Вебинар 5

Операторы, метасимволы и функции в MySQL

Инструкция SELECT

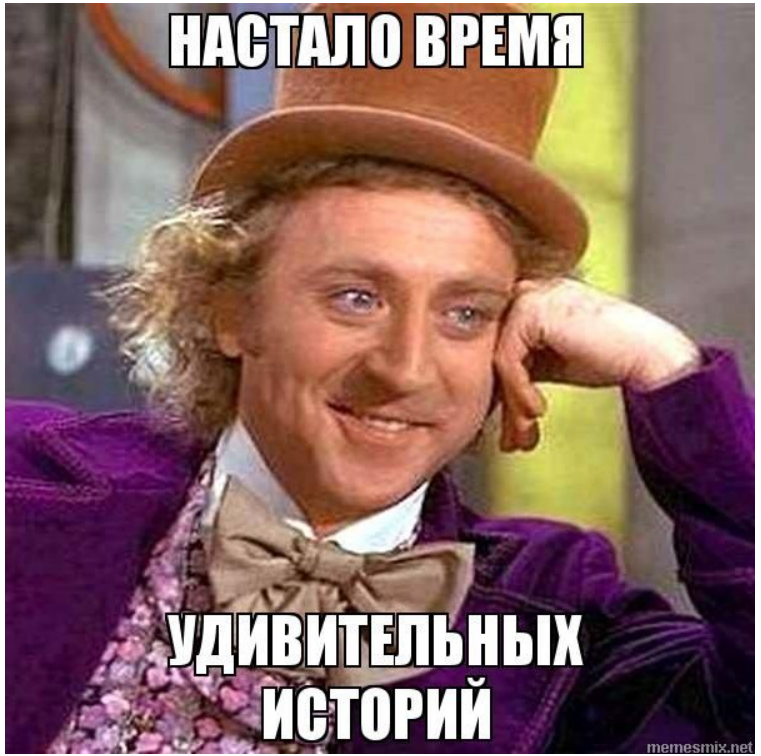
Вспомним, как работает инструкция SELECT.

1. Что надо написать, чтобы вывести всю таблицу сразу?

2. А как мы выводим один отдельный столбец?

3. А если если нам нужно несколько отдельных столбцов?

4. А если мы хотим вывести только уникальные строки?



Инструкция SELECT

Вспомним, как работает инструкция SELECT.

1. Что надо написать, чтобы вывести всю таблицу сразу?

2. А как мы выводим один отдельный столбец?

3. А если если нам нужно несколько отдельных столбцов?

4. А если мы хотим вывести только уникальные строки?

1. Чтобы вывести всю таблицу, надо написать SELECT * FROM таблица;

2. Для отдельного столбца: SELECT столбец FROM таблица

3. Для нескольких столбцов просто указать имена столбцов через запятую: SELECT столбец1, столбец2 FROM таблица

4. А про уникальные строки сейчас поговорим :)

Извлечение уникальных строк

Как мы помним, наши инструкции SELECT возвращают нам абсолютно все значения в выбранных столбцах. А если нам нужно выбрать только уникальные значения?

Для таких случаев есть специальное ключевое слово DISTINCT.

SELECT DISTINCT столбец FROM таблица

Вспомним, как выглядит наша таблица с персонажами:
SELECT * FROM myCharacters;

И применим ключевое слово DISTINCT:
SELECT DISTINCT weapon FROM myCharacters;

Что же получается?

Вся таблица:

Output:

char_id	name	weapon
1	Al-Haitham	Sword
2	Jinx	Gun
3	Steve	Sword
4	Jett	Gun
5	Tracer	Gun
6	Pudge	NULL

Таблица с DISTINCT weapon:

Output:

weapon
Sword
Gun
NULL

А что если..?

А если мы напишем

SELECT DISTINCT * FROM myCharacters; ?

Некоторые компиляторы допускают такую запись, некоторые – нет. Может возникнуть ошибка.

А если мы напишем

SELECT DISTINCT name, weapon FROM myCharacters; ?

Для эксперимента продублируем одну из записей. Например,
INSERT INTO myCharacters(name, weapon)
VALUES ("Al-Haitham", 'Sword');
И дальше напишем вывод:

Output:

char_id	name	weapon
1	Al-Haitham	Sword
2	Jinx	Gun
3	Steve	Sword
4	Jett	Gun
5	Tracer	Gun
6	Pudge	NULL
7	Al-Haitham	Sword

Output:

weapon	name
Sword	Al-Haitham
Gun	Jinx
Sword	Steve
Gun	Jett
Gun	Tracer
NULL	Pudge

Сортировка записей

Для сортировки данных есть предложение ORDER BY. В нем указывается имя одного или нескольких столбцов, по которым и сортируются результаты запроса.

