Федеральное государственное бюджетное образовательное учреждение «Сибирский государственный университет телекоммуникаций и информатики»

кафедра прикладной математики и кибернетики

Лабораторная работа №6 по дисциплине «Визуальное программирование и человеко-машинное взаимодействие»

По теме «Работа с формами»

**Выполнил(а):** студентка 3 курса

Колосова Яна Владимировна

**Проверил(а)**: Преподаватель кафедры ПМиК

Мерзлякова Екатерина Юрьевна

Новосибирск 2020г.

**Цель работы:**

Научиться обмену данными между формами. Использовать стили.

**Задание:**

Разработать приложение для записи в файл информации из формы.

Реализовать связь между двумя формами и создать собственный стиль.

**Листинг программы:**

**form\_kolosova.h:**

#ifndef FORM\_KOLOSOVA\_H

#define FORM\_KOLOSOVA\_H

#include <QWidget>

#include <QFileDialog>

#include <QTextEdit>

#include <QTextStream>

#include <QAbstractButton>

#include <QFileDialog>

#include <QMessageBox>

namespace Ui {

class form\_kolosova;

}

class form\_kolosova : public QWidget

{

Q\_OBJECT

public:

explicit form\_kolosova(QWidget \*parent = 0);

~***form\_kolosova***();

private:

Ui::form\_kolosova \*ui;

QString m\_strFileName;

public slots:

void **recieveData**(QString str);

void **slotSave**();

private slots:

void **on\_buttonBox\_clicked**(QAbstractButton \*button);

signals:

void **changeWindowTitle**(const QString&);

};

#endif // FORM\_KOLOSOVA\_H

**form\_kolosova.cpp:**

#include "form\_kolosova.h"

#include "ui\_form\_kolosova.h"

form\_kolosova::**form\_kolosova**(QWidget \*parent) :

QWidget(parent),

ui(new Ui::form\_kolosova)

{

ui->setupUi(this);

}

form\_kolosova::~***form\_kolosova***()

{

delete ui;

}

void form\_kolosova::**recieveData**(QString str)

{

QStringList lst = str.split("\*");

ui->textEdit->setText(lst.at(1) + "\n" + lst.at(0));

if (lst.size() > 1)

{

QImage image1(lst.at(0));

ui->label->setPixmap(QPixmap::fromImage(image1));

}

}

void form\_kolosova::**slotSave**()

{

if (m\_strFileName.isEmpty())

{

QString str = QFileDialog::getSaveFileName(0, m\_strFileName);

if (!str.isEmpty())

{

m\_strFileName = str;

slotSave();

}

return;

}

QFile file(m\_strFileName);

if(file.*open*(QIODevice::WriteOnly))

{

QTextStream(&file) << ui->textEdit->toPlainText();

file.*close*();

}

QMessageBox::information(0, "Информация", "Файл сохранен!");

}

void form\_kolosova::**on\_buttonBox\_clicked**(QAbstractButton \*button)

{

if (button->text() == "Reset")

{

ui->textEdit->clear();

ui->label->clear();

}

if (button->text() == "Save All")

{

QString str = QFileDialog::getSaveFileName(0, m\_strFileName);

if (!str.isEmpty())

{

m\_strFileName = str;

slotSave();

}

}

if (button->text() == "Open")

{

QString str = QFileDialog::getOpenFileName();

if (str.isEmpty())

{

return;

}

QFile file(str);

if(file.*open*(QIODevice::ReadOnly))

{

QTextStream stream(&file);

QString line = stream.readAll();

file.*close*();

ui->textEdit->setPlainText(line);

}

QStringList inf = ui->textEdit->toPlainText().split("\n");

QImage image2(inf.at(5));

ui->label->setPixmap(QPixmap::fromImage(image2));

}

}

**mainwindow.h:**

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

#include <QWidget>

#include "form\_kolosova.h"

#include <QFileDialog>

#include <QMessageBox>

namespace Ui {

class MainWindow;

}

class MainWindow : public QMainWindow

{

Q\_OBJECT

public:

explicit MainWindow(QWidget \*parent = 0);

~***MainWindow***();

private:

form\_kolosova \*myform;

Ui::MainWindow \*ui;

signals:

void **sendData**(QString str);

private slots:

void **on\_pushButton\_clicked**();

void **on\_pushButton\_2\_clicked**();

};

#endif // MAINWINDOW\_H

**mainwindow.cpp:**

#include "mainwindow.h"

#include "ui\_mainwindow.h"

MainWindow::**MainWindow**(QWidget \*parent) :

QMainWindow(parent),

ui(new Ui::MainWindow)

{

ui->setupUi(this);

myform = new form\_kolosova();

connect(ui->pushButton, SIGNAL(clicked()), myform, SLOT(show()));

connect(ui->pushButton, SIGNAL(clicked()), this, SLOT(onButtonSend()));

connect(this, SIGNAL(sendData(QString)), myform, SLOT(recieveData(QString)));

