Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Синтаксис языка Kotlin. Программирование линейных и разветвляющихся алгоритмов на языке Kotlin.**»

Выполнил: Плоских Александр Павлович

Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2023

**Задание:**

Класс «Покупатель» Поля класса: название покупаемого продукта, цена, количество, кошелек, настроение (начальное значение=10). Методы класса: «Информация о покупке» - отображается название купленного продукта, его количество и стоимость покупки. «Купить» - если человек что-то покупает, то стоимость покупки увеличивается, а наличность кошелька уменьшается. Выдавать сообщение, сколько денег осталось в кошельке после совершения покупки. Добавьте связь с настроением: если денег хватило на совершение покупки, то настроение увеличивается на 50% от суммы покупки, а если не хватило – то настроение уменьшается на количество единиц, равное количеству денег, которых не хватило для совершения покупки. «Посмотреть настроение» - отслеживайте само настроение: если настроение больше 15, то оно «бодрое», если от 5 до 15, то – нормальное, если от -10 до 5 , то – плохое, если меньше -10, то – депрессивное.

**Входные и выходные данные**

**Входные**

Класс Pokupat с полями:

Wallet-тип данных int, хранит в себе сумму денег покупателя

Kol-тип данных int, хранит в себе количество выбранного товара

Tovar-тип данных string,это поле хранит в себе список товаров

**Выходные**

Класс Pokupat с полями:

Nastr-тип данных double,хранит значение настроения покупателя

Name-тип данных string, хранит в себе все товары, которые оплатил покупатель

All\_price-тип данных int,хранит стоимость все совершенных покупок

**Блок-схема**

**Листинг программы**

**Класс**

namespace var6

{

public class Pokupat

{

private double wallet;

private double nastr;

private string[] tovar={ "Картофель","Помидор","Огурец","Блинчики","Колбаса","Сыр","Липтон"};

private int kol;

private int price;

private string name;

private int all\_price;

public Pokupat(int wallet)

{

this.wallet = wallet;

this.nastr = 10;

}

public void Buy(int tovar\_number, int kol)

{

switch (tovar\_number)

{

case 0:

this.price = 20;

break;

case 1:

this.price = 260;

break;

case 2:

this.price = 230;

break;

case 3:

this.price = 250;

break;

case 4:

this.price = 270;

break;

case 5:

this.price = 150;

break;

case 6:

this.price = 79;

break;

}

this.kol=kol;

if (price\*this.kol <= wallet)

{

this.wallet -= this.price\*this.kol;

this.all\_price+= this.price \* this.kol;

if (this.name != null)

{

if (!this.name.Contains(this.tovar[tovar\_number]))

{

this.name += this.tovar[tovar\_number] + " ";

}

}

else

{

this.name += this.tovar[tovar\_number] + " ";

}

this.nastr += this.price \* 0.5;

}

else

{

this.nastr -= this.price\*kol - this.wallet;

}

}

public string Sm\_nastr ()

{

if (this.nastr > 15)

return "Бодрое";

if (this.nastr >= 5 && this.nastr < 15)

return "Хорошее";

if (this.nastr >= -10 && this.nastr < 5)

return "плохое";

else

return "Депрессия";

}

public string Info()

{

return "Куплено \n" + this.name +

"Стоимость\n" + (this.all\_price);

}

}

}

**Form1.cs**

namespace var6

{

public partial class Form1 :Form

{

public Form1 ()

{

InitializeComponent();

}

Pokupat[] pk = new Pokupat[10];

int k=0;

//info

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

{

try

{

listBox1.Items.Add(pk[comboBox2.SelectedIndex].Info());

}

catch { }

}

//buy

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

{

try

{

pk[comboBox2.SelectedIndex].Buy(Convert.ToInt32(comboBox1.SelectedIndex), Convert.ToInt32(numericUpDown1.Value));

}

catch{ }

}

//nastr

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

{

try

{

listBox1.Items.Add(pk[comboBox2.SelectedIndex].Sm\_nastr());

}

catch { }

}

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

{

try

{

pk[k] = new Pokupat(Convert.ToInt32(textBox1.Text));

MessageBox.Show($"Данные занесены давай продолжим в тоже духе\nосталось еще {10 - (k + 1)} покупателей");

k++;

if (k == 10)

{

button4.Visible = false;

button1.Visible = true;

button2.Visible = true;

button3.Visible = true;

listBox1.Visible = true;

comboBox1.Visible = true;

textBox1.Visible = false;

numericUpDown1.Visible = true;

label1.Visible = false;

comboBox2.Visible = true;

button5.Visible = true;

