

Федеральное агентство связи
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Сибирский государственный университет телекоммуникаций и
информатики»

Кафедра прикладной математики и кибернетики

Отчет по лабораторной работе
по дисциплине «Визуальное программирование и человеко-машинное
взаимодействие»

Лабораторная работа №3

Выполнил: студент 3 курса
группы ИП-811
Мироненко К. А

Проверил: доцент кафедры
ПМиК
Мерзлякова Е. Ю.

Оглавление

1. Постановка задачи.....	3
2. Примеры работы программы	4
<i>Приложение</i> Листинг.....	7

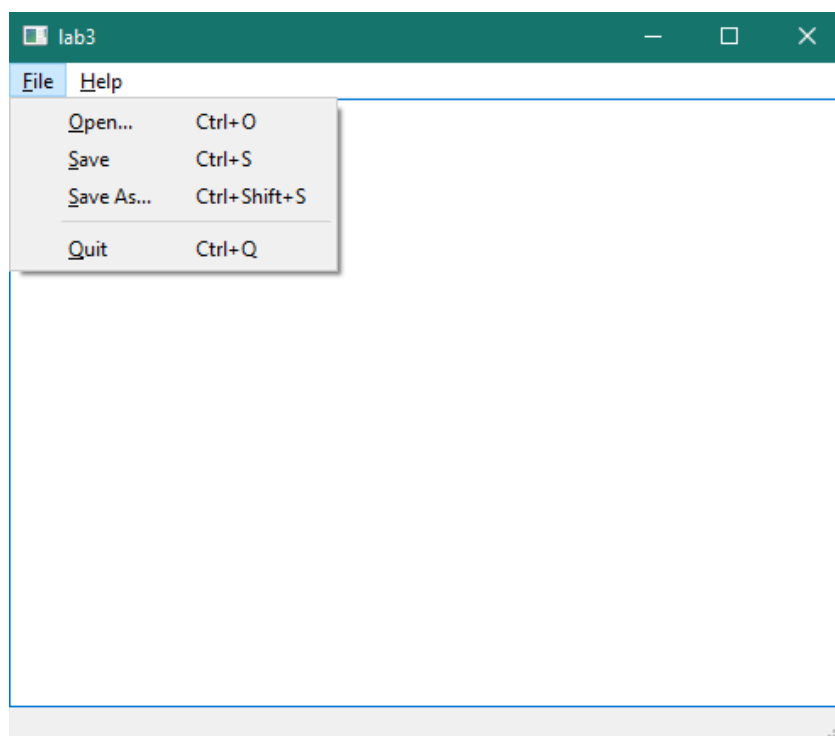
1. Постановка задачи

Цель: Создание SDI – приложения (Single Document Interface, однодокументный интерфейс). Простой текстовый редактор. Окно заставки.

