Министерство образования и науки РФ

Пермский национальный исследовательский политехнический университет

Электротехнический факультет

Кафедра Информационные технологии и автоматизированные системы

Базы данных

Лабораторная работа № 8

Тема: «Реализация хранимых процедур в СУБД MySQL»

Выполнил: студент группы

РИС-22-2б

Прядеин И.А

Проверил: доцент кафедры ИТАС

Петренко А.А.

г. Пермь – 2023

Оглавление

Цель работы 3

Задачи работы 4

Этапы выполнения 5

Заключение 9

Список используемой литературы 10

# Цель работы

Реализовать хранимые процедуры с помощью запросов в СУБД MySQL.

**Задачи работы**

1. Изучить понятие и синтаксис хранимых процедур;
2. Разработать запросы реализующие хранимые процедуры;
3. Проверить работу хранимых процедур.

**Этапы выполнения**

1. Хранимые процедуры (stored procedures) - это предварительно скомпилированные и сохранённые наборы команд SQL, предназначенные для выполнения определённых задач. Процедуры хранятся в базе данных и могут вызываться другими операторами SQL, приложениями или пользователями с соответствующими разрешениями.

Синтаксис создания процедуры:

“CREATE PROCEDURE [IF NOT EXISTS] sp\_name ([proc\_parameter[,...]]) routine\_body;”, где sp\_name - название хранимой процедуры, proc\_parameter - входные параметры процедуры, routine\_body - набор SQL команд.

Вызов процедуры происходит путём использования команды “CALL sp\_name([parameter[,...]])”, где sp\_name - имя процедуры., parameter - входные параметры процедуры.

1. Создание базы данных “lab\_8” и таблицы “employees” реализовано с помощью запросов “CREATE DATABASE” и “CREATE TABLE”.

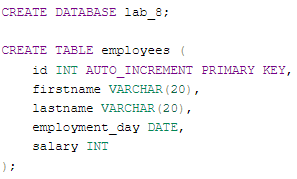


Рис. 1 - Создание исходной таблицы

1. Вставка записей в таблицу “employees” выполнено с помощью запроса “INSERT INTO”.

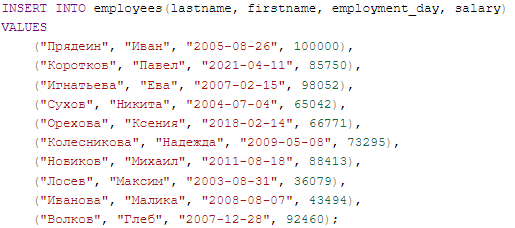


Рис. 2 - Вставка записей в таблицу “employees”

1. Реализация процедуры “get\_employees”, выполняющей запрос “SELECT” для всех записей исходной таблицы “employees”.



Рис. 3 - Создание процедуры “get\_employees”

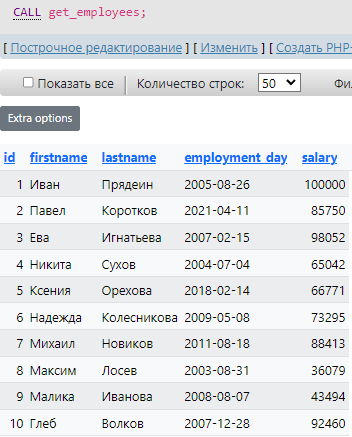


Рис. 4 - Результат вызова процедуры “get\_employees”

1. Также предусмотрено создание процедуры с входными параметрами. Например, процедура “calculate\_avg\_salary” принимает в качестве параметра целое значение. Результатом вызова процедуры будет таблица из одной записи, содержащей среднее округлённое значение зарплат, которые превышают входной параметр.



Рис. 5 - Создание процедуры “calculate\_avg\_salary”

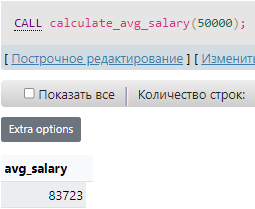


Рис. 6 - Результат вызова процедуры “calculate\_avg\_salary”

1. В процедурах возможно использование условных операторов. Реализация процедуры “calculate\_salary\_expense” выполнена с помощью условного оператора “IF ELSE END IF”. В начале процедуры объявляется переменная “total\_salary”, обозначающая сумму всех зарплат работников. Если переменная “total\_salary” больше входного значения, в качестве сообщения выводиться таблица с одной записью. Для работы процедуры необходимо переопределить разделитель, так как тело процедуры содержит команды, заканчивающиеся точкой с запятой, SQL интерпретирует завершение запроса “CREATE PROCEDURE” до первого символа “;”.

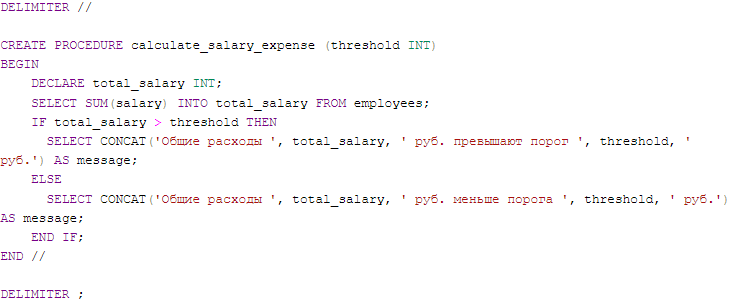


Рис. 7 - Создание процедуры “calculate\_salary\_expense”

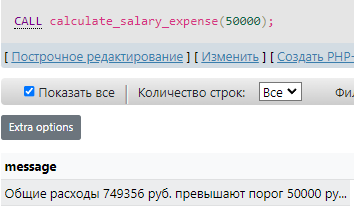


Рис. 8 - Результат вызова процедуры с входным параметром “50000”

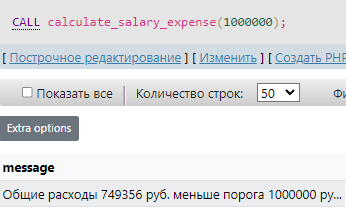


Рис. 9 - Результат вызова процедуры с входным параметром “1000000”

# Заключение

В результате выполнения лабораторной работы были получены навыки разработки запросов на языке SQL в СУБД MySQL.

# Список используемой литературы

Дюбуа П. MySQL. Сборник рецептов. – Пер. с англ. – СПб: СимволПлюс, 2006. – 1056 с., ил. ISBN 5932860707