

2019 年 4 月 25 日 (木) 実施

フォームのデザイン

本日の課題

フォームに様々なコントロールを配置して、その機能を学ぶ。

手順

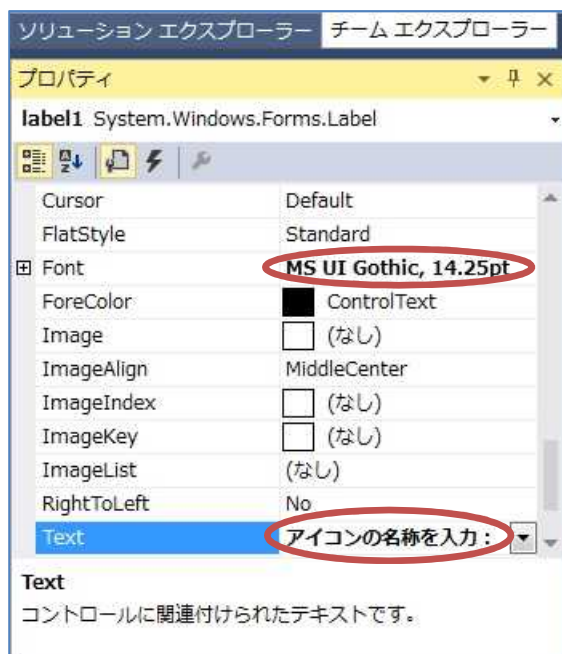
1) プロジェクトの作成

Visual Studio 2013 を起動したら、[ファイル] → [新規作成] → [プロジェクト] と辿って、プロジェクトを作成する。『新しいプロジェクト』ダイアログボックスでは、プログラミング言語を『Visual C#』、プロジェクトテンプレートとしては、『Windows フォームアプリケーション』を選択し、『名前』を「Third」に書き換え、『場所』が「H:¥Documents¥Visual Studio 2013¥Projects」となっていることを確認してから『OK』を押す（詳細は第 1 回の教材を参照）。

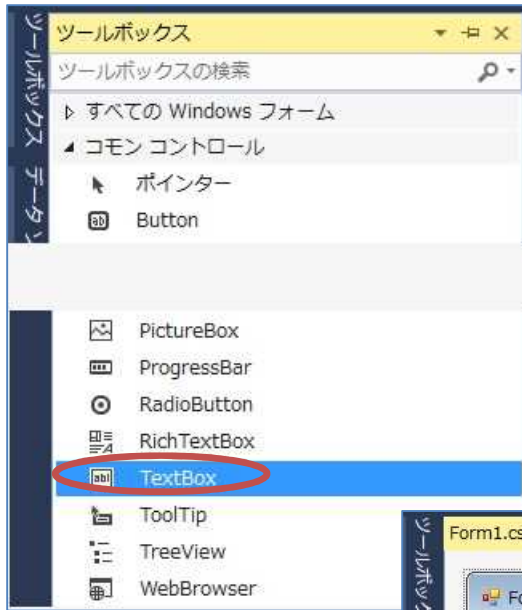


2) コントロールの配置

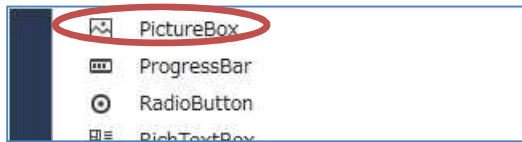
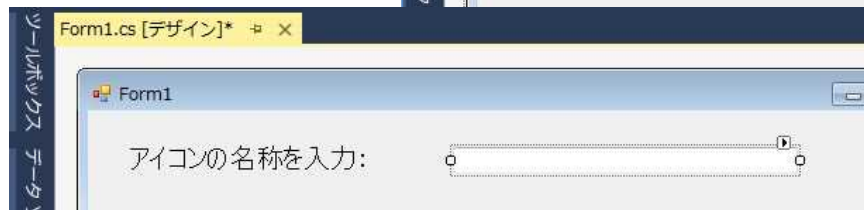
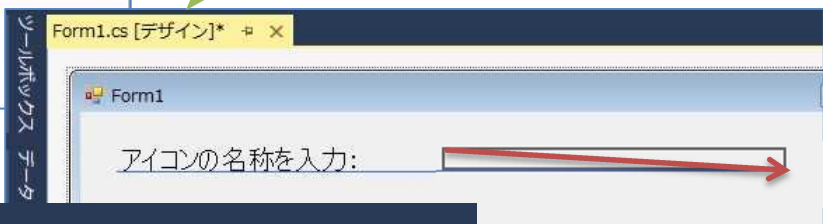
フォームを幅、高さ共に広げる。[表示] → [ツールボックス] と辿って、ツールボックスを表示する。『コモンコントロール』を選択して展開する。『Label1』を選択してから、左にある縦の『ツールボックス』タブをクリックして、メニューを引っ込める。フォームをクリックしてラベルを貼る（『label1』の名前が付く）。フォーム上の『label1』を選択して、プロパティのフォントサイズを 14 ポイントに変更する。（ここまでの詳細は、第 1 回及び第 2 回の教材を参照）