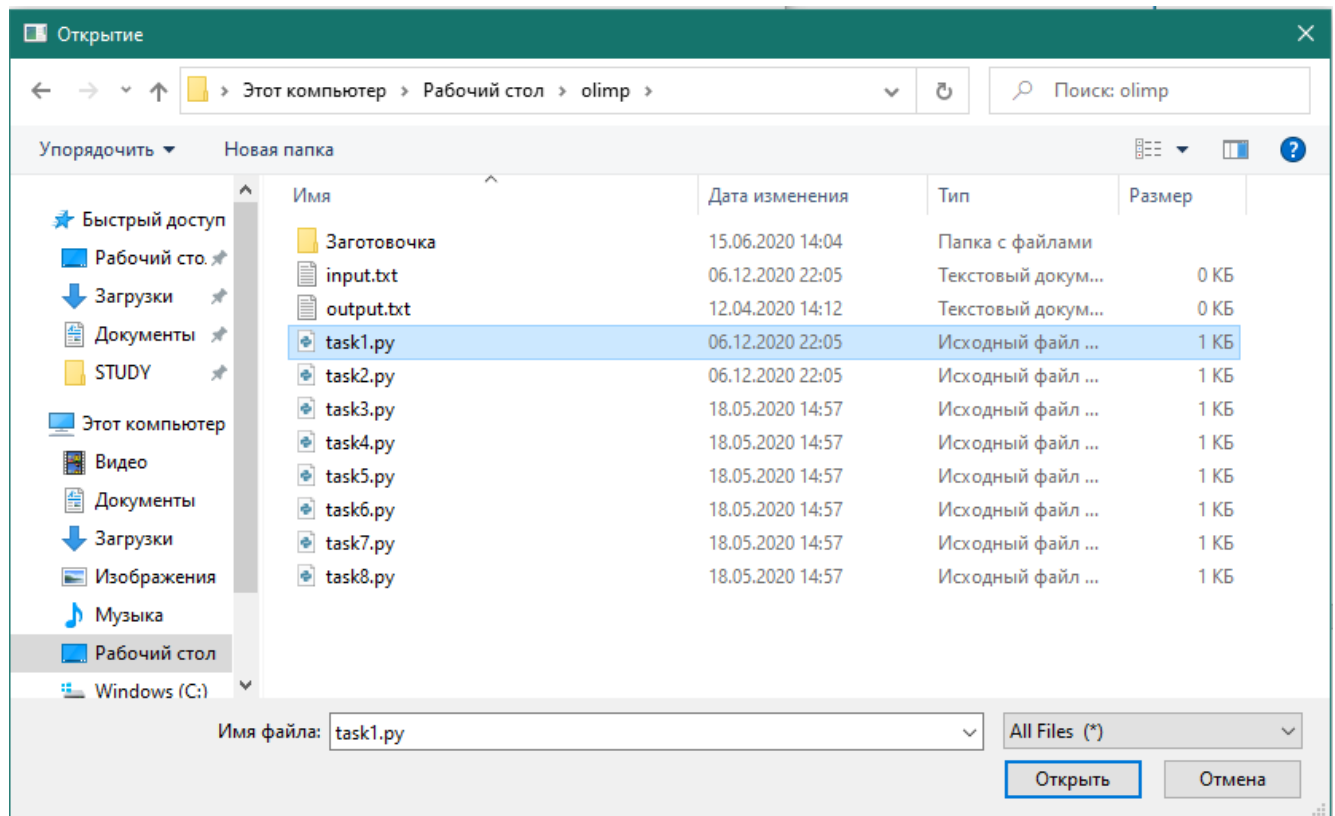
2. Примеры работы программы



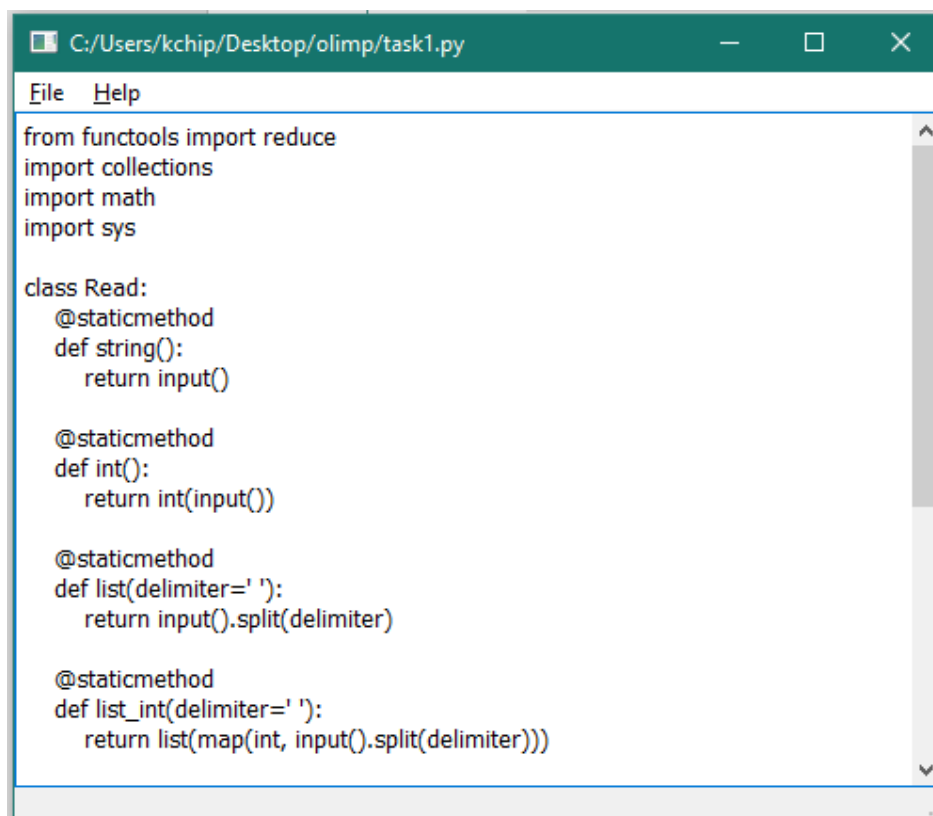
(Экран загрузки)



