALUES  
 ('МОЭВМ', 1, 'Иванов И.И.', 'Доктор наук', 'Профессор'),  
 ('МСК', 5, 'Егорова Е.Е.', 'Кандидат наук', 'Доцент'),  
 ('РЭС', 4, 'Борисов Б.В.', 'Доктор наук', 'Профессор'),  
 ('МВЭ', 2, 'Алексеев А.А.', 'Кандидат наук', 'Доцент'),  
 ('САПР', 1, 'Николаева Е.Н.', 'Доктор наук', 'Профессор'),  
 ('САУ', 3, 'Иванов М.И.', 'Кандидат наук', 'Доцент'),  
 ('СО', 6, 'Смирнов С.С.', 'Доктор наук', 'Профессор');  
  
INSERT INTO telephone(telephone\_number, department\_id)  
 VALUES  
 ('+7 (111) 111-1111', 1),  
 ('+7 (222) 222-2222', 2),  
 ('+7 (333) 333-3333', 3),  
 ('+7 (444) 444-4444', 4),  
 ('+7 (555) 555-5555', 5),  
 ('+7 (666) 666-6666', 6),  
 ('+7 (777) 777-7777', 1),  
 ('+7 (888) 888-8888', 2);  
  
INSERT INTO specialization(specialization\_name, department\_id, specialization\_code,  
 qualification, duration, study\_form)  
 VALUES  
 ('ПИ', 1, '09.03.04', 'Бакалавр', 4, 'Очная'),  
 ('ПМИ', 1, '01.04.02', 'Магистр', 2, 'Заочная'),  
 ('РПА', 2, '2.03.04', 'Бакалавр', 4, 'Очно-заочная'),  
 ('ВАП', 3, '8.04.05', 'Магистр', 2, 'Очная'),  
 ('ВАРВ', 4, '6.03.06', 'Бакалавр', 4, 'Заочная'),  
 ('ПВК', 5, '3.04.03', 'Магистр', 2, 'Очно-заочная'),  
 ('АВКП', 6, '8.03.09', 'Бакалавр', 4, 'Очная'),  
 ('АВ', 7, '10.04.00', 'Магистр', 2, 'Заочная');  
  
INSERT INTO discipline(discipline\_name, lection\_hours,  
 practice\_hours, lab\_hours, cw\_hours, report\_type, semesters)  
 VALUES  
 ('Математический анализ', 40.0, 30.0, 40.0, 50.0, 'Зачет', 3),  
 ('Физика', 35.0, 25.0, 0, 55.0, 'Экзамен', 2),  
 ('Теория электрических цепей', 45.0, 25.0, 30.0, 60.0, 'Зачет', 1),  
 ('Программирование на C++', 30.0, 20.0, 40.0, 0, 'Диф. зачет', 2),  
 ('Инженерная графика', 20.0, 25.0, 35.0, 80.0, 'Экзамен', 1),  
 ('Электроника и микроэлектроника', 50.0, 15.0, 35.0, 60.0, 'Зачет', 1),  
 ('Механика сплошных сред', 40.0, 20.0, 50.0, 0, 'Экзамен', 1),  
 ('Основы автоматизации и управления', 20.0, 15.0, 30.0, 95.0, 'Диф. зачет', 1),  
 ('Базы данных', 20.0, 15.0, 30.0, 0, 'Диф. зачет', 1),  
 ('WEB-технологии', 50.0, 15.0, 35.0, 60.0, 'Диф. зачет', 1);  
  
INSERT INTO discipline\_specialization(discipline\_id, specialization\_id)  
 VALUES  
 (1, 1),  
 (2, 2),  
 (3, 6),  
 (4, 8),  
 (5, 3),  
 (6, 4),  
 (7, 7),  
 (8, 5),  
 (9, 1),  
 (10, 2);  
  
--Названия дисциплин, которые читаются более одного семестра?  
SELECT discipline\_name  
FROM discipline  
WHERE semesters > 1;  
  
  
--Общее количество часов, отводимых на лабораторные работы в одном из семестров, проведение которых обеспечивает определенная кафедра?  
SELECT SUM(discipline.lab\_hours) AS "lab\_hours", department.department\_name  
FROM discipline  
 JOIN discipline\_specialization USING(discipline\_id)  
 JOIN specialization USING(specialization\_id)  
 JOIN department USING(department\_id)  
GROUP BY department.department\_name;  
  
--Название дисциплин, по которым проводятся лабораторные работы на факультете?  
SELECT faculty.faculty\_name, discipline.discipline\_name  
FROM discipline  
 JOIN discipline\_specialization USING(discipline\_id)  
 JOIN specialization USING(specialization\_id)  
 JOIN department USING(department\_id)  
 JOIN faculty USING(faculty\_id)  
WHERE discipline.lab\_hours > 0;  
  
--Разница в часах, отведенных по каждой дисциплине на лабораторные и практические занятия в одном из семестров на заданном факультете?  
SELECT faculty.faculty\_name, discipline.discipline\_name,  
 ABS(discipline.lab\_hours - discipline.practice\_hours) AS "Разница"  
FROM discipline  
 JOIN discipline\_specialization USING(discipline\_id)  
 JOIN specialization USING(specialization\_id)  
 JOIN department USING(department\_id)  
 JOIN faculty USING(faculty\_id);  
  
--Дисциплины, по которым выполняют курсовые работы студенты указанной специальности?  
SELECT specialization.specialization\_name, discipline.discipline\_name  
FROM discipline  
 JOIN discipline\_specialization USING(discipline\_id)  
 JOIN specialization USING(specialization\_id)  
WHERE cw\_hours > 0 AND  
 specialization.specialization\_name IN ('ПИ', 'ПМИ');  
  
--Для каких специальностей читается указанная дисциплина?  
SELECT specialization.specialization\_name, discipline.discipline\_name  
FROM discipline  
 JOIN discipline\_specialization USING(discipline\_id)  
 JOIN specialization USING(specialization\_id)  
WHERE discipline.discipline\_name LIKE 'WEB-технологии'  
 OR discipline.discipline\_name LIKE 'Математический анализ';  
  
--Какое количество дисциплин входит в учебный план подготовки студентов по указанной специальности, и сколько лет осуществляется подготовка?  
SELECT specialization.specialization\_name AS "Специальность", COUNT(discipline.discipline\_name) AS "Количество",  
 specialization.duration AS "Срок обучения"  
FROM discipline  
 JOIN discipline\_specialization USING(discipline\_id)  
 JOIN specialization USING(specialization\_id)  
GROUP BY specialization.specialization\_name, specialization.duration