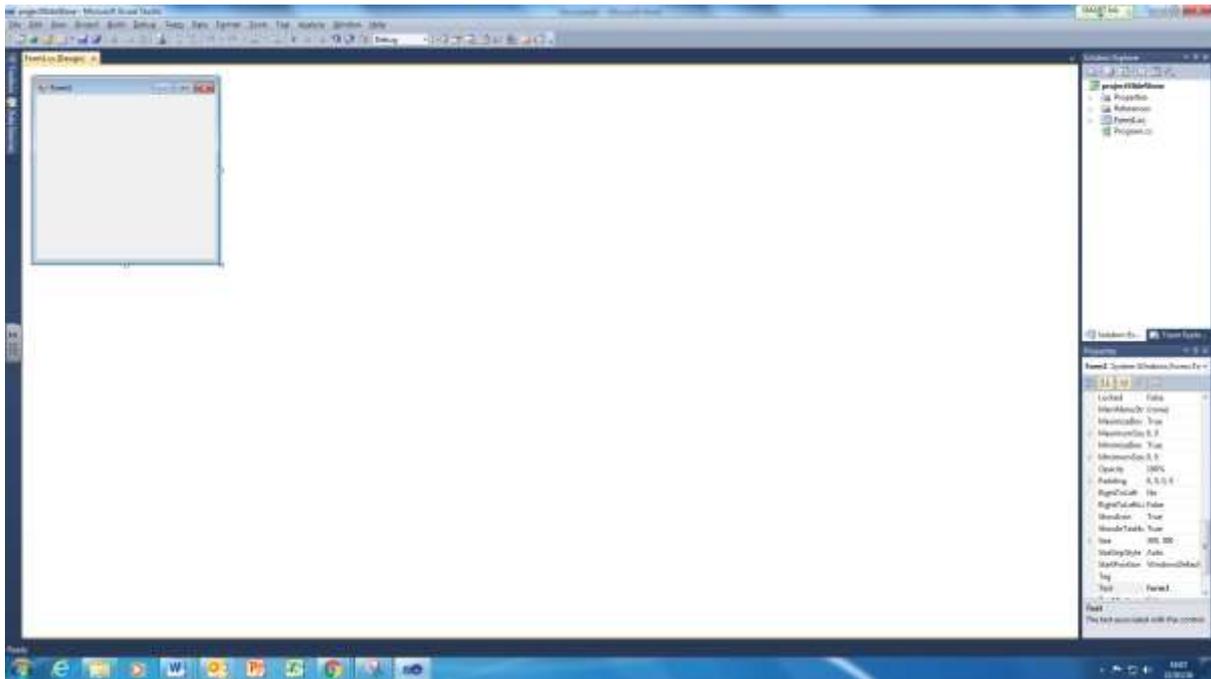


## C# Tutorial – Load all images from a folder Slide Show

In this tutorial we will see how to create a C# slide show where you load everything from a single folder and view them through a timer. This exercise is very useful because it allows us to learn about the various functionalities of C#. Let's get started with this fun tutorial.

First create a new C# Windows Form project in visual studio.

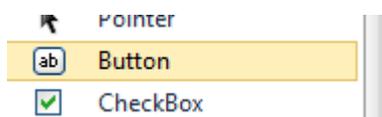
You can call the project anything you like I will be calling this one project slide show.



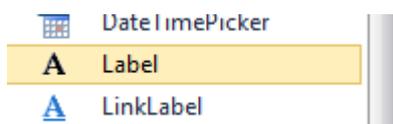
If your screen is the same as mine then the Tool BOX is hiding at the moment. We need to bring it to view. If you already know how to do it then brilliant however if you do not, no worries we will show you.

You can either bring it from the top View -> Toolbox or you can click on the toolbox icon on the left. Either way get it open because we need some windows components to the form.

First thing we need a button on the form.

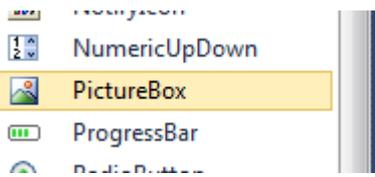


Second thing we need is a Label.

