Министерство науки и высшего образования Российской Федерации

федеральное государственное бюджетное образовательное учреждение   
высшего образования

”Алтайский государственный технический университет им. И. И. Ползунова”

Факультет информационных технологий

наименование подразделения

Кафедра информатики, вычислительной техники

и информационной безопасности

наименование кафедры

Направление Информатика и вычислительная техника

Профиль Программно-техническое обеспечение автоматизированных систем

Отчёт защищен с оценкой\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Я.Б. Ерёмин

(подпись руководителя от вуза) (инициалы, фамилия)

“\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022г.

ОТЧЁТ

по лабораторной работе № 4 тема лабораторной работы

по курсу **Тестирование программного обеспечения автоматизированных систем**

ЛР 09.03.01.02.000 O

обозначение документа

Студент группы ИВТ-91 Е.И. Аненков

инициалы, фамилия

Руководитель работы старший преподаватель Я.Б. Ерёмин

должность, ученое звание инициалы, фамилия

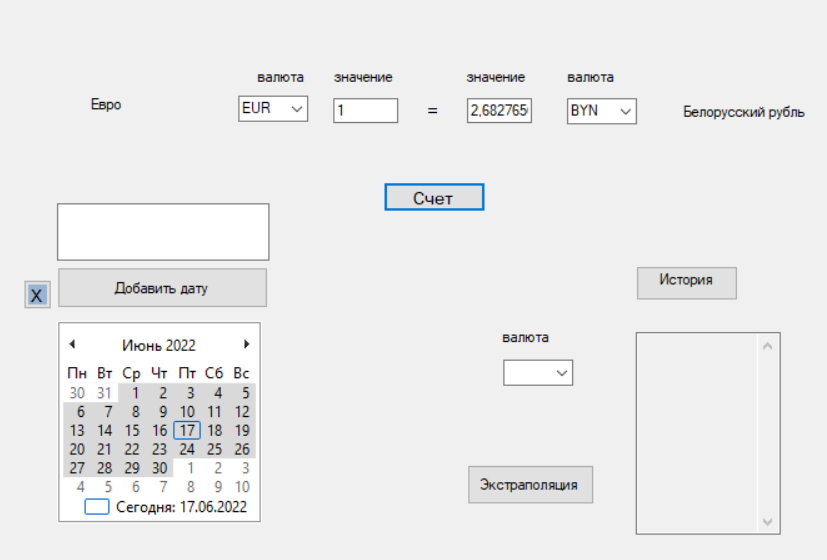
Барнаул 2022

Ссылка на репозиторий: <https://github.com/Egor148/Anenkov/tree/Kurs>

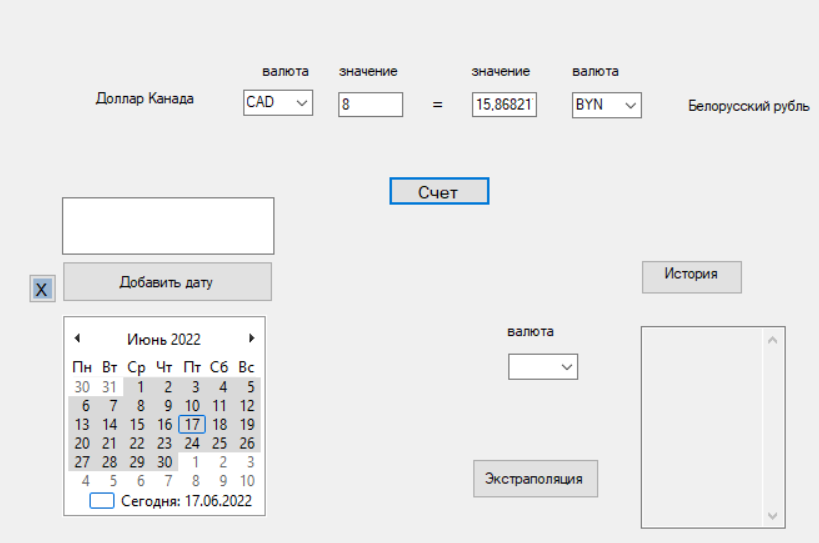
Вариант: Конвертер валют с возможностью предсказания курсов (несколько валют и история курса, можно узнать курс на след. дни экстраполяцией).

