**ТЕМА 2. БАЗОВЫЕ SELECT ЗАПРОСЫ**

1. **Выборка данных с SELECT**

SELECT - это основная команда SQL для выборки данных из таблицы. Пример:

SELECT \*

FROM products;

Этот запрос выбирает все строки из таблицы "products".

SELECT product\_id, product\_name, unit\_price

FROM products;

Этот запрос выбирает столбцы product\_id, product\_name и unit\_price из таблицы "products".

1. **Выборка и простая арифметика в SQL**

Можно выполнять простые арифметические операции в SQL, например:

SELECT product\_id, product\_name, unit\_price \* units\_in\_stock AS total\_price

FROM products;

Этот запрос умножает unit\_price на units\_in\_stock и выводит результат в столбце total\_price.

1. **Использование DISTINCT в запросах**

DISTINCT используется для выбора уникальных значений из столбца:

SELECT DISTINCT title

FROM employees;

Этот запрос вернет уникальные значения из столбца "title" таблицы "employees".

SELECT DISTINCT country, title

FROM employees;

Этот запрос вернет уникальные комбинации значений из столбцов "country" и "title" таблицы "employees".

1. **Использование COUNT в запросах**

COUNT используется для подсчета строк в выборке:

SELECT COUNT(\*)

FROM employees;

Этот запрос вернет общее количество строк в таблице "employees".

SELECT COUNT(DISTINCT country)

FROM employees;

Этот запрос вернет количество уникальных значений в столбце "country" таблицы "employees".

1. **Использование WHERE в запросах**

WHERE используется для фильтрации результатов:

SELECT company\_name, contact\_name, phone, country

FROM customers

WHERE country = 'USA';

Этот запрос вернет данные о клиентах из США.

SELECT \*

FROM products

WHERE unit\_price > 20;

Этот запрос вернет все продукты с ценой выше 20.

SELECT COUNT(\*)

FROM products

WHERE unit\_price < 20;

Этот запрос вернет количество продуктов с ценой ниже 20.

SELECT \*

FROM products

WHERE discontinued = 1;

Этот запрос вернет все продукты, которые были сняты с производства (значение "discontinued" равно 1).

1. **AND и OR**

Можно комбинировать условия с помощью операторов AND и OR:

SELECT \*

FROM products

WHERE unit\_price > 25 AND units\_in\_stock > 40;

Этот запрос вернет продукты с ценой выше 25 и количеством в наличии больше 40.

SELECT \*

FROM customers

WHERE city = 'Berlin' OR city = 'London' OR city = 'San Francisco';

Этот запрос вернет клиентов из городов Берлин, Лондон или Сан-Франциско.

1. **BETWEEN**

BETWEEN используется для выбора значений в заданном диапазоне:

SELECT \*

FROM orders

WHERE freight BETWEEN 20 AND 40;

Этот запрос вернет заказы с стоимостью доставки от 20 до 40.

1. **IN, NOT IN**

IN и NOT IN используются для выбора значений из списка:

SELECT \*

FROM customers

WHERE country IN ('Mexico', 'Germany', 'USA', 'Canada');

Этот запрос вернет клиентов из Мексики, Германии, США или Канады.

SELECT \*

FROM products

WHERE categoryid NOT IN (1, 3, 5, 7);

Этот запрос вернет продукты, которые не принадлежат категориям 1, 3, 5 или 7.

1. **Сортировка с ORDER BY**

ORDER BY используется для сортировки результатов:

SELECT DISTINCT country

FROM customers

ORDER BY country;

Этот запрос вернет уникальные страны клиентов, отсортированные по алфавиту.

SELECT DISTINCT country, city

FROM customers

ORDER BY country DESC, city ASC;

Этот запрос вернет уникальные комбинации страны и города клиентов, сначала отсортированные по убыванию страны, а затем по возрастанию города.

1. **Вычисление минимума, максимума и среднего: MIN, MAX, AVG**

Можно вычислять минимум, максимум и среднее значение столбца:

SELECT MIN(order\_date)

FROM orders

WHERE ship\_city = 'London';

Этот запрос вернет минимальную дату заказа для заказов, доставленных в Лондон.

SELECT MAX(order\_date)

FROM orders

WHERE ship\_city = 'London';

Этот запрос вернет максимальную дату заказа для заказов, доставленных в Лондон.

