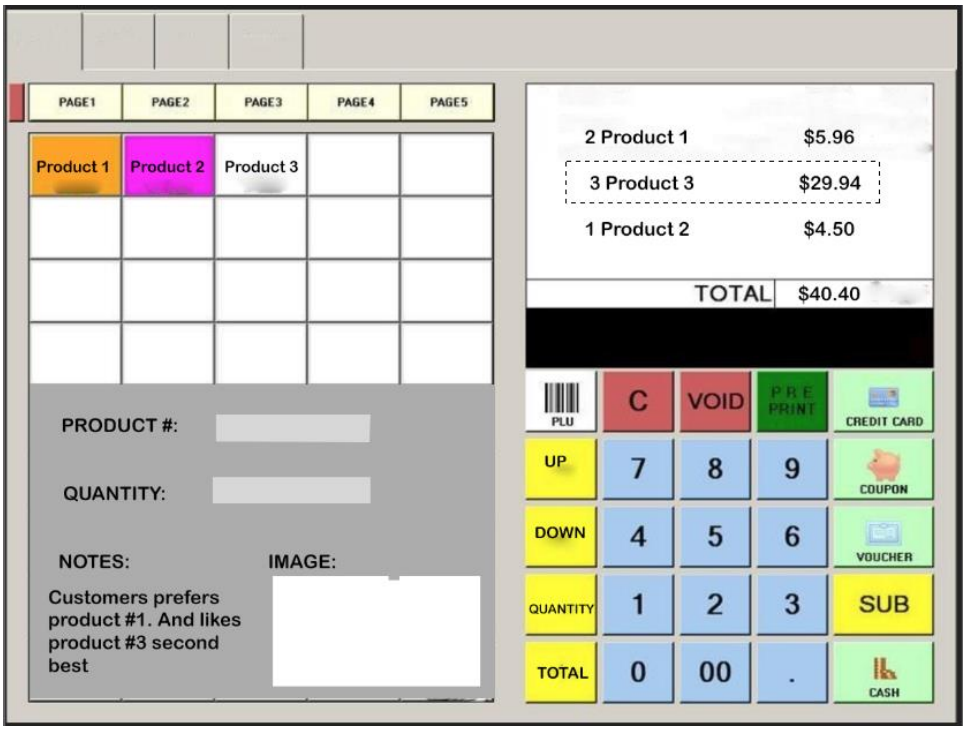
Requirements:

1. Prompt the user for the product number (1,2, or 3)
2. Prompt the user for the quantity sold (use value)
3. Use a Switch Statement
4. Calculate the retail value of each product per the number sold
5. Calculate and display the total retail of all products sold (total)
6. Select the product number which as the largest sold quantity. This is the largest total per each individual product as a separate button and label.

Initial Design



Pseudocode

Initialize product number, quantity sold.

List of products and prices.