**Тесты программы:**

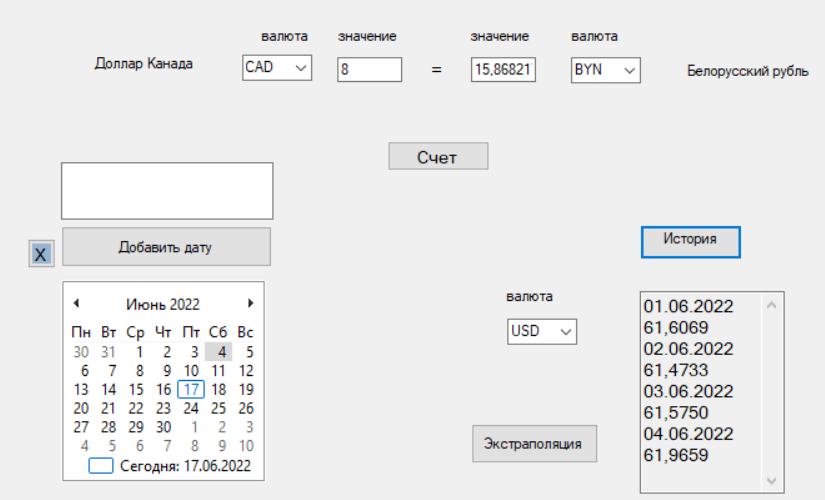
**Вычисление валюты в количестве 1 шт.**



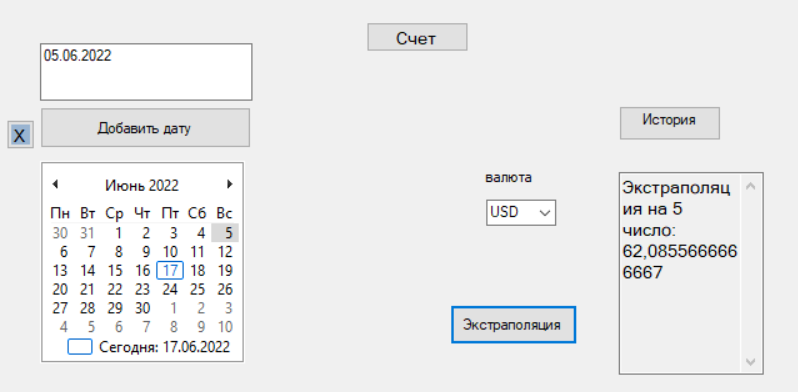
**Вычисление валюты в количестве n шт.**



Выбираем первую дату. Добавляем. Выбираем вторую дату и тоже добавляем. Выбираем валюты и выводим историю курса за выбранный промежуток времени.



Выбираем дату для экстраполяции.



**Код прогаммы**

**Класс Valute**

namespace kurs\_valut

{

public class Valute

{

public string[] name = new[] { "USD", "EUR", "RUB", "BYN", "INR", "KZT", "CAD", "CNY", "UZS" };

public string id; //номер валюты

public List<string> his = new List<string>();

public List<string> date = new List<string>();

}

}

**Класс Form1**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace kurs\_valut

{

public partial class Form1 : Form

{

double c1, c2;

Valute val = new Valute();

public Form1()

{

InitializeComponent();

//объект класса Valute

monthCalendar1.DateChanged += monthCalendar1\_DateSelected;

for (int i = 0; i < val.name.Length; i++)

{

comboBox1.Items.Add(val.name[i]);

comboBox2.Items.Add(val.name[i]);

comboBox3.Items.Add(val.name[i]);

}

comboBox1.SelectedIndexChanged += comboBox1\_SelectedIndexChanged;

comboBox2.SelectedIndexChanged += comboBox2\_SelectedIndexChanged;

comboBox3.SelectedIndexChanged += comboBox3\_SelectedIndexChanged;

textBox1.Text = "1";

textBox4.Text = "1";

}

private void comboBox3\_SelectedIndexChanged(object sender, EventArgs e)

{

//throw new NotImplementedException();

}

private void monthCalendar1\_DateSelected(object sender, DateRangeEventArgs e)

{

// throw new NotImplementedException();

}

private void Form1\_Load(object sender, EventArgs e)

{

}

public string get\_kurs(string name)

{

string url = "http://www.cbr.ru/scripts/XML\_daily.asp";

//string url = "https://tursportopt.ru/category/rybolovnye-tovary-optom/";

//XmlDocument xml\_doc = new XmlDocument();

// xml\_doc.Load(url);

DataSet ds = new DataSet();

ds.ReadXml(url);

