

- а. Напишите SQL запрос который возвращает количество студентов, сгруппированных по их **оценке**. Результат отсортируйте по названию оценки студента. Формула выставления оценки представлена ниже как псевдокод.

ЕСЛИ оценка < 30 ТОГДА неудовлетворительно

ЕСЛИ оценка >= 30 И оценка < 60 ТОГДА удовлетворительно

ЕСЛИ оценка >= 60 И оценка < 85 ТОГДА хорошо

В ОСТАЛЬНЫХ СЛУЧАЯХ отлично

Пример результата ниже. Обратите внимание на именование результирующих столбцов в вашем решении. Курс “Machine Learning”, так как у него нет студентов - проигнорируйте, используя соответствующий тип JOIN.

оценка	количество студентов
неудовлетворительно	2
отлично	3
удовлетворительно	3
хорошо	5

Код запроса:

```
SELECT
CASE
    WHEN student_rating < 30 THEN 'неудовлетворительно'
    WHEN student_rating >= 30 AND student_rating < 60 THEN 'удовлетворительно'
    WHEN student_rating >= 60 AND student_rating < 85 THEN 'хорошо'
    ELSE 'отлично'
END AS оценка,
COUNT(*) AS количество_студентов
```

```
FROM
    student_on_course
JOIN course ON course.id = student_on_course.course_id
WHERE
    course.name != 'Machine Learning'
GROUP BY
    оценка
ORDER BY
    количество_студентов
```

Результат запроса:

	оценка text	количество_студентов bigint
1	неудовлетворительно	2
2	отлично	3
3	удовлетворительно	3
4	хорошо	5

- b. Дополните SQL запрос из задания а), с указанием вывода имени курса и количество оценок внутри курса. Результат отсортируйте по названию курса и оценки студента. Пример части результата ниже.

Обратите внимание на именование результирующих столбцов в вашем решении. Курс “Machine Learning”, так как у него нет студентов - проигнорируйте, используя соответствующий тип JOIN.

Код запроса:

```
SELECT
    course.name,
    CASE
        WHEN student_rating < 30 THEN 'неудовлетворительно'
        WHEN student_rating >= 30 AND student_rating < 60 THEN 'удовлетворительно'
        WHEN student_rating >= 60 AND student_rating < 85 THEN 'хорошо'
        ELSE 'отлично'
    END AS оценка,
    COUNT(*) AS количество_студентов
FROM
    student_on_course
JOIN course ON course.id = student_on_course.course_id
WHERE
    course.name != 'Machine Learning'
GROUP BY
    course.name,
    оценка
ORDER BY
    name
```

Результат запроса:

	name character varying	оценка text	количество_студентов bigint
1	Data Mining	неудовлетворительно	1
2	Data Mining	хорошо	2
3	Актерское мастерство	отлично	2
4	Введение в РСУБД	неудовлетворительно	1
5	Введение в РСУБД	хорошо	2
6	Нейронные сети	удовлетворительно	1
7	Нейронные сети	хорошо	1
8	Цифровая трансформация	отлично	1
9	Цифровая трансформация	удовлетворительно	2

- с. Дополните SQL запрос из задания а), с указанием вывода
- номера строки (сквозная нумерация строк)
 - номера строки для группировки по имени курса (нумерация строк внутри группы)
 - вывода имени курса и соответствующей оценки с указанием количества студентов по оценке и выводом подитога количества по группе

Результат отсортируйте по названию курса и оценки студента.

Пример результата ниже. Курс "Machine Learning", так как у него нет студентов - проигнорируйте, используя соответствующий тип JOIN.