Prompt the user for product number

Prompt the user for quantity number

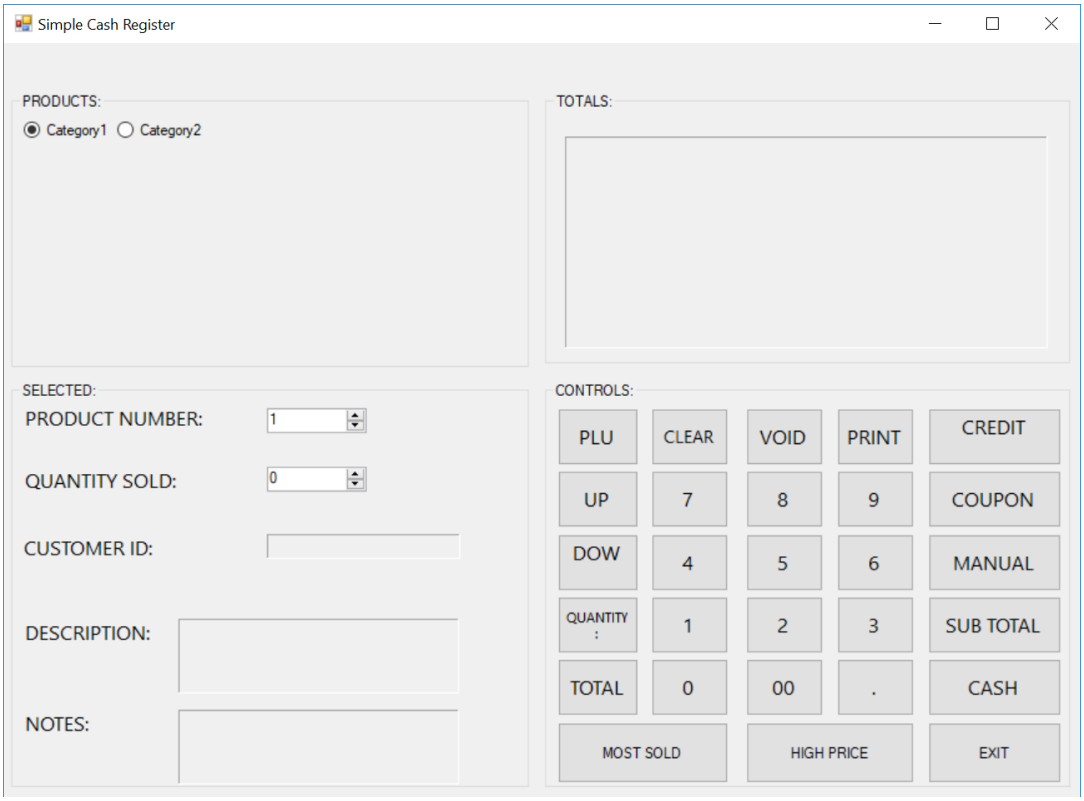
Switch when subtotal is pressed

Calculate subtotal Based on the product number multiply quantity and add with subtotal button.

If total button is pressed all subtotals will be added and displayed.

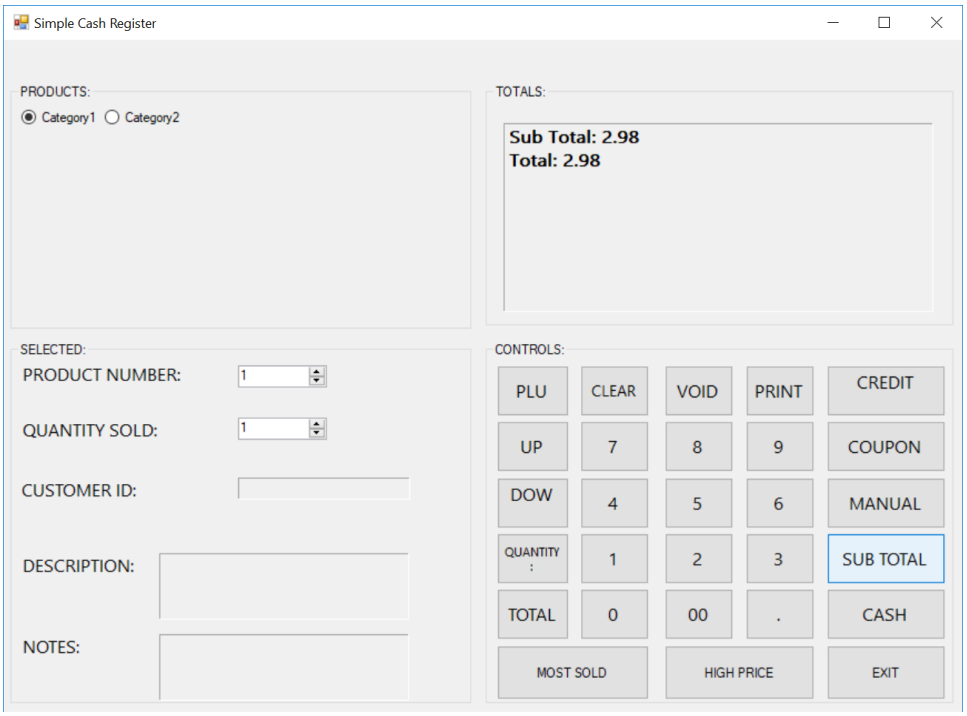
Product with the highest subtotal will be highlighted somehow.

