Visual Basic Tutorial - Slot Machine Game

In this tutorial we will discuss and show how to create a very simple slot machine game using visual studio and visual basic programming language. If you read some of the other Visual Basic we have discussed that visual basic has many different aspects of it one of which is game programming.

Visual Basic is a easy program to learn and syntaxes are similar to other programming languages. Now let's get started.

If you want to create an identical one to mine then download the assets from www.mooict.com and you can import those images as resource to the windows form.

This is what our final product will look like. Check out the Demo video to see the full functionality of the game itself.

First open a new project in Microsoft Visual Studio called slotMachine

more tutorials on www.mooict.com
First we will need 3 picture boxes on the stage

Don’t just drag 3 of them there. Here is an idea drag 1 to the form, let’s change some properties, add some resources and then copy and paste them twice so we won’t have to do it for the rest. Make life easy for yourself.

Right click on the picture box and click on choose image
Select the project resource file and click on import. Import all the pictures from the resource folder which you might have downloaded from mooict.com.

Here we have 9 different images

These are the images we will also I have created an animated GIF combination of all them which will run in a precise section of the game.

Let's move to the next part for now.

Once you have imported the files. Now we need to change a few things in the properties window before moving on to copy and paste. Hold on for a second itchy fingers.

- Change the border style to fixes 3d this gives us the embossed look.
- Change the size to 102, 116 and size mode to stretch image.

Now you can copy and paste them Check the image below.

more tutorials on [www.mooict.com](http://www.mooict.com)
Now we have a nice look to it.

Next we will need 1 button and 6 labels.

Now time to edit these labels. You can edit them easily in the text tab inside the properties window.

I changed the label 1 text to amount to play £5.

I will go over the game rules in a short while. Let’s get the GUI sorted out first.

Change the labels accordingly

Label2 text change to Money
Label3 text change to £££
Label 4 text change to Match 2 = £1
Label 5 text change to Match 3 = £10

more tutorials on [www.mooict.com](http://www.mooict.com)
Now we need to change these text styles so it looks more readable.

Click on a label and you will see the font property in the properties window.

Click on that and it will give you the option to change the font style for the labels.

You can change you what you like or just increase the size of the texts.

Also you can change the colour of the text, look for the fore color option in the properties window. It's under the font option.

So I changed some of the fonts to bold and change the colour of the label £££. Makes it look better don’t you think.

Now for the button.
Here is the button in the toolbox.

If you can’t find the toolbox at any point in the tutorial refer back to the View -> Toolbox

Now back to the button.

Drag it to the form and resize it to your liking. Since the button is the main interaction with the use I will make mine a little bigger. You can change the font and colour of the button if you choose.

Change the buttons text to play

We need to notify the player what they won or lost. So we will Add one last label to the form.

more tutorials on [www.mooict.com](http://www.mooict.com)
This one will feedback to the user after each round to let them know what's the status of the game is at that point.

Now we still have one more thing to add which is of teeny tiny importance the TIMER.

Let's move the timer to the form and look at the properties setting

![Timer properties](image)

Change the properties to the following name to `gameTimer` so we can easily understand it. Interval to 1000. In programming we run things in milliseconds which is 1000 = 1 second. Good to know right.

So now, we have all the necessary GUI in order time to discuss the matter of the actual code.

**Game Logic**

We all know how slot machines work and what they do. However it's time to think like an engineer. We are creating a slot machine we are creating a virtual simulation of one.

So in our game we it will do the follow

1. Reset all the values to default when button is clicked.
2. Generate random numbers for each picture box
3. deduct £5 from the player money [greedy lol]
4. Start animation on each picture box
5. stop animation one after the other not all together
6. check if there is a two match then give user £1 added to the money
7. check if there is a three match then give user £10 added to the money
8. Check if user has enough money to play or tell them to they don't have money to play the game right now.

Here are the rules of our game. Now we just need to adjust the code to run this. We won't need anything fancy to program just plain old simple visual basic.