プロパティの『Text』を「label1」から「アイコンの名称を入力:」に書き換える。

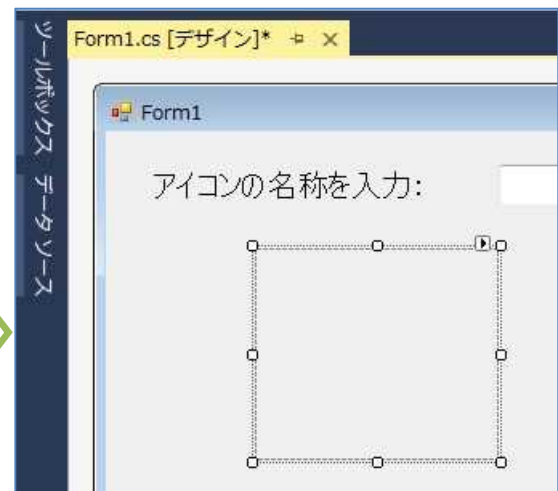
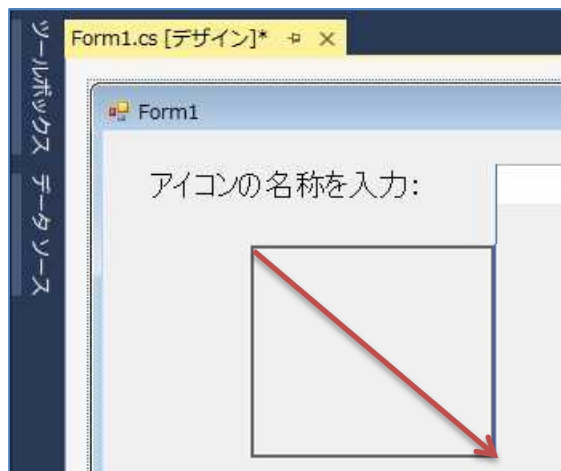


ツールボックスで『TextBox』を選択してから、左にある縦の『ツールボックス』タブをクリックして、メニューを引っ込める。フォーム上でドラッグしてテキストボックスを貼る (『textBox1』の名前が付く)。プロパティのフォントサイズを 14 ポイントに変更する。

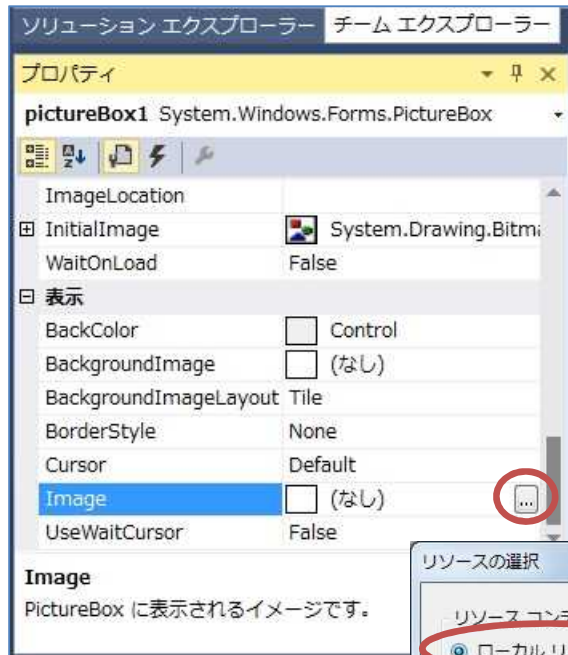


ツールボックスで『PictureBox』を選択してから、左にある縦の『ツールボックス』タブをクリックして、メニューを引っ込める。フォーム上でドラッグしてピクチャーボックスを貼る (『pictureBox1』の名前が付く)。

ピクチャーボックスは、画像を表示するためのコントロールである。



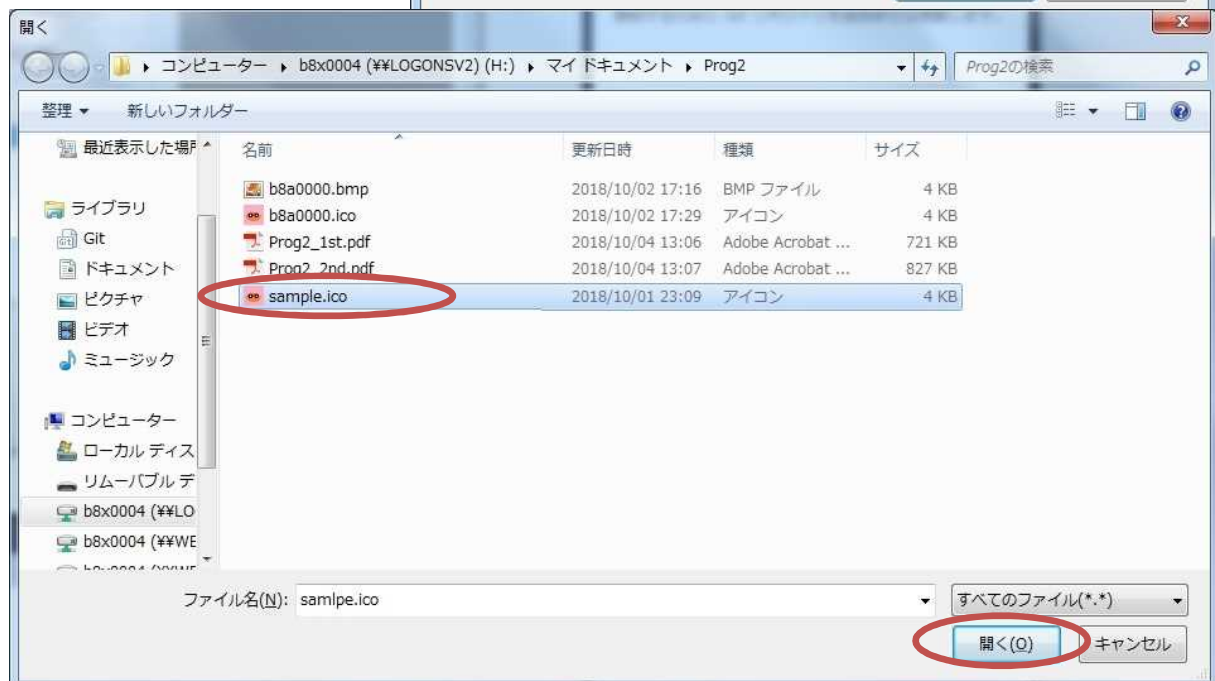
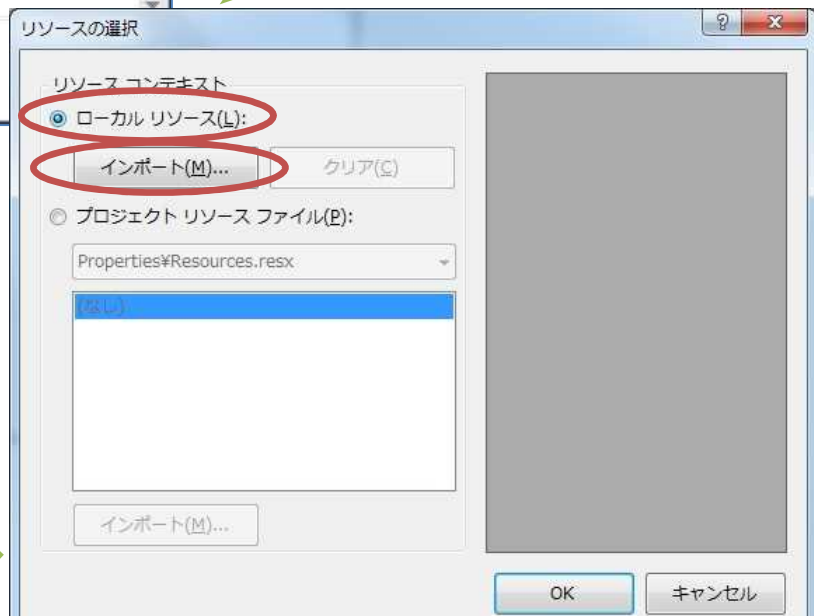
* 教材集から『sample.ico』を『Prog2』フォルダにダウンロードしておくこと。



プロパティの『Image』の右側にある『…』ボタンを押す。

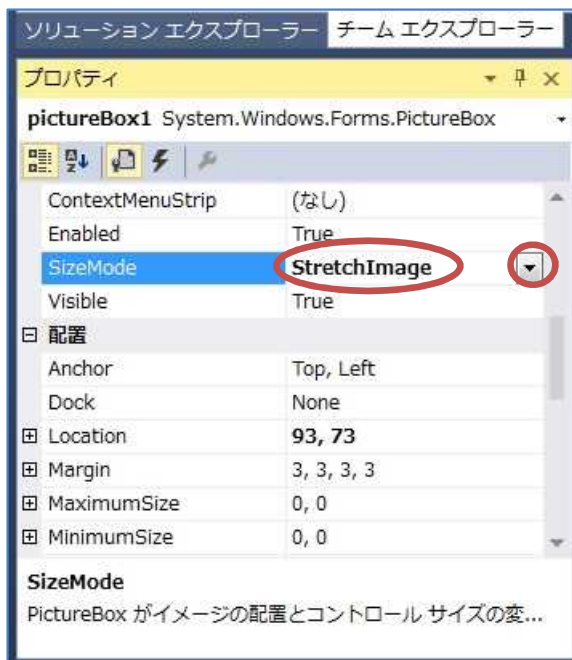
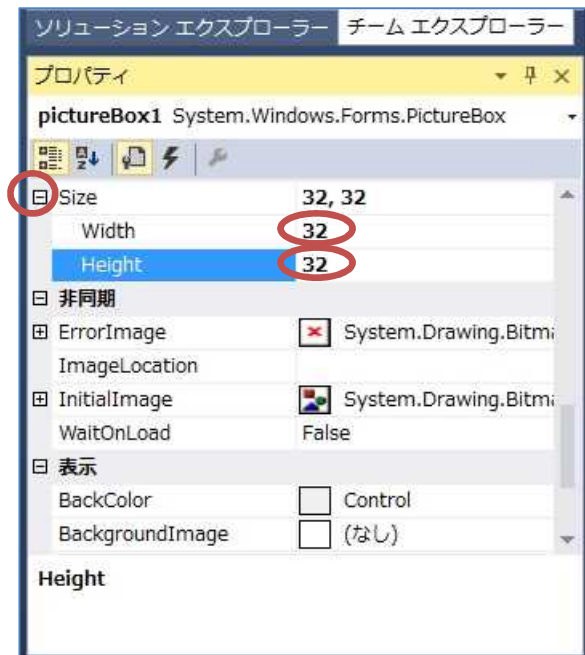
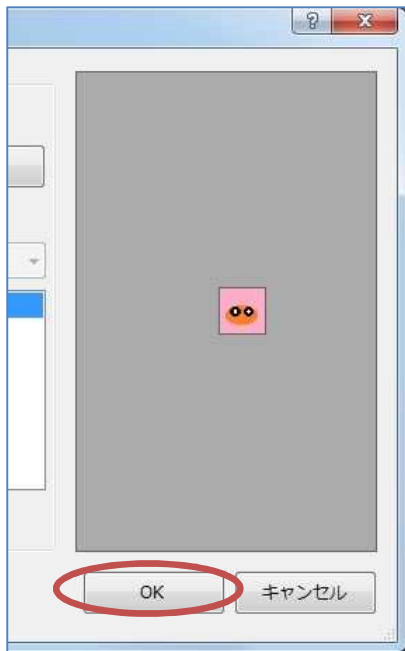
『リソースの選択』ダイアログボックスでは『ローカルリソース』を選択し、『インポート』を押す。

『Prog2』フォルダから『sample.ico』を選択して、『開く』を押す。



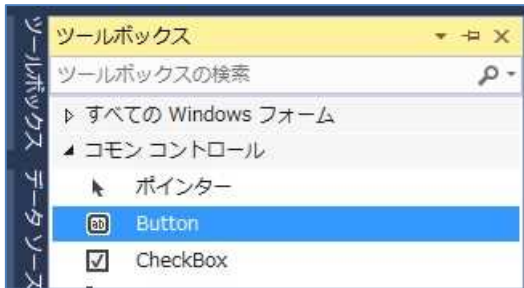
プレビューを確認して『OK』を押す。

プロパティの『Size』を展開して、『Width』及び『Height』の値を「32」に書き換える。



プロパティの『SizeMode』で『StretchImage』を選択する（これで、画像がピクチャーボックスのサイズ一杯に広がる）。





ツールボックスで『Button』を選択してから、左にある縦の『ツールボックス』タブをクリックして、メニューを引っ込める。フォームをクリックしてボタンを貼る (『button1』の名前が付く)。

同様な操作を繰り返して、『button2』及び『button3』をフォームに貼る。

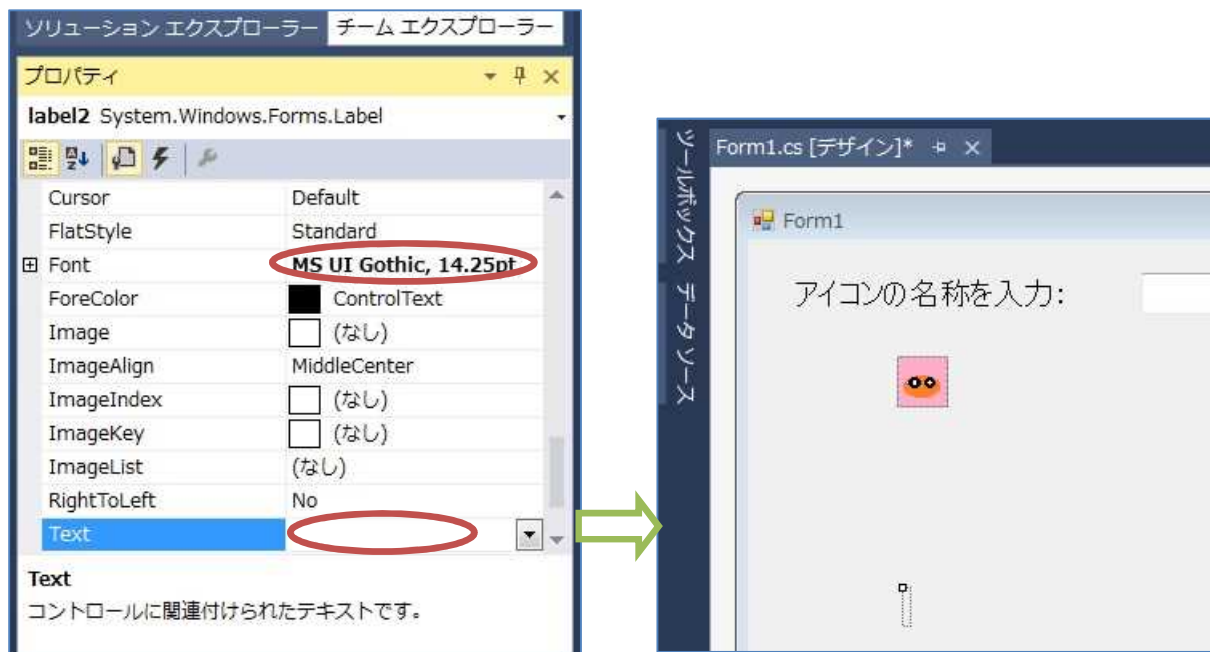


『button1』のプロパティのフォントサイズを 14 ポイントに変更し、プロパティの『Text』を「button1」から「2 倍」に書き換える。



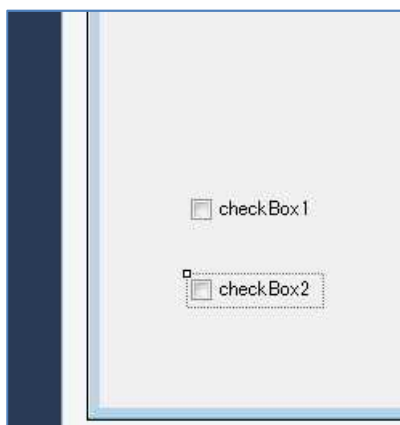
同様に、『button2』及び『button3』のプロパティのフォントサイズを 14 ポイントに変更し、プロパティの『Text』を「button2」、「button3」からそれぞれ「3 倍」、「4 倍」に書き換える。

『label1』と同様な操作で『label2』をフォームに貼り、プロパティのフォントサイズを 14 ポイントに変更する。プロパティの『Text』に書かれている「label2」の文字列を削除する。



ツールボックスで『CheckBox』を選択してから、左にある縦の『ツールボックス』タブをクリックして、メニューを引っ込める。フォームをクリックしてチェックボックスを貼る (『checkBox1』の名前が付く)。

同様の操作を繰り返して、『checkBox2』をフォームに貼る。

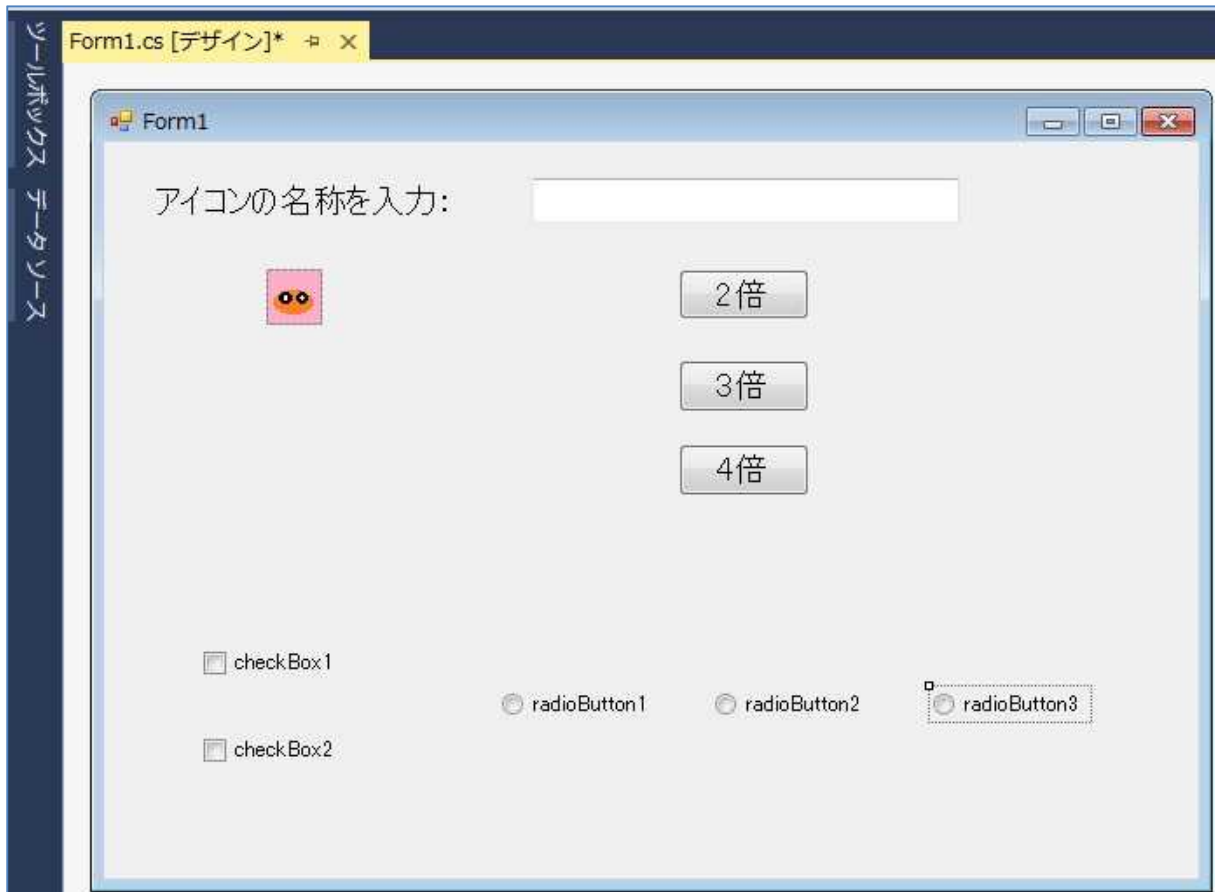


チェックボックスは、クリックしてチェックを入れたり外したりすることで、選択の ON/OFF を切り替えられるコントロールである。



ツールボックスで『RadioButton』を選択してから、左にある縦の『ツールボックス』タブをクリックして、メニューを引っ込める。フォームをクリックしてラジオボタンを貼る（『radioButton1』の名前が付く）。

同様な操作を繰り返して、『radioButton2』及び『radioButton3』をフォームに貼る。



ラジオボタンは、グループの中で一つだけがクリックによって選択可能なコントロールである。

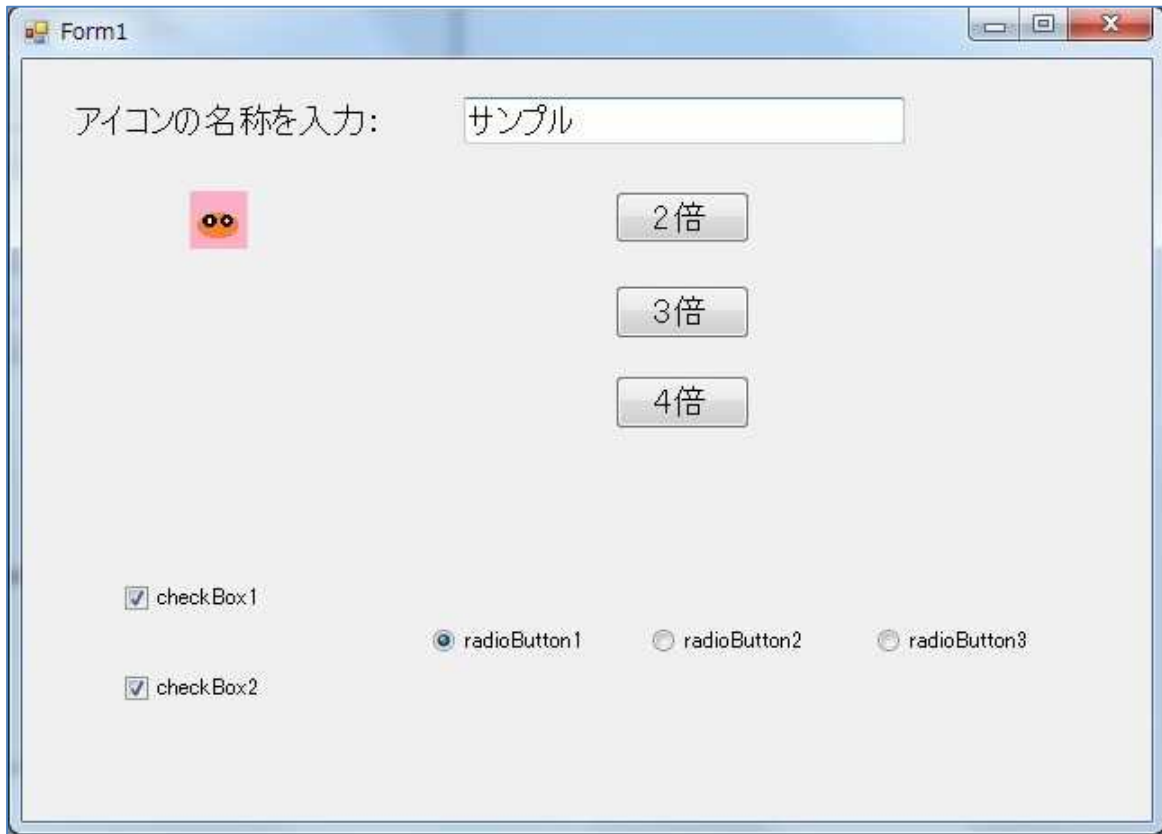
3) プログラムの実行

ここで、『すべてを保存』のボタンを押した後に、『開始』ボタンを押してプログラムを起動してみる（詳細は第 1 回の教材を参照）。

エラーが出ている場合には、修正してから保存、開始と進む。

それぞれのコントロールが機能していることを確認し終わったら、閉じるボタンを押してプログラムを終了する。

(図は次のページ)



4) コーディング

テキストボックスをダブルクリックして、Form1.cs のプログラムのソースコードを表示する。`textBox1_TextChanged` メソッドのブロック内にテキストボックスの文字列が変更された際の処理（赤枠の部分）を記述していく。

```
private void textBox1_TextChanged(object sender, EventArgs e)
{
    label2.Text = textBox1.Text + "ちゃん";
}
```

ここでは、テキストボックスの `Text` プロパティの値に文字列を追加して、ラベルの `Text` プロパティの値として設定している。

入力し終わったら、『すべてを保存』のボタンを押す。

(コードエディタの図は次のページ)


```

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 Third
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

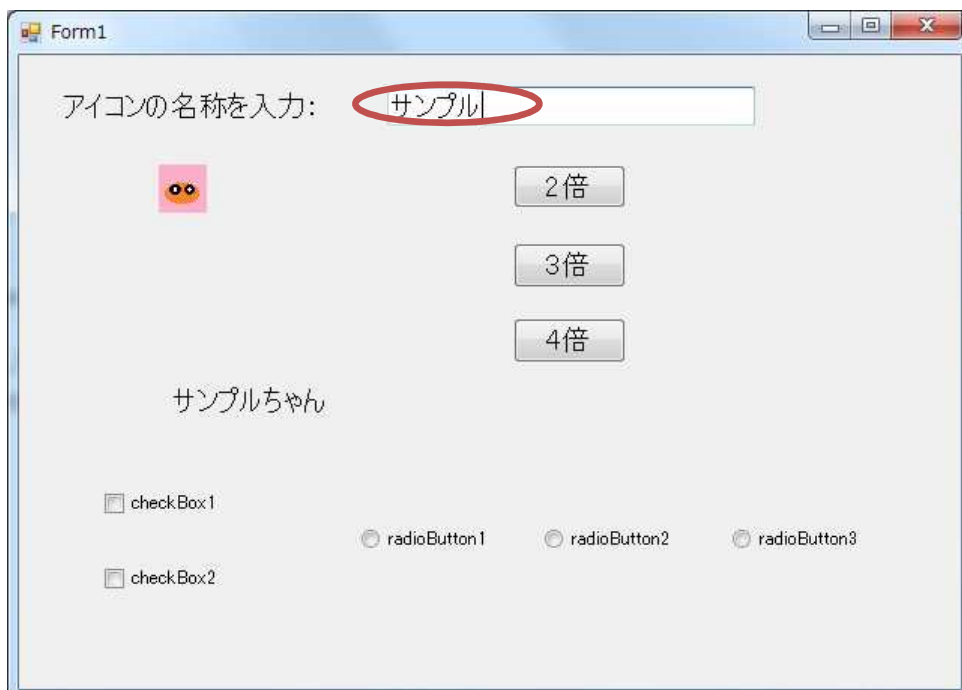
        private void textBox1_TextChanged(object sender, EventArgs e)
        {
            label2.Text = textBox1.Text + "ちゃん";
        }
    }
}
    
```

5) プログラムの2度目の実行

ここで、再度、『開始』ボタンを押してプログラムを起動してみる。

エラーが出ている場合には、修正してから保存、開始と進む。

テキストボックスにアイコンの愛称の文字列「〇〇」を入力して、ピクチャーボックスの下に「〇〇ちゃん」と表示されることを確かめたら、閉じるボタンを押してプログラムを終了する。



フォームデザイナーに戻り、フォーム上の『button1』をダブルクリックして、コードエディタによって、button1_Click メソッドのブロック内に「2 倍」と書かれたボタンがクリックされた際の処理（赤枠の部分）を記述していく。

```
private void button1_Click(object sender, EventArgs e)
{
    pictureBox1.Width = 32;
    pictureBox1.Height = 32;
    pictureBox1.Width = pictureBox1.Width * 2;
    pictureBox1.Height = pictureBox1.Height * 2;
}
```

ここでは、ピクチャーボックスの Width プロパティ及び Height プロパティの値を 2 倍して、Width プロパティ及び Height プロパティを再設定している。初めに Width プロパティ及び Height プロパティを「32」と設定し直しているのは、一度何れかのボタンを押した後は値が初期値とは変わるからである。

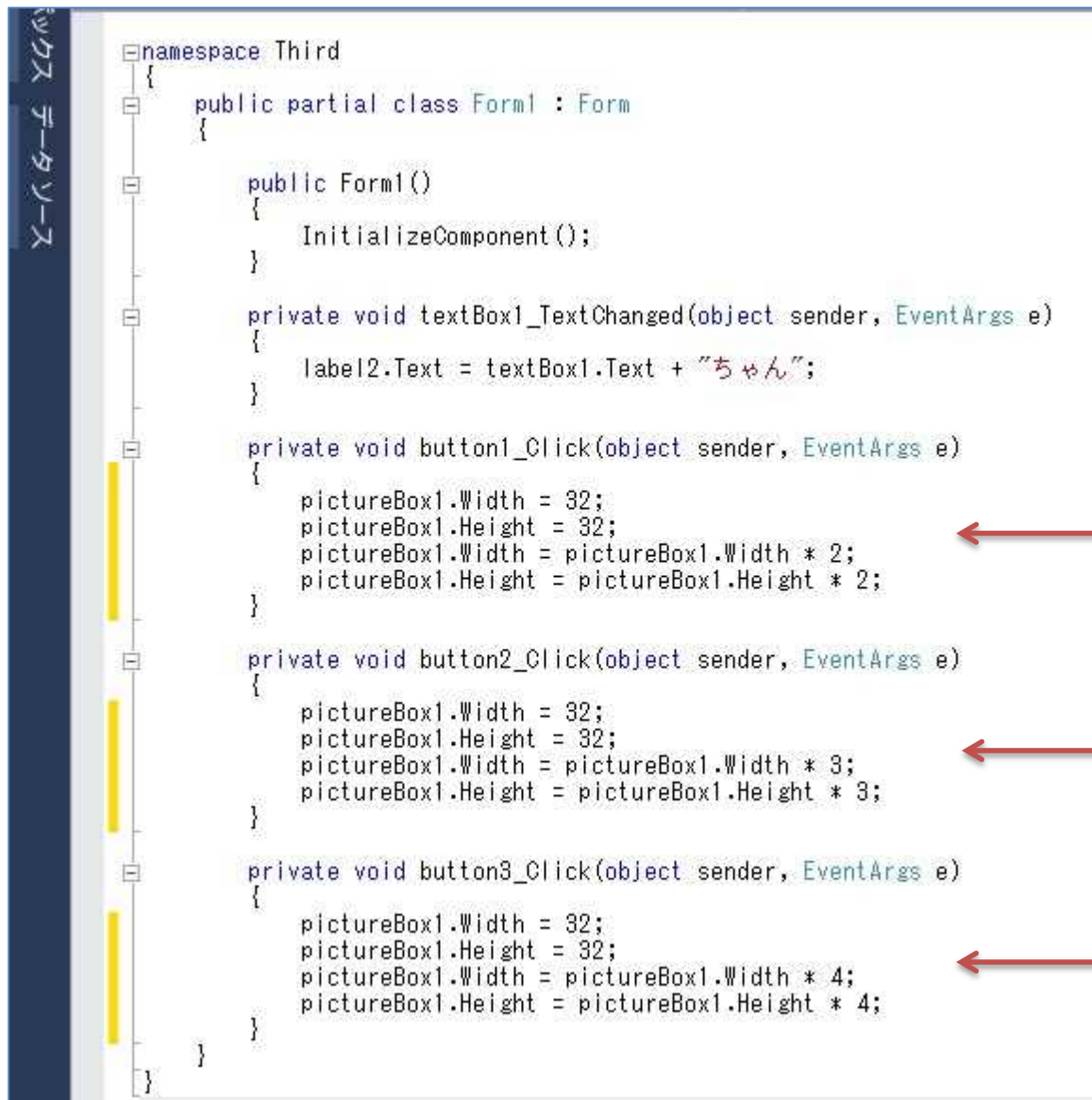
フォームデザイナーに戻り、フォーム上の『button2』をダブルクリックして、コードエディタによって、button2_Click メソッドのブロック内に「3 倍」と書かれたボタンがクリックされた際の処理（赤枠の部分）を記述していく。

```
private void button2_Click(object sender, EventArgs e)
{
    pictureBox1.Width = 32;
    pictureBox1.Height = 32;
    pictureBox1.Width = pictureBox1.Width * 3;
    pictureBox1.Height = pictureBox1.Height * 3;
}
```

フォームデザイナーに戻り、フォーム上の『button3』をダブルクリックして、コードエディタによって、button3_Click メソッドのブロック内に「4 倍」と書かれたボタンがクリックされた際の処理（赤枠の部分）を記述していく。

```
private void button3_Click(object sender, EventArgs e)
{
    pictureBox1.Width = 32;
    pictureBox1.Height = 32;
    pictureBox1.Width = pictureBox1.Width * 4;
    pictureBox1.Height = pictureBox1.Height * 4;
}
```

