**Name Session:**

**Programming II**

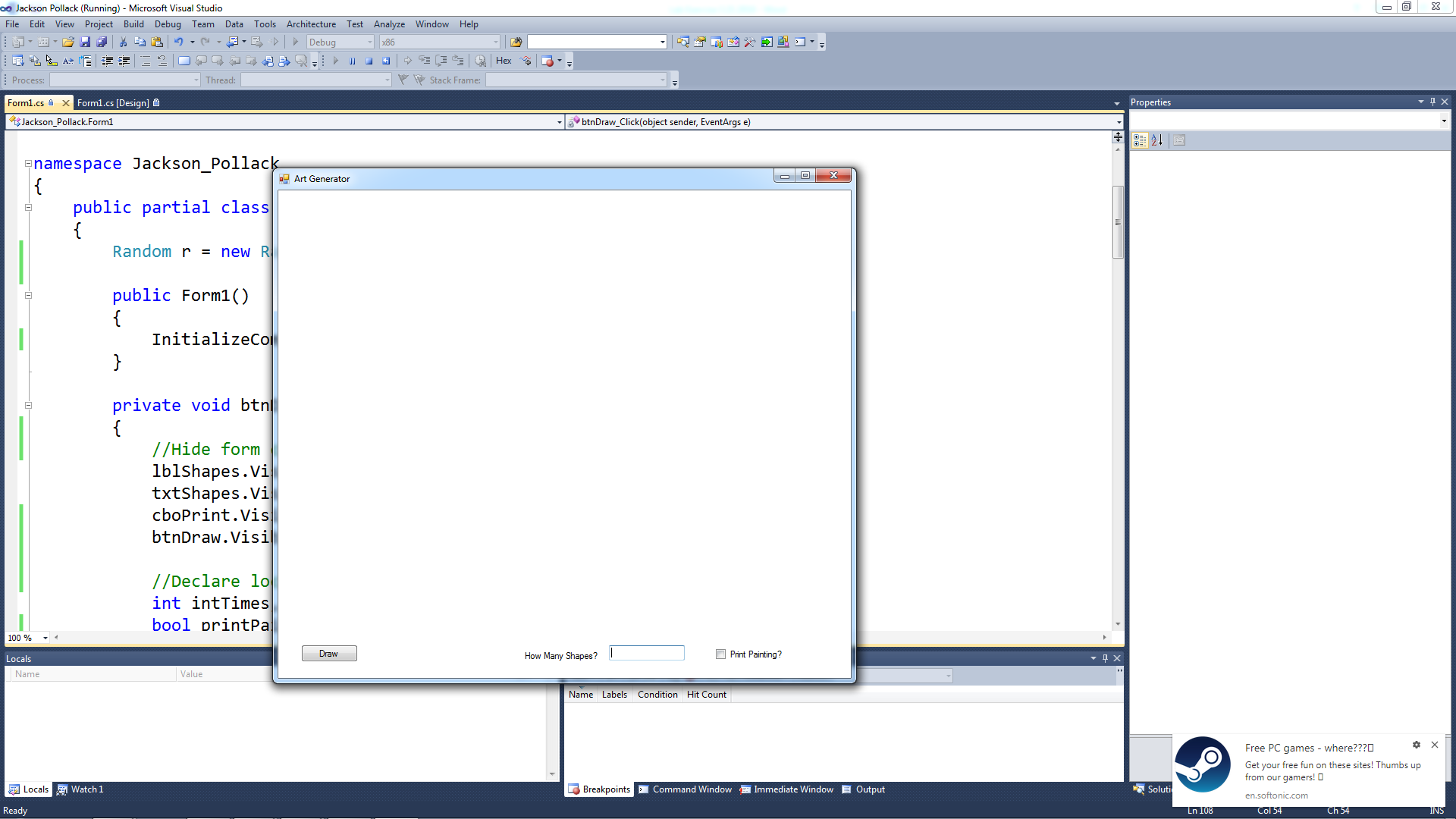
**Lab Exercise 3/23/2020**

**Move Over Jackson Pollack**

I this lab you will create an abstract drawing program that will create computer generated artwork in the spirit of Jackson Pollack. Jackson Pollack was an abstract expressionist whose work was influential in post-World War and whose style attracted the attention of the CIA who saw it as a means of promoting the USA as a haven of free thought and free markets. When you have completed this project, print an abstract drawing, attach to this sheet and turn in.