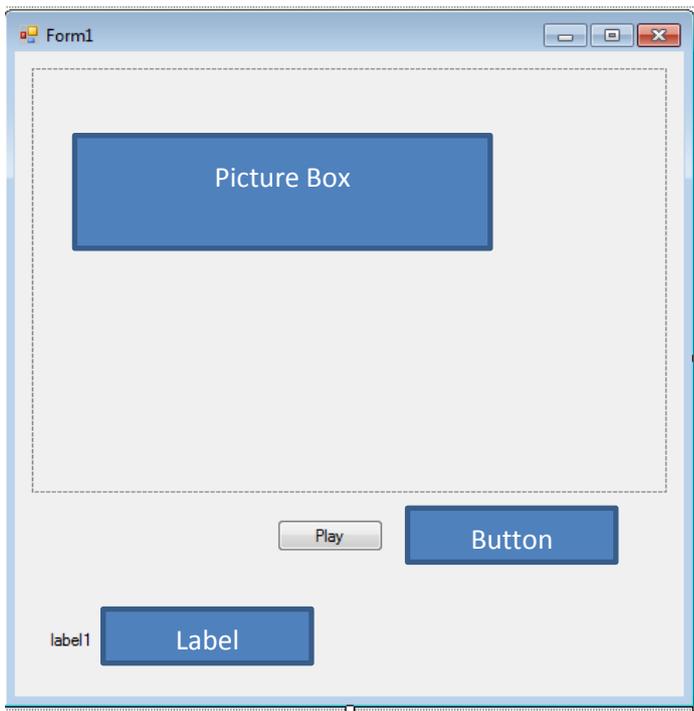
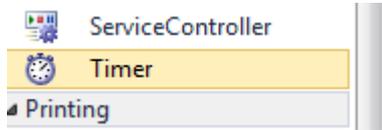


Now add a picture box to the screen

More tutorials on [www.moolct.com](http://www.moolct.com)

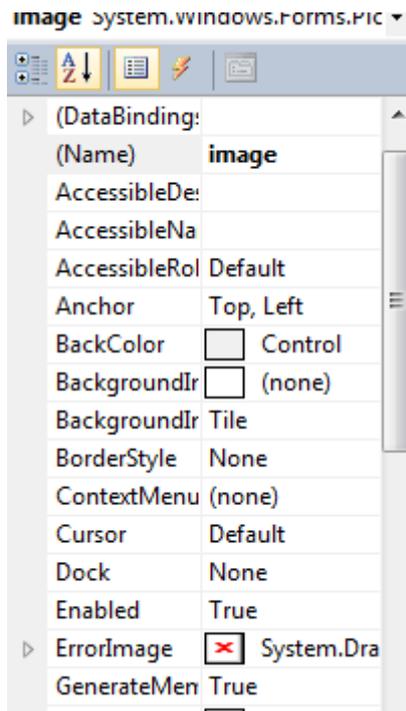


Now lets the final ingredient the Timer to the stage

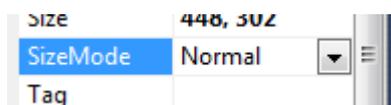


Now to improve the readability of this program we need to give these components individual name instead of the stock names such as pictureBox1 or button1 lets change the names to the following

Component	(Name) ← you will find this in the properties window
<b>Button</b>	Playstop
<b>Picture Box</b>	Image
<b>Label</b>	Filename
<b>Timer</b>	Leave it to default



This is the properties window for the picture box. See where the name is changed from pictureBox1 to image. We also need to change one small thing for this picture box. Find the option that says Size Mode



Change from normal to Stretch



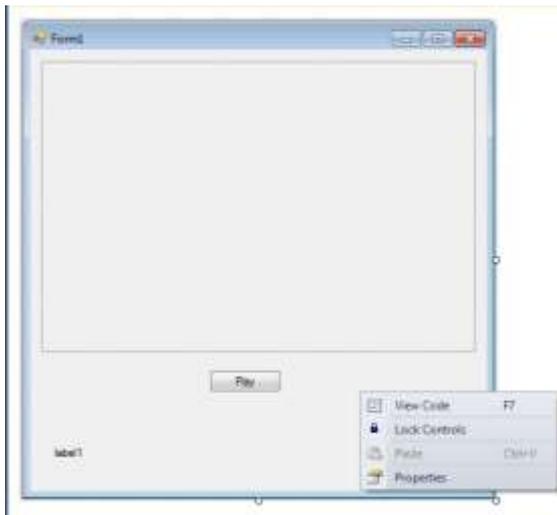
This is auto size the whole image to the size of this picture box.

