

Задание по основам баз данных

Задание выполнялось в **DBeaver**. В репозитории приложены скрипты в формате SQL, также решения продублированы в этом файле.

2. Создать БД academy.

Решение:

```
create database academy with encoding 'UTF8';
```

3. Добавить таблицы по приложенной схеме рис. 1 (названия полей, связи между таблицами должны соответствовать схеме, остальное (ограничения целостности, уникальность, значения по умолчанию, проверки, типы данных) на усмотрение стажера).

Решение:

```
create table Students (  
  s_id INT primary key,  
  name varchar not null,  
  start_year INT  
);  
  
create table Courses (  
  c_no INT primary key,  
  title varchar not null,  
  hours INT  
);  
  
create table Exams (  
  s_id INT,  
  c_no INT,  
  score INT  
);  
  
alter table Exams  
add foreign key (s_id) references Students(s_id);  
  
alter table Exams  
add foreign key (c_no) references Courses(c_no);
```

4. Добавить несколько записей в таблицы.

Решение:

```
INSERT INTO Students (s_id, name, start_year)  
VALUES  
  (1, 'Иванов Александр Алексеевич', 2021),  
  (2, 'Бутаев Даниил Антонович', 2023),  
  (3, 'Зайцева Елизавета Алексеевна', 2023),  
  (4, 'Степанова Ксения Эдуардовна', 2022),
```

```
(5, 'Кириллов Андрей Васильевич', 2024),  
(6, 'Васильев Денис Александрович', 2024);
```

```
INSERT INTO Courses (c_no, title, hours)  
VALUES  
(1, 'Введение в алгоритмы', 102),  
(2, 'Теория вероятностей', 84),  
(3, 'Моделирование технологических процессов', 68),  
(4, 'Философия', 54),  
(5, 'Безопасность жизнедеятельности', 72),  
(6, 'Квантовая физика', 48);
```

```
INSERT INTO Exams (s_id, c_no, score)  
VALUES  
(1, 1, 75),  
(1, 2, 86),  
(1, 6, 70),  
(2, 3, 68),  
(2, 2, 84),  
(3, 3, 90),  
(3, 4, 100),  
(4, 6, 85),  
(4, 5, 91),  
(5, 1, 58);
```

5. Написать запрос, который возвращает всех студентов, которые еще не сдали ни одного экзамена.

Решение:

```
select  
s.*  
from Students s  
where not exists (  
    select  
        s_id  
    from Exams e  
    where e.s_id=s.s_id);
```

6. Написать запрос, который возвращает список студентов и количество сданных им экзаменов. Только для студентов, у которых есть сданные экзамены.

Решение:

```
select e.s_id, s.name, s.start_year, COUNT(*) as passed_exams  
from Exams e  
left join Students s on e.s_id=s.s_id  
group by e.s_id, s.name, s.start_year  
order by e.s_id;
```

7. Вывести список курсов со средним баллом по экзамену. Список отсортирован по убыванию среднего балла.

Решение:

```
select e.c_no , c.title, c.hours, AVG(e.score) as average_point
from Exams e
left join Courses c on e.c_no=c.c_no
group by e.c_no, c.title, c.hours
order by average_point desc;
```