Теперь можно приступить к работе с MYSQL. В настоящей работе создадим небольшую базу данных и несколько хранимых процедур и триггеров. Просмотрим наличие существующих баз данных, как показано на рис.1.1. (команда show databases;) Не забывайте после команды ставить точку с запятой. _ 🗆 x MySQL Command Line Client Enter password: ****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.0.67-community-nt MySQL Community Edition (GPL) Type 'help;' or '\h' for help. Type '\c' to clear the buffer. mysql> show databases; Database information_schema | mysql | test 3 rows in set (0.01 sec) Рис.1.1 Создаем базу командой >create database base1; Подключаемся к базе командой >use base1; Создаем таблицу sklad: >create table sklad(Tovar varchar(25),price int, kol int,postavka Date,Primary >KEY(tovar)); _ 🗆 x MySQL Command Line Client PRIMARY KEY(tovar))' at line 2
mysql> create table sklad(tovar varchar(25),
-> price int, kol int, postavka DATE,
-> PRIMARY KEY(tovar));
Query OK, O rows affected (0.58 sec) mysql> _ Рис.2 Теперь занесем в таблицу данные, набранные в текстовом файле. Пример команды загрузки: mysql> LOAD DATA LOCAL INFILE '/path/pet.txt' INTO TABLE pet; Записи в файле располагаются одна под другой и имеют слндующий примерный вид: Gwen bird \N 1997-12-09 \N Значения полей отделяются символом табуляции (ТАВ). \N соответствует пустому значению. Наш файл такой milk 200 130 2010-01-18 bread 1400 1000 2010-01-17 1800 1200 2010-01-16 curd 3000 500 2010-01-10 Запишем его в таблицу: _ 🗆 x MySQL Command Line Client mysql> load data local infile 'c:/data/my.txt' into table sklad; Query OK, 4 rows affected (0.05 sec) Records: 4 Deleted: 0 Skipped: 0 Warnings: 0

Рис.4 Приступим к созданию хранимых процедур (эта возможность имеется только для сервера MySQL версии 5 и выше). Наша первая процедура такова: _ 🗆 x MySQL Command Line Client mysql> delimiter //
mysql> create procedure findcena(IN tov varchar(25), OUT cen INT)
-> begin
-> select price into cen from sklad where tovar=tov; Query OK, 0 rows affected (0.14 sec) mysql> delimiter ;
mysql> call findcena('bread',@x);
Query OK, 0 rows affected (0.05 sec) 1400 row in set (0.00 sec) Рис.5 Обратим внимание на то, что написанию процедуры предшествует изменение концевого символа-разделителя(delimiter): >delimiter //

_ 🗆 x

_ 🗆 x

Рис.3

mysql> load data local infile 'c:/data/my.txt' into table sklad; Query OK, 4 rows affected (0.05 sec) Records: 4 Deleted: 0 Skipped: 0 Warnings: 0

Теперь можно просмотреть содержимое файла:

>select * from sklad;

MySQL Command Line Client

mysql> select * from sklad;

4 rows in set (0.00 sec)

mysq1> _

tovar | price | kol | postavka |

bread | 1400 | 1000 | 2010-01-17 |
butter | 1800 | 1200 | 2010-01-16 |
curd | 3000 | 500 | 2010-01-10 |
milk | 200 | 130 | 2010-01-18 |

Затем идет собственно текст процедуры:

Затем восстанавливаем разделитель:

>call findcena('bread',@x);

MySQL Command Line Client

>delimiter;

>select @x;

>select price into cen from sklad where Tovar=tov;

>create procedure findcena(IN tov varchar(25), OUT cen INT)

К сожалению, хранимую процедуру нельзя ввести из файла (как данные).

После этого можно вызывать процедуру и просмотреть результат:

mysql> select param_list from mysql.proc where specific_name='findcena';

Просмотреть, какие есть процедуры в вашей базе можно так:

Лабораторная работа 5

Теоретические сведения.

MySQL Command Line Client

MySQL Command Line Client

Enter password: _

mysql> 🔔

ЦЕЛЬ. Изучить основы работы с сервером MYSQL.

mysql -и **имя_пользователя** -p.

Работа в СУБД MySQL. Создание хранимых процедур и функций.

