Министерство образования и науки

Российской Федерации

Федеральное государственное бюджетное

образовательное учреждение

высшего образования

**«Пермский национальный исследовательский**

**политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**ОТЧЕТ**

**по творческой работе**

Дисциплина: «Информатика»

Тема: Калькулятор

Выполнил работу:

студент группы РИС-20-2Б

Морозова Екатерина Максимовна

Проверила:

Доцент кафедры ИТАС

Полякова О.А

**Пермь, 2021**

**Цель работы**

**Постановка задачи**

1. Разработать алгоритм калькулятора в соответствии с заданием.

2. Реализовать алгоритм в виде программы на алгоритмическом языке С++.

3. Разработать интерфейс средствами Qt.

4. Использование анимации, автоматизации, креативной визуализации повышает рейтинг автора проекта и влияет на экзаменационную оценку.

**Условие задачи:** Хранение капиталов в банке.

Банк предлагает своим клиентам условия:

1. Положить деньги на год под p - процентов годовых с начислением процентов по окончанию года

2. Положить деньги на год под p1 – процентов с начислением процентов ежемесячно

3. Положить деньги на год с ежедневным начислением p2 – рублей на каждую полную тысячу вклада

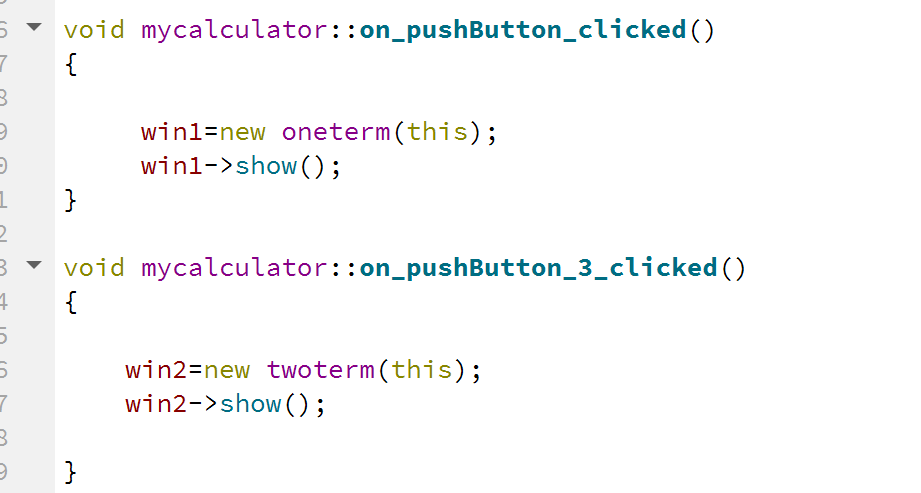
4. Положить деньги на год с ежедневным начислением p3 – процентов на сумму вклада Банк может изменять уговорённые проценты.

Калькулятор получает на вход величину вклада в рублях и копейках или в долларах и центах, а также величину срока в месяцах или годах и вариант условий размещения капитала с вычислением величины вклада к концу указанного срока. Калькулятор должен иметь окна для исходных данных и результатов, для ввода данных, операций и служебных клавиш, клавиши изменения знака числа и вида валюты. Должен позволять корректировать последнее введённое число и выдавать сообщения об ошибках.

**Анализ задачи**

1.Необходимые действия

1) В функции каждой кнопки описать действия для открытия нового окна при нажатии данной кнопки



2)Рассмотрим функционал первого окна:

Изначально я описала функцию для преобразования числового значения суммы. То есть если пользователь ставит галочку в какой денежной системе он собирается положить вклад и хочет эту сумму преобразовать в рубли, то при нажатии кнопки, числовое значение, которое находится в линии для ввода суммы, которую пользователь хочет положить на вклад, изменится на другое число, в зависимости оттого какую денежную систему выбрал пользователь.

1. Считываем данные со строки.

2. С помощью условного оператора определяем у какой денежной системы была поставлена галочка.

3. С помощью цикла for преобразуем данную сумму в рубли.