SELECT name, weapon FROM myCharacters

ORDER BY weapon;

name	weapon
Jinx	Gun
Jett	Gun
Tracer	Gun
Al-Haitham	Sword
Steve	Sword

Как вы видите, у нас вывелись столбцы name и weapon, указанные в инструкции SELECT, а благодаря сортировке - значения столбца weapon стали идти в алфавитном порядке, однако столбец name изменился не полностью!

(обратите внимание на Jinx и Jett)

Сортировка по нескольким столбцам.

Синтаксис использования инструкции сортировки такой же. Разница лишь в том, что вам потребуется перечислить необходимые столбцы через запятую.

SELECT name, weapon FROM myCharacters

ORDER BY weapon, name;

name	weapon
Jett	Gun
Jinx	Gun
Tracer	Gun
Al-Haitham	Sword
Steve	Sword

При данной сортировке, вывелись столбцы name и weapon, указанные в инструкции SELECT, а значения столбца weapon и name стали идти в алфавитном порядке.

(обратите внимание на Jinx и Jett, они поменялись местами)

Примечание:

Приоритет сортировки будет у того столбца, который вы указали первым.

Второй указанный столбец будет “подстраиваться”, при этом не нарушая целостность строк - отсюда и идет смена позиций Jett и Jinx, а также то, что Al-Haitham стоит не в начале списка

Операторы

Как и в C++, в SQL есть приоритет операций (1 - наибольший приоритет, 7 - наименьший):

1. круглые скобки
2. умножение (*), деление (/)
3. сложение (+), вычитание (-)
4. операторы сравнения (=, >, <, >=, <=, <>)
5. NOT
6. AND
7. OR

Нотация	Значение	Нотация	Значение
*	знак умножения	<>	знак неравенства
/	знак деления	>=	больше либо равно
-	знак вычитания	<=	меньше либо равно
+	знак сложения	AND	логическое и
( )	скобки, указывают на порядок действий в выражении или используются при вызове функции	OR	логическое или
>	знак больше	NOT	логическое не
<	знак меньше	BETWEEN a AND b	все значения в промежутке от a до b
=	знак равенства	IN(a, b)	все значения, которые соответствуют списку

Оператор WHERE. Фильтрация данных.

Зачастую необходимо извлекать не все данные из БД, а только те, которые соответствуют определенному условию.

Для фильтрации данных в команде SELECT применяется оператор WHERE, после которого указывается условие.

```
SELECT name, weapon FROM myCharacters
WHERE weapon = "Gun";
```

name	weapon
Jinx	Gun
Jett	Gun
Tracer	Gun

Оператор WHERE. Фильтрация данных.

Добавим к таблице новый столбец, для того, чтобы посмотреть, как работает фильтрация данных с использованием логических операторов:

AND: операция логического И. Она объединяет два выражения.

OR: операция логического ИЛИ. Она также объединяет два выражения.

NOT: операция логического отрицания. Если выражение в этой операции ложно, то общее условие истинно.

```
CREATE TABLE myCharacters (char_id int primary key auto_increment, name char(20), weapon char(20), hp int);

INSERT into mycharacters (name, weapon, hp) values ("Al-Haitham", "Sword", "456"), ("Jinx", "Gun", "782"),

("Steve", "Sword", "783"), ("Jett", "Gun", "873"), ("Tracer", "Gun", "983");

SELECT*FROM myCharacters;

SELECT name, weapon, hp FROM myCharacters WHERE weapon = "Gun" AND hp > 600;
```

name	weapon	hp
Jinx	Gun	782
Jett	Gun	873
Tracer	Gun	983

Оператор LIKE. Фильтрация данных.

Все предыдущие операторы, которые мы рассмотрели, выполняли фильтрацию по известным значениям. Они искали совпадения по одному или нескольким значениям, осуществляли проверку “больше-меньше” или проверку на вхождение в диапазон значений. При этом везде искалось известное значение. Однако фильтрация данных таким способов не всегда работает.

MySQL условие LIKE позволяет использовать шаблоны в операторе WHERE для оператора SELECT, INSERT, UPDATE или DELETE. Это позволяет выполнять сопоставление шаблонов.

Первый MySQL пример LIKE, который мы рассмотрим, включает использование % (подстановочный символ процента).

Рассмотрим как % работает в MySQL условии LIKE. Мы хотим найти все имена персонажей, которые оканчиваются на “cer”.

SELECT name, weapon FROM myCharacters WHERE name LIKE "%cer%";

name	weapon
Tracer	Gun

Примечание:

Если мы хотим посмотреть на имена, которые начинаются с определенных букв, то в таком случае нужно указать “Tra%”, или любые другие буквы.

Ну а больше информации про метасимволы вы узнаете в методичке.