

**Работаем с базой данных учителей teachers.db. Для каждого задания надо сдать только код.**

**1. Найдите общее количество учеников для каждого курса. В отчёт выведите название курса и количество учеников по всем потокам курса. Решите задание с применением оконных функций.**

```
SELECT DISTINCT
courses.name,
SUM(students_amount) OVER (w_streams) AS students_total
FROM streams
INNER JOIN courses
ON streams.course_id = courses.id
WINDOW w_streams AS (PARTITION BY courses.id)
ORDER BY courses.name;
```

**2. Найдите среднюю оценку по всем потокам для всех учителей. В отчёт выведите идентификатор, фамилию и имя учителя, среднюю оценку по всем проведённым потокам. Учителя, у которых не было потоков, также должны попасть в выборку. Решите задание с применением оконных функций.**

```
SELECT DISTINCT
teachers.id,
teachers.surname || teachers.name,
AVG(performance) OVER (w_grades) AS avg_grade
FROM teachers
LEFT JOIN grades
ON teachers.id = grades.teacher_id
WINDOW w_grades AS (PARTITION BY teacher_id)
ORDER BY teachers.id;
```

### 3. Какие индексы надо создать для максимально быстрого выполнения представленного запроса?

```
SELECT  
surname,  
name,  
number,  
performance  
FROM academic_performance  
JOIN teachers  
ON academic_performance.teacher_id = teachers.id  
JOIN streams  
ON academic_performance.stream_id = streams.id  
WHERE number >= 200;
```

```
CREATE UNIQUE INDEX streams_number_uq ON streams(number);
```

```
CREATE INDEX teachers_surname_name_idx ON teachers(surname, name);
```

4. Установите SQLiteStudio, подключите базу данных учителей, выполните в графическом клиенте любой запрос. В качестве отчёта к заданию надо приложить скриншот.