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

{

float result=1.0;

QString sum1=ui→lineEdit\_2→text();

QStringList elem1=sum1.split("\*");

if(ui→checkBox→isChecked())

{

float cons=75.5;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_2->isChecked())

{

float cons=0.75;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_3->isChecked())

{

}

}

Данная операция была выполнена для каждого окна.

Далее рассмотрим вторую функцию первого окна:

Во второй функции происходят вычисления суммы, которая будет на конец срока вложения вклада.

1. Считываем первую строку со значением изначальной суммы.

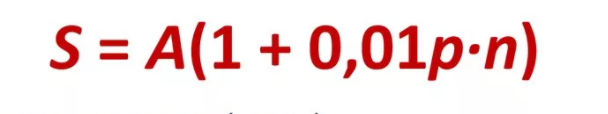
2. Считываем данные со второй строки со значением срока, на который планируется открыть вклад.

3. Присваиваем данные значения к переменным с типом float.

4. По формуле считаем сумму вклада, которую будет иметь пользователь на конец срока.

5. Присваиваем данное значение к строковому типу и выводим его в строке, предназначенной для результата.

Для первого условия используется формула:



S- окончательная сумма вклада;

A- начальная сумма вклада;

p- процент начисления;

n- количество лет;

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

{

float result1=0.0;

float result2=0.0;

QString srok1=ui->srokone->text();

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("+");

QStringList elem2=srok1.split("+");

QString result=ui->lineEdit->text();

for(int i=0;i<elem1.length();i++)

{

result1+=elem1[i].toFloat();

}

for(int i=0;i<elem2.length();i++)

{

result2+=elem2[i].toFloat();

}

for(int i=0;i<result2;i++)

{

result1=result1+result1\*0.0012;

}

QString res2=QString::number(result1);

ui->lineEdit->setText(res2);

}

Для второго, третьего и четвертого окон также содержаться те же самые функции, что и в первом, различие только в формуле, для высчитывания итоговой суммы вклада, которая будет получена по окончанию срока вложения.

Для второго окна используется формула, которая высчитывает количество полных тысяч и умножает это количество на число рублей, которое добавляются ежедневно на каждую полную тысячу вклада.

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

{

float result1=0.0;

float result2=0.0;

QString srok1=ui->srokone->text();

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("+");

QStringList elem2=srok1.split("+");

QString result=ui->lineEdit->text();

for(int i=0;i<elem1.length();i++)

{

result1+=elem1[i].toFloat();

}

for(int i=0;i<elem2.length();i++)

{

result2+=elem2[i].toFloat();

}

float n=0.0;

for(int i=0;i<result2;i++)

{

n=result1/1000;

result1+=n\*5;

}

QString res2=QString::number(result1);

ui->lineEdit->setText(res2);

}

Для третьего окна: формула точно такая же, что и для первого окна, различие только в том, что под переменной n подразумевается количество дней, на которое планируют открыть вклад.

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

{

float result1=0.0;

float result2=0.0;

QString srok1=ui->srokone->text();

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("+");

QStringList elem2=srok1.split("+");

QString result=ui->lineEdit->text();

for(int i=0;i<elem1.length();i++)

{

result1+=elem1[i].toFloat();

}

for(int i=0;i<elem2.length();i++)

{

result2+=elem2[i].toFloat();

}

for(int i=0;i<result2;i++)

{

result1=result1+result1\*0.07;

}

QString res2=QString::number(result1);

ui->lineEdit->setText(res2);

}

Для четвертого окна: формула точно такая же, что и для первого/третьего окна, различие только в том, что под переменной n подразумевается количество месяцев, на которое планируются открыть вклад.

