Содержание

MySQL Администрирование 2

MySQL Запросы 2

MySQL Таблицы и представления(view) 4

MySQL Программирование 4

Индексы, уникальные ограничения и первичные ключи в MySQL 4

Процедуры, функции в MySQL 4

Функции MySQL 4

Процедуры MySQL 6

MySQL Триггеры 8

MySQL Курсоры 8

MySQL Условные операторы 8

MySQL Циклы 8

MySQL условия 8

ТЕСТ 1 и 2 ПРОЦЕДУРЫ И ФУНКЦИИ MY SQL 9

https://quizizz.com/join?gc=44696843 9

https://quizizz.com/join?gc=08652043 9

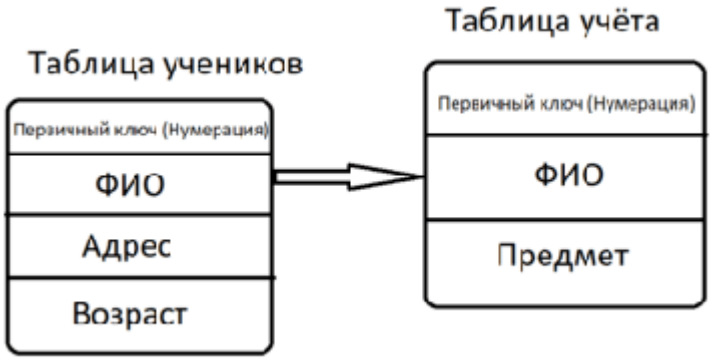
MySQL Администрирование

CREATE USER оператор MySQL	Оператор CREATE USER создает учетную запись базы данных, которая позволяет вам входить в базу данных
DROP USER оператор MySQL	используется для удаления пользователя из базы данных
Изменить пароль пользователя в MySQL	используется для изменения пароля пользователя в базе данных
Переименовать пользователя в MySQL	используется для переименования пользователя в базе данных
Найти пользователя в MySQL	используется для определения пользователей в базе данных
Поиск пользователей вошедших в MySQL	используется для определения пользователей, которые в настоящее время вошли (залогинились) в базу данных
Показывать гранты для пользователя в MySQL	используется для отображения всех грантов для пользователя базы данных
Grant/Revoke привилегии MySQL	Используется для предоставления или отмены привилегий пользователям на таблицы, функции или процедуры

MySQL Запросы

SELECT	извлекает записи из одной или нескольких таблиц
SELECT LIMIT	извлекает записи из одной или нескольких таблиц с ограничением количества возвращаемых записей
FROM	используется для отображения таблиц и любых присоединений
DISTINCT	используется для удаления дубликатов из набора результатов
WHERE	используется для фильтрации результатов в операторах SELECT
ORDER BY	используется для сортировки записей
GROUP BY	используется для группировки записей
HAVING	используется в сочетании с оператором GROUP BY, чтобы ограничить группы возвращаемых строк только тех, чье условие TRUE
INSERT	используется для вставки одной записи или несколько записей в таблицу
UPDATE	используется для обновления существующих записей в таблице
DELETE	используется для удаления одной записи или нескольких записей из таблицы

DELETE LIMIT	используется для удаления записей из таблицы и ограничения количества удаляемых записей
TRUNCATE TABLE	используется для очищения всех записей из таблицы
UNION	возвращает все строки из запроса и удаляет повторяющиеся строки между различными предложениями SELECT
UNION ALL	возвращает все строки из запроса и не удаляет повторяющиеся строки между различными предложениями SELECT
Joins	используются для извлечения данных из нескольких таблиц
Подзапросы	представляет собой запрос в запросе



оператор Select.

сначала мы обращаемся к БД оператором select (от англ. "выбрать") затем мы указываем что, после чего используем оператор from (от англ. "из") и указываем таблицу из которой мы хотим получить данные.

Пример:

select название_столбца from название таблицы

если использовать знак "*" вместо название столбца, то запрос выдаст все столбцы.

если нам нужно из таблицы учеников нужно получить ФИО учеников, необходимо использовать такой запрос:

select ФИО from Таблица учеников

Вывод конкретных данных

Иногда необходимо узнать данные определённой строки, а строк в таблице может быть сотни, а может даже тысячи и десятки тысяч. Для этого был придуман оператор where, который позволяет отсеять столбцы указав условие, которое должно быть выполнено, что бы столбцы были выбраны оператором select.

