Курсоры в MySQL.

Применение и синтаксис. Примеры.

Курсоры прекрасно поддерживаются в хранимых процедурах, функциях и триггерах.  Синтаксис такой же, как и во внедренном SQL. Курсоры пока только для чтения, однонаправленные (т.е по набору можно ходить только вперед без возможности вернуться) и невосприимчивы. Невосприимчивость означает, что сервер может создавать копию результатирующей таблицы, а может и не создавать, формируя ее на лету .

Курсоры должны быть объявлены до их использования. Переменные с условиями объявляются прежде курсоров. Обработчики объявляются строго после объявления курсоров.

Например:

**CREATE PROCEDURE** curdemo()  
**BEGIN**  
 **DECLARE** done **INT DEFAULT** 0;  
 **DECLARE** a **CHAR**(16);  
 **DECLARE** b,c **INT**;  
 **DECLARE** cur1 **CURSOR FOR SELECT** id,data **FROM** test.t1;  
 **DECLARE** cur2 **CURSOR FOR SELECT** i **FROM** test.t2;  
 **DECLARE CONTINUE HANDLER FOR SQLSTATE** '02000' **SET** done = 1;  
  
 **OPEN** cur1;  
 **OPEN** cur2;  
  
**REPEAT**  
 **FETCH** cur1 **INTO** a, b;  
 **FETCH** cur2 **INTO** c;  
 **IF NOT** done **THEN**  
 **IF** b < c **THEN**  
 **INSERT INTO** test.t3 **VALUES** (a,b);  
 **ELSE**  
 **INSERT INTO** test.t3 **VALUES** (a,c);  
 **END IF**;  
 **END IF**;  
**UNTIL** done **END REPEAT**;  
  
 **CLOSE** cur1;  
 **CLOSE** cur2;  
**END**

Объявление курсоров

**DECLARE** cursor\_name **CURSOR FOR** select\_statement

Это выражение объявляет курсор c именем cursor\_name. select\_statement указывает на конструкцию типа SELECT ... FROM ... Можно объявить много курсоров в подпрограмме, но каждый курсор в данном блоке должен иметь уникальное имя. Выражение SELECT не должно содержать указание INTO.

Открывание курсоров

**OPEN** cursor\_name

Выражение открывает ранее объявленный курсор

Выборка из курсора в переменную

**FETCH** cursor\_name **INTO** var\_name [, var\_name] ...

Это выражение выбирает следующую строку (если строка существует), используя указанный открытый курсор, и продвигает указатель курсора. Если более строк не доступно, происходит изменение значения переменной SQLSTATE в 02000. Для отлова этого события вы должны установить обработчик: HANDLER FOR SQLSTATE '02000'

Закрытие курсора

**CLOSE** cursor\_name

Закрывает курсор cursor\_name. Если явно не указано, то курсор закрывается автоматически при закрытии соответствующего блока подпрограммы.

Как использовать курсоры

Применение курсора в процедурах осуществляется путем последовательного выполнения следующих шагов:

* При помощи оператора **DECLARE** объявляется курсор для отдельного оператора **SELECT** или для отдельной процедуры.
* Оператором **OPEN** производится открытие курсора.
* Используя оператор **FETCH**, осуществляется установление указателя на требуемую запись курсора. При этом значения полей текущей записи присваиваются переменным, указываемым в операторе **FETCH**. Обычно это конструкция помещается в итеративный элемент (проще говоря цикл), который прерывается по некоторому условию.
* В процессе перемещения указателя текущей записи курсора при выходе указателя за пределы курсора устанавливается значение **SQLSTATE = 02000**.
* После того как курсор становится ненужным, он закрывается оператором **CLOSE**.

Примеры курсоров

Приведу еще один пример курсора. Курсор предназначен для выборки данных (идентификаторов записей) в строку с разделителем ввиде запятой по переданным параметрам. Курсор находится внутри процедуры get\_pedplan(). В нее передается три параметра: lip-номер лаборатории, ti-номер пары и dt - дата проведения занятия.

01: CREATE DEFINER = 'for\_spammers'@'zoonman.ru' FUNCTION `get\_pedplan`(lip INTEGER(11), ti INTEGER(11), dt DATE)  
02:  RETURNS char(64) CHARSET latin1  
03:  DETERMINISTIC  
04:  CONTAINS SQL  
05:  SQL SECURITY INVOKER  
06:  COMMENT 'Функция возвращает список id из таблицы raspisanie'  
07: BEGIN  
08:   
09:  DECLARE done INT DEFAULT 0;  
10:  DECLARE a INT;  
11:  DECLARE retv CHAR(64);  
12:  DECLARE flg INT;  
13:   
14:  DECLARE cur1 CURSOR FOR SELECT id FROM raspisanie WHERE timeintv=ti AND rdate=dt AND labip=lip ;  
15:  DECLARE CONTINUE HANDLER FOR SQLSTATE '02000' SET done = 1;  
16:  OPEN cur1;  
17:  SET retv:='';  
18:  SET flg:=0;  
19:  REPEAT  
21:  FETCH cur1 INTO a;  
22:  IF NOT done THEN   
23:  IF flg!=0 THEN  
24:  SET retv:= CONCAT(retv,',' , a);  
25:  ELSE  
26:  SET retv:=a;  
27:  SET flg:=1;   
28:  END IF;  
29:  END IF;   
30:  UNTIL done END REPEAT;  
31:  CLOSE cur1;  
32:  RETURN retv;  
33: END;

Курсор определен на строке 14. Открыт на 16 строчке. С 19-й начат проход по выборке полученной курсором. На каждом шаге цикла происходит считывание записи (21-я). Затем, если не достигнут конец выборки (22-я), выполняется проверка флага на первую запись (23-я). Если запись первая, то присваиваем retv текущее значение выборки (26-я) и устанавливаем флаг (27-я), иначе объединяем значение retv с текущим значением выборки (строка 24). После прохода по курсору закрываем его (31-я) и возвращаем значение (32-я строка).