Final Windows Form Design

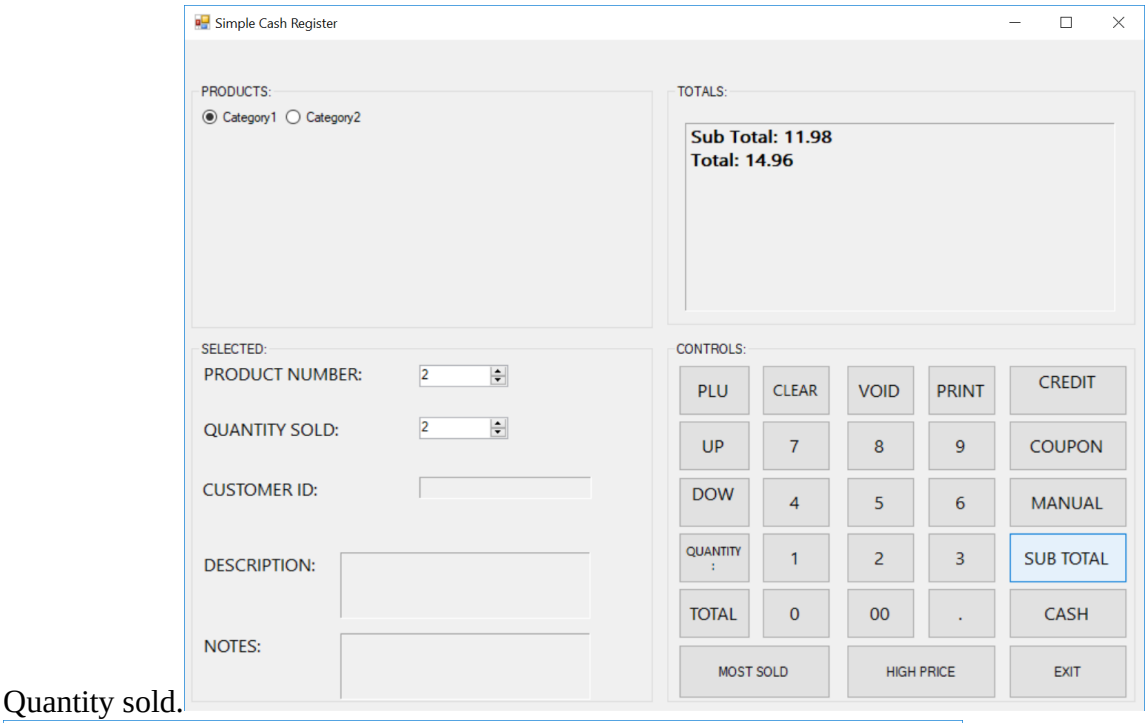


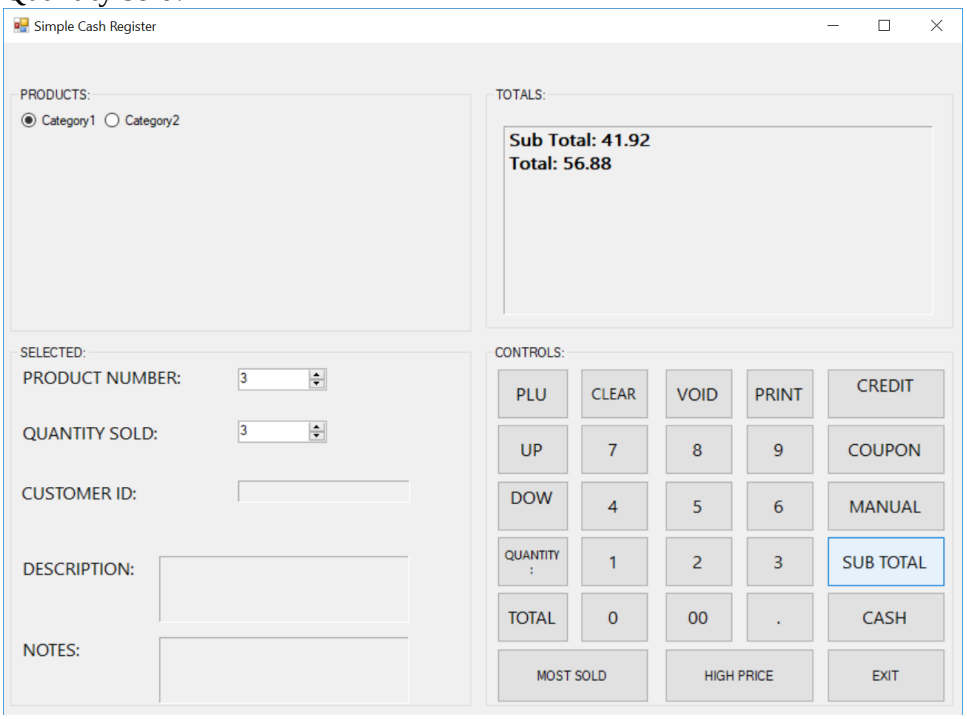
That application reads a series of pairs of numbers using a numericUpDown value:

