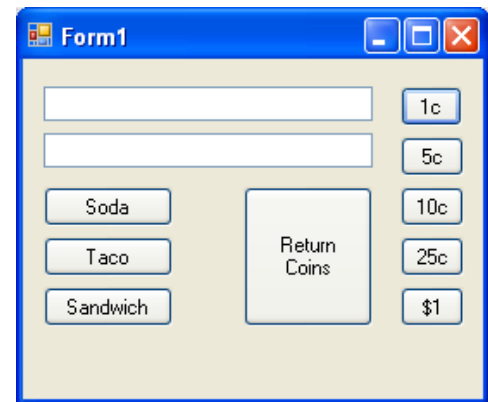


This documentation outlines some steps for button control using an example of “vending machine”.

GUI Layout

The layout of the GUI is shown on the right. Please follow this layout closely.

Refer to tutorial for how to set up the Windows Application environment.

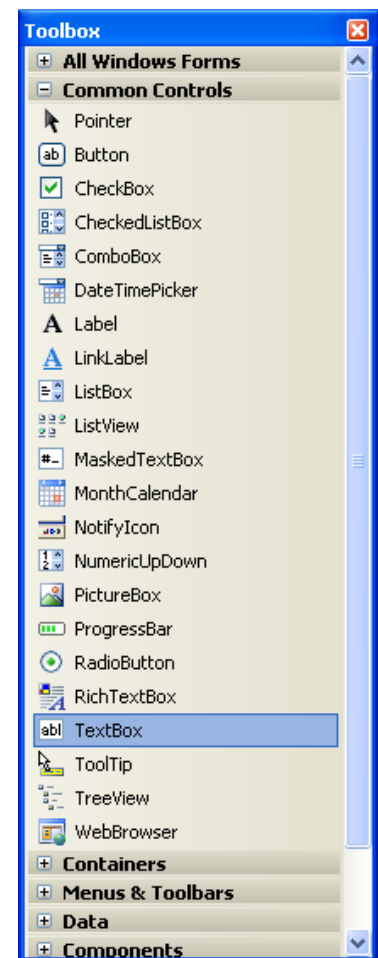
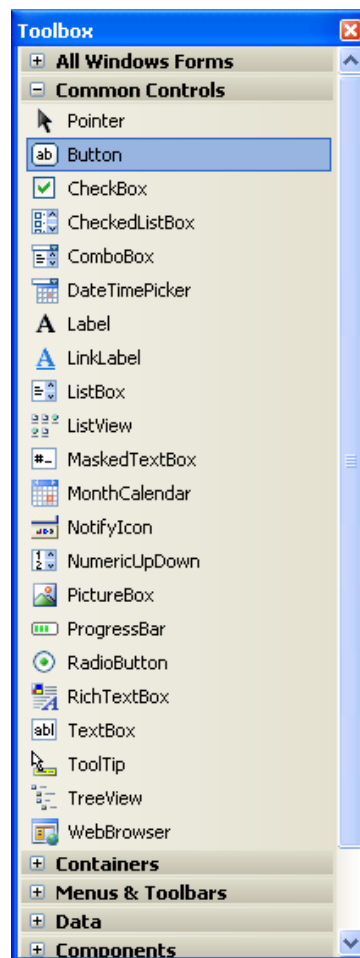


Button and TextBox

Roll over the mouse to the Toolbox tab, the toolbox similar to the right appears.

Button and TextBox are used in this assignment and their locations in toolbox are shown in the figure.

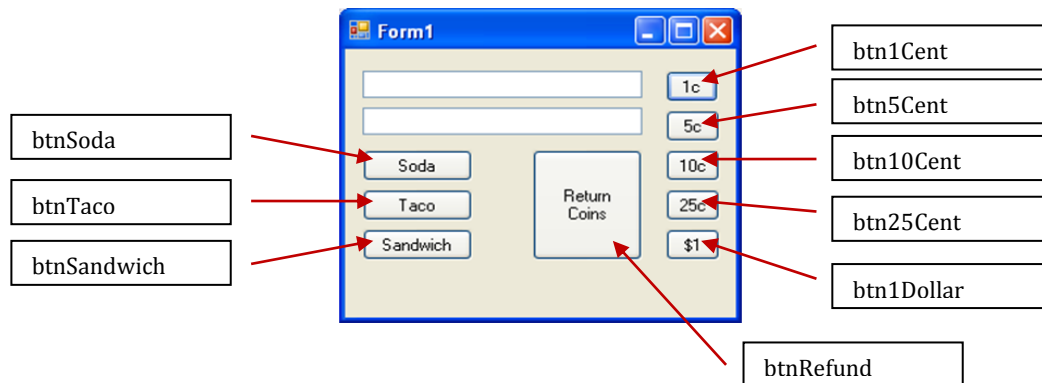
Drag and drop the Button and TextBox onto the form. Resize and position them accordingly.



