

# Лабораторная работа №1

## Связь MYSQL и Lazarus

**Цель работы:** научиться обрабатывать информацию из базы данных MySQL с помощью прикладных программ на примере LAZARUS.

**Ключевые слова:** MYSQL, LAZARUS.

### Задание:

В первую очередь, создаём тестовую базу данных, к которой необходимо будет подключиться. Пусть база называется primer и содержит одну таблицу зарп с полями fio (varchar(20)) и зарп (int). Заполняем ее данными, причем записи вносим как на русском, так и на латинском языке (Рис. 19):

fio	zarп
Иванов	1000
Petrov	2000

Рис. 19 Данные таблицы

Запускаем Lazarus.

На форме размещаем компоненты:

- Button с вкладки Standart (Рис. 20),



Рис. 20 Вкладка Standart

- Data Source с вкладки Data Access (Рис. 21),



Рис. 21 Вкладка Data Access

- DBGrid с вкладки Data Controls (Рис. 22),



Рис. 22 Вкладка Data Controls

- SqlConnection, SimpleDataSet с вкладки dbExpress (Рис. 23).



Рис. 23 Вкладка dbExpress

Форма должна выглядеть следующим образом (Рис. 24):

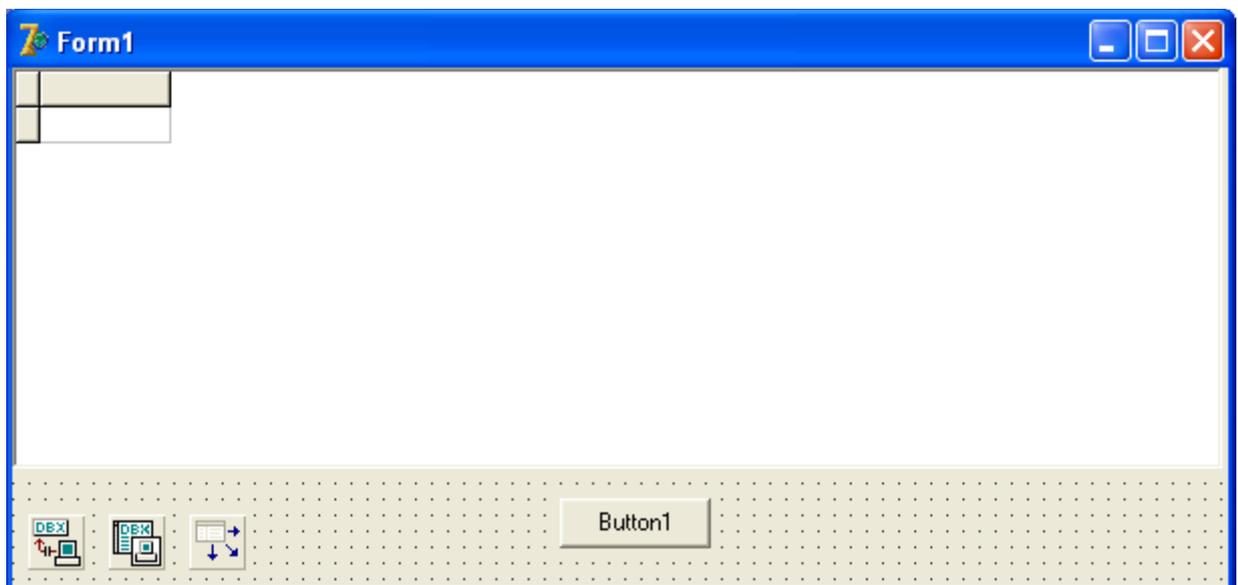


Рис. 24 Внешний вид формы

Сохраняем проект в папку командой File->Save Project As. В эту же папку помещаем dll-библиотеку libmysql.dll (скачать ее можно по ссылке: <http://ru.topdll.com/download/libmysql.dll?q=c71850684e8572b1e6d0a4402372495b>).