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

{

float result1=0.0;

float result2=0.0;

QString srok1=ui->srokone->text();

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("+");

QStringList elem2=srok1.split("+");

QString result=ui->lineEdit->text();

for(int i=0;i<elem1.length();i++)

{

result1+=elem1[i].toFloat();

}

for(int i=0;i<elem2.length();i++)

{

result2+=elem2[i].toFloat();

}

for(int i=0;i<result2;i++)

{

result1=result1+result1\*0.04;

}

QString res2=QString::number(result1);

ui->lineEdit->setText(res2);

}

2. Программный код

**#ifndef MYCALCULATOR\_H**

#define MYCALCULATOR\_H

#include <QMainWindow>

#include "oneterm.h"

#include "twoterm.h"

#include "threeterm.h"

#include "fourterm.h"

QT\_BEGIN\_NAMESPACE

namespace **Ui** { class **mycalculator**; }

QT\_END\_NAMESPACE

class **mycalculator** : public QMainWindow

{

Q\_OBJECT

public:

**mycalculator**(QWidget \*parent = nullptr);

~***mycalculator***();

private slots:

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

void **on\_pushButton\_3\_clicked**();

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

void **on\_pushButton\_4\_clicked**();

private:

Ui::mycalculator \*ui;

oneterm \*win1;

twoterm \*win2;

threeterm \*win3;

fourterm \*win4;

};

#endif // MYCALCULATOR\_H

**#ifndef MYCALCULATOR\_H**

#define MYCALCULATOR\_H

#include <QMainWindow>

#include "oneterm.h"

#include "twoterm.h"

#include "threeterm.h"

#include "fourterm.h"

QT\_BEGIN\_NAMESPACE

namespace **Ui** { class **mycalculator**; }

QT\_END\_NAMESPACE

class **mycalculator** : public QMainWindow

{

Q\_OBJECT

public:

**mycalculator**(QWidget \*parent = nullptr);

~***mycalculator***();

private slots:

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

void **on\_pushButton\_3\_clicked**();

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

void **on\_pushButton\_4\_clicked**();

private:

Ui::mycalculator \*ui;

oneterm \*win1;

twoterm \*win2;

threeterm \*win3;

fourterm \*win4;

};

#endif // MYCALCULATOR\_H

**#include "mycalculator.h"**

#include "ui\_mycalculator.h"

#include "oneterm.h"

#include "twoterm.h"

#include "threeterm.h"

#include "fourterm.h"

mycalculator::**mycalculator**(QWidget \*parent)

: QMainWindow(*parent*)

, ui(new Ui::mycalculator)

{

ui->setupUi(this);

QPalette palette;

QPixmap pan("C:/MyGRAPPH/green.jpg");

pan = pan.scaled(this->size(), Qt::IgnoreAspectRatio);

palette.setBrush(QPalette::Background, pan);

this->setPalette(palette);

}

mycalculator::~***mycalculator***()

{

delete ui;

}

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

{

win1=new oneterm(this);

win1->show();

}

void mycalculator::**on\_pushButton\_3\_clicked**()

{

win2=new twoterm(this);

win2->show();

}

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

{

win3=new threeterm(this);

win3->show();

}

void mycalculator::**on\_pushButton\_4\_clicked**()

{

win4=new fourterm(this);

win4->show();

}

**#ifndef ONETERM\_H**

#define ONETERM\_H

#include <QDialog>

namespace **Ui** {

class **oneterm**;

}

class **oneterm** : public QDialog

{

Q\_OBJECT

public:

explicit **oneterm**(QWidget \*parent = nullptr);

~***oneterm***();

private slots:

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

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

private:

Ui::oneterm \*ui;

};

#endif // ONETERM\_H

**#include "oneterm.h"**

#include "ui\_oneterm.h"

oneterm::**oneterm**(QWidget \*parent) :

QDialog(*parent*),

ui(new Ui::oneterm)

{

ui->setupUi(this);

QPalette palette;

QPixmap pan("C:/MyGRAPPH/green.jpg");

pan = pan.scaled(this->size(), Qt::IgnoreAspectRatio);

palette.setBrush(QPalette::Background, pan);

this->setPalette(palette);

}

oneterm::~***oneterm***()

{

delete ui;

}

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

{

float result=1.0;

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("\*");

if(ui->checkBox->isChecked())

{

float cons=75.5;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_2->isChecked())

{

float cons=0.75;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_3->isChecked())

{

}

}

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

{

float result1=0.0;

float result2=0.0;

QString srok1=ui->srokone->text();

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("+");

QStringList elem2=srok1.split("+");

QString result=ui->lineEdit->text();

for(int i=0;i<elem1.length();i++)