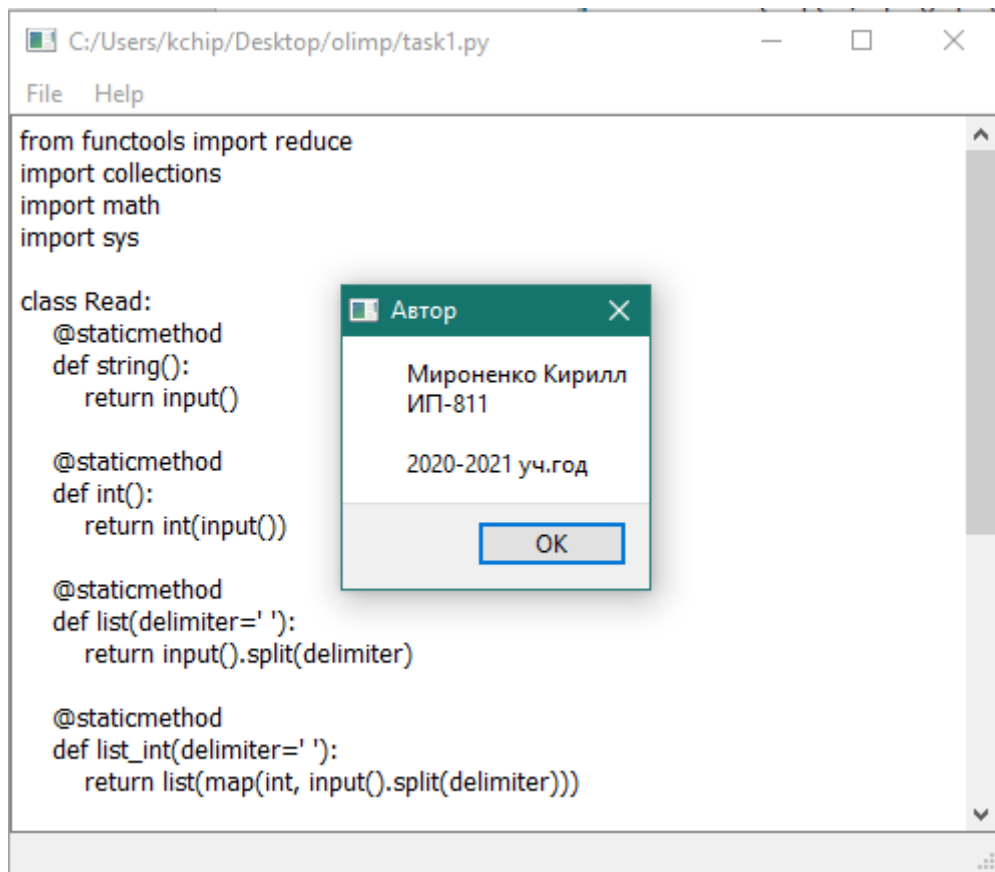
(Всплывающее меню «File»)



(Диалоговое окно с открытием файла)



(Основное окно)



(Диалоговое окно с авторами)

Приложение Листинг

main.cpp

```
#include "sdiprogram_mironenko.h"
#include <QApplication>
#include <QSplashScreen>
#include <QTime>

void LoadModules(QSplashScreen* psplash){
    QTime time;
    time.start();
    for(int i=0; i<100 ; ){
        if(time.elapsed() > 40){
            time.start();
            ++i;
        }
        psplash->showMessage("Loading modules: " + QString::number(i)+"%", Qt::AlignCenter |
Qt::AlignBottom, Qt::cyan);
        qApp->processEvents();
    }
}

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    QSplashScreen splash(QPixmap("backGaara.jpg"));
    SDIProgram_Mironenko w;

    splash.show();
    LoadModules(&splash);
    splash.finish(&w);
    w.show();

    return a.exec();
}
```

docwindow_mironenko.h

```
#ifndef DOCWINDOW_MIRONENKO_H
#define DOCWINDOW_MIRONENKO_H

#include<QTextEdit>
#include<QFileDialog>
#include <QTextStream>

class DocWindow_Mironenko : public QTextEdit
{
    Q_OBJECT
private:
    QString m_strFileName;
public:
```

```

    DocWindow_Mironenko(QWidget * pwgt = nullptr);
signals:
    void changeWindowTitle(const QString&);

public slots:
    void slotLoad();
    void slotSave();
    void slotSaveAs();
};

#endif // DOCWINDOW_MIRONENKO_H