Произведем настройку компонентов.

Выделяем компонент SqlConnection. В контекстном меню выбираем пункт «Edit Connection Properties» (Рис 25):

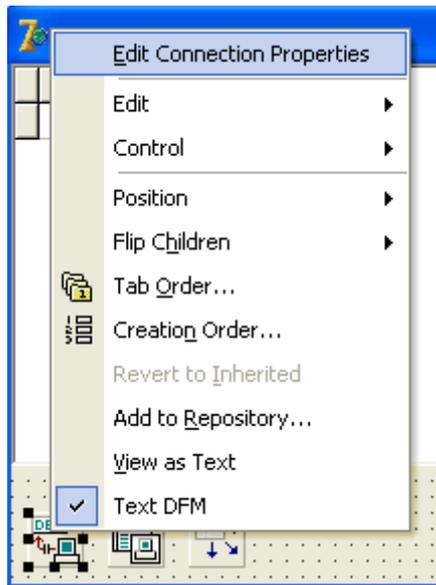


Рис. 25 Контекстное меню

Выбираем в поле «Connection Name» тип MySqlConnection. В настройках подключения (Connection Settings) устанавливаем следующие свойства: HostName – адрес сервера (localhost для локального подключения), Database – имя базы данных, User\_Name – имя пользователя при подключении к базе данных, Password – пароль. Нажимаем ОК (Рис. 26).

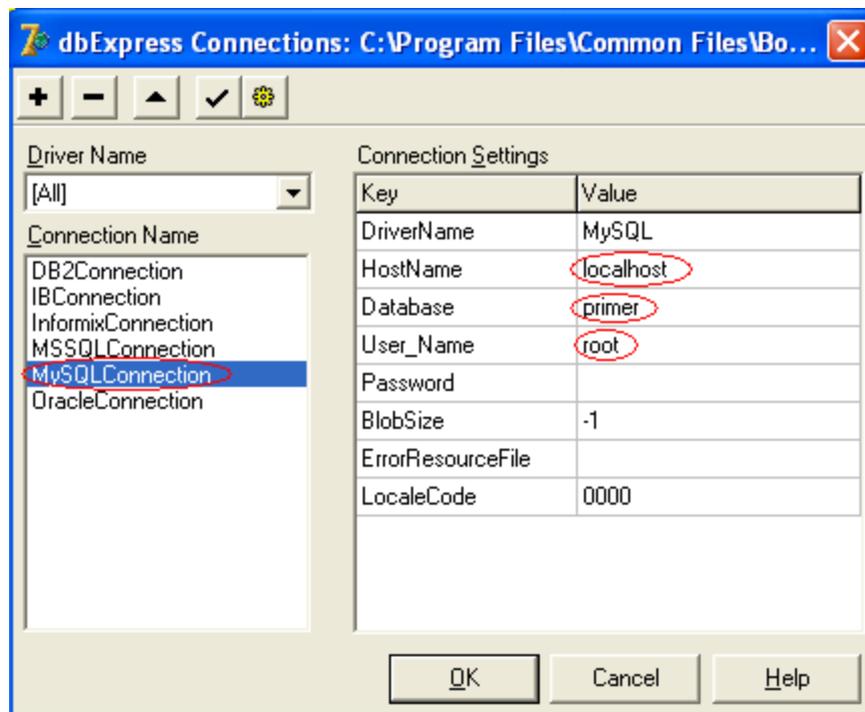


Рис. 26 Настройка соединения с базой данных.

Проверяем, что выделен по-прежнему компонент SqlConnection, устанавливаем в инспекторе объектов (Object Inspector) свойство

«LoginPrompt» в false (это позволит отключить запрос пароля при каждом подключении к базе), а также «Connected» в true (Рис. 27).