You program should do the following.

1. Create a graphics surface to draw on.
2. Select a random shape (Rectangle, Ellipse, Line, Filled Rectangle, Filled Ellipse)
3. Select a random starting point for your shape
4. Select a random width and height for your shape
5. Select a random color for your shape
6. Select a random pen width for your non-filled shapes
7. Draw the shape
8. Repeat steps 2 – 7 a user specified number of times
9. Create a new Windows Application



Notice that the form has a Draw button, a TextBox to specify the number of shapes to draw and a CheckBox to specify printing generated artwork.

These should have the following properties set:

|  |  |  |
| --- | --- | --- |
| Control | Name | Text |
| Form1 | Form1 | Art Generator |
| Button | btnDraw | Draw |
| Textbox | txtShapes |  |
| Checkbox | ckbPrint | Print Painting? |

1. When you click on the draw button, your program will hide all of the form controls, draw a number of random shapes, print if desired and restore the form to its original state.
2. Here is the btnDraw\_Click event code:

//Add comment here

lblShapes.Visible = false;

txtShapes.Visible = false;

cboPrint.Visible = false;

btnDraw.Visible = false;

// Add comment here

int intTimes, rShape, rX, rY, rWidth, rHeight, rColor, index;

bool printPainting = false;

// Add comment here

Graphics FormSurface = CreateGraphics();

FormSurface.Clear(Color.White);

// Add comment here

intTimes = Convert.ToInt32(txtShapes.Text);

// Add comment here

Pen myPen = new Pen(Color.Red, 5);

SolidBrush myBrush = new SolidBrush(Color.Red);

Color myColor = new Color();

//Add comment here

for (index = 1; index <= intTimes; index++)

{

rShape = r.Next(1,5); //Add comment here

rColor = r.Next(1,5); //Add comment here

rX = r.Next(0, 800); //Add comment here

rY = r.Next(0, 800); //Add comment here

rWidth = r.Next(100,300); //Add comment here

rHeight = r.Next(100, 300); //Add comment here

//Add comment here

switch (rColor)

{

case 1:

myColor = Color.Red;

break;

case 2:

myColor = Color.Green;

break;

case 3:

myColor = Color.Blue;

break;

case 4:

myColor = Color.Yellow;

break;

case 5:

myColor = Color.Orange;

break;

}

//Add comment here

myPen.Color = myColor;

myBrush.Color = myColor;

//Add comment here

switch (rShape)

{

case 1:

FormSurface.DrawLine(myPen, rX, rY, rX + rWidth, rY + rHeight);

break;

case 2:

FormSurface.DrawEllipse(myPen, rX, rY, rWidth, rHeight);

break;

case 3:

FormSurface.DrawRectangle(myPen, rX, rY, rWidth, rHeight);

break;

case 4:

FormSurface.FillRectangle(myBrush, rX, rY, rWidth, rHeight);

break;

case 5:

FormSurface.FillEllipse(myBrush, rX, rY, rWidth, rHeight);

break;

}

}

//Add comment here

if (ckbPrint.Checked)

printPainting = true;

//Add comment here

if (printPainting)

printForm1.Print();

//Add comment here

Thread.Sleep(5000);

////Add comment here

btnDraw.Visible = true;

lblShapes.Visible = true;

txtShapes.Visible = true;

cboPrint.Visible = true;

cboPrint.Checked = false;

txtShapes.Text = "";

txtShapes.Focus();

FormSurface.Clear(Color.White);