ЛАБОРАТОРНАЯ РАБОТА №4

**ПОДЗАПРОСЫ. ОПЕРАТОР CASE**

**ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

# **Подзапросы**

Подзапрос — это запрос, использующийся в другом SQL запросе. Подзапрос всегда заключён в круглые скобки и обычно выполняется перед основным запросом.

Как и любой другой SQL запрос, подзапрос возвращает результирующий набор, который может быть одним из следующих:

* одна строка и один столбец;
* нескольких строк с одним столбцом;
* нескольких строк с несколькими столбцами.

В зависимости от типа результирующего набора подзапроса определяются операторы, которые могут использоваться в основном запросе.

**Скалярный подзапрос** возвращает одну строку и один столбец. В условиях ограничений выборки (оператор WHERE) он используется с использованием операторов сравнения (=, <>, >, <).

***Пример использования:*** *вывести информацию о продавцах с максимальными комиссионными в таблице.*

*SELECT \* FROM sal  
WHERE comm =   
(SELECT MAX(comm) FROM sal);*

**Подзапрос, возвращающий несколько строк,** нельзя просто использовать с операторами сравнения, для работы с ними существуют операторы IN, ALL, ANY.

С помощью оператора **ALL** можно сравнивать отдельное значение с каждым значением в наборе, полученным подзапросом. Условие с ALL вернёт TRUE, только если все сравнения отдельного значения со значениями в наборе вернут TRUE. Условное выражение с **ANY** имеет схожее поведение, но оно возвращает TRUE, если хотя бы одно сравнение отдельного значения со значением в наборе вернёт TRUE. Оператор **IN** проверяет входит ли конкретное значение в набор значений (результат подзапроса).

***Пример использования:*** *вывести все информацию о продавцах, у которых не было ни одного заказа.*

*SELECT \* FROM sal  
WHERE snum <> ALL   
(SELECT snum FROM sal);*

**Коррелированный подзапрос –** это подзапрос, который ссылается на значения из внешнего запроса и выполняется многократно, по одному разу для каждой строки внешнего запроса. Он использует значения (поля) из внешнего запроса в своем выражении или фильтре.

**Некоррелированный подзапрос** не зависит от внешнего запроса, выполняется независимо, в первую очередь и один раз.

***Пример коррелированного подзапроса:*** *вывести для всех продавцов их номер и количество заказов, если это количество заказов меньше 10.*

*SELECT snum, (SELECT count(\*) FROM ord WHERE ord.snum = sal.snum)   
FROM sal   
WHERE (SELECT count(\*) FROM ord WHERE ord.snum = sal.snum) < 10*

Один и тот же запрос может быть переписан с использованием коррелированного или некоррелированного подзапроса. Коррелированные подзапросы зачастую менее эффективны, поскольку требуют больше времени на многократное исполнение.

# **Оператор CASE**

Оператор CASE в SQL подобен аналогичному оператору в различных ЯП и представляет собой множественный условный оператор.

*CASE  
 WHEN условие\_1 THEN возвращаемое\_значение\_1*

*…  
 WHEN условие\_n THEN возвращаемое\_значение\_n  
 [ELSE возвращаемое\_значение\_по\_умолчанию]  
END*

Оператор может иметь и более простой, но менее гибкий синтаксис:

*CASE значение  
 WHEN сравниваемое\_значение\_1 THEN возвращаемое\_значение\_1  
 …  
 WHEN сравниваемое\_значение\_n THEN возвращаемое\_значение\_n  
 [ELSE возвращаемое\_значение\_по\_умолчанию]*

*END*

***Пример использования:*** *вывести для каждого заказа его номер и категорию: «малый заказ», если его объем меньше или равен 5, «большой заказ»,   
если больше 5".*

*SELECT onum, amt,  
 CASE  
 WHEN amt<=5 THEN 'малый заказ'  
 ELSE 'большой заказ'  
 END as category  
FROM ord*

В SQL также существуют другие менее универсальные условные операторы, например, NULLIF и другие.

**ПРАКТИЧЕСКОЕ ЗАДАНИЕ**

Данная практическая работа направлена на освоение работы с подзапросами, поэтому выполнять задания необходимо БЕЗ ИСПОЛЬЗОВАНИЯ ОПЕРАТОРА СОЕДИНЕНИЯ ТАБЛИЦ JOIN.

*Вариант 1*

1. Вывести товары, вес которых больше среднего веса всех товаров.
2. Вывести общее количество проданных товаров для каждого продавца, у которого это количество больше, чем у продавца с номером 3003.
3. Выбрать продукты, которые заказывались покупателем с номером 2002 у продавцов, которые когда-либо делали заказы для покупателя 2001.
4. Вывести имена всех покупателей, которые никогда не заказывали у продавцов из Екатеринбурга.
5. Вывести покупателей, рейтинг которых больше, чем рейтинг каждого покупателя из Москвы.
6. Вывести покупателей, которые заказывали товары из своего города.
7. Вывести всю информацию о продуктах с дополнительным столбцом «категория товара» со следующими вариантами значений: «легкий», если вес меньше 500, «средний», если вес от 500 до 1000, «тяжелый», если вес больше 1000.
8. Вывести из таблицы заказов уникальные номера покупателей и показатель активности каждого из них на основе количества заказов: «активный» — 4+ заказа, «умеренный» — 2-3 заказа, «неактивный» — 1 заказ.