Пример:

select FIO from Ycheniki where name='Ivan'

Такой оператор выберет из всей таблицы только значения столбика FIO где FIO равняется Ivan. Так же стоит заметить, что для столбцов с типом "текст" необходимо использовать кавычки как на примере.

MySQL Таблицы и представления(view)

CREATE TABLE	позволяет создавать и определять таблицу
CREATE TABLE AS	используется для создания таблицы из существующей таблицы путем копирования столбцов существующей таблицы
ALTER TABLE	используется для добавления, изменения или удаления столбцов в таблице
DROP TABLE	позволяет стереть или удалить таблицу из базы данных
VIEW	представляет собой виртуальную таблицу, созданную запросом, соединяющим одну или несколько таблиц

MySQL Программирование

- Типы данных MySQL
- Алиасы в MySQL
- Sequences (последовательности) MySQL
- Литералы
- Объявление переменных
- Комментарии

Индексы, уникальные ограничения и первичные ключи в MySQL

- Indexes
- Unique Constraints
- Primary Keys

Процедуры, функции в MySQL

- Как создать или удалить функцию

Функции MySQL

Это учебное пособие объясняет, как создавать и удалять **функции в MySQL** с синтаксисом и примерами.

Определение

В MySQL функция - это хранимая программа, в которую вы можете передавать параметры и возвращать значение.

Создать функцию (create function)

Как и на других языках программирования, вы можете создавать свои собственные функции в MySQL.

Синтаксис

Синтаксис создания функции в MySQL:

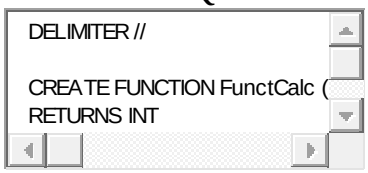
CREATE FUNCTION function_name [(parameter datatype [, parameter datatype])]
RETURNS return_datatype
BEGIN
declaration_section
executable_section
END;

Параметры или аргументы

function_name - наименование функции в MySQL
parameter - один или несколько параметров передаются в функцию. При создании функции все параметры считаются параметрами IN (не OUT или IN OUT), где параметры могут ссылаться на функцию, но не могут быть перезаписаны функцией.
return_datatype - тип данных возвращаемого значения функции.
declaration_section - место в функции, где вы объявляете локальные переменные.
executable_section - место в функции, в которой вы создаете код функции.

Пример

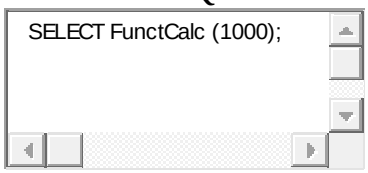
Рассмотрим пример, как создать функцию в MySQL:
Oracle PL/SQL




```
1  DELIMITER //
2
3  CREATE FUNCTION FunctCalc ( starting_value INT )
4  RETURNS INT
5
6  BEGIN
7
8    DECLARE income INT;
9
10   SET income = 0;
11
12   label1: WHILE income <= 8000 DO
13     SET income = income + starting_value;
14   END WHILE label1;
15
16   RETURN income;
17
18 END; //
19
20 DELIMITER ;
```

Затем вы можете ссылаться на свою новую функцию следующим образом:

Oracle PL/SQL




```
1 SELECT FunctCalc (1000);
```

УДАЛИТЬ ФУНКЦИЮ (DROP FUNCTION)

После того, как вы создали функцию в MySQL, вам может понадобиться удалить ее из базы данных.

Синтаксис

Синтаксис удаления функция в MySQL:

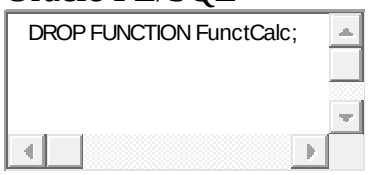
DROP FUNCTION [IF EXISTS] function_name;

function_name - наименование функции, которую вы хотите удалить.

Пример

Рассмотрим пример, как удалить функцию в MySQL.

Oracle PL/SQL




```
1 DROP FUNCTION FunctCalc;
```

В этом примере удаляем функцию под названием FunctCalc.

- Как создать или удалить процедуру

Процедуры MySQL

В MySQL процедура - это хранимая программа, в которую вы можете передавать параметры. Процедура не возвращает значение, как функция.