{

result1+=elem1[i].toFloat();

}

for(int i=0;i<elem2.length();i++)

{

result2+=elem2[i].toFloat();

}

for(int i=0;i<result2;i++)

{

result1=result1+result1\*0.0012;

}

QString res2=QString::number(result1);

ui->lineEdit->setText(res2);

}

**#ifndef TWOTERM\_H**

#define TWOTERM\_H

#include <QDialog>

namespace **Ui** {

class **twoterm**;

}

class **twoterm** : public QDialog

{

Q\_OBJECT

public:

explicit **twoterm**(QWidget \*parent = nullptr);

~***twoterm***();

private slots:

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

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

private:

Ui::twoterm \*ui;

};

#endif // TWOTERM\_H

**#include "twoterm.h"**

#include "ui\_twoterm.h"

twoterm::**twoterm**(QWidget \*parent) :

QDialog(*parent*),

ui(new Ui::twoterm)

{

ui->setupUi(this);

QPalette palette;

QPixmap pan("C:/MyGRAPPH/green.jpg");

pan = pan.scaled(this->size(), Qt::IgnoreAspectRatio);

palette.setBrush(QPalette::Background, pan);

this->setPalette(palette);

}

twoterm::~***twoterm***()

{

delete ui;

}

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

{float result=1.0;

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("\*");

if(ui->checkBox->isChecked())

{

float cons=75.5;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_2->isChecked())

{

float cons=0.75;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_3->isChecked())

{

}

}

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

{

float result1=0.0;

float result2=0.0;

QString srok1=ui->srokone->text();

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("+");

QStringList elem2=srok1.split("+");

QString result=ui->lineEdit->text();

for(int i=0;i<elem1.length();i++)

{

result1+=elem1[i].toFloat();

}

for(int i=0;i<elem2.length();i++)

{

result2+=elem2[i].toFloat();

}

for(int i=0;i<result2;i++)

{

result1=result1+result1\*0.07;

}

QString res2=QString::number(result1);

ui->lineEdit->setText(res2);

}

**#ifndef THREETERM\_H**

#define THREETERM\_H

#include <QDialog>

namespace **Ui** {

class **threeterm**;

}

class **threeterm** : public QDialog

{

Q\_OBJECT

public:

explicit **threeterm**(QWidget \*parent = nullptr);

~***threeterm***();

private slots:

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

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

private:

Ui::threeterm \*ui;

};

#endif // THREETERM\_H

**#include "threeterm.h"**

#include "ui\_threeterm.h"

threeterm::**threeterm**(QWidget \*parent) :

QDialog(*parent*),

ui(new Ui::threeterm)

{

ui->setupUi(this);

QPalette palette;

QPixmap pan("C:/MyGRAPPH/green.jpg");

pan = pan.scaled(this->size(), Qt::IgnoreAspectRatio);

palette.setBrush(QPalette::Background, pan);

this->setPalette(palette);

}

threeterm::~***threeterm***()

{

delete ui;

}

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

{

float result=1.0;

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("\*");

if(ui->checkBox->isChecked())

{

float cons=75.5;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_2->isChecked())

{

float cons=0.75;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_3->isChecked())

{

}

}

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

{

float result1=0.0;

float result2=0.0;

QString srok1=ui->srokone->text();

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("+");

QStringList elem2=srok1.split("+");

QString result=ui->lineEdit->text();

for(int i=0;i<elem1.length();i++)

{

result1+=elem1[i].toFloat();

}

for(int i=0;i<elem2.length();i++)

{

result2+=elem2[i].toFloat();

}

float n=0.0;

for(int i=0;i<result2;i++)

{

n=result1/1000;

result1+=n\*5;

}

QString res2=QString::number(result1);

ui->lineEdit->setText(res2);

}

**#ifndef FOURTERM\_H**

#define FOURTERM\_H

#include <QDialog>

namespace **Ui** {

class **fourterm**;

}

class **fourterm** : public QDialog

{

Q\_OBJECT

public:

explicit **fourterm**(QWidget \*parent = nullptr);

~***fourterm***();

private slots:

