SQL-запросы

Интернет · 14.01.2026 · PDF-версия статьи с сайта cms.dolinais.ru

SQL — это язык запросов для управления базами данных.

Основные команды SQL для управления базами данных

Что такое SQL?

SQL (Structured Query Language) — это стандартный язык запросов для взаимодействия с реляционными базами данных. Он предназначен для выполнения операций над данными, такими как выборка, обновление, удаление и создание объектов базы данных.

Типы запросов в SQL

Существует четыре основных категории запросов в SQL:

1. DDL (Data Definition Language) — определяет структуру данных.

   Примеры команд: CREATE, DROP, ALTER.

2. DML (Data Manipulation Language) — управляет данными в таблицах.

   Примеры команд: SELECT, INSERT, UPDATE, DELETE.

3. DCL (Data Control Language) — контролирует доступ к данным.

   Примеры команд: GRANT, REVOKE.

4. TCL (Transaction Control Language) — управляет транзакциями.

   Примеры команд: COMMIT, ROLLBACK.

Структура SQL-запроса

Классический SQL-запрос включает шесть наиболее часто используемых операторов:

SELECT: выбирает столбцы или таблицу целиком.

FROM: указывает источник данных (таблицу).
WHERE: задаёт фильтры для отбора данных.
GROUP BY: группирует данные по указанному полю.
HAVING: применяется для фильтрации сгруппированных данных.
ORDER BY: сортирует результат по указанным полям.

Практические примеры SQL-запросов

Создание первой базы данных

Начнем с простого примера создания базы данных и добавления записей.

Шаг 1: Создаем базу данных

```
CREATE DATABASE cats_db;
```

Шаг 2: Создаем таблицу с параметрами

```
USE cats_db;
CREATE TABLE cats (
    cat_id INT PRIMARY KEY AUTO_INCREMENT,
    cat_name VARCHAR(50),
    age INT,
    color VARCHAR(50)
);
```

Шаг 3: Заполняем таблицу данными

```
INSERT INTO cats(cat_name, age, color) VALUES ('Барсик', 3, 'Серый');
INSERT INTO cats(cat_name, age, color) VALUES ('Матроскин', 5, 'Белый');
```

Шаг 4: Выборка данных

```
SELECT * FROM cats WHERE age > 4;
```

Агрегатные функции

Для обработки больших объемов данных SQL поддерживает ряд встроенных функций:

COUNT(): подсчитывает количество строк.
SUM(): сумма значений в столбце.
AVG(): среднее арифметическое.

MIN(): минимальное значение.
MAX(): максимальное значение.

Пример использования агрегирования:

SELECT AVG(age) AS average_age FROM cats;

Итоговые выводы

Основные моменты, которые важно помнить:

SQL необходим для эффективного управления данными в реляционных базах.
Различаются запросы по типу выполняемых действий: изменение структуры, манипуляции данными, контроль доступа и транзакций.
Базовая структура SQL-запроса строится вокруг обязательных операторов SELECT и FROM.
Существует множество полезных дополнительных команд и функций, позволяющих эффективно обрабатывать большие объемы данных.

Таким образом, изучив основы SQL, вы сможете уверенно взаимодействовать с любыми реляционными системами хранения данных.

Источник: https://task.dolinais.ru/sql-zaprosy