ЛАБОРАТОРНАЯ РАБОТА №3

**АГРЕГАТНЫЕ ФУНКЦИИ. ГРУППИРОВАНИЕ.   
ОПЕРАТОР HAVING**

**ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

# **Агрегатные функции**

Агрегатные функции выполняют вычисление на наборе строк и возвращают единственное результирующее значение.

**Основные агрегатные функции:**

|  |  |  |
| --- | --- | --- |
| **Функция** | **Описание** | **Пример** |
| COUNT() | Возвращает количество строк в группе. | COUNT(\*) |
| SUM() | Возвращает сумму значений в столбце. | SUM(цена) |
| AVG() | Возвращает среднее арифметическое значений в столбце. | AVG(счет) |
| MIN() | Возвращает минимальное значение в столбце. | MIN(вес) |
| MAX() | Возвращает максимальное значение в столбце. | MAX(возраст) |

**Примеры использования:**

1. ***Общее количество записей в таблице:***

*SELECT COUNT(\*) AS total\_orders  
FROM ord;*

1. ***Сумма и среднее значение по столбцу:***

*SELECT SUM(amt) AS total\_amount, AVG(amt) AS avg\_amount  
FROM ord;*

**Важно:** При использовании агрегатных функций в одном запросе с обычными столбцами **без** GROUP BY возникает ошибка, так как СУБД не знает, как сопоставить множество строк с одним итоговым значением.

# **Оператор GROUP BY**

## Оператор GROUP BY группирует строки, имеющие одинаковые значения в указанных столбцах, в сводные строки. Он всегда используется вместе с агрегатными функциями для получения отчета по каждой группе. Можно группировать по комбинации нескольких полей.

*SELECT поле\_1, агрегатная\_функция (поле\_2)  
FROM имя\_таблицы  
GROUP BY поле\_1;*

**Примеры использования:**

***Общее количество заказов по каждому товару:***

*SELECT pnum, sum (amt)  
FROM ord  
GROUP BY pnum;*

# **Оператор HAVING**

## Оператор HAVING предназначен для фильтрации результатов группировки. Он применяется ПОСЛЕ того, как строки сгруппированы ( GROUP BY ) и вычислены агрегатные функции.

*SELECT поле\_1, агрегатная\_функция (поле\_2)  
FROM имя\_таблицы  
GROUP BY поле\_1  
HAVING условие;*

**Ключевое отличие HAVING от WHERE:**

|  |  |
| --- | --- |
| **WHERE** | **HAVING** |
| Фильтрует отдельные строки до их группировки. | Фильтрует уже сгруппированные результаты после группировки. |
| Может использоваться без GROUP BY . | Используется только в паре с GROUP BY. |
| Не может работать с агрегатными функциями. | Может работать с агрегатными функциями из SELECT . |

## ***Полный пример запроса с WHERE, GROUP BY и HAVING:***

## Вывести номера всех продавцов, у которых общее число заказов больше 2. Считать только заказы с количеством товаров amt более 1 штуки.

## *SELECT cnum, count(onum) FROM ord WHERE amt>1* – Фильтрация строк ДО группировки *GROUP BY cnum* – Группировка по номеру продавца *HAVING count(onum)>2;* – Фильтрация по агрегатной функции

**ПРАКТИЧЕСКОЕ ЗАДАНИЕ**

Перед выполнением практического задания 3 необходимо добавить в таблицу ord новые записи:

**insert** **into** **ord** **values**

(4021, 1001, 2001, 3001, 9, '2025-09-10'),

(4022, 1001, 2002, 3001, 1, '2025-09-14'),

(4023, 1001, 2006, 3002, 9, '2025-09-15'),

(4024, 1001, 2003, 3003, 2, '2025-09-12'),

(4025, 1001, 2001, 3004, 5, '2025-09-13'),

(4026, 1002, 2001, 3005, 7, '2025-09-30'),

(4027, 1002, 2002, 3001, 1, '2025-09-11'),

(4028, 1003, 2001, 3002, 2, '2025-09-21'),

(4029, 1003, 2006, 3003, 8, '2025-09-21'),

(4030, 1003, 2007, 3004, 1, '2025-09-21'),

(4031, 1003, 2007, 3003, 6, '2025-09-12'),

(4032, 1004, 2007, 3001, 6, '2025-09-20'),

(4033, 1004, 2001, 3004, 4, '2025-09-14'),

(4034, 1004, 2001, 3004, 4, '2025-09-16'),

(4035, 1004, 2001, 3003, 1, '2025-09-22'),

(4036, 1004, 2004, 3005, 3, '2025-09-16'),

(4037, 1004, 2007, 3002, 7, '2025-09-19'),

(4038, 1005, 2001, 3005, 5, '2025-09-17'),

(4039, 1006, 2004, 3005, 2, '2025-09-18'),

(4040, 1006, 2003, 3002, 38, '2025-09-01');