}

}

catch (FormatException)

{

MessageBox.Show("Sorry");

}

}

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

{

for(int i = 1; i <= 10; i++)

{

comboBox2.Items.Add($"Клиент {i}");

}

}

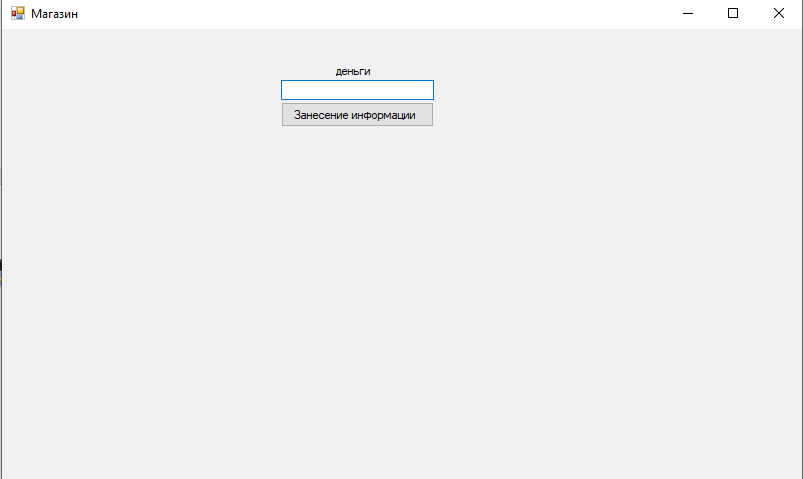
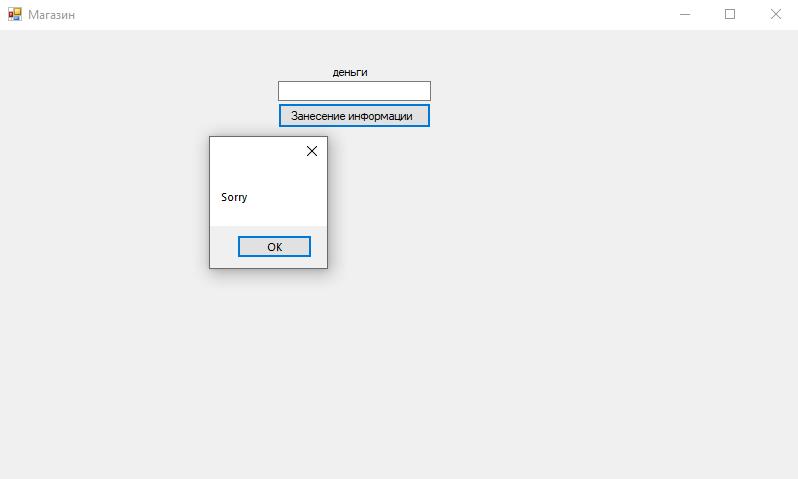
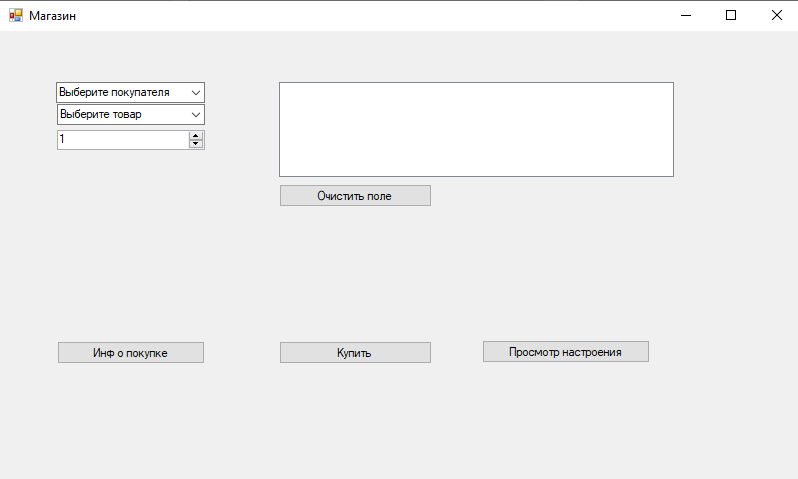