SELECT AVG(unit\_price)

FROM products

WHERE discontinued <> 1;

Этот запрос вернет среднюю цену продуктов, исключая те, которые сняты с производства (значение "discontinued" не равно 1).

1. **LIKE в запросах для поиска строк**

LIKE используется для поиска строк по шаблону:

SELECT last\_name, first\_name

FROM employees

WHERE first\_name LIKE '%n';

Этот запрос вернет сотру

дников, у которых имя содержит букву "n" в любом месте.

SELECT last\_name, first\_name

FROM employees

WHERE last\_name LIKE 'B%';

Этот запрос вернет сотрудников, у которых фамилия начинается с буквы "B".

SELECT last\_name, first\_name

FROM employees

WHERE last\_name LIKE 'Buch%';

Этот запрос вернет сотрудников, у которых фамилия начинается с "Buch".

SELECT last\_name, first\_name

FROM employees

WHERE last\_name LIKE '\_uch%';

Этот запрос вернет сотрудников, у которых фамилия начинается с любой буквы, затем следует "uch".

1. **LIMIT**

LIMIT используется для ограничения количества возвращаемых строк:

SELECT product\_name, unit\_price

FROM products

WHERE discontinued <> 1

ORDER BY unit\_price DESC

LIMIT 10;

Этот запрос вернет только первые 10 продуктов, исключая те, которые сняты с производства.

1. **Проверка на NULL: IS NULL, IS NOT NULL**

IS NULL и IS NOT NULL используются для проверки наличия или отсутствия значений NULL:

SELECT ship\_city, ship\_region, ship\_country

FROM orders

WHERE ship\_region IS NULL;

Этот запрос вернет заказы, у которых поле "ship\_region" равно NULL.

SELECT ship\_city, ship\_region, ship\_country

FROM orders

WHERE ship\_region IS NOT NULL;

Этот запрос вернет заказы, у которых поле "ship\_region" не равно NULL.

1. **Группировка GROUP BY**

GROUP BY используется для группировки результатов:

SELECT ship\_country, COUNT(\*)

FROM orders

WHERE freight > 50

GROUP BY ship\_country

ORDER BY COUNT(\*) DESC;

Этот запрос вернет количество заказов для каждой страны доставки, где стоимость доставки больше 50, отсортированных по убыванию количества.

SELECT category\_id, SUM(UnitsInStock)

FROM products

GROUP BY category\_id

ORDER BY SUM(units\_in\_stock) DESC

LIMIT 5;

Этот запрос вернет сумму остатков продуктов для каждой категории, сгруппированных по категориям и отсортированных по убыванию суммы. Ограничено выводом первых 5 результатов.

1. **Постфильтрация с HAVING**

HAVING используется для фильтрации результатов после группировки:

SELECT category\_id, SUM(unit\_price \* units\_in\_stock) AS total\_value

FROM products

WHERE discontinued <> 1

GROUP BY category\_id

HAVING total\_value > 5000

ORDER BY total\_value DESC;

Этот запрос вернет сумму стоимости продуктов в каждой категории (исключая снятые с производства), сгруппированных по категориям, и оставит только те категории, где суммарная стоимость больше 5000.

Результаты отсортированы по убыванию суммарной стоимости.

1. **Объединение, пересечение, разница в SQL - UNION, INTERSECT, EXCEPT**

UNION, INTERSECT и EXCEPT используются для объединения, пересечения и разницы результатов соответственно:

-- UNION & UNION ALL - одинаковое количество столбцов в обоих запросах

SELECT country

FROM customers

UNION

SELECT country

FROM employees;

SELECT country

FROM customers

UNION ALL

SELECT country

FROM employees;

-- если нужна уникальность, используйте UNION

SELECT DISTINCT country

FROM customers

UNION

SELECT DISTINCT country

FROM employees;

SELECT country

FROM customers

INTERSECT

SELECT country

FROM suppliers;

SELECT country

FROM customers

EXCEPT

SELECT country

FROM suppliers;

-- EXCEPT возвращает все строки, которые есть в результате query1, но отсутствуют в результате query2. -- (Иногда это называют разницей между двумя запросами.)

-- Опять же, дубликаты удаляются, если не используется EXCEPT ALL.

SELECT country

FROM customers

EXCEPT ALL

SELECT country

FROM suppliers;

Этот запрос вернет уникальные идентификаторы клиентов и сотрудников.