}

MainWindow::~***MainWindow***()

{

delete ui;

}

void MainWindow::**on\_pushButton\_2\_clicked**()

{

QString filename = QFileDialog::getOpenFileName(0, "Выберите изображение", QDir::currentPath(), "\*.png \*.jpg \*.gif \*.jpeg");

ui->lineEdit\_4->setText(filename);

QImage image1(filename);

ui->label\_4->setPixmap(QPixmap::fromImage(image1));

}

void MainWindow::**on\_pushButton\_clicked**()

{

QString st = ui->lineEdit\_4->text() + "\*" + ui->lineEdit->text()

+ "\n" + ui->lineEdit\_2->text() + "\n" + ui->dateEdit->text()

+ "\n" + ui->lineEdit\_3->text();

if (ui->radioButton->isChecked() == true)

st += "\nПол: мужской";

else

st += "\nПол: женский";

if (ui->lineEdit->text() == "" || ui->lineEdit\_2->text() == ""

|| ui->lineEdit\_3->text() == "" || ui->lineEdit\_4->text() == "")

{

QMessageBox::information(0, "Ошибка", "Некоторые поля не заполнены");

}

else

{

emit sendData(st);

}

}

**main.cpp:**

#include "mainwindow.h"

#include "form\_kolosova.h"

#include <QApplication>

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

{

QApplication a(argc, argv);

qApp->setStyleSheet("QWidget { background-color: white; }"

"QMainWindow { background-color: white; }"

"QDialogButtonBox { background-color: fuchsia; }"

"QPushButton { background-color: fuchsia; }"

"QLineEdit { border-color: fuchsia; border-width: 1px; border-style: solid;}"

"QTextEdit { border-color: fuchsia; border-width: 1px; border-style: solid;}"

"QDateEdit { border-color: fuchsia; border-width: 1px; border-style: solid;}");

MainWindow w;

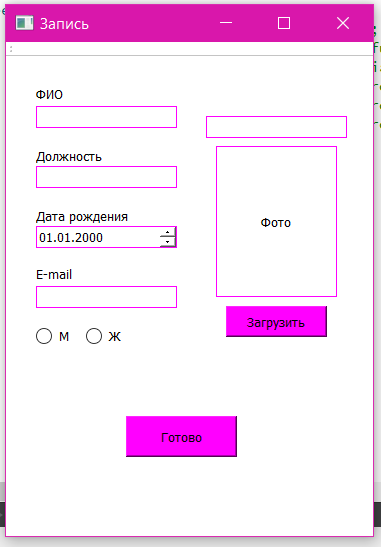
w.show();

return a.exec();

