

Printer Caps 6



```
/*  
Professional Windows GUI Programming Using C#  
by Jay Glynn, Csaba Torok, Richard Conway, Wahid Choudhury,  
Zach Greenvoss, Shripad Kulkarni, Neil Whitlow
```

```
Publisher: Peer Information
```

```
ISBN: 1861007663
```

```
*/
```

```
using System;  
using System.Drawing;  
using System.Collections;  
using System.ComponentModel;  
using System.Windows.Forms;  
using System.Data;  
  
using System.Drawing.Printing;  
using System.Drawing.Drawing2D;
```

```
namespace PrinterCaps6
```

```
{  
public class PrinterCaps6 : System.Windows.Forms.Form  
{  
//Saved PageSettings Object  
private PageSettings oPageSettings;  
  
private int m_nCurrPrinter;  
private int m_nCurrPage;  
  
private System.Drawing.Printing.PrintDocument printDoc;  
private System.Windows.Forms.PrintPreviewDialog ppDialog;  
private System.Windows.Forms.Button btnPrint;  
private System.Windows.Forms.PrintDialog oPrintDialog;  
private System.Windows.Forms.PageSetupDialog oPageSetup;  
private System.Windows.Forms.Button btnSetup;
```

```

private System.ComponentModel.Container components = null;

public PrinterCaps6()
{
    InitializeComponent();

    //Create PageSettings object to store page information
    oPageSettings = new PageSettings();
}

///

/// Clean up any resources being used.
///
protected override void Dispose( bool disposing )
{
    if( disposing )
    {
        if (components != null)
        {
            components.Dispose();
        }
    }
    base.Dispose( disposing );
}

#region Windows Form Designer generated code
///

/// Required method for Designer support – do not modify
/// the contents of this method with the code editor.
///
private void InitializeComponent()
{
    // System.Resources.ResourceManager resources = new
    System.Resources.ResourceManager(typeof(PrinterCaps6));
    this.printDoc = new System.Drawing.Printing.PrintDocument();
    this.ppDialog = new System.Windows.Forms.PrintPreviewDialog();
    this.btnPrint = new System.Windows.Forms.Button();
}

```

```

this.oPrintDialog = new System.Windows.Forms.PrintDialog();
this.oPageSetup = new System.Windows.Forms.PageSetupDialog();
this.btnSetup = new System.Windows.Forms.Button();
this.SuspendLayout();
//
// printDoc
//
this.printDoc.PrintPage += new
System.Drawing.Printing.PrintPageEventHandler(this.printDoc_Pr
intPage);
//
// ppDialog
//
this.ppDialog.AutoScrollMargin = new System.Drawing.Size(0,
0);
this.ppDialog.AutoScrollMinSize = new System.Drawing.Size(0,
0);
this.ppDialog.ClientSize = new System.Drawing.Size(400, 300);
this.ppDialog.Enabled = true;
//
this.ppDialog.Icon =
((System.Drawing.Icon)(resources.GetObject("ppDialog.Icon")));
this.ppDialog.Location = new System.Drawing.Point(326, 19);
this.ppDialog.MaximumSize = new System.Drawing.Size(0, 0);
this.ppDialog.Name = "ppDialog";
this.ppDialog.Opacity = 1;
this.ppDialog.TransparencyKey = System.Drawing.Color.Empty;
this.ppDialog.Visible = false;
//
// btnPrint
//
this.btnPrint.Location = new System.Drawing.Point(52, 44);
this.btnPrint.Name = "btnPrint";
this.btnPrint.TabIndex = 0;
this.btnPrint.Text = "Print";
this.btnPrint.Click += new
System.EventHandler(this.btnPrint_Click);
//

```

```

// oPrintDialog
//
this.oPrintDialog.AllowSomePages = true;
this.oPrintDialog.PrintToFile = true;
//
// btnSetup
//
this.btnSetup.Location = new System.Drawing.Point(52, 8);
this.btnSetup.Name = "btnSetup";
this.btnSetup.TabIndex = 1;
this.btnSetup.Text = "Page Setup";
this.btnSetup.Click += new
System.EventHandler(this.btnSetup_Click);
//
// PrinterCaps6
//
this.AutoScaleBaseSize = new System.Drawing.Size(5, 13);
this.ClientSize = new System.Drawing.Size(176, 86);
this.Controls.AddRange(new System.Windows.Forms.Control[] {
this.btnSetup,
this.btnPrint});
this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.FixedDialog;
this.Name = "PrinterCaps6";
this.StartPosition =
System.Windows.Forms.FormStartPosition.CenterScreen;
this.Text = "Printer Caps";
this.ResumeLayout(false);

}
#endregion

///

/// The main entry point for the application.
///
[STAThread]
static void Main()