В качестве имени пользователя введите *root*. После запуска программа запросит пароль:

Enter password: ***
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 1 to server version: 5.0.20-nt

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

Введите пароль, который вы установили при установке MySQL. Если введен верный пароль, система выдаст следующее сообщение:

Монитор mysql запускаем из меню "Пуск" → "Программы" → "MySQL" → "MySQL Server 5.0" → "MySQL Command Line Client", или через командную строку путем ввода:

| param_list IN tov varchar(25), OUT cen INT row in set (0.00 sec) mysql> select specific_name from mysql.proc where name='findcena'; | specific_name | row in set (0.00 sec) mysql> select name from mysql.proc where db='base1'; name findcena row in set (0.00 sec) Ри.6 >select name from mysql.proc where db='basel'; Содержимое процедуры можно просмотреть таким образом: >show create procedure findcena; Или так: >select body from mysql.proc where name='findcena'; _ 🗆 x MySQL Command Line Client | specific_name | findcena row in set (0.00 sec) mysql> select name from mysql.proc where db='base1'; | findcena | 1 row in set (0.00 sec)

mysql> select body from mysql.proc where name='findcena';

Рис.7

Следующая процедура использует курсор для выборки товара с минимальной ценой:

| begin | select price into cen from sklad where tovar=tov; | end |

Продемонстрируем сейчас работу с курсором.

mysql> create procedure proccount(OUT p int)

-> open cur; -> fetch cur into p;

Query OK, O rows affected (0.00 sec)

mysql> delimiter ; mysql> call proccount(@x); Query OK, O rows affected (0.00 sec)

-> close cur;

-> end;

mysql> SELECT 0x;

Пример с циклом:

row in set (0.00 sec)

MySQL Command Line Client

row in set (0.00 sec)

Общий формат таков:

MySQL Command Line Client

Query OK, 0 rows affected (0.02 sec)

mysql> SElect @x=tax('bread');

NULL :

Создать хранимые процедуры для

1) Вставки записи и ее обновления.

2) Подсчета налога со всех товаров.

2) Вычисления средней цены товара по таблице

! @x=tax('bread') !

Создать функции для:

ЗАДАНИЕ.

>return val;

>create function myfun(x INT) returns int

mysql> source d:\work\dat.txt Query OK, 0 rows affected (0.00 sec)

Query OK, O rows affected (0.00 sec)

Рис.11

Рассмотрим теперь, как создавать функции MySQL.

Таким образом можно облегчить работу по набору текста процедур.

Рис.12

1) Вычисления налога в зависимости от цены и количества товара

-> begin
-> Declare cur CURSOR for select count(tovar) from sklad;

Рис.9

row in set (0.00 sec)

_ 🗆 x MySQL Command Line Client mysql> delimiter //
mysql> create procedure thecheepest(OUT tov varchar(25>)
-> begin
-> DECLARE cur CURSOR for Select tovar from sklad
-> where price=(Select min(price) from sklad);
-> open cur;
-> fetch cur into tov;
-> close cur; -> close cur; -> end; Query OK, 0 rows affected (0.00 sec) mysql> delimiter ; mysql> call thecheepest(@x); Query OK, O rows affected (0.05 sec) mysql> Select @x; | milk | 1 row in set (0.00 sec) Рис.8 Курсор необходимо объявить как предложение SELECT: DECLARE cur CURSOR for Select Tovar from sklad ... ; Для работы курсор следует открыть Open cur; А потом и закрыть Close cur; Другой пример подсчета числа записей в курсоре: MySQL Command Line Client

MySQL Command Line Client mysql> create procedure cycl2(OUT p INT) -> begin
-> declare n INT;
-> declare cur CURSOR for select price from sklad; -> open cur; -> set n=1; -> aaa: LOOP -> fetch cur into p; \rightarrow if n>2 then -> leave aaa; -> end if; -> set n=n+1; -> end loop aaa; -> end; Query OK, O rows affected (0.00 sec) mysql> call cycl2(@y); -> // Query OK, O rows affected (0.00 sec) mysql> delimiter ; mysql> select @y; Рис.10 Цикл организуется между aaa: LOOP End LOOP aaa; Внутри цикла выполняется преждевременный выход по команде leave aaa; если срабатывает условие: Ясно, что набирать текст процедур крайне неудобно с консоли. Поэтому можно набрать текст в текстовом файле, например, такой: delimiter // create procedure proc3(OUT p int) declare cur CURSOR for select count(tovar) from sklad; open cur; fetch cur into p; end; delimiter; call proc3(@z); select @z; Назовем этот файл dat.txt. Теперь можно из консоли выдать следующую команду: source d:\work\dat.txt

Значение функции возвращаем оператором return. Рассмотрим пример. Enter password: *
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 29
Server version: 5.0.67-community-nt MySQL Community Edition (GPL) Type 'help;' or '\h' for help. Type '\c' to clear the buffer. mysql> use mydb;
Database changed
mysql> delimiter //
mysql> create function tax(x varchar(25)) returns float
-> begin
-> declare z float;
-> select price*1.3 into z from sklad where tovar=x;
-> return z;
-> end: 3) Получения точного названия товара по первым буквам. 4) Работы с курсором (подсчитать число записей с одинаковым названием товара).