DataTable currency = ds.Tables["Valute"];

foreach (DataRow row in currency.Rows)

{

if (row["CharCode"].ToString() == name)//Ищу нужный код валюты

{

return row["Value"].ToString(); //Возвращаю значение курсы валюты

}

}

return "";

}

public void get\_history\_kurs(string date1, string date2, string id)

{

string url = "https://cbr.ru/scripts/XML\_dynamic.asp?date\_req1=" + date1 + "&date\_req2=" + date2 + "&VAL\_NM\_RQ=" + id;

DataSet ds = new DataSet();

ds.ReadXml(url);

DataTable currency = ds.Tables["Record"];

foreach (DataRow row in currency.Rows)

{

val.his.Add(row["Date"].ToString());

val.his.Add(row["Value"].ToString());

}

}

public double read\_kurs(double c1, double c2, double n)

{

double res;

res = ((c1 / c2) \* n);

return res;

}

private void kurs() //отношение курсов

{

string selectedState1 = comboBox1.SelectedItem.ToString();

string selectedState2 = comboBox2.SelectedItem.ToString();

if (selectedState1 != "RUB" && selectedState2 != "RUB")

{

string selected = get\_box1();

string val = get\_kurs(selected);

c1 = Convert.ToDouble(val); // наше значение 1 колонки

selected = get\_box2();

val = get\_kurs(selected);

c2 = Convert.ToDouble(val); // значение 2 колонки

string nn = textBox1.Text;

double n = Convert.ToDouble(nn);

double res = read\_kurs(c1, c2, n);

textBox4.Text = Convert.ToString(res);

}

else

{

string selected1 = get\_box1();

string selected2 = get\_box2();

if (selected1 == "RUB") // если первая колонка рубль

{

string val2 = get\_kurs(selected2);

c2 = Convert.ToDouble(val2); // значение 2 колонки

textBox4.Text = "1";

string nn = textBox1.Text;

double n = Convert.ToDouble(nn);

textBox1.Text = Convert.ToString(c2);

}

else if (selected2 == "RUB")

{

string val1 = get\_kurs(selected1);

c1 = Convert.ToDouble(val1); // значение 2 колонки

textBox1.Text = "1";

string nn = textBox4.Text;

double n = Convert.ToDouble(nn);

textBox4.Text = Convert.ToString(c1);

}

}

}

public double get\_text1()

{

string s = textBox1.Text;

return Convert.ToDouble(s);

}

public double get\_text2()

{

string s = textBox1.Text;

return Convert.ToDouble(s);

}

public string get\_box1()//выбор comboBox1

{

string selectedState = comboBox1.SelectedItem.ToString();

return selectedState;

}

public string get\_box2()//выбор comboBox2

{

string selectedState = comboBox2.SelectedItem.ToString();

return selectedState;

}

public string get\_box3() // валюта для истории курса валюты

{

string selectedState = comboBox3.SelectedItem.ToString();

return selectedState;

}

private void comboBox2\_SelectedIndexChanged(object sender, EventArgs e)

{

string selectedState = comboBox2.SelectedItem.ToString();

if (selectedState != "RUB")

{

string val = get\_kurs(selectedState);

c2 = Convert.ToDouble(val);

}

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

string selectedState = comboBox1.SelectedItem.ToString();

if (selectedState != "RUB")

{

string val = get\_kurs(selectedState);

c1 = Convert.ToDouble(val);

}

}

private void name\_valute1(object sender, EventArgs e)

{//"EUR", "RUB", "BYN", "INR", "KZT", "CAD", "CNY", "UZS"

switch (get\_box1())

{

case "USD":

label4.Text = "Доллар США";

break;

case "EUR":

label4.Text = "Евро";

break;

case "RUB":

label4.Text = "Рубль";

break;

case "BYN":

label4.Text = "Белорусский рубль";

break;

case "INR":

label4.Text = "Рупий";

break;

case "KZT":

label4.Text = "Тенге";

break;

case "CAD":

label4.Text = "Доллар Канада";

break;

case "CNY":

label4.Text = "Юань";

break;

case "UZS":

label4.Text = "Узб. Сумы";

break;

}

}

private void button1\_Click(object sender, EventArgs e)

{

kurs();//вызов функции для расчета отношения валют

}

private void button3\_Click(object sender, EventArgs e)

{

if (val.date.Count != 2)

{

val.date.Add(textBox3.Text);

}

textBox3.Clear();

