

Лабораторная работа №3. Хранимые процедуры и триггеры.

ППСУБД
2курс. 5 группа.
Кушнеров А.В. 2017-2018 г.

Цель работы:

Изучить механизмы построения и применения хранимых процедур и триггеров языка SQL на примере СУБД MySQL. Освоить технологию курсора и её применение.

Минимальные теоретические сведения

Хранимые процедуры и функции в языке SQL предназначены для объединения часто повторяемых запросов и сохранения их на сервере. Триггер – разновидность хранимой процедуры, которая вызывается сервером автоматически.

Общий вид:

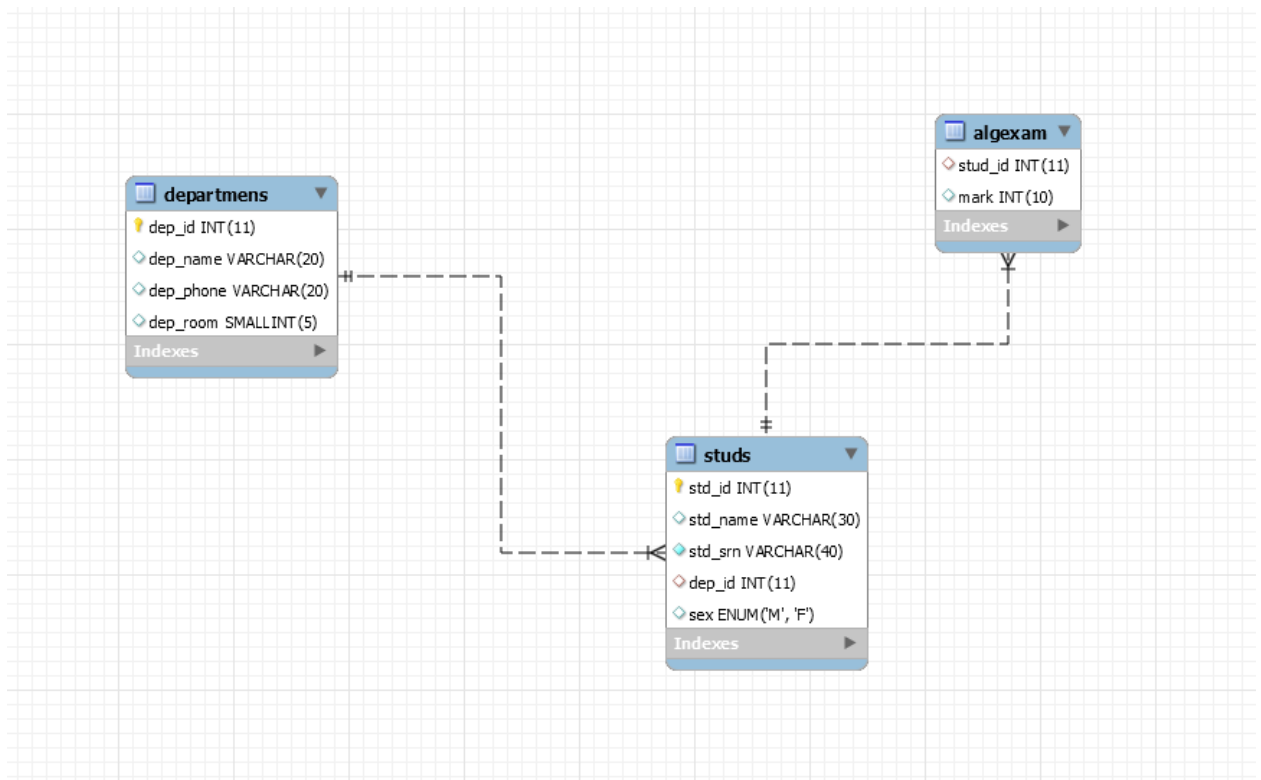
```
CREATE PROCEDURE sp_name ([parameter[, ...]])  
    [characteristic ...] routine_body
```

```
CREATE FUNCTION sp_name ([parameter[, ...]])  
    RETURNS type  
    [characteristic ...] routine_body
```

```
CREATE TRIGGER trigger_name trigger_time trigger_event  
    ON tbl_name FOR EACH ROW trigger_stmt
```

Примеры:

Задана учебная БД students. В ней 3 таблицы (кафедры, студенты, экзамен по алгебре).
Схема представлена ниже.



1. Хранимая процедура для получения всех записей из таблицы studs.

```

Delimiter //
create procedure givestuds()
begin
select * from studs;
end//
Delimiter ;
  
```

2. Хранимая процедура для получения всех студентов, получивших определённую оценку на экзамене. Эта оценка поступает на вход процедуры.

```

Delimiter //
create procedure givestdbymark(IN value1 int)
begin
select s.std_name, s.std_srn from studs as s
inner join algexam as ae
on s.std_id=ae.stud_id
where ae.mark=value1;
end//
Delimiter ;

```

3. Хранимая процедура с записью результата в новую таблицу с помощью курсора.

```

Delimiter //
create procedure givestdbymark1(IN value1 int)
begin
DECLARE name1 varchar(30);
DECLARE surname1 varchar(30);
declare is_end INT default 0;

declare curs cursor for
select s.std_name, s.std_srn from studs as s
inner join algexam as ae
on s.std_id=ae.stud_id
where ae.mark=value1;

declare continue handler for not found set is_end=1;
open curs;
reading: LOOP
fetch curs INTO name1,surname1;
if is_end then leave reading;
end if;
insert into stdres values(name1,surname1);
end loop reading;
close curs;
end//
Delimiter ;

```

4. Триггер для подсчёта количества добавленных строк.

```
Delimiter //
create trigger ins_count AFTER INSERT ON departmens
for each row
begin
SET @col=@col+1;
end//
Delimiter ;
```

ОСОБОЕ ВНИМАНИЕ! Сложность при построении хранимых процедур возникает с символом « ; ». Синтаксис требует заканчивать им строки процедуры, но в тоже время этот символ заканчивает запрос. Поэтому мы временно меняем символ конца запроса с помощью атрибута DELIMITER.

Задания для самостоятельной работы

1. Продолжите реализацию учебной БД, представленной на схеме выше. Добавьте все кафедры ММФ, добавьте таблицы для всех экзаменов, которые вы сдавали в последнюю сессию. Добавьте в БД информацию о стипендии студентов.

2. Реализуйте следующие хранимые процедуры или функции.

- 1. Хранимая процедура для получения всех студенток, которые закончили сессию без пересдач.*
- 2. Хранимая процедура для получения 3-ёх лучших студентов какой-либо кафедры по среднему баллу.*
- 3. Хранимая процедура для изменения размера стипендии каждого студента в зависимости от среднего балла.*
- 4. Выведите средний балл студентов по каждой из кафедр. С помощью курсора, запишите эти данные в отдельную таблицу.*
- 5. Хранимая процедура для получения всех студентов, получивших пересдачу, по каждому из предметов.*
- 6. Хранимая процедура для отчисления студента и удаления всех упоминаний о нём.*

3. Реализуйте следующие триггеры.

- 1. Триггер для автоматического подсчёта количества поступающих студентов.*
- 2. Триггер для проверки данных об оценках на экзамене на корректность.*
- 3. Триггер, реализующий потерю стипендии при пересдаче, а также её восстановлении, в случае исправления оценки.*