

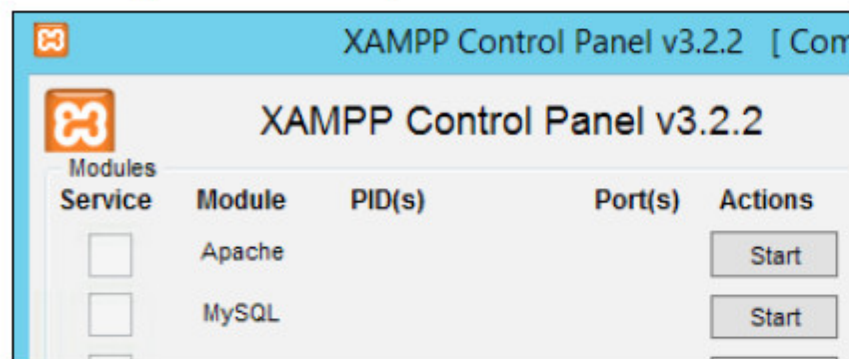
Курс по программированию на языке СИ/C++

### **Практическая работа № 6**

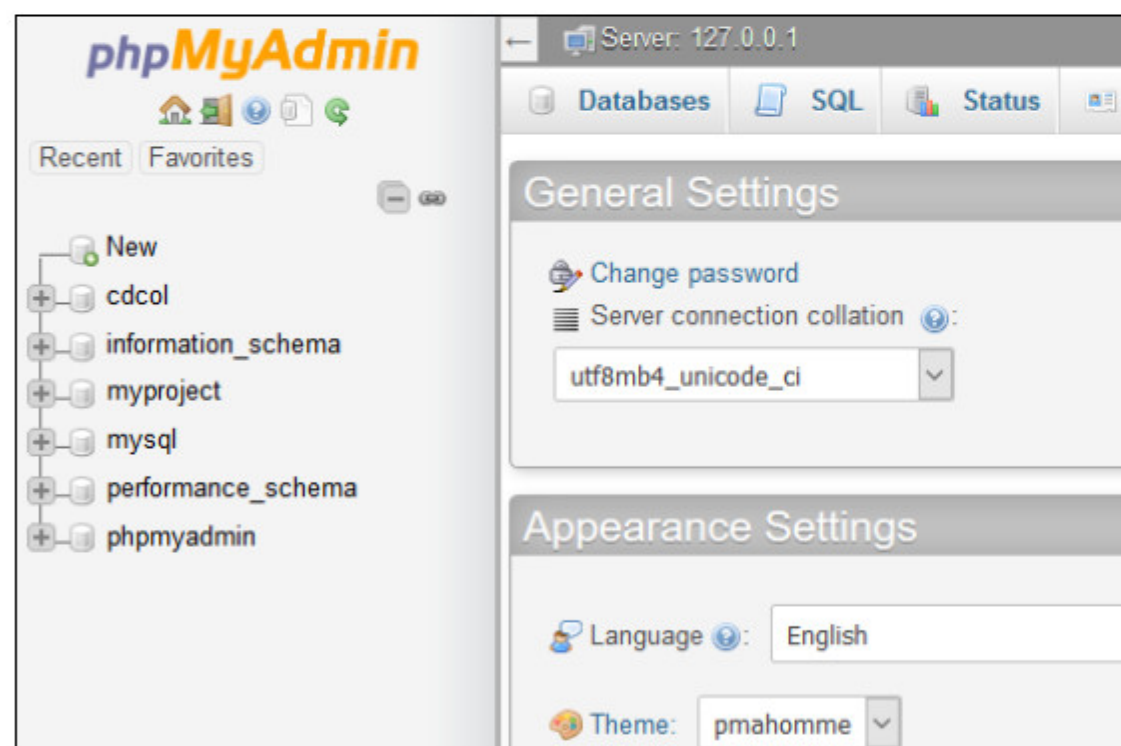
Тема: Работа с базой данных MySQL в среде Qt .

## Ход работы:

1. Запустите приложение XAMPP. Нажмите кнопки Start напротив Apache и MySQL.



После запуска веб-сервера Apache и сервера базы данных MySQL откройте веб-браузер и перейдите на страницу <http://localhost/phpmyadmin>



2. Создайте новую базу данных и в ней таблицу employee, состоящую из 5 столбцов
3. Заполните таблицу данными (не менее трех строк)

Column	Type	Function	Null	Value
id	int(11)			1
name	varchar(50)			John Doe
age	int(11)			42
gender	int(11)			0
married	tinyint(1)			1

#### 4. Перейдите в папку установки Qt и найдите папку sqldrivers

Например: C:\Qt\5.5\mingw492\_32\plugins\sqldrivers

Скопируйте всю папку sqldrivers в каталог сборки вашего проекта. Можно удалить все лишнее, кроме qsqlmysql.dll и qsqlmysqld.dll

Поместите эти библиотеки в соответствующие подкаталоги:

../builds/debug/sqldrivers/qsqlmysqld.dll

../builds/release/sqldrivers/qsqlmysql.dll

Вам также может понадобиться библиотека Libmysql.dll, скопируйте её из каталога, в котором находится MySQL сервер

