**HttpResponse Class**

Encapsulates HTTP-response information from an ASP.NET operation.

**Namespace:**  [System.Web](http://msdn.microsoft.com/en-us/library/system.web.aspx)  
**Assembly:**  System.Web (in System.Web.dll)

http://i.msdn.microsoft.com/Global/Images/clear.gif Syntax

Visual Basic (Declaration)

<AspNetHostingPermissionAttribute(SecurityAction.LinkDemand, Level := AspNetHostingPermissionLevel.Minimal)> \_

Public NotInheritable Class HttpResponse

Visual Basic (Usage)

Dim instance As HttpResponse

C#

[AspNetHostingPermissionAttribute(SecurityAction.LinkDemand, Level = AspNetHostingPermissionLevel.Minimal)]

public sealed class HttpResponse

Visual C++

[AspNetHostingPermissionAttribute(SecurityAction::LinkDemand, Level = AspNetHostingPermissionLevel::Minimal)]

public ref class HttpResponse sealed

JScript

public final class HttpResponse

http://i.msdn.microsoft.com/Global/Images/clear.gif Remarks

The methods and properties of the **HttpResponse** class are exposed through the [Response](http://msdn.microsoft.com/en-us/library/system.web.httpapplication.response.aspx) property of the [HttpApplication](http://msdn.microsoft.com/en-us/library/system.web.httpapplication.aspx), [HttpContext](http://msdn.microsoft.com/en-us/library/system.web.httpcontext.aspx), [Page](http://msdn.microsoft.com/en-us/library/system.web.ui.page.aspx), and [UserControl](http://msdn.microsoft.com/en-us/library/system.web.ui.usercontrol.aspx) classes.

The following methods of the **HttpResponse** class are supported only in postback scenarios and not in asynchronous postback scenarios:

* [BinaryWrite](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.binarywrite.aspx)
* [Clear](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.clear.aspx)
* [ClearContent](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.clearcontent.aspx)
* [ClearHeaders](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.clearheaders.aspx)
* [Close](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.close.aspx)
* [End](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.end.aspx)
* [Flush](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.flush.aspx)
* [TransmitFile](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.transmitfile.aspx)
* [Write](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.write.aspx)
* [WriteFile](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.writefile.aspx)
* [WriteSubstitution](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.writesubstitution.aspx)

Partial-page updates are enabled when you use [UpdatePanel](http://msdn.microsoft.com/en-us/library/system.web.ui.updatepanel.aspx) controls to update selected regions of a page instead of updating the whole page with a postback. For more information, see [UpdatePanel Control Overview](http://msdn.microsoft.com/en-us/library/bb386454.aspx) and [Partial-Page Rendering Overview](http://msdn.microsoft.com/en-us/library/bb386573.aspx).

**HttpResponse** is introduced in the .NET Framework version 3.5. For more information, see [.NET Framework 3.5 Architecture](http://msdn.microsoft.com/en-us/library/bb822049.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Examples

The following example draws three overlapping rectangles when the page is requested. The code begins by setting the [ContentType](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.contenttype.aspx) property to image/jpeg, so that the entire page will be rendered as a JPEG image. The code then calls the [Clear](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.clear.aspx) method to ensure that no extraneous content is sent with this response. Next, the code sets the [BufferOutput](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.bufferoutput.aspx) property to true so that the page is completely processed before it is sent to the requesting client. Two objects used to draw the rectangles are then created: a [Bitmap](http://msdn.microsoft.com/en-us/library/system.drawing.bitmap.aspx) and a [Graphics](http://msdn.microsoft.com/en-us/library/system.drawing.graphics.aspx) object. The variables created in the page are used as coordinates to draw the rectangles and a string that appears inside the largest rectangle.