---

more tutorials on [www.mooict.com](http://www.mooict.com)
If you know how to get to the code view then do so if you don't simply right click on the form and click on view code or press **F7**. See picture below.

![View Code](image1)

This is what the code looks like now.

![Form1.vb](image2)

Nice and empty pure silence

Thing to remember. All the we are doing goes inside the class which means nothing goes outside that END CLASS line.

We will cover the CLASS system in another tutorial for now let's get this thing working.

```vbnet
Public Class Form1

    Dim slotA As Integer
    Dim slotB As Integer
    Dim slotC As Integer
    Dim money As Integer = 50
    Dim result As Integer
    Dim counter As Integer
    Dim cost As Integer = 5
    Dim rn As New Random

End Class
```

Here we have added some variables for this game. We need these to check what the outcome of the game will be and how we can use those to either deduct the money or reward the player.
Slot A B C will contain individual numbers and we will link those numbers to an image.

Money variable will hold the player's money in the beginning of the game.

Result will contain what the player won or lost in the game.

Counter will be used with the timer to help simulate an animation for the slot machine.

Cost is how much we are going to charge the player for each round.

Rn is a variable which will generate a random number for us.

Now let's start adding some events to this game. Our first component for interactivity is the button on stage. Double click on the button.

![Button](image)

Visual studio will automatically a function for that button. So each time it's clicked it will trigger this function.

```vbnet
Public Class Form1

    Dim slotA As Integer
    Dim slotB As Integer
    Dim slotC As Integer
    Dim money As Integer = 50
    Dim result As Integer
    Dim counter As Integer
    Dim cost As Integer = 5
    Dim rn As New Random

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

    End Sub

End Sub
```

In Visual Basic the SUB means a function so we start with a sub and end the sub to determine where it starts or ends.

Inside it we will start adding our values.

```vbnet
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    slotA = 1
End Sub
```

More tutorials on [www.mooict.com](http://www.mooict.com)
slotA = 1  
slotB = 1  
slotC = 1

If money < cost Then
  Button1.Enabled = False
  Label6.Text = "You don't have any money left"
  counter = 99
  gameTimer.Stop()
Else
  End If
End Sub

Now we have given the Slot A B C a value of one. Just for testing purposes at the moment.

There is a IF statement which is checking if the amount of money is less than the cost then what do we do. In any game the most important thing is to end the game and know when to quit. So as long as the player has more than the cost per round they can play otherwise we don't let them. Strict rules but hey why not.

We will be creating a nesting if statements inside this one remember where to start and end.

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
  slotA = 1  
  slotB = 1  
  slotC = 1

  If money < cost Then
    Button1.Enabled = False
    Label6.Text = "You don't have any money left"
    counter = 99
    gameTimer.Stop()
  Else
    End If
End Sub

Let's take another look at the newly added code on the screen.

If the players money is less than the cost

First we are disabling the button.

Label 6 is the status label we added last we are giving feedback to the user about what happened.

we are giving the counter a value of 99

Game timer is being stopped so it doesn't run the timer function which we will add to the game soon.

Notice we added a small ELSE end of the if statement before the END IF. All out game logic where the player has the money to play each round will go inside this.

If money < cost Then
  Button1.Enabled = False
  Label6.Text = "You don't have any money left"
  counter = 99
  gameTimer.Stop()

more tutorials on www.mooict.com
Else
    money = money - 5
    result = 0
    counter = 0
    gameTimer.Start()
    Button1.Enabled = False
End If

Here we are going to add some reset variable to the else statement.

We will deduct the £5 from the user first. House rules

Then we change the result to 0 and counter to 0

we are starting the game timer by using the START() function which is embedded into the timer itself. Its the same as STOP() used in the timer.

Finally we will disable the button. We are doing this so the player cannot cheat by pressing the button multiple times. One click each round.

