функции агрегирования **здесь ключевую роль играет GROUP BY, без него возможно эти функции вообще не работают, поэтому и называется групповые функции, сначала находятся уникальные элементы путём разбиения на группы и берётся только одно значение из каждой группы, так находится уникальность, и для каждого уникального значения подщитывается групповая функция**

**В запросы с групповыми функциями можно включать условие отбора строк, которое в обычных запросах записывается после WHERE. В запросах с групповыми функциями вместо WHERE используется ключевое слово HAVING , которое размещается после оператора GROUP BY.**

• COUNT определяет количество строк (в столбце)или значений поля, выбранных посредством

запроса и не являющихся NULL-значениями;

• SUM вычисляет арифметическую сумму всех выбранных значений данного поля;

• АVG вычисляет среднее значение для всех выбранных значений данного поля;

• МАХ вычисляет наибольшее из всех выбранных значений данного поля;

• MIN вычисляет наименьшее из всех выбранных значений данного поля.

Для SUM и AVG могут использоваться только цифровые поля. Для COUNT, МАХ и

MIN -циф­ровые и символьные поля.

SELECT COUNT(ODATE) FROM orders;<--вернёт число 10

SELECT COUNT(DISTINCT ODATE)FROM orders;<--вернёт число 5(только уникальные значения)

Для подсчета об­щего количества строк в таблице следует использовать функцию COUNT

со звездочкой COUNT со звездочкой включает как NULL-значения, так и повторяющиеся

значения, DISTINCT в этом случае не применим

SELECT COUNT (ALL rating) FROM Customers;<--пощитает строки с учётом дубликатов но не

учитывая null-значения.

Заметка: Если в выражении используются агрегированные ф-ции надо использовать group by, либо

необходимо чтобы все перечисленные столбцы в SELECT использовали агрегированные ф-ции.

**How to use the COUNT function in MySQL**

The COUNT function returns the number of rows in a table. With this function, you can count all the rows that fulfill(выполнить) a specified condition.

COUNT has three forms: COUNT(\*), COUNT(expression), and COUNT(DISTINCT expression).

Пример: сколько экземпляров книг каждого автора хранится на складе

**SELECT author, sum(amount)**

**FROM book**

**Group by author**

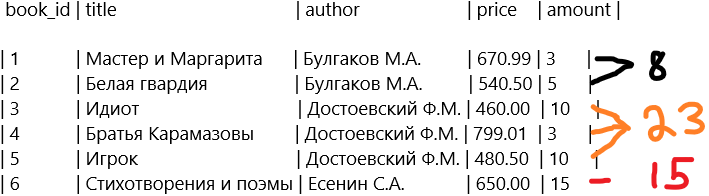
Group BY делит таблицу на группы, в одну группу добовляет только одного автора, сколько раз бы он не встречался в таблице, в другую группу другого автора, в конце получается сколько уникальных авторов, стоько и групп. Берётся только по одному автору из каждой группы, таким способом находятся уникальные значения по заданному столбцу(здесь author). затем sum(amount) берёт 1 уникальное значение из того столбца что было задано и проходит по каждому ряду в таблице, и где значение автора совподает с уникальным значением по которому сейчас происходит подсчёт, сумируется значение из колонки amount.

author | sum(amount)

| Булгаков М.А. | 8

| Достоевский Ф.М. | 23

| Есенин С.А. | 15



Пример:

Посчитать, количество различных книг и количество экземпляров книг каждого автора , хранящихся на складе. Столбцы назвать Автор, Различных\_книг и Количество\_экземпляров соответственно.

**SELECT author AS 'Автор',COUNT(title)AS 'Различных\_книг',SUM(amount)AS 'Количество\_экземпляров'**

**FROM book**

**GROUP BY author**

| Автор | Различных\_книг | Количество\_экземпляров

| Булгаков М.А. | 2 | 8

| Достоевский Ф.М. | 3 | 23

| Есенин С.А. | 1 | 15

**MIN()**

Вывести минимальную цену книги каждого автора, здесь ключевую роль играет GROUP BY, без него MIN() возможно вообще не работает, поэтому и называется групповые функции, сначала находятся уникальные элементы путём разбиения на группы и берётся только одно значение из каждой группы, так находится уникальность, и для каждого уникального значения подщитывается групповая функция

**SELECT author, MIN(price) AS min\_price**

**FROM book**

**GROUP BY author;**

author | min\_price |

| Булгаков М.А. | 540.50 |

| Достоевский Ф.М. | 460.00 |

| Есенин С.А. | 650.00

Пример: Вывести фамилию и инициалы автора, минимальную, максимальную и среднюю цену книг каждого автора . Вычисляемые столбцы назвать Минимальная\_цена, Максимальная\_цена и Средняя\_цена соответственно.

**SELECT author, MIN(price)AS'Минимальная\_цена', MAX(price)AS'Максимальная\_цена',AVG(price)AS'Средняя\_цена'**

**FROM book**

**GROUP BY author**

| author | Минимальная\_цена | Максимальная\_цена | Средняя\_цена |

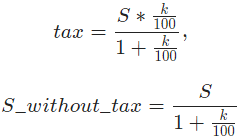
| Булгаков М.А. | 540.50 | 670.99 | 605.745000 |

| Достоевский Ф.М. | 460.00 | 799.01 | 579.836667 |

| Есенин С.А. | 650.00 | 650.00 | 650.000000

Сложный(или нет) пример:

Для каждого автора вычислить суммарную стоимость книг S (имя столбца Стоимость), а также вычислить налог на добавленную стоимость для полученных сумм (имя столбца НДС ) , который включен в стоимость и составляет 18% (k=18), а также стоимость книг (Стоимость\_без\_НДС) без него. Значения округлить до двух знаков после запятой. В запросе для расчета НДС(tax) и Стоимости без НДС(S\_without\_tax) использовать следующие формулы:



**SELECT author, SUM(amount\*price)AS'Стоимость',**

**ROUND(SUM((amount\*price)\*(18/100))/(1+(18/100)),2)AS'НДС',**

**ROUND(SUM(amount\*price)/(1+(18/100)),2)AS'Стоимость\_без\_НДС'**

**FROM book**

**GROUP BY author**

author | Стоимость | НДС | Стоимость\_без\_НДС

| Булгаков М.А. | 4715.47 | 719.31 | 3996.16

| Достоевский Ф.М. | 11802.03 | 1800.31 | 10001.72 |

| Есенин С.А. | 9750.00 | 1487.29 | 8262.71

пример:

Посчитать общее количество экземпляров книг на складе и их стоимость

**SELECT SUM(amount), SUM(amount\*price)**

**FROM book**

Результат:

| Количество | Стоимость |

| 46 | 26267.50

разбор:

В общем sum берёт **из одного ряда** price и amount и перемножает их между собой. Потом берёт следующий ряд, делает тоже самое. Результат умножения из первого ряда складывает с результатом умножения второго ряда и так до конца.