*Вариант 2*

1. Вывести покупателей, рейтинг которых больше среднего рейтинга всех покупателей.
2. Вывести общее число выполненных заказов для каждого продавца, у которого это число больше, чем у продавца с номером 3002.
3. Выбрать продукты, которые поставлялись продавцом с номером 3002 для покупателей, которые когда-либо делали заказы у продавца 3005.
4. Вывести названия всех продавцов, которые никогда не заказывали товары из Санкт-Петербурга.
5. Вывести заказы, в которых количество товара больше, чем количество товара в каждом из заказов за 1 сентября 2025.
6. Вывести товары, которые никогда не заказывались покупателями из города, в котором произведен этот товар.
7. Вывести всю информацию о покупателях с дополнительным столбцом «категория покупателя» со следующими вариантами значений: «новый», если рейтинг меньше 200, «постоянный», если рейтинг от 200 до 300, «ВИП-клиент», если рейтинг больше 300.
8. Вывести из таблицы заказов уникальные номера покупателей и показатель активности каждого из них по датам заказов: «постоянный» — заказы в более 5 разных дат, «частый» — заказы в 3-5 различных дат, «редкий» — заказы в 1-2 даты.

*Вариант 3*

1. Вывести продавцов, комиссионные которых не менее средних комиссионных всех продавцов.
2. Вывести общее количество купленных товаров для каждого покупателя, у которого это количество не больше, чем у покупателя с номером 2003.
3. Выбрать продавцов, которые продавали продукты с номером 1001 для покупателей, которые когда-либо покупали продукт с номером 1005.
4. Вывести названия всех товаров, которые никогда не продавали продавцы из Москвы.
5. Вывести покупателей, рейтинг которых не меньше. чем рейтинг хотя бы одного покупателя из Москвы.
6. Вывести продавцов, которые не продавали товары покупателям из своего города.
7. Вывести всю информацию о продавцах с дополнительным столбцом «вид комиссии» со следующими вариантами значений: «низкая», если комиссионные меньше 0.12, «средняя», если комиссионные от 0.12 до 0.13, «высокая», если комиссионные больше 0.13.
8. Вывести из таблицы заказов уникальные номера продавцов и показатель активности каждого из них на основе количества заказов: «активный» — 4+ заказа, «умеренный» — 2-3 заказа, «неактивный» — 1 заказ.

*Вариант 4*

1. Вывести продавцов, которые имеют минимальные комиссионные среди всех продавцов.
2. Вывести общее число совершенных заказов для каждого покупателя, у которого это число не меньше, чем у покупателя с номером 2004.
3. Выбрать продавцов, которые поставляли заказ покупателю с номером 2006 и с продуктами, которые когда-либо заказывались покупателем с номером 2002.
4. Вывести названия всех товаров, которые никогда не заказывали покупатели из Москвы.
5. Вывести заказы, в которых количество товара больше, чем количество товара хотя бы в одном из заказов за 1 сентября 2025.
6. Вывести покупателей, которые делали заказы у продавцов из своего города.
7. Вывести всю информацию о заказах с дополнительным столбцом «размер партии товара» со следующими вариантами значений: «мелкая», если количество товара в заказе меньше 5, «средняя», если количество товара от 5 до 9, «крупная», если количество товара больше 10.
8. Вывести из таблицы заказов уникальные номера продавцов и показатель активности каждого из них по датам заказов: «постоянный» — заказы в более 5 разных дат, «частый» — заказы в 3-5 различных дат, «редкий» — заказы в 1-2 даты.

*Вариант 5*

1. Вывести покупателей, которые имеют минимальный рейтинг среди всех покупателей.
2. Для каждого товара вывести общее количество заказов с этим товаром, только если это количество заказов больше, чем у товара с номером 1001.
3. Выбрать покупателей, которые покупали продукты с номером 1004 у продавцов, которые когда-либо продавали продукт с номером 1002.
4. Вывести названия всех продавцов, которые никогда не работали с покупателями из Новосибирска.
5. Вывести товары, вес которых больше веса хотя бы одного товара из города Новосибирск.
6. Вывести товары, которые никогда не продавались продавцами из города, в котором произведен этот товар.
7. Вывести всю информацию о продавцах с дополнительным столбцом «округ продавца» со следующими вариантами значений: «северо-западный», если город Санкт-Петербург, «центральный», если город Москва, «сибирский», если город Новосибирск, или «другой» для остальных городов.
8. Вывести из таблицы заказов уникальные номера продуктов и показатель популярности каждого из них на основе количества заказов: «популярный» — 4+ заказа, «умеренный» — 2-3 заказа, «непопулярный» — 1 заказ.

*Вариант 6*

1. Вывести заказы, в которых количество товара выше, чем среднее количество товаров во всех заказах.
2. Для каждого товара вывести общее количество этого товара во всех заказах, только если это количество меньше, чем у товара с номером 1002.
3. Выбрать покупателей, которые осуществляли заказ у продавца с номером 3006 и с продуктами, которые когда-либо продавались продавцом с номером 3002.
4. Вывести имена всех покупателей, которые никогда не заказывали товары из города Обнинск.
5. Вывести товары, вес которых больше веса каждого из товаров из города Новосибирск.
6. Найти продавцов, которые продавали товары из своего города.
7. Вывести всю информацию о продуктах с дополнительным столбцом «статус города продукта» со следующими вариантами значений: «федерального значения», если город Санкт-Петербург или Москва, «миллионник», если город Новосибирск или Екатеринбург, или «другое» для остальных городов.
8. Вывести из таблицы заказов уникальные номера продуктов и показатель популярности каждого из них по датам заказов: «ежедневный» — заказы в более 5 разных дат, «сезонный» — заказы в 3-5 различных дат, «редкий» — заказы в 1-2 даты.