1. Product number.



2. Quantity sold.



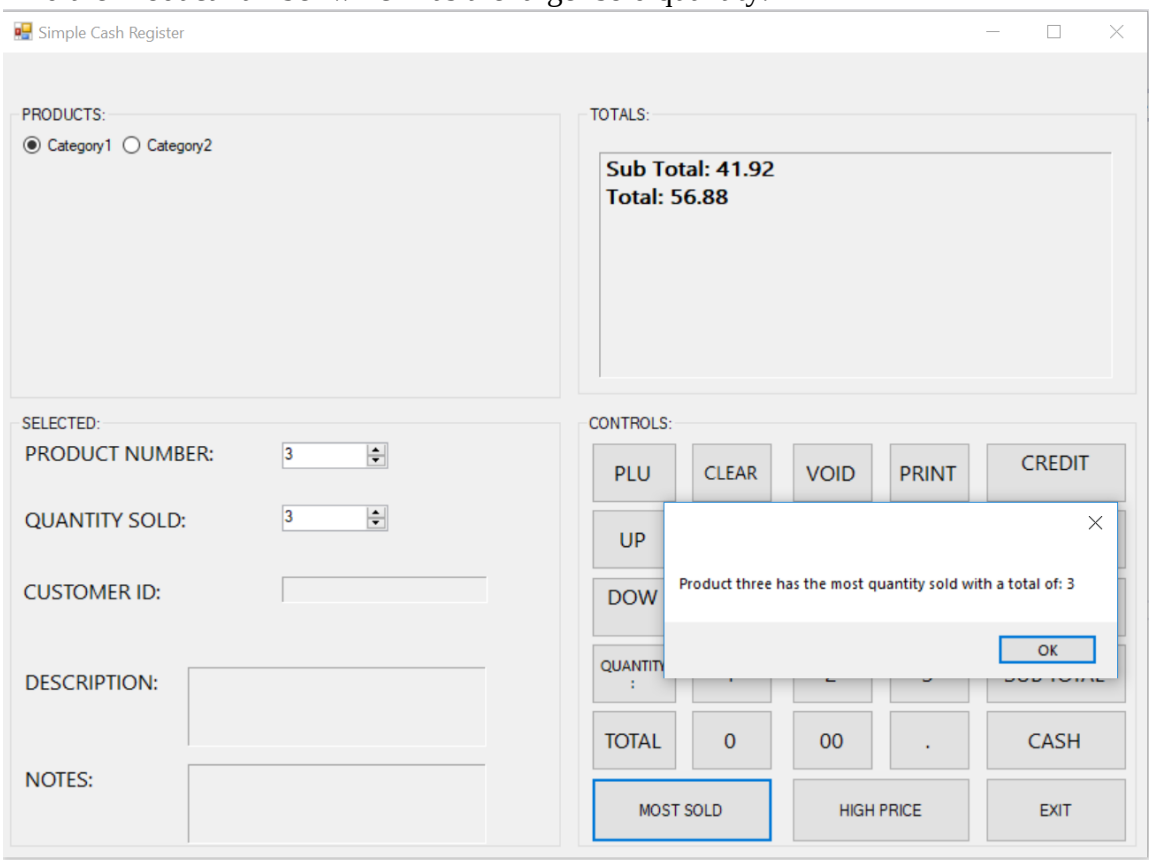


In the totals panel you notice a running subtotal and total are updated on every click.

Your app should use a switch statement to determine the retail price for each product. It should:

a. Calculate and display the total retail value of all products sold.

b. Find the Product number which has the larger sold quantity.



Code

checkout.cs | checkout.cs [Design]

Super Simple Cash Register | Super_Simple_Cash_Register.checkout | checkout_FormClosing(object sender, FormClosin

