**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»**

(СПбГУТ)

ФАКУЛЬТЕТ ИНФОРМАЦИОНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ **(ИТПИ)**

Кафедра программной инженерии и вычислительной техники **(ПИиВТ)**

Дисциплина: «Базы данных»

Лабораторная работа №3.

**Тема: «Case, встроенные функции MySQL»**

Выполнил:

  Студент группы ИКПИ-23

Харлова А.А.

Подпись \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Принял:

Дятлов Д.А.

Подпись \_\_\_\_\_\_\_\_\_\_\_\_\_\_

2024 г.

**Цель работы:**

Научиться работать с транзакциями, использовать точки сохранения, поработали с подтверждением изменений.

**Постановка задачи:**

1. Создать или открыть ранее сохранённую базу данных «university» в программе-дизайнере MySQL Workbench.

2. В базе данных «university» создать таблицу «tasks» с полями:

**• id тип int, ключ (PK), счетчик (AI);**

**• taskname тип varchar(45), не нулевое (NN);**

**• taskmonth тип varchar(45);**

**• taskday тип varchar(45);**

**• u\_id тип int.**

3. Создать хранимую процедуру с именем createTask (раздел Routines в MySQL Workbech).

4. Заполнить тело процедуры согласно образцу:

**DELIMITER //**

**CREATE PROCEDURE `university`.`createTask` (IN tname VARCHAR(45), IN tdate DATETIME,**

**OUT muchdays VARCHAR(45))**

**BEGIN**

**DECLARE tmonth VARCHAR(45);**

**SELECT CONCAT('Task month is: ',**

**(CASE MONTH(tdate)**

**WHEN 1 THEN 'Jan'**

**WHEN 2 THEN 'Feb'**

**WHEN 3 THEN 'Mar'**

**WHEN 4 THEN 'Apr'**

**WHEN 5 THEN 'May'**

**WHEN 6 THEN 'Jun'**

**WHEN 7 THEN 'Jul'**

**WHEN 8 THEN 'Aug'**

**WHEN 9 THEN 'Sep'**

**WHEN 10 THEN 'Oct'**

**WHEN 11 THEN 'Nov'**

**WHEN 12 THEN 'Dec'**

**ELSE 'None'**

**END**

**)) INTO tmonth;**

**INSERT INTO tasks (taskname, taskday, taskmonth) VALUES (tname, DAY(tdate), tmonth);**

**SELECT CONCAT('Remains days: ', DATEDIFF(tdate, CURDATE())) INTO muchdays;**

**END//**

5. Сохранить созданную в программе-дизайнере схему базы данных на локальный компьютер.

6. Запустить генерацию базы данных на сервере MySQL. (Пункт меню: Database->Forward Engineer. В опциях необходимо поставить галки напротив пунктов DROP Objects Before Each

**CREATE Object и Generate INSERT Statements for Tables).**

7. Подключиться к базе данных MySQL через терминал (mysql –u root –p).

8. Активизировать базу данных «university» (use).

9. В командной строке выполнить следующую последовательность команд:

**• CALL createTask(‘Database optimization’, ‘2009-11-01’, @days);**

**• SELECT CONCAT(‘Optimization ‘, @days);** (результаты вывода записать в черновик)

**• CALL createTask(‘Database replication ’, ‘2009-09-14’, @days);**

**• SELECT CONCAT(‘Replication ‘, @days);** (результаты вывода записать в черновик)

**• CALL createTask(‘<Ввести свою задачу>’, ‘<Ввести свою дату>’, @days);**

**• SELECT \* FROM tasks;** (результаты вывода записать в черновик)

**Ход работы:**

1. Создать или открыть ранее сохранённую базу данных «university».

2. В базе данных «university» создать таблицу «tasks» с полями:

**• id тип int, ключ (PK), счетчик (AI);**

**• taskname тип varchar(45), не нулевое (NN);**

**• taskmonth тип varchar(45);**

**• taskday тип varchar(45);**

**• u\_id тип int.**

3. Создать хранимую процедуру с именем createTask.

4. Заполнить тело процедуры согласно образцу:

**DELIMITER //**

**CREATE PROCEDURE `university`.`createTask` (IN tname VARCHAR(45), IN tdate DATETIME,**

**OUT muchdays VARCHAR(45))**

**BEGIN**

**. . .**

**END**

**)) INTO tmonth;**

**INSERT INTO tasks (taskname, taskday, taskmonth) VALUES (tname, DAY(tdate), tmonth);**

**SELECT CONCAT('Remains days: ', DATEDIFF(tdate, CURDATE())) INTO muchdays;**

**END//**

5. Сохранить созданную в программе-дизайнере схему базы данных на локальный компьютер.

6. Запустить генерацию базы данных на сервере MySQL.

**CREATE Object и Generate INSERT Statements for Tables)**

7. Подключиться к базе данных MySQL через терминал.

8. Активизировать базу данных «university».

9. В командной строке выполнить следующую последовательность команд:

**• CALL createTask(‘Database optimization’, ‘2009-11-01’, @days);**

**• SELECT CONCAT(‘Optimization ‘, @days);**

**• CALL createTask(‘Database replication ’, ‘2009-09-14’, @days);**

**• SELECT CONCAT(‘Replication ‘, @days);**

**• CALL createTask(‘<Ввести свою задачу>’, ‘<Ввести свою дату>’, @days);**

**• SELECT \* FROM tasks; (результаты вывода записать в черновик)**

**Вывод:**

Мы научились работать с транзакциями, использовать точки сохранения, поработали с подтверждением изменений.