Now its time to nest another IF statement within in our ELSE

If money < cost Then
    Button1.Enabled = False
    Label6.Text = "You don't have any money left"
    counter = 99
    gameTimer.Stop()
Else
    money = money - 5
    result = 0
    counter = 0
    gameTimer.Start()
    Button1.Enabled = False
    If slotA = slotB And slotB = slotC Then
        result = 10
        money = money + result
    End If
End If

Here we have a nested IF statement. Now we will check if all the three variable contain the same value. Is that is true then we can reward user with £10 and add that money to their money variable.

Look closely at that IF statement first we are checking if slot A and B are the same then we are checking if Slot B and C are the same. If they are its a Jackpot. Now you can change the reward if you wish but lets complete this game first.

If money < cost Then
    Button1.Enabled = False
    Label6.Text = "You don't have any money left"
    counter = 99
    gameTimer.Stop()
Else
    money = money - 5
    result = 0
End If

more tutorials on [www.mooict.com](http://www.mooict.com)
Now it's time to check if TWO of the slots are the same. We have seen how to use a AND condition in the IF statement here is the perfect example of OR. Or does what it says on the box it checks a condition or another condition or another to see which matches and takes the appropriate action from it.

Here we need to find a match for only two out of three so that opens up a new issue however by using we can simply check between the slots which are matching with others.

If A is equals to B or is A is equals to C or is B equals to C then we give the user £1 reward for the effort and add that to their money variable.

Here is the full button click function right now.

```vbnet
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    slotA = 1
    slotB = 1
    slotC = 1

    If money < cost Then
        Button1.Enabled = False
        Label6.Text = "You don't have any money left"
        counter = 99
        gameTimer.Stop()
    Else
        money = money - 5
        result = 0
        counter = 0
        gameTimer.Start()
        Button1.Enabled = False
        If slotA = slotB And slotB = slotC Then
            result = 10
            money = money + result
        ElseIf slotA = slotB Or slotA = slotC Or slotB = slotC Then
            result = 1
            money = money + result
        End If
    End If
End Sub
```

Feast your eyes on it :)

There is nothing more to add to this button 1 function for now. However We will come back to it once we added our next function.

more tutorials on [www.mooict.com](http://www.mooict.com)
Now we will create a new function which will respond to the random numbers from each slot.

```vbnet
Private Sub changeImageA(ByVal pp As PictureBox, ByVal num As Integer)
    Select Case num
        Case 1
            pp.Image = Me.Resources.apple
        Case 2
            pp.Image = Me.Resources.banana
        Case 3
            pp.Image = Me.Resources.cherry
        Case 4
            pp.Image = Me.Resources.lemon
        Case 5
            pp.Image = Me.Resources.melon
        Case 6
            pp.Image = Me.Resources.pear
        Case 7
            pp.Image = Me.Resources.bar
        Case 0
            pp.Image = Me.Resources.animation
    End Select
End Sub
```

Give the code above a very good look. So far we have used functions that are related to an event or an empty argument function. This one however passes a variable through it and it will use that to show a certain resource which we imported earlier.

Let’s take a close look at the beginning of the function.

```vbnet
Private Sub changeImageA(ByVal pp As PictureBox, ByVal num As Integer)
```

its begins just as any function `Private Sub changeImageA` Then inside it you can see we are requesting two different variables. first one is `pp as PictureBox` since we have 3 different picture boxes to control on the form we can use this one function to control them 3. Next variable is `num as integer` this will pass the number we have stored in each slot and change the images appropriately. If you are wondering about the `ByVal` keyword before each variable in the argument this was added automatically by visual basic. If yours not doing that then just add it yourself. It simply implies that we will use the values from each variables in the function. Well duuh right.

```vbnet
Select Case num
```

This line of code works similar to the if statement of switch statement from C#. In visual basic we use the Select Case. This select condition starts with a case, the case of what if inside of num variable. We will manually write what to do in each case.

```vbnet
Case 1
    pp.Image = Me.Resources.apple
```

If the number is 1 then we change the pp picture box to the apple image from our resources. Now you might be wondering what’s this PP picture box and why am I writing the code for it. Well all will be cleared soon enough. Just follow along for now.

We do the same thing for each case up to 7 and also we put a case in for 0 which is out animated GIF in the resources folder.
End Select

Don't forget to end the select condition before the function ending.