### Application Requirement

OK so we have gone to the part where the GUI is set up right now but we need to set up the code. Before we do that we need to first understand what it that this program must do.

1. *Select a folder*
2. *Load all images from folder to the program*
3. *Change image every 3 seconds*
4. *Load only JPG images from folder*
5. *Loop through the images doesn't stop until the button is clicked again.*

To enter the code view either right click on the form and click view code or press F7



Before we begin we need to import the system IO class for this program. IO will allow us to use the folder selection class.

This is your current view on the code

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

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

Now add the IO under the System.Windows.Forms; line

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

More tutorials on [www.moolct.com](http://www.moolct.com)

```

using System.Windows.Forms;
using System.IO;

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

```

Now time to add our necessary variables for the program

```

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

namespace projectSlideShow
{
    public partial class Form1 : Form
    {
        int counter = 1;
        FolderBrowserDialog fbd = new FolderBrowserDialog();
        bool playing = false;

        public Form1()
        {
            InitializeComponent();
        }
    }
}

```

We are declaring an integer called counter and gave it a value of 1.

Since we imported the System.IO into our program we can use Folder Browser Dialog Class to ensure we have a folder dialog box to choose the images.

Last variable is a Boolean called playing. This will determine what to do when the slide show is playing or not playing. For now lets leave it on False.

Now we need to add an event to the button. Go back to the design view and double click on the button.



Double click this

Once you done that. Visual studio will add the following code automatically in your program.

More tutorials on [www.moolct.com](http://www.moolct.com)

```

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

namespace projectSlideShow
{
    public partial class Form1 : Form
    {
        int counter = 1;
        FolderBrowserDialog fbd = new FolderBrowserDialog();
        bool playing = false;

        public Form1()
        {
            InitializeComponent();
        }

        private void playstop_Click(object sender, EventArgs e)
        {
        }
    }
}

```

Within the curly brackets { } we can do our magic.

```

private void playstop_Click(object sender, EventArgs e)
{
    if (!playing)
    {
        DialogResult result = fbd.ShowDialog();
        playstop.Text = "Stop";
        timer1.Start();
        playing = true;
    }
    else
    {
        playstop.Text = "Play";
        playing = false;
        timer1.Stop();
    }
}

```

Lets look at the first part.

Inside that if statement we are checking if the playing is false when the button is clicked then we will do the following to it.

First we will load the dialog box to choose a folder by initiating the DialogResult class and adding a showDialog() to it.

Then we will change the button text to Stop instead of playing

More tutorials on [www.moolct.com](http://www.moolct.com)

Then we can start our timer that we added before to the form.

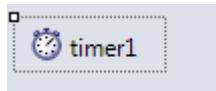
And finally we will set the playing variable to true.

```
private void playstop_Click(object sender, EventArgs e)
{
    if (!playing)
    {
        DialogResult result = fbd.ShowDialog();
        playstop.Text = "Stop";
        timer1.Start();
        playing = true;
    }
    else
    {
        playstop.Text = "Play";
        playing = false;
        timer1.Stop();
    }
}
```

Now we do the opposite to that first IF in the ELSE because if the slide show is playing then we need to stop it. So we are going to change the button text back to Play and set the Boolean playing to false and finally stop the timer 1.

That wasn't complicated was it?