When the three rectangles and the string that appears inside them are drawn, the [Bitmap](http://msdn.microsoft.com/en-us/library/system.drawing.bitmap.aspx) is saved to the [Stream](http://msdn.microsoft.com/en-us/library/system.io.stream.aspx) object that is associated with the [OutputStream](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.outputstream.aspx) property and its format is set to JPEG. The code calls the [Dispose](http://msdn.microsoft.com/en-us/library/system.drawing.image.dispose.aspx) and [Dispose](http://msdn.microsoft.com/en-us/library/system.drawing.graphics.dispose.aspx) methods to release the resources used by the two drawing objects. Lastly, the code calls the [Flush](http://msdn.microsoft.com/en-us/library/system.web.httpresponse.flush.aspx) method to send the buffered response to the requesting client.

Visual Basic

[[http://i.msdn.microsoft.com/Global/Images/clear.gif](javascript:CopyCode('ctl00_rs1_mainContentContainer_ctl91VisualBasic');)Copy Code](javascript:CopyCode('ctl00_rs1_mainContentContainer_ctl91VisualBasic');)

<%@ Page Language="VB" %>

<%@ import Namespace="System.Drawing" %>

<%@ import Namespace="System.Drawing.Imaging" %>

<%@ import Namespace="System.Drawing.Drawing2D" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<script runat="server">

Private Sub Page\_Load(sender As Object, e As EventArgs)

' Set the page's content type to JPEG files

' and clear all response headers.

Response.ContentType = "image/jpeg"

Response.Clear()

' Buffer response so that page is sent

' after processing is complete.

Response.BufferOutput = True

' Create a font style.

Dim rectangleFont As New Font( \_

"Arial", 10, FontStyle.Bold)

' Create integer variables.

Dim height As Integer = 100

Dim width As Integer = 200

' Create a random number generator and create

' variable values based on it.

Dim r As New Random()

Dim x As Integer = r.Next(75)

Dim a As Integer = r.Next(155)

Dim x1 As Integer = r.Next(100)

' Create a bitmap and use it to create a

' Graphics object.

Dim bmp As New Bitmap( \_

width, height, PixelFormat.Format24bppRgb)

Dim g As Graphics = Graphics.FromImage(bmp)

g.SmoothingMode = SmoothingMode.AntiAlias

g.Clear(Color.LightGray)

' Use the Graphics object to draw three rectangles.

g.DrawRectangle(Pens.White, 1, 1, width - 3, height - 3)

g.DrawRectangle(Pens.Aquamarine, 2, 2, width - 3, height - 3)

g.DrawRectangle(Pens.Black, 0, 0, width, height)

' Use the Graphics object to write a string

' on the rectangles.

g.DrawString("ASP.NET Samples", rectangleFont, SystemBrushes.WindowText, New PointF(10, 40))

' Apply color to two of the rectangles.

g.FillRectangle( \_

New SolidBrush( \_

Color.FromArgb(a, 255, 128, 255)), \_

x, 20, 100, 50)

g.FillRectangle( \_

New LinearGradientBrush( \_

New Point(x, 10), \_

New Point(x1 + 75, 50 + 30), \_

Color.FromArgb(128, 0, 0, 128), \_

Color.FromArgb(255, 255, 255, 240)), \_

x1, 50, 75, 30)

' Save the bitmap to the response stream and

' convert it to JPEG format.

bmp.Save(Response.OutputStream, ImageFormat.Jpeg)

' Release memory used by the Graphics object

' and the bitmap.

g.Dispose()

bmp.Dispose()

' Send the output to the client.

Response.Flush()

End Sub 'Page\_Load

</script>

<html >

<head>

<title>ASP.NET Example</title>

</head>

<body>

<form id="form1" runat="server">

</form>

</body>

</html>

C#

[[http://i.msdn.microsoft.com/Global/Images/clear.gif](javascript:CopyCode('ctl00_rs1_mainContentContainer_ctl92CSharp');)Copy Code](javascript:CopyCode('ctl00_rs1_mainContentContainer_ctl92CSharp');)

<%@ Page Language="C#" %>

<%@ import Namespace="System.Drawing" %>

<%@ import Namespace="System.Drawing.Imaging" %>

<%@ import Namespace="System.Drawing.Drawing2D" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<script runat="server">

private void Page\_Load(object sender, EventArgs e)

{

// Set the page's content type to JPEG files

// and clear all response headers.

Response.ContentType = "image/jpeg";

Response.Clear();

// Buffer response so that page is sent

// after processing is complete.

Response.BufferOutput = true;

// Create a font style.

Font rectangleFont = new Font(

"Arial", 10, FontStyle.Bold);

// Create integer variables.

int height = 100;

int width = 200;

// Create a random number generator and create

// variable values based on it.

Random r = new Random();

int x = r.Next(75);

int a = r.Next(155);

int x1 = r.Next(100);

// Create a bitmap and use it to create a

// Graphics object.

Bitmap bmp = new Bitmap(

width, height, PixelFormat.Format24bppRgb);

Graphics g = Graphics.FromImage(bmp);

g.SmoothingMode = SmoothingMode.AntiAlias;

g.Clear(Color.LightGray);

// Use the Graphics object to draw three rectangles.

g.DrawRectangle(Pens.White, 1, 1, width-3, height-3);

g.DrawRectangle(Pens.Aquamarine, 2, 2, width-3, height-3);

g.DrawRectangle(Pens.Black, 0, 0, width, height);

// Use the Graphics object to write a string

// on the rectangles.

g.DrawString(

"ASP.NET Samples", rectangleFont,

SystemBrushes.WindowText, new PointF(10, 40));

// Apply color to two of the rectangles.

g.FillRectangle(

new SolidBrush(

Color.FromArgb(a, 255, 128, 255)),

x, 20, 100, 50);

g.FillRectangle(

new LinearGradientBrush(

new Point(x, 10),

new Point(x1 + 75, 50 + 30),

Color.FromArgb(128, 0, 0, 128),

Color.FromArgb(255, 255, 255, 240)),

x1, 50, 75, 30);

// Save the bitmap to the response stream and

// convert it to JPEG format.

bmp.Save(Response.OutputStream, ImageFormat.Jpeg);

// Release memory used by the Graphics object

// and the bitmap.

g.Dispose();

bmp.Dispose();

// Send the output to the client.

Response.Flush();

}

</script>

<html >

<head>

<title>ASP.NET Example</title>

</head>

<body>

<form id="form1" runat="server">

</form>

</body>

</html>

http://i.msdn.microsoft.com/Global/Images/clear.gif Permissions

* [AspNetHostingPermission](http://msdn.microsoft.com/en-us/library/system.web.aspnethostingpermission.aspx)

for operating in a hosted environment. Demand value: [LinkDemand](http://msdn.microsoft.com/en-us/library/system.security.permissions.securityaction.linkdemand.aspx). Permission value: [Minimal](http://msdn.microsoft.com/en-us/library/system.web.aspnethostingpermissionlevel.minimal.aspx)

http://i.msdn.microsoft.com/Global/Images/clear.gif Inheritance Hierarchy

[System..::.Object](http://msdn.microsoft.com/en-us/library/system.object.aspx)  
  **System.Web..::.HttpResponse**

http://i.msdn.microsoft.com/Global/Images/clear.gif Thread Safety

Any public **static** (**Shared** in Visual Basic) members of this type are thread safe. Any instance members are not guaranteed to be thread safe.

http://i.msdn.microsoft.com/Global/Images/clear.gif Platforms

Windows Vista, Windows XP SP2, Windows XP Media Center Edition, Windows XP Professional x64 Edition, Windows XP Starter Edition, Windows Server 2003, Windows Server 2000 SP4, Windows Millennium Edition, Windows 98

The .NET Framework and .NET Compact Framework do not support all versions of every platform. For a list of the supported versions, see [.NET Framework System Requirements](http://msdn.microsoft.com/en-us/library/8z6watww.aspx).

http://i.msdn.microsoft.com/Global/Images/clear.gif Version Information

**.NET Framework**

Supported in: 3.5, 3.0, 2.0, 1.1, 1.0