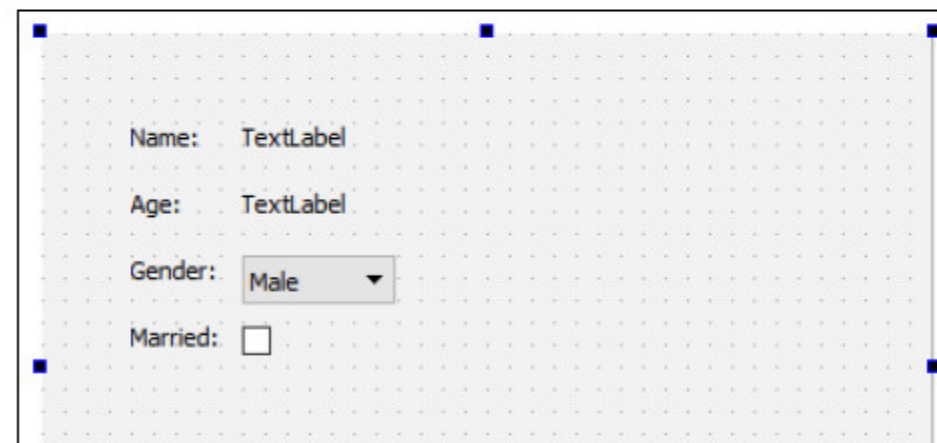
Она должна находиться вместе с .exe файлом программы.

#### 5. Откройте Qt Creator и создайте новый проект приложения Qt Widgets Application.

Откройте файл проекта (.pro) и добавьте в свой проект модуль SQL:

**QT += core gui sql**

Затем откройте mainwindow.ui и постройте интерфейс как на рисунке:





6. Откройте mainwindow.h и добавьте следующие заголовки ниже

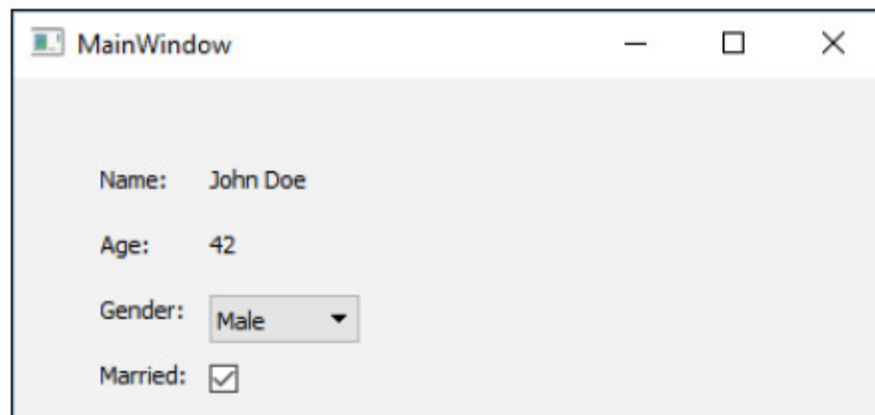
```
#include <QMainWindow>
#include <QtSql>
#include <QSqlDatabase>
#include <QSqlQuery>
#include <QDebug>
```

Затем откройте файл mainwindow.cpp и наберите следующий код:

```
MainWindow::MainWindow(QWidget *parent) :
    QMainWindow(parent), ui(new Ui::MainWindow)
{
    ui->setupUi(this);
    QSqlDatabase db = QSqlDatabase::addDatabase("QMYSQL");
    db.setHostName("127.0.0.1");
    db.setUserName("yourusername");
    db.setPassword("yourpassword");
    db.setDatabaseName("databasename");
    if (db.open())
    {
        QSqlQuery query;
        if (query.exec("SELECT name, age, gender, married FROM
employee"))
        {
            while (query.next())
            {
                qDebug() << query.value(0) << query.value(1) <<
                query.value(2) << query.value(3);
                ui->name->setText(query.value(0).toString());
                ui->age->setText(query.value(1).toString());
                ui->gender->setCurrentIndex(query.value(2).toInt());
                ui->married->setChecked(query.value(3).toBool());
            }
        }
        else
        {
            qDebug() << query.lastError().text();
        }

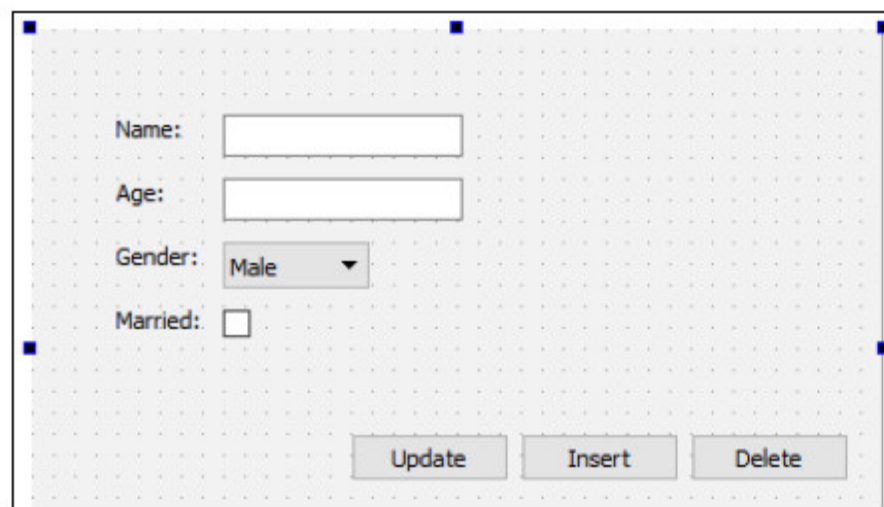
        db.close();
    }
    else
    {
        qDebug() << "Failed to connect to database.";
    }
}
```

7. Скомпилируйте и запустите свой проект  
Если все было сделано верно, данные из базы будут показаны.



## Задание:

Откройте `mainwindow.ui` и замените виджеты имени и возраста виджетами редактирования. Затем добавьте три кнопки и назовите их `Update`, `Insert` и `Delete`:



Напишите функции Добавления, Обновления и Удаления данных из базы по нажатию на соответствующие кнопки.