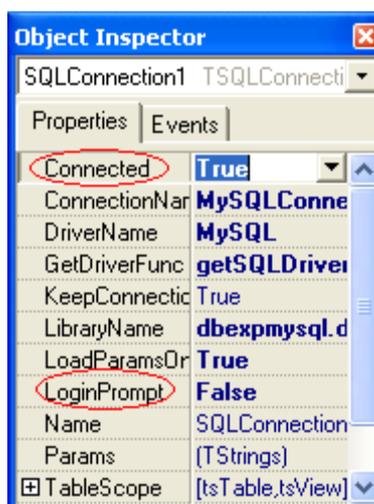


Рис. 27 Окно свойств

Теперь настроим компонент SimpleDataSet. В Object Inspector для свойства «Connection» выберите из выпадающего списка значение SQLConnection1 (рис. 28).



Рис. 28 Свойство Connection

Далее раскрываем свойство «DataSet» и в строке «CommandText» пишем запрос к базе данных. Например, «Select \* from zarp» (Рис. 29).



Рис. 29 Свойство CommandText

Запрос можно прописать вручную или с помощью редактора запросов «CommandText Editor» (для его вызова нужно нажать на ) (Рис. 30):

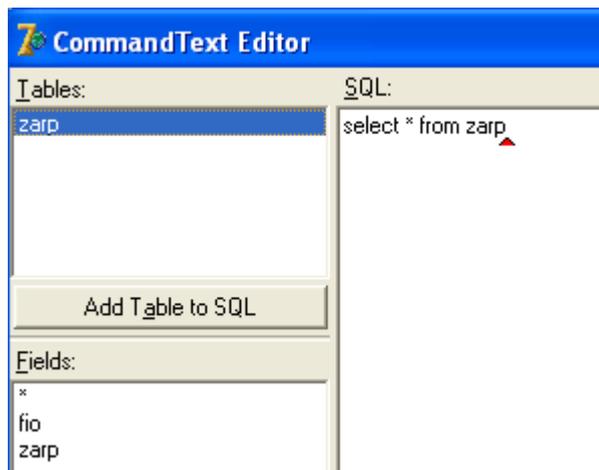


Рис. 30 Создание запросов

Настроим теперь компонент DataSource. Устанавливаем свойство «DataSet» в SimpleDataSet1 (Рис. 31):



Рис. 31 Свойство DataSet

Последний компонент, который нужно настроить – DBGrid. Устанавливаем свойство «DataSource» в DataSource1 (Рис. 32).