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

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

private:

Ui::fourterm \*ui;

};

#endif // FOURTERM\_H

**#include "fourterm.h"**

#include "ui\_fourterm.h"

fourterm::**fourterm**(QWidget \*parent) :

QDialog(*parent*),

ui(new Ui::fourterm)

{

ui->setupUi(this);

QPalette palette;

QPixmap pan("C:/MyGRAPPH/green.jpg");

pan = pan.scaled(this->size(), Qt::IgnoreAspectRatio);

palette.setBrush(QPalette::Background, pan);

this->setPalette(palette);

}

fourterm::~***fourterm***()

{

delete ui;

}

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

{

float result=1.0;

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("\*");

if(ui->checkBox->isChecked())

{

float cons=75.5;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_2->isChecked())

{

float cons=0.75;

for(int i=0;i<elem1.length();i++)

{

result\*=elem1[i].toFloat()\*cons;

}

QString res=QString::number(result);

ui->lineEdit\_2->setText(res);

}

if(ui->checkBox\_3->isChecked())

{

}

}

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

{

float result1=0.0;

float result2=0.0;

QString srok1=ui->srokone->text();

QString sum1=ui->lineEdit\_2->text();

QStringList elem1=sum1.split("+");

QStringList elem2=srok1.split("+");

QString result=ui->lineEdit->text();

for(int i=0;i<elem1.length();i++)

{

result1+=elem1[i].toFloat();

}

for(int i=0;i<elem2.length();i++)

{

result2+=elem2[i].toFloat();

}

for(int i=0;i<result2;i++)

{

result1=result1+result1\*0.04;