Now go back to design view and double click on the timer icon.



```
private void timer1_Tick(object sender, EventArgs e)
{
}
}
```

This code above will be added your project. As we did with our button before we can add our code in between the curly brackets { }

Add the following code to the timer function.

```
private void timer1_Tick(object sender, EventArgs e)
{
    counter++;

    string[] images = Directory.GetFiles(fbd.SelectedPath, "*.*");

    if (counter > images.Length - 1)
    {
        counter = 0;
    }

    image.Image = Image.FromFile(images[counter]);
    filename.Text = images[counter]; //show name of the image
}
```

First we have a line called counter++ this is basically going to add 1 to the integer counter which we declared way up top.

Second we are declaring a string array where we will store the file names.

IMPORTANT

```
string[] images = Directory.GetFiles(fbd.SelectedPath, "*.*");
```

Look at the line above carefully notice we are using the Directory function to get the files and inside the brackets we are using fbd or file browser directory which we declared earlier in the tutorial. Inside that we are asking for it find a selected PATH and then we will load all file types from that folder.

NOW MORE IMPORTANT

If you want your program to run smoothly then have a folder where you only have image files for example JPG, GIF or PNG files. If you have other documents such as a word document, text file or spreadsheet the program will crash because it will not be able to load that file in to the picture box.

```
if (counter > images.Length - 1)
{
    counter = 0;
}
```

Here we are going to tell the slide show to loop once it reached the last image. So if the counter is larger than the file names in folder meaning we have reached the end then we will reset the counter to 0.

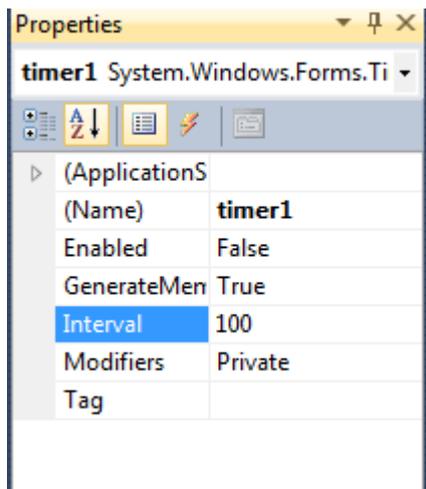
```
image.Image = Image.FromFile(images[counter]);
```

This line will load the individual images to the picture box on form.

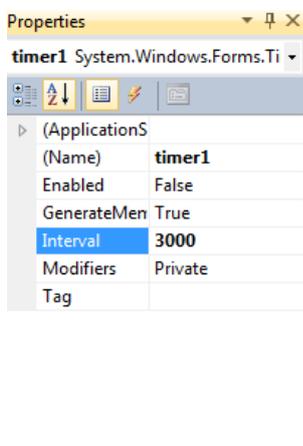
```
filename.Text = images[counter]; //show name of the image
```

And this line will load the file name of those images into the label.

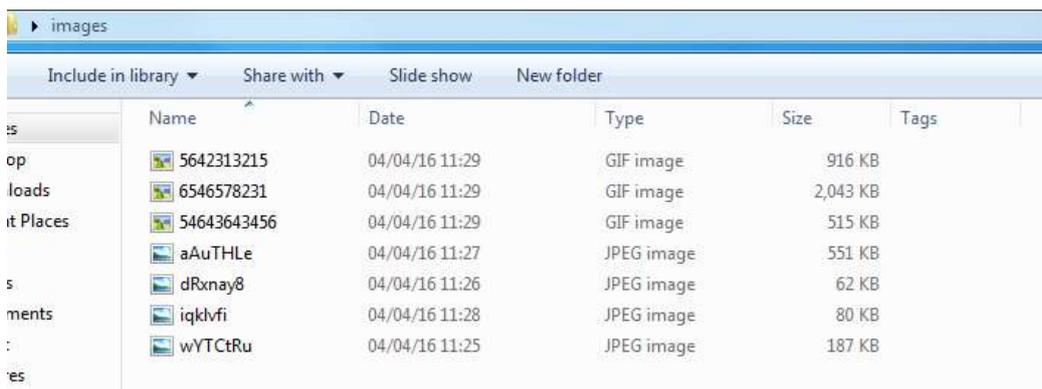
Once thing to do before we test the program. Look into the properties panel of the Timer we added to form.



Change the interval from 100 to 3000 meaning our slide show to should change every 3 seconds.



Now with that set up let's take a look the folder we have the pictures saved in.

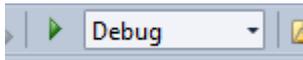


I have created an images folder with 4 JPEG and 3 GIF images. Since we will load all of the files from this folder we don't need to specify the names of the files only the types. You know the old saying programming should make your life easier not difficult. Well it's only true sometimes.

This folder is saved on the desktop. So hopefully once we click on play it should give us the option to select a folder and then it can start playing the images. We also need to keep in mind that since we will be allowing all files to be loaded into the picture box lets only save image files in the folder. If we have a DOC or TXT or PDF file in there the program can potentially crash.

More tutorials on [www.moolct.com](http://www.moolct.com)

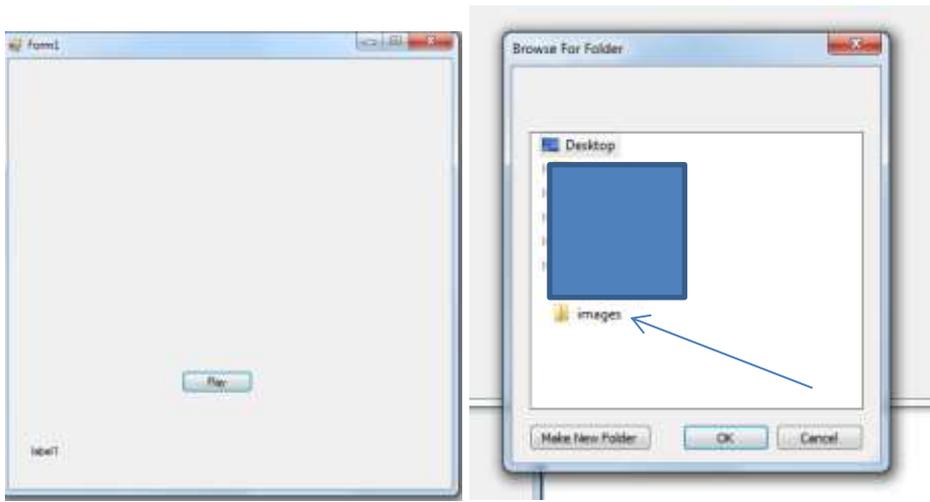
Time for testing



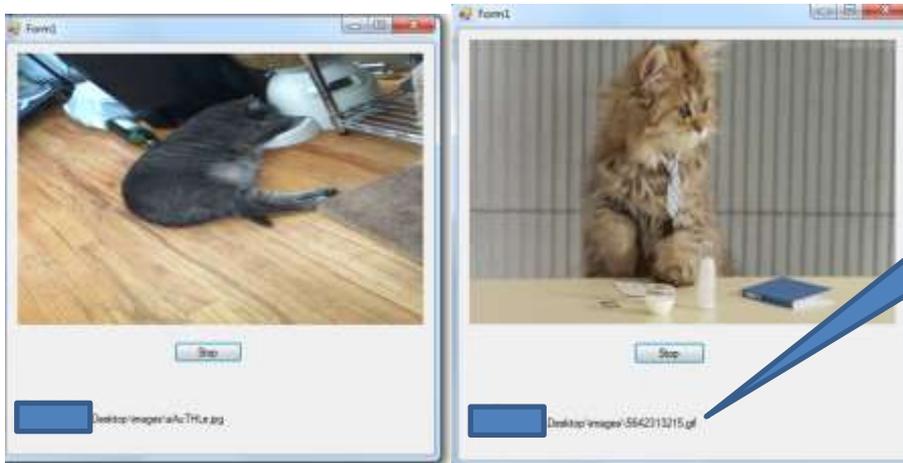
You can click on the play button to start debugging the application



Or you can click on debug on the top and click on start debugging.



Once we clicked on play the open folder dialog box comes up



The first image is a JPEG and second image is GIF both playing fine.



First image is a JPEG and second image is a GIF



Now we have stopped the slide show. Its sooo cuteeee OMG.

Hopefully you had fun creating this Slideshow.

See you soon on MoolCT.com