*Вариант 1*

1. Вывести количество покупателей с рейтингом больше 500.
2. Вывести первый и последний по алфавиту город из таблицы продавцов.
3. Вывести для каждого продавца наименьший номер покупателя, которому он продавал заказы.
4. Вывести число уникальных продавцов, которые продавали продукты с номерами 1001-1003.
5. Вывести для каждого продавца его номер и общее количество проданных продуктов, если это количество больше, чем 5.
6. Вывести количество заказов для всех уникальных пар покупатель-продавец, между которыми осуществлялись заказы.
7. Вывести все даты, в которые в заказах участвовали более 1 продавца   
   и более 1 покупателя.

*Вариант 2*

1. Вывести количество покупателей с именами, начинающими на букву A-D.
2. Вывести наименьший и наибольший рейтинг из таблицы покупателей.
3. Вывести для каждого покупателя наибольший номер продавца, у которого он совершал заказы.
4. Вывести число уникальных покупателей, которые покупали продукты с номерами 1004-1005.
5. Вывести для каждого покупателя его номер и общее количество проданных продуктов, если это количество меньше, чем 10.
6. Вывести количество заказов для всех уникальных пар продукт-продавец,   
   между которыми осуществлялись заказы.
7. Вывести все даты, в которые в заказах участвовали ровно 1 продавец   
   и 1 покупатель.

*Вариант 3*

1. Вывести количество продуктов, в названии которых больше 7 букв.
2. Вывести наименьшее и наибольшее значение комиссионных из таблицы продавцов.
3. Вывести для каждого продавца наименьший номер продукта, который он продавал.
4. Вывести число уникальных продуктов, которые продавались продавцами   
   с номерами 3001-3004.
5. Вывести для каждого продукта его номер и общее количество продукта во всех заказах, если это количество не менее 15.
6. Вывести количество заказов для всех уникальных пар продукт-покупатель, между которыми осуществлялись заказы.
7. Вывести все даты, в которые в заказах участвовали ровно 2 продавца   
   и более 1 продукта.

*Вариант 4*

1. Вывести число заказов, в которых количество продуктов не более 5.
2. Вывести самую раннюю и самую позднюю дату заказов.
3. Вывести для каждого покупателя наибольший номер продукта, который он покупал.
4. Вывести число уникальных продуктов, которые покупались покупателями  
   с номерами 2003-2006.
5. Вывести для каждого продавца его номер и среднее количество проданных продуктов, если это количество больше, чем 6.
6. Вывести количество заказов для всех уникальных пар продавец-покупатель, между которыми осуществлялись заказы.
7. Вывести все даты, в которые в заказах участвовали не менее 2 покупателей   
   и не менее 2 продуктов.

*Вариант 5*

1. Вывести количество продавцов из городов, начинающихся на буквы I-N.
2. Вывести наименьший и наибольший вес из таблицы продуктов.
3. Вывести для каждого продукта наименьший номер продавца, у которого этот продукт продавался.
4. Вывести число уникальных продавцов, которые продавали товары покупателям с номерами 2005 и 2004.
5. Вывести для каждого покупателя его номер и общее количество проданных продуктов, если это количество не меньше, чем 15.
6. Вывести количество заказов для всех уникальных пар продавец-продукт,   
   между которыми осуществлялись заказы.
7. Вывести все даты, в которые в заказах участвовали ровно 3 покупателя   
   и 1 продукт.

*Вариант 6*

1. Вывести количество продавцов с комиссионными менее 0.16.
2. Вывести первый и последний по алфавиту город из таблицы продуктов.
3. Вывести для каждого продукта наибольшее количество этого продукта в таблице заказов.
4. Вывести число уникальных покупателей, которые покупали товары у продавцов с номерами 3001, 3002 и 2006.
5. Вывести для каждого продукта его номер и общее количество продукта во всех заказах, если это количество менее 20.
6. Вывести количество заказов для всех уникальных пар покупатель-продукт,   
   между которыми осуществлялись заказы.
7. Вывести все даты, в которые в заказах участвовали более 3 покупателей   
   и любое число продуктов.