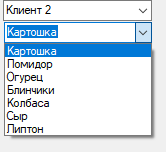
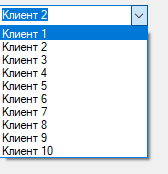
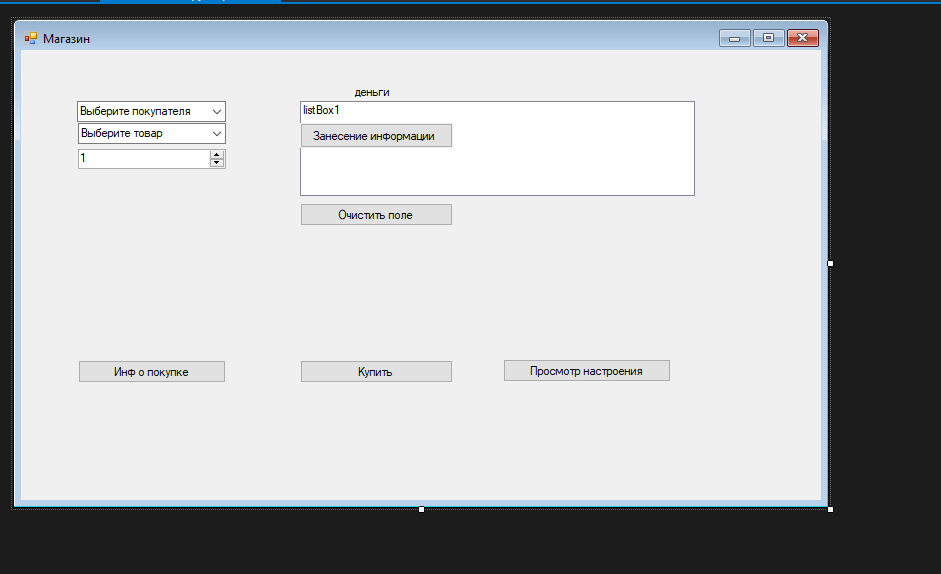
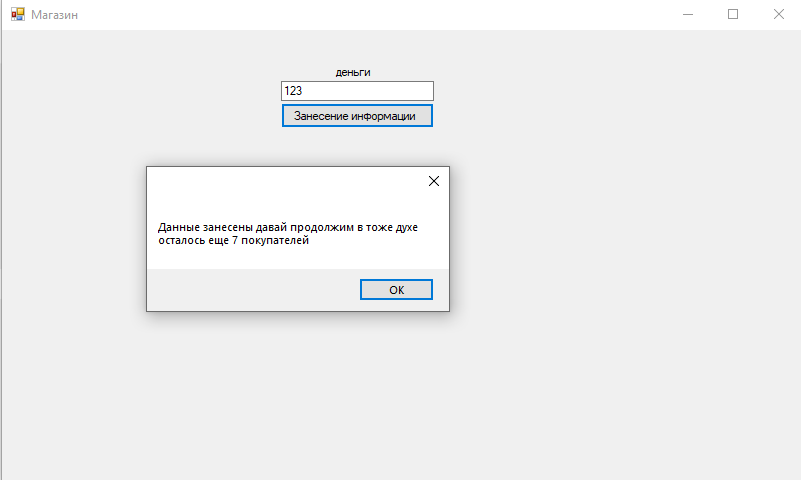
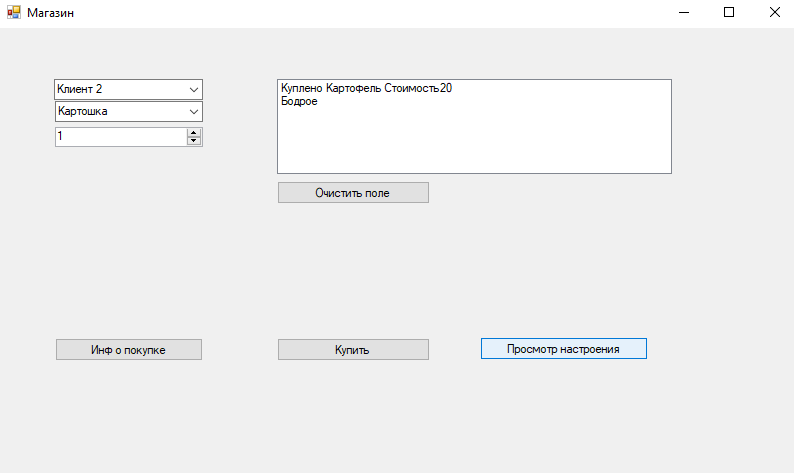
private void button5\_Click (object sender, EventArgs e)

{

listBox1.Items.Clear();

}

}

**Тестовые ситуации**