Button Properties

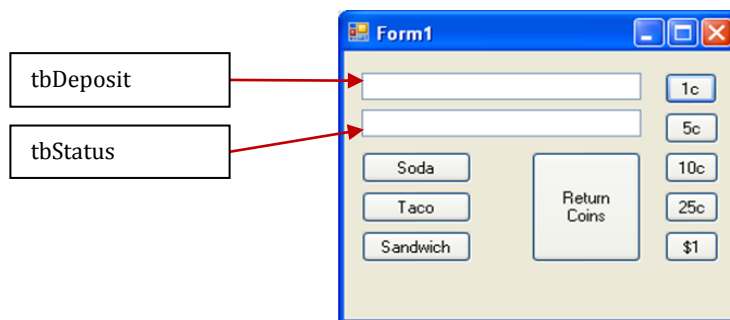
Before adding code to the objects (button and textbox) we just created, we need to rename the button. There are two fields for naming; one is for the Appearance and one is for the Design.

The **Text** field under Appearance specifies what we see in the GUI. The **(Name)** field under Design declared the variable (handler) we use in the code. Right-click on the Button object and choose **Properties**. Please use the following variables and texts for this assignment (case sensitive).



TextBox Properties

Similar to the procedures above, variables are declared.



Form Properties

Rename the **Form1** to **Vending Machine**, the variable name remained unchanged.

The location of Text and (Name) in the Properties windows are shown below.

Properties Form1 System.Windows.Forms.Form

Accessibility

Appearance

BackColor Control

BackgroundImage (none)

BackgroundImageLayout Tile

Cursor Default

Font Microsoft Sans Serif, 8.25pt

ForeColor ControlText

FormBorderStyle Sizable

RightToLeft No

RightToLeftLayout False

Text **Vending Machine**

UseWaitCursor False

Behavior

Data

Design

(Name) **Form1**

Language (Default)

Localizable False

Locked False

Focus

CausesValidation True

Layout

AutoScaleMode **Font**

AutoScroll False

AutoScrollMargin 0, 0

AutoScrollMinSize 0, 0

AutoSize False

AutoSizeMode GrowOnly

Location 0, 0

MaximumSize 0, 0

MinimumSize 0, 0

Padding 0, 0, 0, 0

Size **275, 231**

Text

The text associated with the control.

Properties Form1 System.Windows.Forms.Form

Accessibility

Appearance

BackColor Control

BackgroundImage (none)

BackgroundImageLayout Tile

Cursor Default

Font Microsoft Sans Serif, 8.25pt

ForeColor ControlText

FormBorderStyle Sizable

RightToLeft No

RightToLeftLayout False

Text **Vending Machine**

UseWaitCursor False

Behavior

Data

Design

(Name) **Form1**

Language (Default)

Localizable False

Locked False

Focus

CausesValidation True

Layout

AutoScaleMode **Font**

AutoScroll False

AutoScrollMargin 0, 0

AutoScrollMinSize 0, 0

AutoSize False

AutoSizeMode GrowOnly

Location 0, 0

MaximumSize 0, 0

MinimumSize 0, 0

Padding 0, 0, 0, 0

Size **275, 231**

(Name)

Indicates the name used in code to identify the object.

Writing Codes

After the naming procedure, we add codes to make the application running the way we want.

The code listing below shows where to place the variables such as deposit, change. We can treat them as global variables and can be accessed anywhere within the class Form1. Notice the suffix F for the constant in type float.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace gui_ex01
{
    public partial class Form1 : Form
    {
        public float deposit = 0;
        public float change;
        public float item1 = 1.25F;
        public float item2 = 1.75F;
        public float item3 = 2.25F;

        public Form1()
        {
            InitializeComponent();
        }

        ...
    }
}
```

To add code for the button such as depositing coins, double-click on the Button and the skeleton code will be generated for you. All you need to do is to write the code to perform the intended action, for example, accumulating the variable **deposit**.

Display Message

To show message in the textbox, we can use the follow syntax.

```
tbDeposit.Text = "Deposit: " + deposit.ToString();
```

Variable tbDeposit refers to the variable name we declared earlier. To show number on the textbox, we can use ToString() method, follow the syntax above. The operator + is used for concatenating two strings.

Other Syntax

The rest of the program is very similar to C language. Condition statement, case statement, for-loop and many others are the same as ANSI C.