



МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технологический университет
«СТАНКИН»
(ФГБОУ ВО «МГТУ «СТАНКИН»)

**Институт
информационных
систем и технологий**

**Кафедра
информационных систем**

Основная образовательная программа 09.03.02
«Информационные системы»

дисциплина «Технологии программирования»

Ответ на задание
к практическому занятию
«Реализация. Спецификации. Формы представления»

Студент
группы ИДБ-19-06

Макаров М.А.

Преподаватель
К.Т.Н., доц.

Волкова О. Р.

Москва, 2021 г.

Задание.

В соответствии с разработанной диаграммой взаимодействия (UseCase) или диаграммой потоков данных (DFD) составить миниспецификацию процесса разрабатываемого программного продукта. Тело миниспецификации должно содержать псевдокод или flow-формы.

Для миниспецификации должен быть представлен код реализации на конкретном языке программирования, выбранном для реализации проекта

Дополнительное условие:

Количество процессов для которых составляются миниспецификации должно равняться количеству студентов в группе разработчиков. Для каждой миниспецификации должен быть представлен код реализации на конкретном языке программирования.

Миниспецификация №1. Кнопка “Всего”.

Миниспецификация предназначена для подсчета итоговой стоимости выбранных товаров.

Реализация миниспецификации программно:

```
private void button1_Click(object sender, EventArgs e)
{ //btnTotal
    double lat, mocha, espr, Raf, cappu, afri, mTea, cTea;
    double cCake, rValvet, bFor, bCream, souf, Choco, cheese, rain;
    double tax;
    tax = 0.1;

    lat = 49; mocha = 50; espr = 80; Raf = 59; cappu = 70; afri = 85; mTea = 40; cTea = 55; //coffee
    pries
    cCake = 50; rValvet = 100; bFor = 60; bCream = 40; souf = 79; Choco = 109; cheese = 85; rain = 89;
    //cake prices
    CultureInfo culture = CultureInfo.CreateSpecificCulture("ru-RU");

    //Coffee
    double latteCof = Convert.ToDouble(txtLatte.Text);
    double mochaCof = Convert.ToDouble(txtMocha.Text);
    double espressoCof = Convert.ToDouble(txtEspresso.Text);
    double RafCof = Convert.ToDouble(txtRafCoffee.Text);
    double cappCof = Convert.ToDouble(txtCappu.Text);
    double afriCof = Convert.ToDouble(txtAfricanCoffee.Text);
    double miTea = Convert.ToDouble(txtMilkTea.Text);
    double cineseTea = Convert.ToDouble(txtChineseTea.Text);
    //Cakes
    double cofCake = Convert.ToDouble(txtCoffeCake.Text);
    double redValvet = Convert.ToDouble(txtRedValvetCake.Text);
    double bForest = Convert.ToDouble(txtBlackForestCake.Text);
    double bostonCream = Convert.ToDouble(txtBostonCream.Text);
```

```
double souffle = Convert.ToDouble(txtSouffleCake.Text);
double kilbChoco = Convert.ToDouble(txtKillburnChoco.Text);
double cheeseCak = Convert.ToDouble(txtCheeseCake.Text);
double rainbow = Convert.ToDouble(txtRainbowCake.Text);
```

```
Cafe eat_in_cafe = new Cafe(latteeCof, mochaCof, espressoCof, RafCof, cappCof, afriCof, miTea,
cineseTea, cofCake, redValvet, bForest, bostonCream, souffle, kilbChoco, cheeseCak, rainbow);
```

```
double drinkCosts = (latteeCof * lat * 1.1) + (mochaCof * mocha * 1.1) + (espressoCof * espr *
1.1) + (RafCof * Raf * 1.1)
    + (cappCof * cappu * 1.1) + (afriCof * afri * 1.1) + (miTea * mTea * 1.1) + (cineseTea * cTea *
1.1);
lblDrinkCost.Text = Convert.ToString(drinkCosts);
double cakeCosts = (cofCake * cCake * 1.1) + (redValvet * rValvet * 1.1) + (bForest * bFor * 1.1) +
(bostonCream * bCream * 1.1)
    + (souffle * souf * 1.1) + (kilbChoco * Choco * 1.1) + (cheeseCak * cheese * 1.1) + (rainbow *
rain * 1.1);
lblCakeCost.Text = Convert.ToString(cakeCosts);
```

```
double drinkCostsnotax = (latteeCof * lat) + (mochaCof * mocha) + (espressoCof * espr) + (RafCof
* Raf) + (cappCof * cappu) + (afriCof * afri) + (miTea * mTea) + (cineseTea * cTea);
lblDrinkCostnotax.Text = Convert.ToString(drinkCostsnotax);
double cakeCostsnotax = (cofCake * cCake) + (redValvet * rValvet) + (bForest * bFor) +
(bostonCream * bCream) + (souffle * souf) + (kilbChoco * Choco) + (cheeseCak * cheese) + (rainbow *
rain);
lblCakeCostnotax.Text = Convert.ToString(cakeCostsnotax);
```

```
lblTotalNoTax.Text = Convert.ToString(cakeCostsnotax + drinkCostsnotax );
lblTax.Text = Convert.ToString(tax);
lblTotal.Text = Convert.ToString(cakeCosts + drinkCosts);
```

```
lblDrinkCost.Text = String.Format(culture, "{0:C}", drinkCosts);
lblCakeCost.Text = String.Format(culture, "{0:C}", cakeCosts);
lblTotalNoTax.Text = String.Format(culture, "{0:C}", (cakeCostsnotax + drinkCostsnotax ));
lblTax.Text = String.Format(culture, "{0:P0}", tax);
lblTotal.Text = String.Format(culture, "{0:C}", (cakeCosts + drinkCosts));
}
```