```

docwindow_mironenko.cpp

```

#include "docwindow_mironenko.h"

DocWindow_Mironenko::DocWindow_Mironenko(QWidget* pwgt) : QTextEdit(pwgt)
{
}

void DocWindow_Mironenko::slotLoad()
{
    QString str = QFileDialog::getOpenFileName();
    if (str.isEmpty())
        return;

    QFile file(str);
    if (file.open(QIODevice::ReadOnly)){
        QTextStream stream(&file);
        setPlainText(stream.readAll());
        file.close();

        m_strFileName = str;

        emit changeWindowTitle(m_strFileName);
    }
}

void DocWindow_Mironenko::slotSaveAs(){
    QString str = QFileDialog::getSaveFileName(nullptr, m_strFileName);
    if (!str.isEmpty()){
        m_strFileName=str;
        slotSave();
    }
}

void DocWindow_Mironenko::slotSave(){
    if (m_strFileName.isEmpty()){
        slotSaveAs();
        return;
    }
}

```



```

    QFile file(m_strFileName);
    if (file.open(QIODevice::WriteOnly)){
        QTextStream(&file) << toPlainText();
        file.close();

        emit changeWindowTitle(m_strFileName);
    }
}

```

sdiprogram_mironenko.h

```

#ifndef SDIPROGRAM_MIRONENKO_H
#define SDIPROGRAM_MIRONENKO_H

#include <QMainWindow>
#include <QMessageBox>

#include "docwindow_mironenko.h"

class SDIProgram_Mironenko : public QMainWindow
{
    Q_OBJECT

public:
    SDIProgram_Mironenko(QWidget *parent = nullptr);
    ~SDIProgram_Mironenko();

public slots:
    void slotAbout();
    void slotChangeWindowTitle(const QString&);
};

#endif // SDIPROGRAM_MIRONENKO_H

```

sdiprogram_mironenko.cpp

```

#include "sdiprogram_mironenko.h"
#include "ui_sdiprogram_mironenko.h"

SDIProgram_Mironenko::SDIProgram_Mironenko(QWidget *parent) : QMainWindow(parent)
{
    QMenu *pmnuFile = new QMenu("&File");
    QMenu *pmnuHelp = new QMenu("&Help");

    DocWindow_Mironenko* pdoc = new DocWindow_Mironenko;

    pmnuFile->addAction("&Open...",
        pdoc,
        SLOT(slotLoad()),
        QKeySequence("CTRL+O")
    );
    pmnuFile->addAction("&Save",
        pdoc,

```

```

        SLOT(slotSave()),
        QKeySequence("CTRL+S")
    );
    pmnuFile->addAction("&Save As...",
        pdoc,
        SLOT(slotSaveAs()),
        QKeySequence("CTRL+Shift+S")
    );
    pmnuFile->addSeparator();
    pmnuFile->addAction("&Quit",
        qApp,
        SLOT(quit()),
        QKeySequence("CTRL+Q")
    );
    pmnuHelp->addAction("&Help",
        this,
        SLOT(slotAbout()),
        QKeySequence(Qt::Key_F1)
    );

    menuBar()->addMenu(pmnuFile);
    menuBar()->addMenu(pmnuHelp);

    setCentralWidget(pdoc);

    connect(pdoc, SIGNAL(changeWindowTitle(const QString&)), SLOT(slotChangeWindowTitle(const
QString&)));

    statusBar()->showMessage("Ready",2000);
}

void SDIProgram_Mironenko::slotAbout(){
    QMessageBox::about(this, "Автор", "Мироненко Кирилл\nИП-811\n\n2020-2021 уч.год");
}

void SDIProgram_Mironenko::slotChangeWindowTitle(const QString& str){
    setWindowTitle(str);
}

SDIProgram_Mironenko::~SDIProgram_Mironenko()
{
}

```