

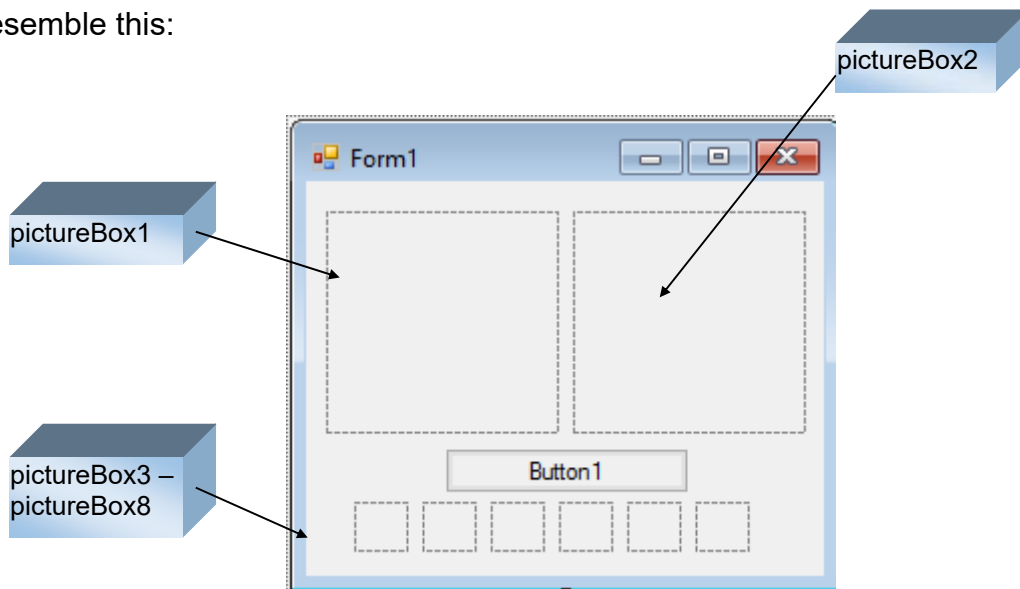
Project 3 – Dice Rolling

Project Design

It happens all the time. You get your favorite game out and the dice are missing! This program comes to the rescue – it uses the C# random number generator to roll two dice for you. Simply click a button to see the two dice displayed. A group of picture box controls will hold the six possible die values. This project is saved as **DiceRoll** in the project folder (**\BeginVCS\BVCS Projects**).

Place Controls on Form

Start a new project in Visual C#. Place two large picture box controls (to display the dice) and six small picture box controls (to hold the six possible die pictures) on the form. Place one button on the form. When done, your form should resemble this:



Set Control Properties

Set the control properties using the properties window:

Form1 Form:

Property Name	Property Value
Text	Dice Rolling
BackColor	Red
FormBorderStyle	FixedSingle
StartPosition	CenterScreen

pictureBox1 Picture Box:

Property Name	Property Value
Name	picDice1
SizeMode	StretchImage

pictureBox2 Picture Box:

Property Name	Property Value
Name	picDice2
SizeMode	StretchImage

pictureBox3 Picture Box:

Property Name	Property Value
Name	picDots1
Image	Dice1.gif (in \VCSKids\VCSK Projects\DiceRoll folder)
SizeMode	StretchImage
Visible	False

pictureBox4 Picture Box:

Property Name	Property Value
Name	picDots2
Image	Dice2.gif (in \VCSKids\VCSK Projects\DiceRoll folder)
SizeMode	StretchImage
Visible	False

pictureBox5 Picture Box:

Property Name	Property Value
Name	picDots3
Image	Dice3.gif (in \VCSKids\VCSK Projects\DiceRoll folder)
SizeMode	StretchImage
Visible	False

pictureBox6 Picture Box:

Property Name	Property Value
Name	picDots4
Image	Dice4.gif (in \VCSKids\VCSK Projects\DiceRoll folder)
SizeMode	StretchImage
Visible	False

pictureBox7 Picture Box:

Property Name	Property Value
Name	picDots5
Image	Dice5.gif (in \VCSKids\VCSK Projects\DiceRoll folder)
SizeMode	StretchImage
Visible	False

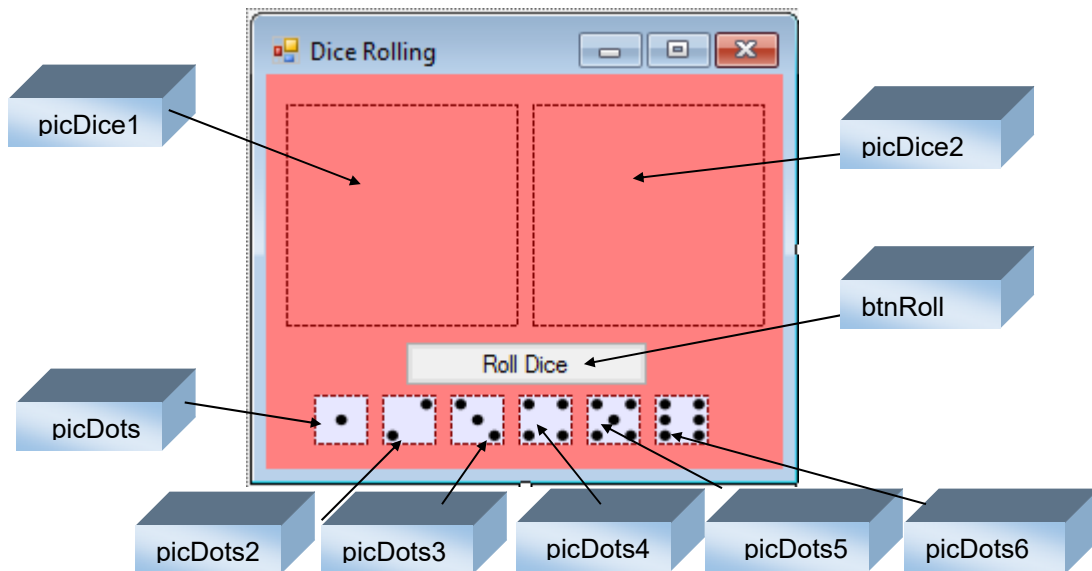
pictureBox8 Picture Box:

Property Name	Property Value
Name	picDots6
Image	Dice6.gif (in \VCSKids\VCSK Projects\DiceRoll folder)
SizeMode	StretchImage
Visible	False

button1 Button:

Property Name	Property Value
Name	btnRoll
Text	Roll Dice

When, done my form looks like this:



Notice we use two sets of picture boxes. The first, picDice1 and picDice2, is used to display the two dice. The second, picDots1 – picDots6, is used to store the six possible die pictures. This second group has a Visible property of False. Hence, you only see them displayed at design time.

Write Event methods

To roll the dice, simply click **Roll Dice**.

Declare an array of images (named **dots**) in the **general declarations** area. This array will be used to choose which of the six possible images to display. You also need a random number object:

```
Image[] dots = new Image[6];  
Random myRandom = new Random();
```

Add this code to the **Form1_Load** event. Here, we establish the image array and 'click' the **btnRoll** button to 'roll' the dice before the display is activated:

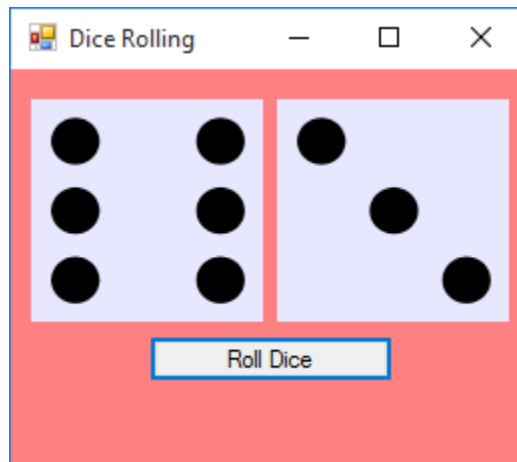
```
private void Form1_Load(object sender, EventArgs e)  
{  
    // initialize display  
    dots[0] = picDots1.Image;  
    dots[1] = picDots2.Image;  
    dots[2] = picDots3.Image;  
    dots[3] = picDots4.Image;  
    dots[4] = picDots5.Image;  
    dots[5] = picDots6.Image;  
    btnRoll.PerformClick();  
}
```

The **btnRoll_Click** event method:

```
private void btnRoll_Click(object sender, EventArgs e)  
{  
    // Roll Dice 1 and set display  
    picDice1.Image = dots[myRandom.Next(6)];  
    // Roll Dice 2 and set display  
    picDice2.Image = dots[myRandom.Next(6)];  
}
```

Run the Project

Save your work. Run the project. Click **Roll Dice** to see the dice change with each click. Look at the code to see how the random number (1 through 6) is generated and how the image array (**Dots**) sets the display. Here's one of my rolls:



Other Things to Try

The game of Yahtzee requires 5 dice. Modify the project to roll and display five dice. Or, let the user decide how many dice to display (you could more 'display' picture boxes and use the **Visible** property to specify whether a particular die is displayed). Add a label control that displays the sum of the displayed dice.

A fun change would be to have the die displays updated by a **Timer** control to give the appearance of rolling dice. You would need a Timer control for each die (every 100 milliseconds or so, randomly display from 1 to 6 dots). And, then you would need a Timer control to stop the 'rolling' (use an **Interval** of about 2000 milliseconds). The **btnRoll** button would control enabling on the Timer controls. All Timer controls are turned off (Enabled is set to false) by the Timer event that stops the rolling.