1. 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:**

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

1. Дополните SQL запрос из задания a), 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:**

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

1. Дополните SQL запрос из задания a), 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:**

Изображение выглядит как текст, число, документ, чек

Автоматически созданное описание

1. Дополните SQL запрос из задания a), с указанием вывода

* плотного ранжирования (dense\_rank) по оценкам курсов, основанных на формуле из задания a)
* 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:**

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание