**Лабораторная работа №7**

**«ИССЛЕДОВАНИЕ СПОСОБОВ РЕАЛИЗАЦИИ СЕТЕВЫХ ЗАПРОСОВ В QT-ПРИЛОЖЕНИЯХ»**

**Цель работы**

Исследование способов работы Qt-приложения с HTTP GET/POST запросами. Приобретение навыков разработки простейших сетевых приложений в Qt.

**Постановка задачи**

1. Изучить основные возможности модуля QtNetwork для реализацииь HTTP-запросов (выполняется в ходе самостоятельной подготовки к лабораторной работе).

2. Создать Qt GUI приложение.

3. Реализовать следующую логику приложения :

Все элементы, кроме кнопок выбора типа запроса изначально скрыты. После выбора метода отобразить скрытые элементы. При этом кнопки выбора метода – скрыть. QLabel с текстом “Enter Data:” и соответствующее поле ввода должно отображаться только для POST запроса.

По нажатию кнопки Submit реализовать отправку соответствующего запроса и вывод результата в поле Response. Для POST запроса использовать данные из поля Enter Data.

4. Исследовать работу программы при отправке GET и POST запросов, проанализировать формат получаемых HTTP-ответов.

**Ход работы**

1. Было реализовано приложения для отправки GET- и POST- запросов с отображением соответствующих ответов. На рисунках 1–2 продемонстрировано выполнение программы.

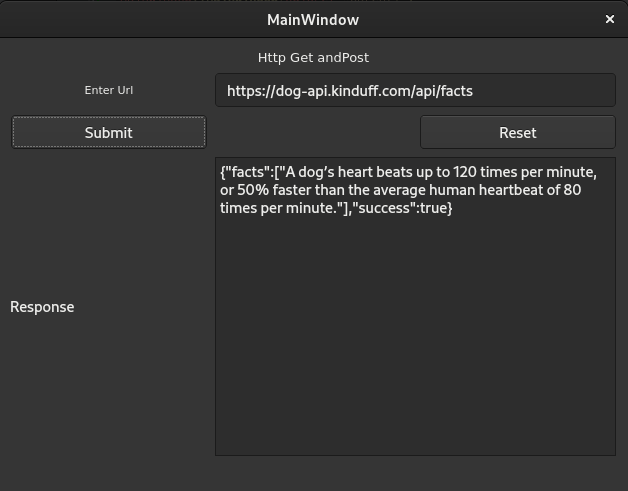


Рисунок 1 – Выполнение GET-запроса

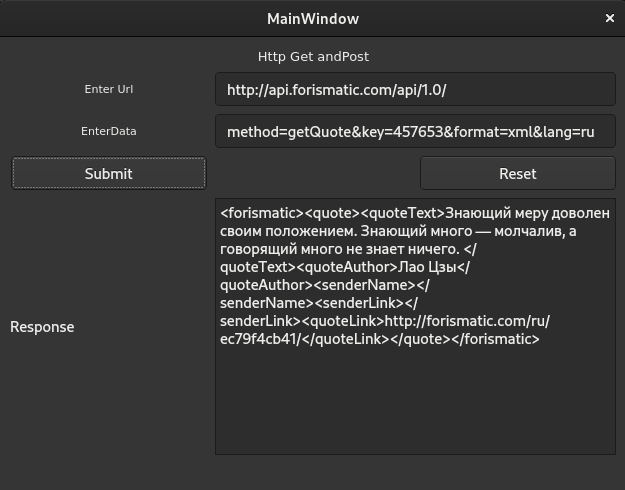


Рисунок 2 – Выполнение POST-запроса

**Выводы**

В ходе выполнения лабораторной работы были исследованы способы работы Qt-приложения с HTTP GET/POST запросами. Приобретены навыки разработки простейших сетевых приложений в Qt.

Было выяснено, что использование стандартных классов QT позволяет легко и удобно реализовывать сетевые запросы в приложениях.

Также было установлено, что для обработки ответов от сервера можно использовать различные методы, такие как сигналы и слоты, а также классы QNetworkReply.

В результате проведенной работы можно сделать вывод, что QT предоставляет широкие возможности для реализации сетевых запросов в приложениях.

**ПРИЛОЖЕНИЕ А**

**«ИСХОДНЫЙ КОД ПРОГРАММЫ»**

Листинг А1 – Файл 7th.pro

QT += core gui network widgets

greaterThan(QT\_MAJOR\_VERSION, 4): QT += widgets

CONFIG += c++17

# You can make your code fail to compile if it uses deprecated APIs.

# In order to do so, uncomment the following line.

#DEFINES += QT\_DISABLE\_DEPRECATED\_BEFORE=0x060000 # disables all the APIs deprecated before Qt 6.0.0

SOURCES += \

main.cpp \

mainwindow.cpp

HEADERS += \

mainwindow.h

FORMS += \

mainwindow.ui

# Default rules for deployment.

qnx: target.path = /tmp/$${TARGET}/bin

else: unix:!android: target.path = /opt/$${TARGET}/bin

!isEmpty(target.path): INSTALLS += target

Листинг А2 – Файл main.cpp

#include "mainwindow.h"

#include <QApplication>

int main(int argc, char \*argv[])

{

QApplication a(argc, argv);

MainWindow w;

w.show();

return a.exec();

}

Листинг А3 – Файл mainwindow.h

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

#include <QMainWindow>

#include <QNetworkAccessManager>

#include <QNetworkReply>

#include <QMessageBox>

#include <QByteArray>

QT\_BEGIN\_NAMESPACE

namespace Ui { class MainWindow; }

QT\_END\_NAMESPACE

class MainWindow : public QMainWindow

{

Q\_OBJECT

public:

MainWindow(QWidget \*parent = 0);

~MainWindow();

protected:

void changeEvent(QEvent \*e);

private slots:

void finished(QNetworkReply \*reply);

void DoHttpGet();

void activateGetWidgets();

void HideWidgets();

void clearWidgets();

void activatePostWidgets();

private:

Ui::MainWindow \*ui;

QNetworkAccessManager \*nam;

};

#endif // MAINWINDOW\_H

Листинг А4 – Файл mainwindow.cpp

#include "mainwindow.h"

#include "ui\_mainwindow.h"

// https://dog-api.kinduff.com/api/facts

// http://api.forismatic.com/api/1.0/ method=getQuote&key=457653&format=xml&lang=ru

MainWindow::MainWindow(QWidget \*parent) :

QMainWindow(parent),

ui(new Ui::MainWindow)

{

ui->setupUi(this);

nam = new QNetworkAccessManager(this);

HideWidgets();

connect(ui->getButton,SIGNAL(clicked()),this,SLOT(activateGetWidgets()));

connect(ui->submitButton,SIGNAL(clicked()),this,SLOT(DoHttpGet()));

connect(ui->resetButton,SIGNAL(clicked()),this,SLOT(clearWidgets()));

connect(nam,SIGNAL(finished(QNetworkReply\*)),this,SLOT(finished(QNetworkReply\*)));

connect(ui->postButton,SIGNAL(clicked()),this,SLOT(activatePostWidgets()));

}

MainWindow::~MainWindow()

{

delete ui;

}

void MainWindow::changeEvent(QEvent \*e)

{

QMainWindow::changeEvent(e);

switch (e->type()) {

case QEvent::LanguageChange:

ui->retranslateUi(this);

break;

default:

break;

}

}

void MainWindow::activateGetWidgets()

{

ui->urlLabel->setHidden(false);

ui->urlLine->setHidden(false);

ui->submitButton->setHidden(false);

ui->textBrowser->setHidden(false);

ui->responseTitleLabel->setHidden(false);

ui->getButton->setHidden(true);

ui->postButton->setHidden(true);

}

void MainWindow::activatePostWidgets()

{

ui->dataLabel->setHidden(false);

ui->dataLine->setHidden(false);

activateGetWidgets();

}

void MainWindow::finished(QNetworkReply \*reply)

{

if(reply->error() == QNetworkReply::NoError)

{

ui->textBrowser->setText(reply->readAll());

}

else

{

QMessageBox::critical(this, tr("Network communication error"), reply->errorString());

}

}

void MainWindow::DoHttpGet()

{

ui->resetButton->setHidden(false);

QString url = ui->urlLine->text();

QString data = ui->dataLine->text();

QByteArray postData;

postData.append(data.toLatin1());

if(postData.isEmpty() == true)

{

nam->get(QNetworkRequest(QUrl(url)));

}

else

{

nam->post(QNetworkRequest(QUrl(url)), postData);

}

}

void MainWindow::HideWidgets()

{

ui->urlLabel->setHidden(true);

ui->urlLine->setHidden(true);

ui->dataLabel->setHidden(true);

ui->dataLine->setHidden(true);

ui->submitButton->setHidden(true);

ui->responseTitleLabel->setHidden(true);

ui->textBrowser->setHidden(true);

ui->resetButton->setHidden(true);

}

void MainWindow::clearWidgets()

{

ui->urlLabel->setHidden(true);

ui->urlLine->setHidden(true);

ui->dataLabel->setHidden(true);

ui->dataLine->setHidden(true);

ui->submitButton->setHidden(true);

ui->responseTitleLabel->setHidden(true);

ui->textBrowser->setHidden(true);

ui->resetButton->setHidden(true);

ui->urlLine->clear();

ui->textBrowser->clear();

ui->dataLine->clear();

ui->getButton->setHidden(false);

ui->postButton->setHidden(false);

}

Листинг А5 – Файл mainwindow.ui

<?xml version="1.0" encoding="UTF-8"?>

<ui version="4.0">

<class>MainWindow</class>

<widget class="QMainWindow" name="MainWindow">

<property name="geometry">

<rect>

<x>0</x>

<y>0</y>

<width>360</width>

<height>530</height>

</rect>

</property>

<property name="windowTitle">

<string>MainWindow</string>

</property>

<widget class="QWidget" name="centralWidget">

<layout class="QVBoxLayout" name="verticalLayout">

<item>

<widget class="QLabel" name="label">

<property name="text">

<string>&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML 4.0//EN&quot;

&quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;&gt;

&lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;1&quot; /&gt;&lt;style type=&quot;text/css&quot;&gt; p, li { white-space: pre-wrap; }

&lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'MS Shell Dlg 2'; font-size:8.25pt; font-weight:400; font-style:normal;&quot;&gt;

&lt;p align=&quot;center&quot; style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; textindent:0px;&quot;&gt;&lt;span style=&quot; font-size:10pt; fontweight:600;&quot;&gt;Http Get and

Post&lt;/span&gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>

</property>

</widget>

</item>

<item>

<layout class="QGridLayout" name="gridLayout">

<item row="0" column="0">

<widget class="QPushButton" name="getButton">

<property name="text">

<string>Get</string>

</property>

</widget>

</item>

<item row="0" column="2">

<widget class="QPushButton" name="postButton">

<property name="text">

<string>Post</string>

</property>

</widget>

</item>

<item row="1" column="0">

<widget class="QLabel" name="urlLabel">

<property name="text">

<string>&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML 4.0//EN&quot;

&quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;&gt;

&lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;1&quot; /&gt;&lt;style type=&quot;text/css&quot;&gt;

p, li { white-space: pre-wrap; }

&lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'MS Shell Dlg 2';

font-size:8.25pt; font-weight:400; font-style:normal;&quot;&gt;

&lt;p align=&quot;center&quot; style=&quot; margin-top:0px; margin-bottom:0px;

margin-left:0px; margin-right:0px; -qt-block-indent:0; text-

indent:0px;&quot;&gt;Enter Url&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>

</property>

</widget>

</item>

<item row="1" column="1" colspan="2">

<widget class="QLineEdit" name="urlLine"/>

</item>

<item row="2" column="0">

<widget class="QLabel" name="dataLabel">

<property name="text">

<string>&lt;!DOCTYPE HTML PUBLIC &quot;-//W3C//DTD HTML 4.0//EN&quot;

&quot;http://www.w3.org/TR/REC-html40/strict.dtd&quot;&gt;

&lt;html&gt;&lt;head&gt;&lt;meta name=&quot;qrichtext&quot; content=&quot;1&quot; /&gt;&lt;style type=&quot;text/css&quot;&gt; p, li { white-space: pre-wrap; }

&lt;/style&gt;&lt;/head&gt;&lt;body style=&quot; font-family:'MS Shell Dlg 2'; font-size:8.25pt; font-weight:400; font-style:normal;&quot;&gt;

&lt;p align=&quot;center&quot; style=&quot; margin-top:0px; margin-bottom:0px; margin-left:0px; margin-right:0px; -qt-block-indent:0; text-

indent:0px;&quot;&gt;&lt;span style=&quot; font-size:8pt;&quot;&gt;Enter

Data&lt;/span&gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;</string>

</property>

</widget>

</item>

<item row="2" column="1" colspan="2">

<widget class="QLineEdit" name="dataLine"/>

</item>

<item row="3" column="0">

<widget class="QPushButton" name="submitButton">

<property name="text">

<string>Submit</string>

</property>

</widget>

</item>

<item row="3" column="2">

<widget class="QPushButton" name="resetButton">

<property name="text">

<string>Reset</string>

</property>

</widget>

</item>

<item row="4" column="0">

<widget class="QLabel" name="responseTitleLabel">

<property name="text">

<string>Response</string>

</property>

</widget>

</item>

<item row="4" column="1" colspan="2">

<widget class="QTextBrowser" name="textBrowser"/>

</item>

</layout>

</item>

</layout>

</widget>

<widget class="QStatusBar" name="statusBar"/>

</widget>

<layoutdefault spacing="6" margin="11"/>

<resources/>

<connections/>

</ui>