Создать процедуру (create Procedure)

Как и на других языках программирования, вы можете создавать свои собственные процедуры в MySQL. Рассмотрим подробнее.

Синтаксис

Синтаксис создания процедуры в MySQL:

CREATE PROCEDURE procedure_name [(parameter datatype [, parameter datatype])]

BEGIN

declaration_section

executable_section

END;

Параметры или аргументы

procedure_name - наименование процедуры в MySQL.

parameter - необязательный. Один или несколько параметров передаются в процедуру. При создании процедуры могут быть объявлены три типа параметров:

IN - Параметр может ссылаться на процедуру. Значение параметра не может быть перезаписано процедурой.

OUT - Параметр не может ссылаться на процедуру, но значение параметра может быть перезаписано процедурой.

IN OUT - Параметр может ссылаться на процедуру, и значение параметра может быть перезаписано процедурой.

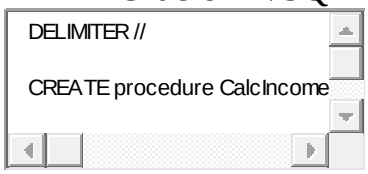
declaration_section - место в процедуре, где вы объявляете локальные переменные.

executable_section - место в процедуре, в котором вы создаете код процедуры.

Пример

Рассмотрим пример, как создать процедуру в MySQL:

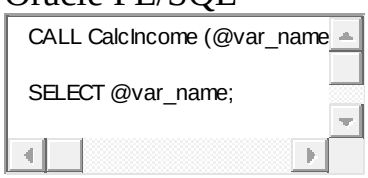
Oracle PL/SQL




```
1  DELIMITER //
2
3  CREATE procedure CalcIncome ( OUT ending_value INT )
4
5  BEGIN
6
7    DECLARE income INT;
8
9    SET income = 50;
10
11   label1: WHILE income <= 3000 DO
12     SET income = income * 2;
13   END WHILE label1;
14
15   SET ending_value = income;
16
17 END; //
18
19 DELIMITER ;
```

Затем вы можете ссылаться на свою новую процедуру следующим образом:

Oracle PL/SQL




```
1 CALL CalcIncome (@var_name);
2
3 SELECT @var_name;
```

УДАЛИТЬ процедуру (DROP procedure)

После того, как вы создали процедуру в MySQL, вам может понадобиться удалить ее из базы данных.

Синтаксис

Синтаксис удаления процедуры в MySQL:

DROP procedure [IF EXISTS] procedure_name;

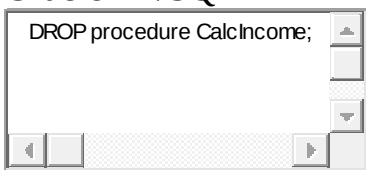
procedure_name - наименование процедуры, которую вы хотите удалить.

Пример

Рассмотрим пример удаления процедуру в MySQL.

Например:

Oracle PL/SQL



DROP procedure CalcIncome;

Этот пример приведет к удалению процедуры CalcIncome.

MySQL Триггеры

Создать триггер MySQL	
BEFORE INSERT	AFTER INSERT
BEFORE UPDATE	AFTER UPDATE
BEFORE DELETE	AFTER DELETE
Удалить триггер MySQL	
DROP TRIGGER	

MySQL Курсоры

Declare a Cursor		
OPEN	FETCH	CLOSE
Обработка для условия Cursor NOT FOUND		

MySQL Условные операторы

- IF-THEN-ELSE
- CASE

MySQL Циклы

LOOP	ITERATE
WHILE	LEAVE
REPEAT	RETURN

MySQL условия

AND	логический оператор и
OR	логический оператор или

AND и OR	логический операторы и & или
LIKE	сопоставляет данные с шаблоном
IN	определяет, соответствует ли значение или список значений выражению в указанном наборе
NOT	логический оператор не
IS NULL	проверка значения NULL
IS NOT NULL	проверка на значения NOT NULL
BETWEEN	используется для получения значений в пределах диапазона
EXISTS	используется в сочетании с подзапросом (subquery)
Операторы сравнения MySQL	Такие как =, <=>, <>, !=, >, >=, <, <=.

ТЕСТ 1 и 2 ПРОЦЕДУРЫ И ФУНКЦИИ MY SQL

https://quizizz.com/join?gc=44696843
https://quizizz.com/join?gc=08652043