1. Напишите SQL запрос который вернет имена студентов, курс на котором они учатся, названия их родных университетов (в которых они официально учатся) и соответствующий рейтинг по курсу. С условием что рассматриваемый рейтинг студента должен быть строго больше (>) 50 баллов и размер соответствующего ВУЗа должен быть строго больше (>) 5000 студентов. Результат необходимо отсортировать по первым двум столбцам. Обратите внимание на часть ответа ниже с учетом **именования выходных атрибутов** вашего запроса

**Код запроса:**

SELECT

student.name as student\_name,

course.name as course\_name,

college.name as student\_college,

student\_on\_course.student\_rating

FROM

student\_on\_course

JOIN student ON student.id = student\_on\_course.student\_id

JOIN course ON course.id = student\_on\_course.course\_id

JOIN college ON college.id = student.college\_id

WHERE student\_on\_course.student\_rating > 50 and college.size > 5000

ORDER BY

student\_name, course\_name

**Результат выполнения кода:**

Изображение выглядит как текст, Шрифт, снимок экрана, число

Автоматически созданное описание

1. Выведите уникальные семантические пары студентов, родной город которых один и тот же. Результат необходимо отсортировать по первому столбцу. Семантически эквивалентная пара является пара студентов например (Иванов, Петров) = (Петров, Иванов), в этом случае должна быть выведена одна из пар. Обратите внимание на ответ ниже с учетом **именования выходных атрибутов** вашего запроса

**Код запроса:**

SELECT DISTINCT s1.name AS student\_1, s2.name AS student\_2, s1.city

FROM student s1

JOIN student s2 ON s1.city = s2.city AND s1.id < s2.id -- исключение дублирования пар

ORDER BY student\_2

**Результат запроса:**

Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание

1. Выведите имена студентов и названия соответствующих учебных курсов, которые преподаются в тех же университетах, в которых учатся студенты. Результат необходимо отсортировать по первому столбцу Обратите внимание на ответ ниже с учетом **именования выходных атрибутов** вашего запроса

**Код запроса:**

SELECT student.name as student\_name, course.name as course\_date

FROM

student\_on\_course as sc

JOIN course ON course.id = sc.course\_id

JOIN student ON student.id = sc.student\_id

WHERE course.college\_id = student.college\_id

ORDER BY student\_name

**Результат выполнения запроса:**

Изображение выглядит как текст, Шрифт, линия, число

Автоматически созданное описание

1. Выведите статистику (min, max, avg, sum) по показателю рейтинга всех студентов по всем курсам. Примените округление до 2 цифр после запятой (функция round). Обратите внимание на ответ ниже с учетом **именования выходных атрибутов** вашего запроса

**Код запроса:**

SELECT

MIN(student\_rating) AS min\_student\_rating,

MAX(student\_rating) AS max\_student\_rating,

ROUND(AVG(student\_rating), 2) AS avg\_student\_rating,

SUM(student\_rating) AS sum\_student\_rating

FROM student\_on\_course

**Результат выполнения запроса:**

Изображение выглядит как текст, Шрифт, линия, снимок экрана

Автоматически созданное описание

1. Выведите города, количество проживающих студентов в которых равно 2. Результат отсортируйте по имени города. Обратите внимание на ответ ниже с учетом **именования выходных атрибутов** вашего запроса.

Код запроса:

SELECT city

FROM student

GROUP BY city

HAVING COUNT(\*) = 2

ORDER BY city

Результат запроса:

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

1. Перед выполнением этого задания пожалуйста проверьте что таблица *course* содержит курс “Machine Learning”, который мы добавили в рамках Лабораторной Работы №4 задание d). Если запись отсутствует, тогда пожалуйста сделайте задание d) из ЛР №4 повторно. Выведите **все зарегистрированные** оффлайн курсы и соответствующие имена студентов которые на них учатся. Если на курсе не зарегистрированы студенты - то необходимо вывести значение “курс не используется”. Результат отсортируйте по первым двум полям. Обратите внимание на ответ ниже с учетом **именования выходных атрибутов** вашего запроса.

**Код запроса:**

SELECT cs.name AS course\_name, COALESCE(st.name, 'курс не используется') AS student\_name

FROM course cs

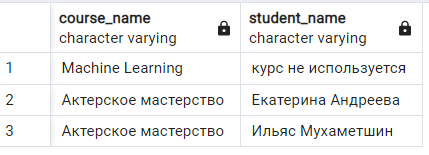
FULL OUTER JOIN student\_on\_course sc ON cs.id = sc.course\_id

FULL OUTER JOIN student st ON sc.student\_id = st.id

WHERE cs.is\_online = false

ORDER BY course\_name

**Результат выполнения запроса:**



1. (\*\* задача опциональная) Выведите доказательство, что количество онлайн курсов больше чем количество оффлайн курсов. Результат представьте как строку в одном столбце. Обратите внимание на ответ ниже с учетом **именования выходных атрибутов** вашего запроса.

**Код запроса:**

SELECT

CASE

WHEN COUNT(\*) > 0 THEN 'количество онлайн курсов больше'

ELSE 'количество оффлайн курсов больше'

END AS result

FROM

course

WHERE

is\_online = true

HAVING

COUNT(\*) > (SELECT COUNT(\*) FROM course WHERE is\_online = false)

**Результат выполнения запроса:**

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание