PrintDocument Class

Namespace: System.Drawing.Printing

Assembly: System.Drawing.Common.dll

Defines a reusable object that sends output to a printer, when printing from a Windows Forms application.

In this article

Definition

Examples

Remarks

Constructors

Properties

Methods

Events

Applies to

See also

C#	Copy

```csharp
public class PrintDocument : System.ComponentModel.Component
```

Inheritance   Object → MarshalByRefObject → Component → PrintDocument

Examples

The following code example prints the file named C:\My Documents\MyFile.txt on the default printer. To run the example, create a new Windows Forms project and paste the example code into the form, replacing the file contents. For C#, you will need to delete the Form1.Designer.cs file. Also, change the path to the file you want to print.

> ⓘ **Note**
>
> The example requires that each line fits within the page width.

C#   ⎙ Copy

```csharp
using System;
using System.IO;
using System.Drawing;
using System.Drawing.Printing;
using System.Windows.Forms;

public partial class Form1 : System.Windows.Forms.Form
{
    private System.ComponentModel.Container components;
    private System.Windows.Forms.Button printButton;
    private Font printFont;
    private StreamReader streamToPrint;

    public Form1()
    {
        // The Windows Forms Designer requires the following call.
        InitializeComponent();
    }

    // The Click event is raised when the user clicks the Print button.
    private void printButton_Click(object sender, EventArgs e)
    {
        try
        {
            streamToPrint = new StreamReader
               ("C:\\My Documents\\MyFile.txt");
            try
            {
                printFont = new Font("Arial", 10);
                PrintDocument pd = new PrintDocument();
                pd.PrintPage += new PrintPageEventHandler
                    (this.pd_PrintPage);
                pd.Print();
            }
            finally
            {
                streamToPrint.Close();
            }
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message);
        }
    }

    // The PrintPage event is raised for each page to be printed.
    private void pd_PrintPage(object sender, PrintPageEventArgs ev)
    {
        float linesPerPage = 0;
        float yPos = 0;
        int count = 0;
```

```
        float leftMargin = ev.MarginBounds.Left;
        float topMargin = ev.MarginBounds.Top;
        string line = null;

        // Calculate the number of lines per page.
        linesPerPage = ev.MarginBounds.Height /
           printFont.GetHeight(ev.Graphics);

        // Print each line of the file.
        while (count < linesPerPage &&
           ((line = streamToPrint.ReadLine()) != null))
        {
            yPos = topMargin + (count *
               printFont.GetHeight(ev.Graphics));
            ev.Graphics.DrawString(line, printFont, Brushes.Black,
               leftMargin, yPos, new StringFormat());
            count++;
        }

        // If more lines exist, print another page.
        if (line != null)
            ev.HasMorePages = true;
        else
            ev.HasMorePages = false;
    }

    // The Windows Forms Designer requires the following procedure.
    private void InitializeComponent()
    {
        this.components = new System.ComponentModel.Container();
        this.printButton = new System.Windows.Forms.Button();

        this.ClientSize = new System.Drawing.Size(504, 381);
        this.Text = "Print Example";

        printButton.ImageAlign =
           System.Drawing.ContentAlignment.MiddleLeft;
        printButton.Location = new System.Drawing.Point(32, 110);
        printButton.FlatStyle = System.Windows.Forms.FlatStyle.Flat;
        printButton.TabIndex = 0;
        printButton.Text = "Print the file.";
        printButton.Size = new System.Drawing.Size(136, 40);
        printButton.Click += new System.EventHandler(printButton_Click);

        this.Controls.Add(printButton);
    }
}
```

Remarks

Typically, you create an instance of the PrintDocument class, set properties such as the DocumentName and PrinterSettings, and call the Print method to start the printing process. Handle the PrintPage event where you specify the output to print, by using the GraphicsGraphics property of the PrintPageEventArgs.

For more information about printing from a Windows Form application, see the Windows Forms Print Support. If you wish to print from a Windows Presentation Foundation application, see the System.Printing namespace.

Constructors

PrintDocument()	Initializes a new instance of the PrintDocument class.

Properties

CanRaiseEvents	Gets a value indicating whether the component can raise an event. (Inherited from Component)
Container	Gets the IContainer that contains the Component. (Inherited from Component)
DefaultPageSettings	Gets or sets page settings that are used as defaults for all pages to be printed.
DesignMode	Gets a value that indicates whether the Component is currently in design mode. (Inherited from Component)
DocumentName	Gets or sets the document name to display (for example, in a print status dialog box or printer queue) while printing the document.
Events	Gets the list of event handlers that are attached to this Component. (Inherited from Component)
OriginAtMargins	Gets or sets a value indicating whether the position of a graphics object associated with a page is located just inside the user-

specified margins or at the top-left corner of the printable area of the page.

PrintController	Gets or sets the print controller that guides the printing process.
PrinterSettings	Gets or sets the printer that prints the document.
Site	Gets or sets the ISite of the Component. (Inherited from Component)

Methods

CreateObjRef(Type)	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object. (Inherited from MarshalByRefObject)
Dispose()	Releases all resources used by the Component. (Inherited from Component)
Dispose(Boolean)	Releases the unmanaged resources used by the Component and optionally releases the managed resources. (Inherited from Component)
Equals(Object)	Determines whether the specified object is equal to the current object. (Inherited from Object)
GetHashCode()	Serves as the default hash function. (Inherited from Object)
GetLifetimeService()	**Obsolete.** Retrieves the current lifetime service object that controls the lifetime policy for this instance. (Inherited from MarshalByRefObject)
GetService(Type)	Returns an object that represents a service provided by the Component or by its Container. (Inherited from Component)
GetType()	Gets the Type of the current instance. (Inherited from Object)

InitializeLifetimeService()	**Obsolete.** Obtains a lifetime service object to control the lifetime policy for this instance. (Inherited from MarshalByRefObject)
MemberwiseClone()	Creates a shallow copy of the current Object. (Inherited from Object)
MemberwiseClone(Boolean)	Creates a shallow copy of the current MarshalByRefObject object. (Inherited from MarshalByRefObject)
OnBeginPrint(PrintEventArgs)	Raises the BeginPrint event. It is called after the Print() method is called and before the first page of the document prints.
OnEndPrint(PrintEventArgs)	Raises the EndPrint event. It is called when the last page of the document has printed.
OnPrintPage(PrintPageEventArgs)	Raises the PrintPage event. It is called before a page prints.
OnQueryPageSettings(QueryPageSettingsEventArgs)	Raises the QueryPageSettings event. It is called immediately before each PrintPage event.
Print()	Starts the document's printing process.
ToString()	Provides information about the print document, in string form.

Events

BeginPrint	Occurs when the Print() method is called and before the first page of the document prints.
Disposed	Occurs when the component is disposed by a call to the Dispose() method. (Inherited from Component)
EndPrint	Occurs when the last page of the document has printed.
PrintPage	Occurs when the output to print for the current page is needed.

QueryPageSettings	Occurs immediately before each PrintPage event.

Applies to

Product	Versions
.NET	5.0, 6.0 preview 3
.NET Core	3.0, 3.1
.NET Framework	1.1, 2.0, 3.0, 3.5, 4.0, 4.5, 4.5.1, 4.5.2, 4.6, 4.6.1, 4.6.2, 4.7, 4.7.1, 4.7.2, 4.8
.NET Platform Extensions	2.1, 2.2, 3.0, 3.1, 5.0, 6.0 preview 3

See also

- Graphics
- PageSettings
- PrintController
- PrinterSettings
- How to: Print a Multi-Page Text File in Windows Forms

Is this page helpful?

👍 Yes   👎 No