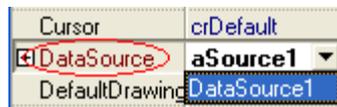


Рис. 32 Свойство DataSource

Программируем кнопку так, чтобы при ее нажатии активировался написанный нами запрос: создаем для кнопки обработчик OnClick прописываем в процедуре код:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  DataSource1.DataSet.Active:=true;
end;
```

Запускаем проект на исполнение (F9). Нажимаем на кнопку. Таблица должна заполниться данными из базы. При этом данные, записанные кириллицей, могут отображаться знаками вопроса «?» (Рис. 33).

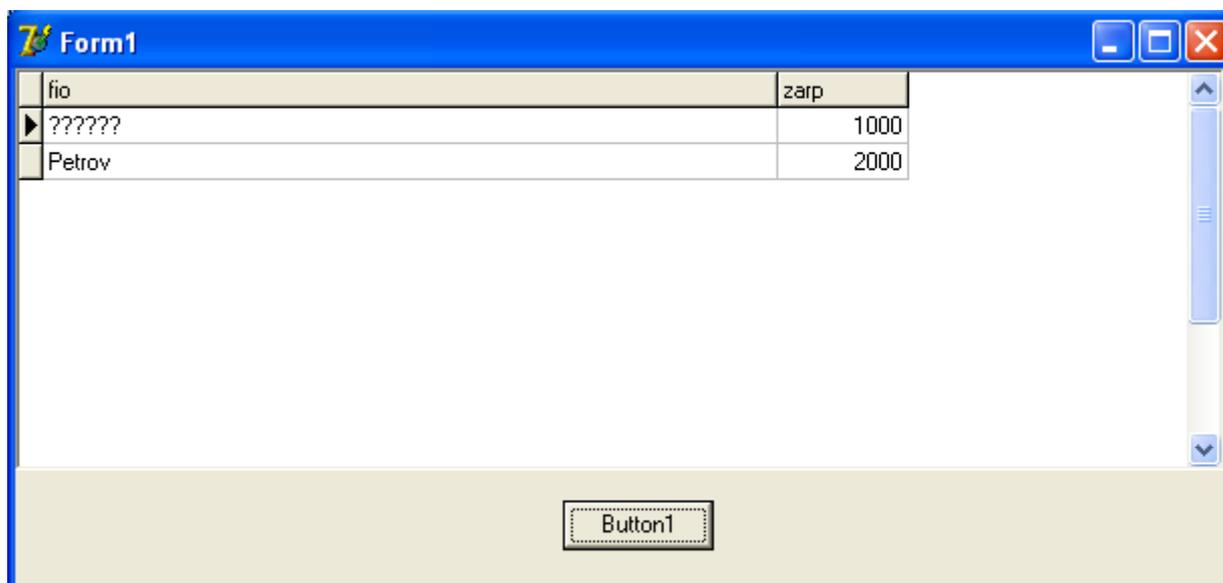


Рис. 33 Вывод таблицы на форму

Для исправления этого необходимо до выполнения запроса установить параметры кодировки. Итак, создаем обработчик события формы OnCreate и прописываем код:

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  SQLConnection1.ExecuteDirect(' SET names cp1251');
  SQLConnection1.ExecuteDirect(' SET CHARACTER SET cp1251');
end;
```

Запускаем, проверяем работоспособность проекта.

Далее программу можно усложнить, добавив возможность поиска информации, ее сортировки и т.д.

## Лабораторная работа №2

### Создание web-приложений

---

**Цель работы:** научиться создавать web-приложения с помощью базы данных MySQL и прикладной программы на примере LAZARUS.

**Ключевые слова:** MYSQL, LAZARUS.

1. **Создайте веб-приложение**, реализующее калькулятор (минимум четыре действия).
2. **Создайте веб-приложение**, анализирующее введенное в поле ввода число по следующим параметрам: четность, знак (положительное\отрицательное) и количество разрядов. Если введено не число, то сообщить об этом.
3. **Создайте веб-приложение**, позволяющее проверить информацию, введенную в регистрационную форму, состоящую из следующих полей: Имя (максимум 10 символов) и адрес электронной почты (наличие обязательного символа @ и минимум двух точек). Если хотя бы одно поле содержит ошибку, то вывести соответствующее сообщение, если же вся информация введена верно, то поздравить с успешной регистрацией.
4. **Создайте веб-приложение**, реализующее тест на основе выпадающих списков.
5. **Создайте веб-приложение**, позволяющее определить название браузера пользователя. Выведите результат в удобочитаемом виде, например: «Вы используете Google».
6. **Создайте веб-приложение**, позволяющее определить операционную систему, в которой запущен браузер пользователя. Выведите результат в удобочитаемом виде, например: «Вы используете ОС ...»
7. **Создайте веб-приложение** позволяющее выстроить возрастающую последовательность из чисел, переданных через поля ввода формы.
8. **Создайте веб-приложение**, позволяющее выстроить убывающую последовательность из чисел переданных через поля ввода формы.
9. **Создайте веб-приложение**, позволяющее вычислить среднее арифметическое, а так же найти максимальный элемент последовательности чисел, переданных через поля ввода формы.
10. **Создайте веб-приложение**, позволяющее вычислить сумму элементов, а так же найти минимальный элемент последовательности чисел переданных через поля ввода формы.