(コードエディタの図は次のページ)



```
namespace Third
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void textBox1_TextChanged(object sender, EventArgs e)
        {
            label2.Text = textBox1.Text + "ちゃん";
        }

        private void button1_Click(object sender, EventArgs e)
        {
            pictureBox1.Width = 32;
            pictureBox1.Height = 32;
            pictureBox1.Width = pictureBox1.Width * 2;
            pictureBox1.Height = pictureBox1.Height * 2;
        }

        private void button2_Click(object sender, EventArgs e)
        {
            pictureBox1.Width = 32;
            pictureBox1.Height = 32;
            pictureBox1.Width = pictureBox1.Width * 3;
            pictureBox1.Height = pictureBox1.Height * 3;
        }

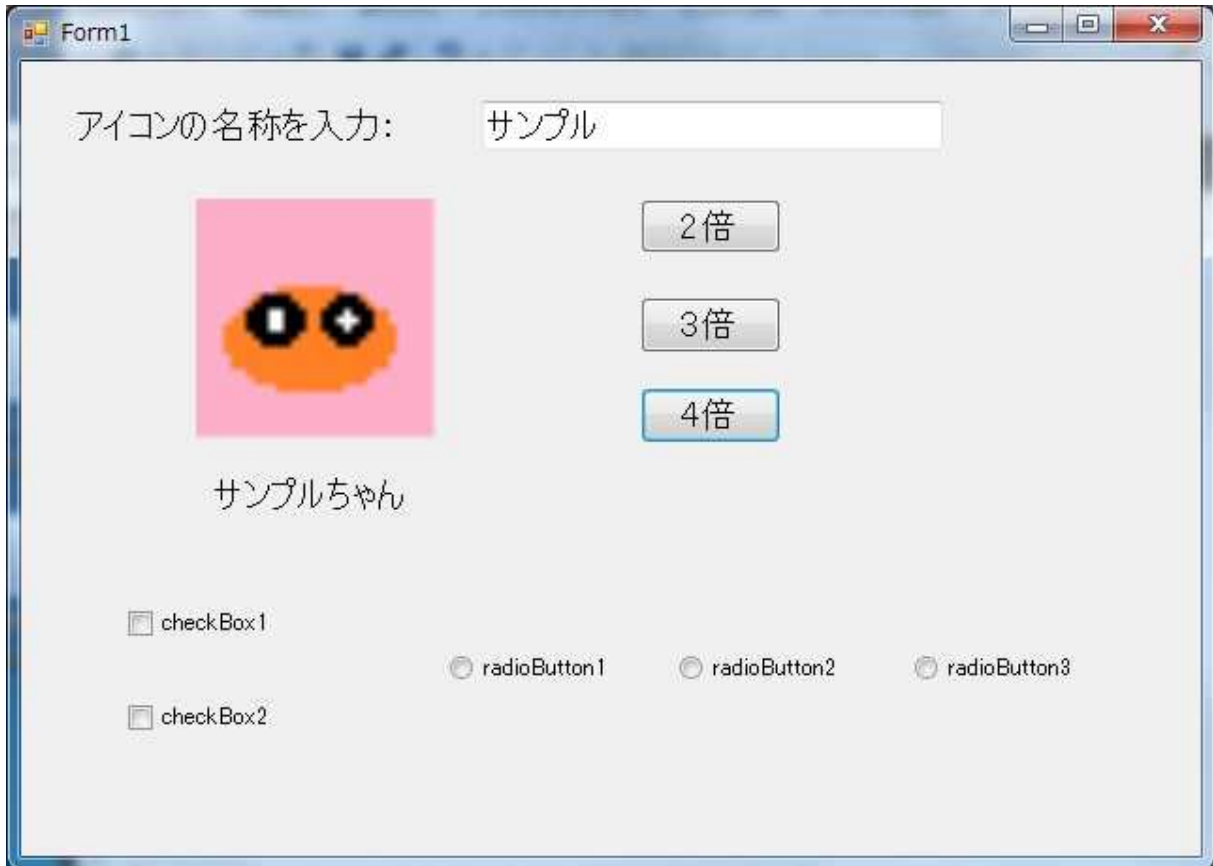
        private void button3_Click(object sender, EventArgs e)
        {
            pictureBox1.Width = 32;
            pictureBox1.Height = 32;
            pictureBox1.Width = pictureBox1.Width * 4;
            pictureBox1.Height = pictureBox1.Height * 4;
        }
    }
}
```

6) プログラムの実行・最終確認

『すべてを保存』ボタンを押してから、『開始』ボタンを押して、プログラムを実行する。
エラーが出ている場合には、修正してから保存、開始と進む。

それぞれのボタンをクリックして、アイコン画像がそれぞれの倍率で拡大されることを確認し
終わったら、閉じるボタンを押してプログラムを終了する。

(図は次のページ)

**【余裕のある人向け】**

Form1 の『Text』プロパティの値を「Form1」から「Third Appli」に書き換える。また、Form1 の『Icon』プロパティを第 2 回の授業で作成したオリジナルアイコンに設定する。

【ファイルが保存されている場所】 H:¥Documents¥Visual Studio 2013¥Projects¥Third¥Third

提出物：

- 1) フォームのデザインファイル **Form1.Designer.cs** をメールに添付して提出する。
- 2) フォームを含むソースファイル **Form1.cs** をメールに添付して提出する。
- 3) 質問を記述したファイル **Questions_3rd.txt** に解答を書き込んで保存し、メールに添付して提出する。