}

private void button5\_Click(object sender, EventArgs e)

{

textBox3.Clear();

}

private void monthCalendar1\_DateChanged(object sender, DateRangeEventArgs e)

{

this.monthCalendar1.MaxSelectionCount = 1;

// Это возвратит только дату без времени

textBox3.Text += get\_kalend();

}

private string get\_kalend()

{

string s;

s = String.Format("{0}", this.monthCalendar1.SelectionRange.Start.ToShortDateString() + " ");

return s;

}

public void Replace()

{

for (int i = 0; i < val.date.Count; i++)

{

string s = val.date[i];

s = s.Replace(".", "/");

val.date[i] = s;

}

for (int i = 0; i < val.his.Count; i++)

{

string s = val.his[i];

s = s.Replace(".", "/");

val.his[i] = s;

}

}

public void his\_Replace()

{

for (int i = 0; i < val.his.Count; i++)

{

string s = val.his[i];

s = s.Replace(".", "/");

val.his[i] = s;

}

}

private void button2\_Click(object sender, EventArgs e)

{

textBox2.Clear();

Replace();//функция убирающая точки из даты и вставляющая /

switch (get\_box3())

{

case "USD":

val.id = "R01235";

break;

case "EUR":

val.id = "R01239";

break;

case "RUB":

break;

case "BYN":

val.id = "R01090B";

break;

case "INR":

val.id = "R01270";

break;

case "KZT":

val.id = "R01335";

break;

case "CAD":

val.id = "R01350";

break;

case "CNY":

val.id = "R01375";

break;

case "UZS":

val.id = "R01717";

break;

}

//switch case с названием валюты и кодом

get\_history\_kurs(val.date[0], val.date[1], val.id);

for (int i = 0; i < val.his.Count; i++)

{

if (val.his[i] == "") break;

textBox2.AppendText(val.his[i] + Environment.NewLine);

}

}

public double extrapolate(double[,] d, double x)

{

double y = d[0, 1] + (x - d[0, 0]) / (d[1, 0] - d[0, 0]) \* (d[1, 1] - d[0, 1]);

return y;

}

private void button4\_Click(object sender, EventArgs e)

{

his\_Replace();//передаем список

double d1, d2, price1 = 0, price2 = 0;

string s = val.date[0].Substring(0, val.date[0].IndexOf('/'));//обрезаем только число из даты

d1 = Convert.ToDouble(s);

string s2 = val.date[1].Substring(0, val.date[1].IndexOf('/'));//обрезаем только число из даты

d2 = Convert.ToDouble(s2);

string pr1, pr2;

//поиск цены в his[i]

for (int i = 0; i < val.his.Count; i += 2)

{

if (s.Equals(val.his[i].Substring(0, (val.his[i]).IndexOf('/'))))

{

pr1 = val.his[i + 1];

price1 = Convert.ToDouble(pr1);

}

else if (s2.Equals(val.his[i].Substring(0, (val.his[i]).IndexOf('/'))))

{

pr2 = val.his[i + 1];

price2 = Convert.ToDouble(pr2);

}

}

double[,] d = { { d1, price1 }, { d2, price2 } }; //воткнуть даты придумать как считать разницу в функции

//взять цену из функции get\_kurs\_history добавить еще один аргумент bool.

// если выполняем экстраполяцию то type(bool) = true

// и там в обработке ищем дату и возвращаем цену, подставляем цену и делаем экстраполяцию

// или найти цену в his[i]

string x = textBox3.Text;

s = x.Substring(0, x.IndexOf('.'));//обрезаем только число из даты

double xx = Convert.ToDouble(s); //нужная дата

textBox2.Clear();

textBox2.AppendText("Экстраполяция на " + xx + " число: " + extrapolate(d, xx));

}

private void name\_valute2(object sender, EventArgs e)

{

switch (get\_box2())

{

case "USD":

label5.Text = "Доллар США";

break;

case "EUR":

label5.Text = ("Евро");

break;

case "RUB":

label5.Text = "Рубль";

break;

case "BYN":

label5.Text = "Белорусский рубль";

break;

case "INR":

label5.Text = "Рупий";

break;

case "KZT":

label5.Text = "Тенге";

break;

case "CAD":

label5.Text = "Доллар Канада";

break;

case "CNY":

label5.Text = "Юань";

break;

case "UZS":

label5.Text = "Узб. Сумы";

break;