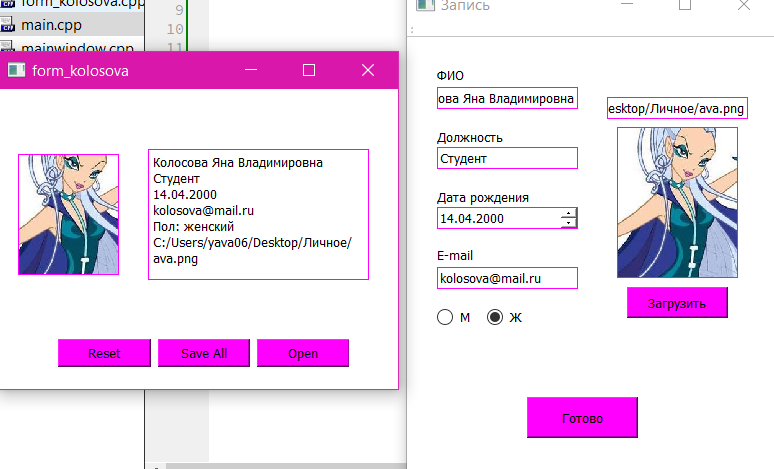
}

**Результат работы программы:**

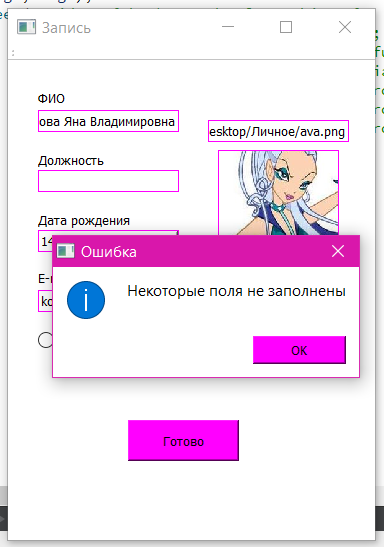
1. Начало работы



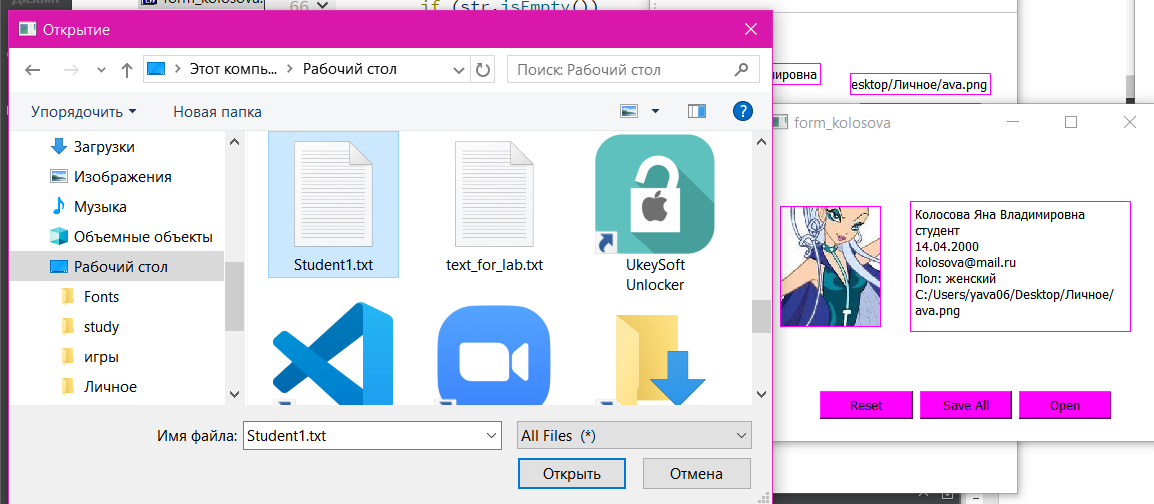
1. Передача данных в другую форму

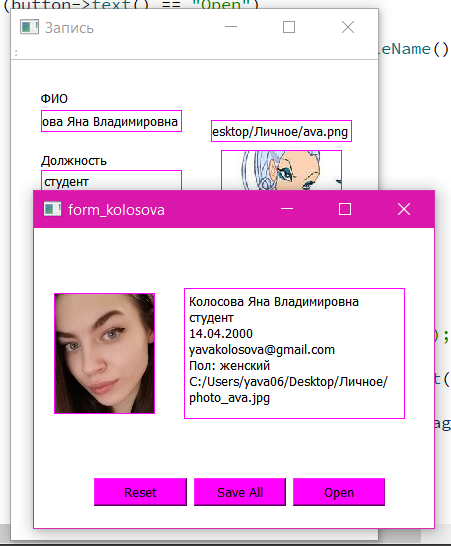


1. Запрет при пустых полях

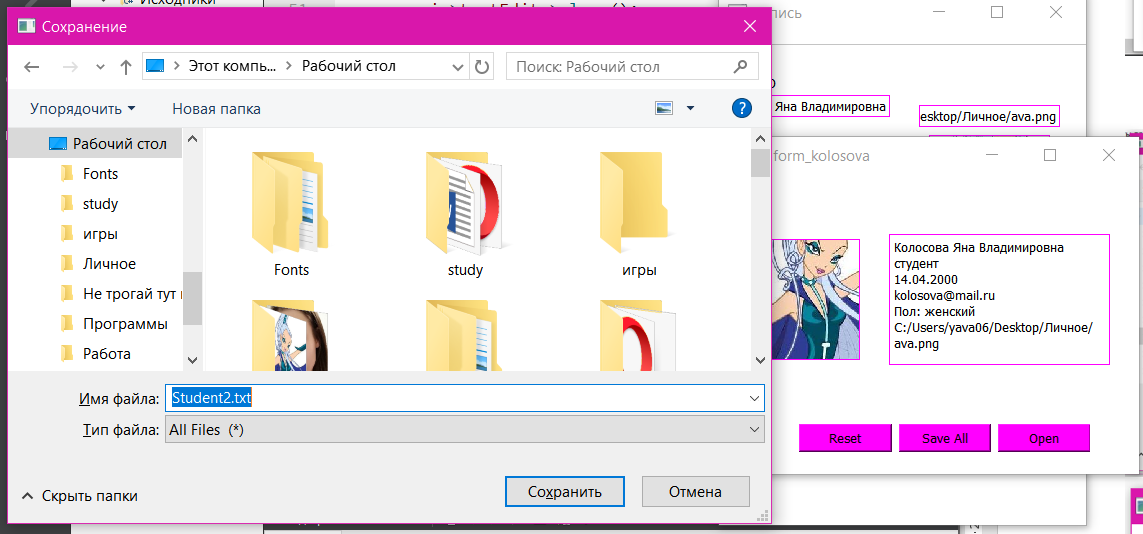
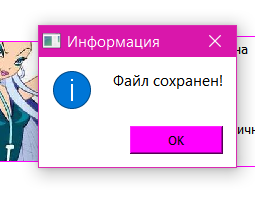


1. Открытие файла:





1. Сохранение файла

1. Кнопка «Reset»