Код запроса:

```
WITH student_counts AS (  
    SELECT  
        course.name AS курс,  
        CASE  
            WHEN student_rating < 30 THEN 'неудовлетворительно'  
            WHEN student_rating >= 30 AND student_rating < 60 THEN  
'удовлетворительно'  
            WHEN student_rating >= 60 AND student_rating < 85 THEN  
'хорошо'  
            ELSE 'отлично'  
        END AS оценка,  
        COUNT(*) AS "количество студентов"  
    FROM  
        student_on_course  
        INNER JOIN course ON course.id = student_on_course.course_id  
    GROUP BY  
        курс, оценка),  
student_assessments AS (  
    SELECT  
        CASE
```

```

        WHEN student_rating < 30 THEN 'неудовлетворительно'
        WHEN student_rating >= 30 AND student_rating < 60 THEN
'удовлетворительно'
        WHEN student_rating >= 60 AND student_rating < 85 THEN
'хорошо'
        ELSE 'отлично'
    END AS "оценка",
    COUNT(*) AS "количество студентов"
FROM
    course
    RIGHT JOIN student_on_course ON course.id =
student_on_course.course_id
GROUP BY
    оценка
ORDER BY
    оценка),
student_group AS (
    SELECT
        курс,
        оценка,
        "количество студентов"
    FROM student_counts
    UNION ALL
    SELECT
        курс,
        NULL,
        SUM("количество студентов")
    FROM student_counts
    GROUP BY
        курс
    UNION ALL
    SELECT
        NULL,
        оценка,
        "количество студентов"
    FROM student_assessments
    UNION ALL
    SELECT
        NULL,
        NULL,
        SUM("количество студентов")
    FROM student_assessments
)
SELECT
    ROW_NUMBER() OVER () AS "# строки",
    ROW_NUMBER() OVER (PARTITION BY курс ORDER BY оценка) AS "# строки в
группе",
    курс,
    оценка,
    "количество студентов"

```

```
FROM
    student_group
GROUP BY курс, оценка, "количество студентов"
ORDER BY курс, оценка;
```

Результат запроса:

	# строки bigint	# строки в группе bigint	курс character varying	оценка text	количество студентов numeric
1	1	1	Data Mining	неудовлетворительно	1
2	2	2	Data Mining	хорошо	2
3	3	3	Data Mining	[null]	3
4	4	1	Актерское мастерство	отлично	2
5	5	2	Актерское мастерство	[null]	2
6	6	1	Введение в РСУБД	неудовлетворительно	1
7	7	2	Введение в РСУБД	хорошо	2
8	8	3	Введение в РСУБД	[null]	3
9	9	1	Нейронные сети	удовлетворительно	1
10	10	2	Нейронные сети	хорошо	1
11	11	3	Нейронные сети	[null]	2
12	12	1	Цифровая трансформация	отлично	1
13	13	2	Цифровая трансформация	удовлетворительно	2
14	14	3	Цифровая трансформация	[null]	3
15	15	1	[null]	неудовлетворительно	2
16	16	2	[null]	отлично	3
17	17	3	[null]	удовлетворительно	3
18	18	4	[null]	хорошо	5
19	19	5	[null]	[null]	13

- d. Дополните SQL запрос из задания а), с указанием вывода
- плотного ранжирования (dense_rank) по оценкам курсов, основанных на формуле из задания а)
 - имени курса с соответствующей оценке по нему

Результат отсортируйте по ранжированию оценок за курсы.
Пример результата ниже. Курс "Machine Learning", так как у него нет студентов - проигнорируйте, используя соответствующий тип JOIN.

Код запроса:

```
SELECT
    DENSE_RANK() OVER (ORDER BY CASE
        WHEN student_rating < 30 THEN 'неудовлетворительно'
        WHEN student_rating >= 30 AND student_rating < 60 THEN 'удовлетворительно'
        WHEN student_rating >= 60 AND student_rating < 85 THEN 'хорошо'
        ELSE 'отлично'
    END) AS ranking,
    course.name,
    CASE
```

```

    WHEN student_rating < 30 THEN 'неудовлетворительно'
    WHEN student_rating >= 30 AND student_rating < 60 THEN 'удовлетворительно'
    WHEN student_rating >= 60 AND student_rating < 85 THEN 'хорошо'
    ELSE 'отлично'
END AS оценка

```

```

FROM
    student_on_course
JOIN course ON course.id = student_on_course.course_id
WHERE
    course.name != 'Machine Learning'
GROUP BY
    course.name,
CASE
    WHEN student_rating < 30 THEN 'неудовлетворительно'
    WHEN student_rating >= 30 AND student_rating < 60 THEN 'удовлетворительно'
    WHEN student_rating >= 60 AND student_rating < 85 THEN 'хорошо'
    ELSE 'отлично'
END
ORDER BY
    ranking

```

Результат запроса:

	ranking bigint	name character varying	оценка text
1	1	Введение в РСУБД	неудовлетворительно
2	1	Data Mining	неудовлетворительно
3	2	Цифровая трансформация	отлично
4	2	Актерское мастерство	отлично
5	3	Цифровая трансформация	удовлетворительно
6	3	Нейронные сети	удовлетворительно
7	4	Нейронные сети	хорошо
8	4	Введение в РСУБД	хорошо
9	4	Data Mining	хорошо