```

```

{
Application.Run(new PrinterCaps6());
}

private void printDoc_PrintPage(object sender,
System.Drawing.Printing.PrintPageEventArgs e)
{
PrinterSettings pSettings = new PrinterSettings();
Font printFont = new Font("Arial", 12);

//Use the Margins
int nTextPosY = e.MarginBounds.Top;
int nTextPosX = e.MarginBounds.Left;
int nHeight = (int)printFont.GetHeight(e.Graphics);

//Height of a printer block
int nBlockHeight = 9 * nHeight;

if(e.PageSettings.PrinterSettings.PrintRange ==
PrintRange.SomePages)
{
while(m_nCurrPage < e.PageSettings.PrinterSettings.FromPage) {
//Move printer to next page block m_nCurrPrinter +=
(int)(e.MarginBounds.Height / nBlockHeight); m_nCurrPage++;
if(m_nCurrPrinter > PrinterSettings.InstalledPrinters.Count)
return;
}

if(m_nCurrPage > e.PageSettings.PrinterSettings.ToPage)
{
//Don't print anything more
return;
}
}

//Print Background Graphic
LinearGradientBrush aBrush =
new
LinearGradientBrush(e.MarginBounds,Color.FromArgb(100,Color.Li

```

```

ghtBlue),
Color.FromArgb(100,Color.Blue),LinearGradientMode.ForwardDiagonal);
e.Graphics.FillRectangle(aBrush,e.MarginBounds);

//Loop through using indexor now
//Start with the previous index in m_nCurrPrinter
for(int x = m_nCurrPrinter;x<
PrinterSettings.InstalledPrinters.Count; x++)//each(string
sPtr in PrinterSettings.InstalledPrinters) {
pSettings.PrinterName = PrinterSettings.InstalledPrinters[x];
if(pSettings.IsValid) { //Ensure this printer block can fit on
the page if(nTextPosY + nBlockHeight < e.MarginBounds.Bottom)
{ //Print the caps of the printer
e.Graphics.DrawString(PrinterSettings.InstalledPrinters[x],
printFont,Brushes.Black, nTextPosX, nTextPosY + 5);
e.Graphics.DrawString("Can Duplex: " +
pSettings.CanDuplex.ToString(), printFont,Brushes.Black,
nTextPosX + 10, nTextPosY + (5 + nHeight));
e.Graphics.DrawString("Is Default: " +
pSettings.IsDefaultPrinter.ToString(),
printFont,Brushes.Black, nTextPosX + 10, nTextPosY + (5 +
nHeight*2)); e.Graphics.DrawString("Is Plotter: " +
pSettings.IsPlotter.ToString(), printFont,Brushes.Black,
nTextPosX + 10, nTextPosY + (5 + nHeight*3));
e.Graphics.DrawString("Landscape Angle: " +
pSettings.LandscapeAngle.ToString(), printFont,Brushes.Black,
nTextPosX + 10, nTextPosY + (5 + nHeight*4));
e.Graphics.DrawString("Maximum Copies: " +
pSettings.MaximumCopies.ToString(), printFont,Brushes.Black,
nTextPosX + 10, nTextPosY + (5 + nHeight*5));
e.Graphics.DrawString("Maximum Page: " +
pSettings.MaximumPage.ToString(), printFont,Brushes.Black,
nTextPosX + 10, nTextPosY + (5 + nHeight*6));
e.Graphics.DrawString("Minimum Page: " +
pSettings.MinimumPage.ToString(), printFont,Brushes.Black,
nTextPosX + 10, nTextPosY + (5 + nHeight*7));

```

```

e.Graphics.DrawString("Supports Color: " +
pSettings.SupportsColor.ToString(), printFont,Brushes.Black,
nTextPosX + 10, nTextPosY + (5 + nHeight*8)); nTextPosY =
nTextPosY + ((5 + nHeight*8) + nHeight); //Draw line after
each
e.Graphics.DrawLine(System.Drawing.Pens.Black,nTextPosX,nTextP
osY,e.MarginBounds.Right - 10,nTextPosY);
e.Graphics.FillEllipse(System.Drawing.Brushes.Black,e.MarginBo
unds.Right - 10,nTextPosY-5,10,10); } else { //Couldn't fit
block on the page - need more pages m_nCurrPrinter = x;
e.HasMorePages = true; return; } } } //Last page if we reached
here e.HasMorePages = false; m_nCurrPrinter = 0; return; }
private void btnPrint_Click(object sender, System.EventArgs e)
{ //Set to defaults m_nCurrPrinter = 0; m_nCurrPage = 1;
oPrintDialog.Document = printDoc; if
(oPrintDialog.ShowDialog() == DialogResult.OK) { try {
//Ensure the correct PrinterSettings object is used
oPageSettings.PrinterSettings = printDoc.PrinterSettings;
//Assign PageSettings object to all pages
printDoc.DefaultPageSettings = oPageSettings;
ppDialog.Document = printDoc; ppDialog.ShowDialog(); }
catch(Exception ex) { MessageBox.Show(ex.ToString()); } } }
private void btnSetup_Click(object sender, System.EventArgs e)
{ oPageSetup.PageSettings = oPageSettings;
if(oPageSetup.ShowDialog() == DialogResult.OK) { oPageSettings
= oPageSetup.PageSettings; } } } } [/csharp]

```