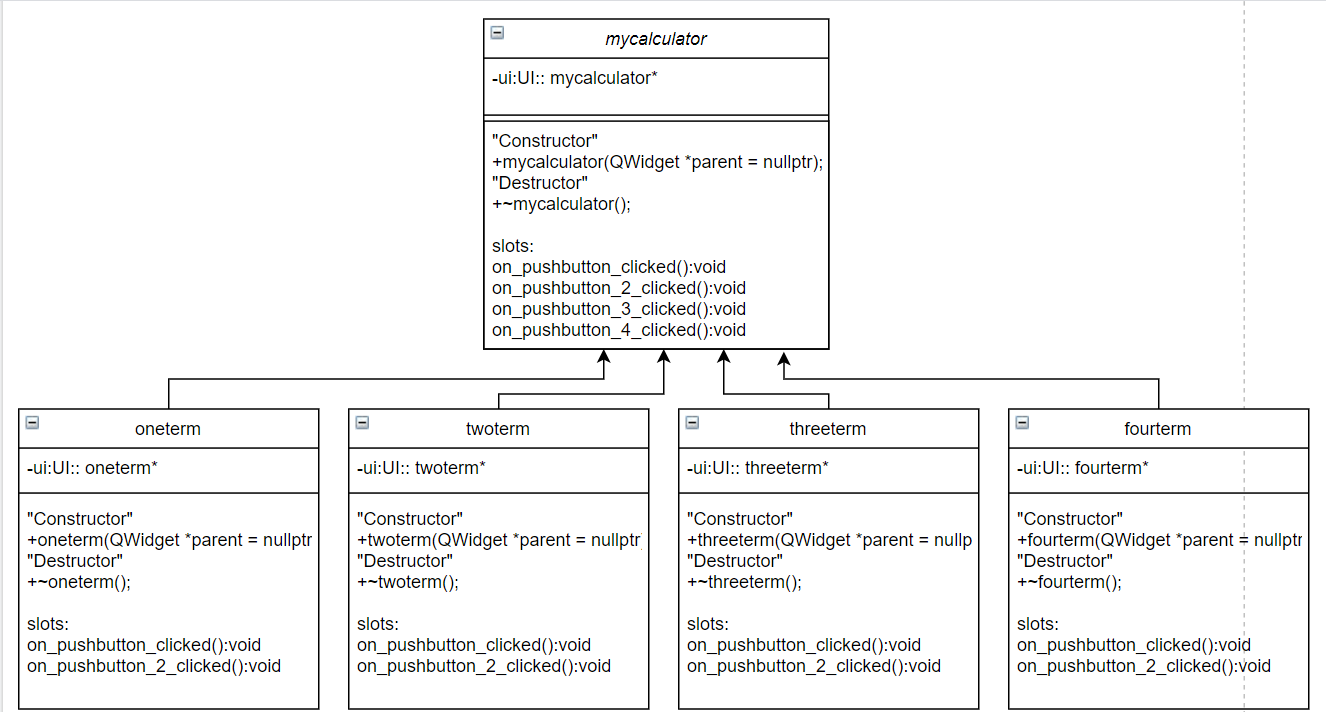
}

QString res2=QString::number(result1);

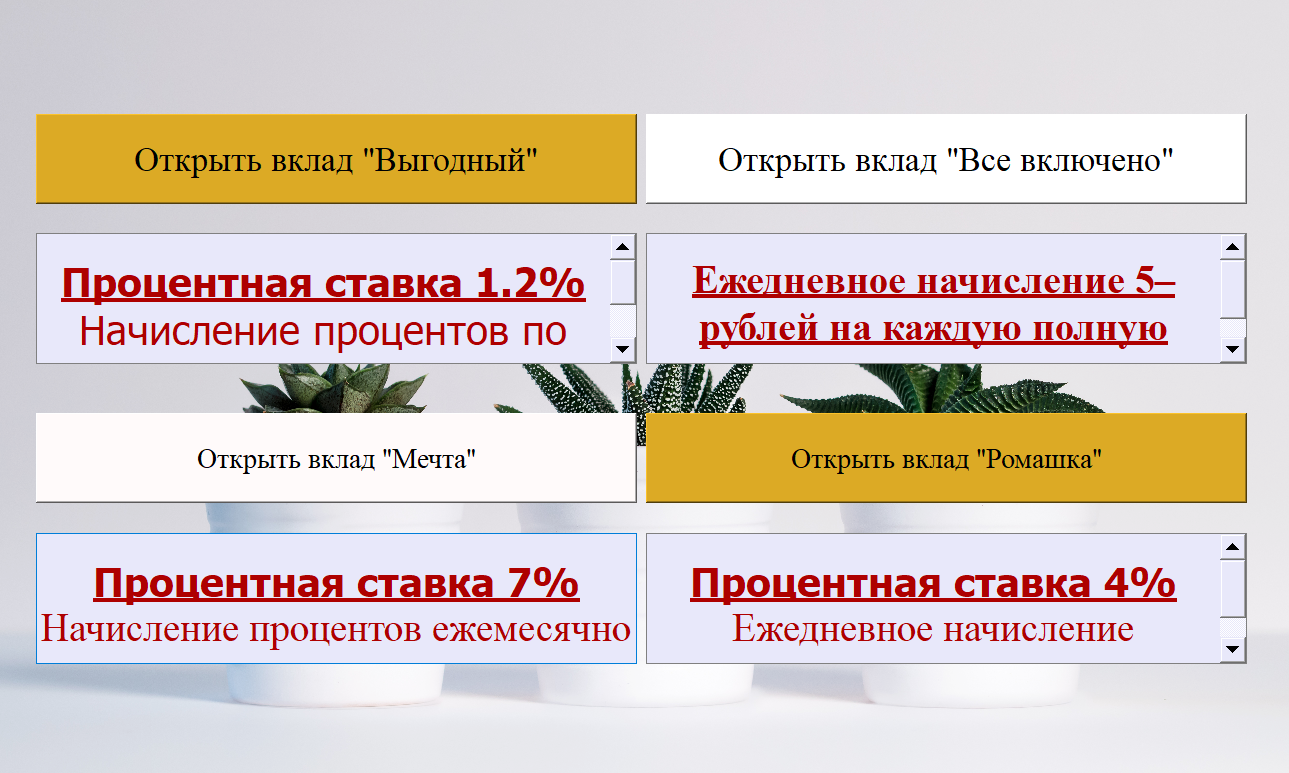
ui->lineEdit->setText(res2);

}

**UML-диаграмма**

****

**Результат**

****