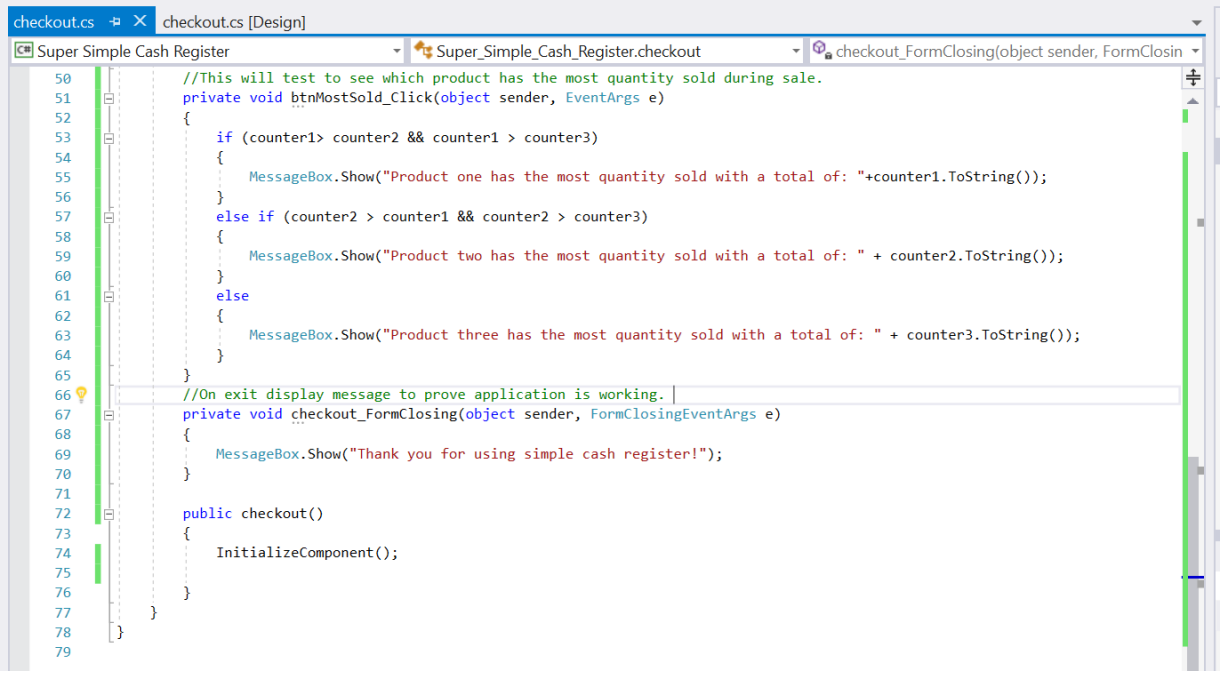
```
using System;
using System.Windows.Forms;

namespace Super_Simple_Cash_Register
{
    public partial class checkout : Form
    {
        //global variables for calculations of subtotal, total, and most sold.
        int productNumber = 0;
        int quantitySold = 0;
        double total = 0.0, subTotal = 0.0;
        int counter1 = 0;
        int counter2 = 0;
        int counter3 = 0;

        private void btnSubTotal_Click(object sender, EventArgs e)
        {
            // Getting Value from the user using up down value to make sure to validate input
            productNumber = Convert.ToInt32(nmUpDownProduct.Value);
            quantitySold = Convert.ToInt32(nmUpDownQuantity.Value);

            //Every time sub total button is pressed that sale will be added to subtotal.
            //Then subtotal will be added to total.
            //Counter is used to keep track of the quantity of each product.
            switch (productNumber)
            {
                case 1:
                    const double price = 2.98;
                    subTotal = subTotal + (price * quantitySold);
                    total = total + subTotal;
                    lblTotals.Text = "Sub Total: " + subTotal.ToString() + "\nTotal: " + total.ToString();
                    counter1= counter1 +quantitySold;
                    break;
                case 2:
                    const double price2 = 4.50;
                    subTotal = subTotal + (price2 * quantitySold);
                    total = total + subTotal;
                    lblTotals.Text = "Sub Total: " + subTotal.ToString() + "\nTotal: " + total.ToString();
                    counter2 = counter2 + quantitySold;
                    break;
                case 3:
                    const double price3 = 9.98;
                    subTotal = subTotal + (price3 * quantitySold);
                    total = total + subTotal;
                    lblTotals.Text = "Sub Total: " + subTotal.ToString() + "\nTotal: " + total.ToString();
                    counter3 = counter3 + quantitySold;
                    break;
            }
        }
        //This will test to see which product has the most quantity sold during sale
```

87 %




```
50          //This will test to see which product has the most quantity sold during sale.
51          private void btnMostSold_Click(object sender, EventArgs e)
52          {
53              if (counter1> counter2 && counter1 > counter3)
54              {
55                  MessageBox.Show("Product one has the most quantity sold with a total of: "+counter1.ToString());
56              }
57              else if (counter2 > counter1 && counter2 > counter3)
58              {
59                  MessageBox.Show("Product two has the most quantity sold with a total of: " + counter2.ToString());
60              }
61              else
62              {
63                  MessageBox.Show("Product three has the most quantity sold with a total of: " + counter3.ToString());
64              }
65          }
66          //On exit display message to prove application is working.
67          private void checkout_FormClosing(object sender, FormClosingEventArgs e)
68          {
69              MessageBox.Show("Thank you for using simple cash register!");
70          }
71
72          public checkout()
73          {
74              InitializeComponent();
75
76          }
77      }
78  }
79
```