Миниспецификация №2. Кнопка “Чек”.

Миниспецификация предназначена для вывода на экран кассового чека.

Реализация миниспецификации программно:

```
private void button2_Click(object sender, EventArgs e)
{
    //btnReceipt
    rtfReceipt.Clear();

    // rtfReceipt.AppendText(Environment.NewLine);
    rtfReceipt.AppendText("\t\t" + "Кассовый чек" + Environment.NewLine);
    rtfReceipt.AppendText(Environment.NewLine + "ПРИХОД" + "\t\t" + Environment.NewLine);
    rtfReceipt.AppendText("-----" +
Environment.NewLine);
    rtfReceipt.AppendText("\t\t" + " Starbucks" + Environment.NewLine);
    rtfReceipt.AppendText("-----" +
Environment.NewLine);
    rtfReceipt.AppendText("Продукт" + "\t\t" + "Кол-во" + Environment.NewLine);
    rtfReceipt.AppendText("-----" +
Environment.NewLine);
    if (chkLatte.Checked == true)
    { rtfReceipt.AppendText("Латте \t\t\t" + txtLatte.Text + Environment.NewLine); }
    else
    { }
    if (chkEspresso.Checked == true)
    { rtfReceipt.AppendText("Эспрессо \t\t\t" + txtEspresso.Text + Environment.NewLine); }
    else
    { }
    if (chkMocha.Checked == true)
    { rtfReceipt.AppendText("Мокко \t\t\t\t" + txtMocha.Text + Environment.NewLine); }
    else
    { }
    if (chkRafCoffee.Checked == true)
    { rtfReceipt.AppendText("Раф \t\t\t\t" + txtRafCoffee.Text + Environment.NewLine); }
    else
    { }
    if (chkCappuccino.Checked == true)
    { rtfReceipt.AppendText("Капучино \t\t\t" + txtCappu.Text + Environment.NewLine); }
    else
    { }
    if (chkAfricanCoffe.Checked == true)
    { rtfReceipt.AppendText("Айс кофе \t\t\t" + txtAfricanCoffee.Text + Environment.NewLine); }
    else
    { }
    if (chkMilkTea.Checked == true)
    { rtfReceipt.AppendText("Молочный чай \t\t" + txtMilkTea.Text + Environment.NewLine); }
```

```

else
{ }
if (chkChineseTea.Checked == true)
{ rtfReceipt.AppendText("Китайский чай \t\t" + txtChineseTea.Text + Environment.NewLine); }
else
{ }
if (chkCoffe.Checked == true)
{ rtfReceipt.AppendText("Кофейный торт \t\t" + txtCoffeCake.Text + Environment.NewLine); }
else
{ }
if (chkRedValvet.Checked == true)
{ rtfReceipt.AppendText("Красный бархат \t\t" + txtRedValvetCake.Text +
Environment.NewLine); }
else
{ }
if (chkBlackForest.Checked == true)
{ rtfReceipt.AppendText("Черный лес \t\t\t" + txtBlackForestCake.Text + Environment.NewLine); }
}

else
{ }
if (chkBostonCream.Checked == true)
{ rtfReceipt.AppendText("Кремовый торт \t\t" + txtBostonCream.Text + Environment.NewLine); }
else
{ }
if (chkSouffle.Checked == true)
{ rtfReceipt.AppendText("Птичье молоко \t\t" + txtSouffleCake.Text + Environment.NewLine); }
else
{ }
if (chkKilburnChoco.Checked == true)
{ rtfReceipt.AppendText("Шоколадный торт \t\t" + txtKillburnChoco.Text +
Environment.NewLine); }
else
{ }
if (chkCheese.Checked == true)
{ rtfReceipt.AppendText("Чизкейк \t\t\t" + txtCheeseCake.Text + Environment.NewLine); }
else
{ }
if (chkRainbowCake.Checked == true)
{ rtfReceipt.AppendText("Радужный торт \t\t" + txtRainbowCake.Text + Environment.NewLine); }
else
{ }
rtfReceipt.AppendText("-----" +
Environment.NewLine);
rtfReceipt.AppendText("Итого \t\t\t\t=" + lblTotal.Text + Environment.NewLine);
rtfReceipt.AppendText("Сумма без НДС \t\t=" + lblTotalNoTax.Text + Environment.NewLine);
rtfReceipt.AppendText("-----" +
Environment.NewLine);