Now let's get back to our design view.

Find the game time and double click on it.

```
Private Sub gameTimer_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles gameTimer.Tick
    End Sub
```

This function will be added to the code by visual studio.

```
Private Sub gameTimer_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles gameTimer.Tick
    counter = counter + 1
    If counter = 1 Then
        changeImageA(PictureBox1, slotA)
    ElseIf counter = 3 Then
        changeImageA(PictureBox2, slotB)
    ElseIf counter = 5 Then
        changeImageA(PictureBox3, slotC)
        gameTimer.Stop()
        Label6.Text = money
        Button1.Enabled = True
        Label6.Text = "you won £" + Str(result)
        Label3.Text = money
    End If
End Sub
```

inside the function all the following code.

First we will call counter variable and increase it by one. We are going to increase it by checking the amount in the counter and adding 1 on top of that.

Now we will check the values in the counter. If the value is one then we will call that function we created earlier.

See how we are calling it. by using one function we can change all 3 picture boxes.

```
changeImageA(PictureBox1, slotA)
```

inside the change image A function we are doing the following changeImageA (name of the picture box, number which the select statement will respond to.)

Now run the program.

We will test to see if we win £10 and if we can match 2 to win £1

more tutorials on www.mooict.com
Since our Slot A B and C is set to 1 they all match and it wins us £10. Also you can see the 10 is added to our 50 minus the 5 for playing a round.

Now lets change slot C to 3 and try it

That worked nicely.

Now its time to put some final touch to the program and it will be working flawlessly.

Inside the button 1 function add the three lines below.

more tutorials on [www.mooict.com](http://www.mooict.com)
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    slotA = 1
    slotB = 1
    slotC = 3
    
    If money < cost Then
        Button1.Enabled = False
        Label6.Text = "You don't have any money left"
        counter = 99
        gameTimer.Stop()
    Else
        money = money - 5
        result = 0
        counter = 0
        gameTimer.Start()
        Button1.Enabled = False
        If slotA = slotB And slotB = slotC Then
            result = 10
            money = money + result
        ElseIf slotA = slotB Or slotA = slotC Or slotB = slotC Then
            result = 1
            money = money + result
        End If
    End If
End Sub

Add these 3 change image functions to the button 1. Once the button is clicked we will play the animation before we set it up. Run it now and see how it works.

Now the final ingredient to the program. Once again it will be residing inside the button 1 function.

Remember that Dim rn As New Random line in the program earlier. We will use this random class to generate a random number between 1 and 7 and each will be residing inside the button 1 function from there we can create an unpredictable change and miss game.

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    slotA = rn.Next(1, 7)
    slotB = rn.Next(1, 7)
    slotC = rn.Next(1, 7)
    
    If money < cost Then
        Button1.Enabled = False
        Label6.Text = "You don't have any money left"
        counter = 99
        gameTimer.Stop()
    Else
        money = money - 5
        result = 0
        counter = 0
        gameTimer.Start()
        Button1.Enabled = False
        If slotA = slotB And slotB = slotC Then
            result = 10
            money = money + result
        ElseIf slotA = slotB Or slotA = slotC Or slotB = slotC Then
            result = 1
            money = money + result
        End If
    End If
End Sub

more tutorials on www.mooict.com
money = money + result
End If
End If
changeImageA(PictureBox1, 0)
changeImageA(PictureBox2, 0)
changeImageA(PictureBox3, 0)
End Sub

look at the slot A B and C we are using rn.next(1 to 7) line to generate a random number between 1 and 7 and then we will pass the value to our timer which will animate each picture box.

Now lets try the game.

There we played 3 rounds of the game and we only won £1 huh its closer to a real slot machine lol.

Now you can do the following to the program.

You can research to see if you can add sound to each time the image is loaded. That will add a good auditory response for the user. You can have a each round end sound and a final sound for different wins.

We came a long way in this tutorial. Make sure to have fun with all this.

Moo for now. :)

more tutorials on www.mooict.com