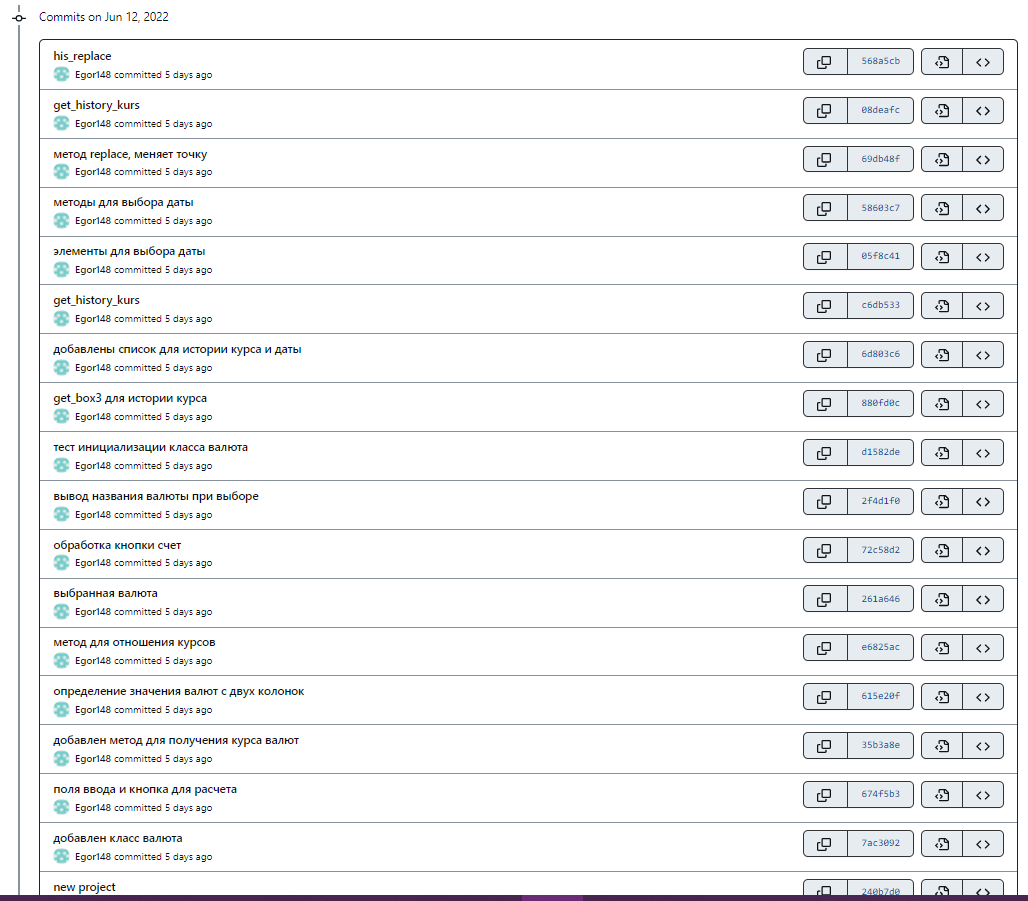
}

}

}

}

**Коммиты**

****

**Юнит тесты**

using Microsoft.VisualStudio.TestTools.UnitTesting;

using kurs\_valut;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace kurs\_valut.Tests

{

[TestClass()]

public class Form1Tests

{

[TestMethod()]

public void extrapolateTest()

{

Form1 f = new Form1();

double[,] d = { { 1, 70 }, { 3, 75 } };

double expected = 77.5;

Assert.AreEqual(expected, f.extrapolate(d, 4));

}

[TestMethod()]

public void read\_kursTest()

{

Form1 f = new Form1();

double c1, c2, n;

c1 = 25;

c2 = 5;

n = 3;

double expected = 15;

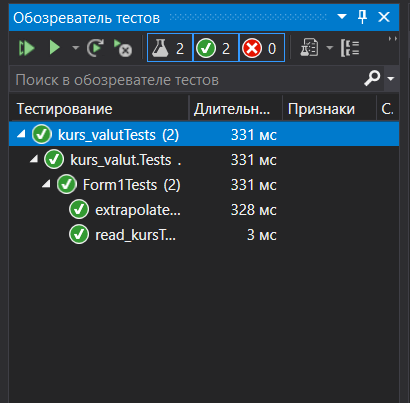
double res = f.read\_kurs(c1, c2, n);

Assert.AreEqual(expected, res);

}

}

}

****