```

```
rtfReceipt.AppendText(lblTimer.Text + "\t\t" + lblDate.Text); }
```

Миниспецификация №3. Кнопка “Сброс”.

Миниспецификация предназначена для очистки всех заполненных данных.

Реализация миниспецификации программно:

```
private void button3_Click(object sender, EventArgs e)
```

```
{ //btnReset
```

```
txtLatte.Text = "0";
```

```
txtEspresso.Text = "0";
```

```
txtMocha.Text = "0";
```

```
txtRafCoffee.Text = "0";
```

```
txtCappu.Text = "0";
```

```
txtAfricanCoffee.Text = "0";
```

```
txtMilkTea.Text = "0";
```

```
txtChineseTea.Text = "0";
```

```
txtCoffeCake.Text = "0";
```

```
txtRedVelvetCake.Text = "0";
```

```
txtBlackForestCake.Text = "0";
```

```
txtBostonCream.Text = "0";
```

```
txtSouffleCake.Text = "0";
```

```
txtKillburnChoco.Text = "0";
```

```
txtCheeseCake.Text = "0";
```

```
txtRainbowCake.Text = "0";
```

```
lblCakeCost.Text = "0";
```

```
lblDrinkCost.Text = "0";
```

```
lblCakeCostnotax.Text = "0";
```

```
lblDrinkCostnotax.Text = "0";
```

```
lblTax.Text = "0";
```

```
lblTotalNoTax.Text = "0";
```

```
lblTotal.Text = "0";
```

```
chkLatte.Checked = false;
```

```
chkEspresso.Checked = false;
```

```
chkMocha.Checked = false;
```

```
chkRafCoffee.Checked = false;
```

```
chkCappucino.Checked = false;
```

```
chkAfricanCoffe.Checked = false;
```

```
chkMilkTea.Checked = false;
```

```
chkChineseTea.Checked = false;
```

```
chkRedVelvet.Checked = false;
```

```
chkCoffe.Checked = false;
```

```
chkBlackForest.Checked = false;
```

```
chkBostonCream.Checked = false;
```

```
chkSouffle.Checked = false;
```

```
chkKilburnChoco.Checked = false;  
chkCheese.Checked = false;  
chkRainbowCake.Checked = false;  
}
```

Миниспецификация №4. Кнопка “Выход”.

Миниспецификация предназначена для выхода из программы.

Реализация миниспецификации программно:

```
private void btnExit_Click(object sender, EventArgs e)  
{  
    Application.Exit();  
    double latteCof = Convert.ToDouble(txtLatte.Text);  
}
```

Миниспецификация №5. Кнопка “Сохранить чек”.

Миниспецификация предназначена для сохранения чека в отдельный файл формата .txt.

Реализация миниспецификации программно:

```
private void saveToolStripButton_Click(object sender, EventArgs e)  
{  
    //this code will save text files  
    SaveFileDialog saveFile = new SaveFileDialog();  
  
    saveFile.FileName = "Notepad Text";  
    saveFile.Filter = "Text Files (*.txt) | *.txt | All files (*.*) | *.*";  
  
    if (saveFile.ShowDialog() == System.Windows.Forms.DialogResult.OK)  
    {  
        using (System.IO.StreamWriter sw = new System.IO.StreamWriter(saveFile.FileName))  
            sw.WriteLine(rtfReceipt.Text);  
    }  
}
```