

# EditWorks v4.0 Setup Guide

<http://www.editworkspro.com>

## Table Of Contents

Legal Disclaimer .....	2
Introduction .....	3
Web Server Requirements .....	4
Uploading the Files .....	5
Testing EditWorks .....	6
EditWorks Functions .....	7
EditWorks Image Manager .....	28
EditWorks Spell Checker .....	30
Using Multiple EditWorks Controls .....	31
Conclusion .....	33

## Legal Disclaimer

Software Version: 4.0 - 04/02/2003 (Usage Guide). Copyright © 2003 EditWorksPro.com (referred to as EditWorks from here on in) - All Rights Reserved. THIS COPYRIGHT INFORMATION MUST REMAIN INTACT AND MAY NOT BE MODIFIED IN ANY WAY.

When you purchased this software you agreed to accept the terms of this Agreement. This Agreement is a legal contract, which specifies the terms of the license and warranty limitation between you and 'EditWorks'. You should carefully read the following terms and conditions before installing or using this software. Unless you have a different license agreement obtained from 'EditWorks', installation or use of this software indicates your acceptance of the license and warranty limitation terms contained in this Agreement. If you do not agree to the terms of this Agreement, promptly delete and destroy all copies of the Software.

Versions of the Software: Only one licensed copy of EditWorks may be used on one web site.

License to Redistribute: Distributing the software and/or documentation with other products (commercial or otherwise) by any means without prior written permission from 'EditWorks' is forbidden. All rights to the EditWorks software and documentation not expressly granted under this Agreement are reserved to 'EditWorks'.

Disclaimer of Warranty: THIS SOFTWARE AND ACCOMPANYING DOCUMENTATION ARE PROVIDED "AS IS" AND WITHOUT WARRANTIES AS TO PERFORMANCE OF MERCHANTABILITY OR ANY OTHER WARRANTIES WHETHER EXPRESSED OR IMPLIED. BECAUSE OF THE VARIOUS HARDWARE AND SOFTWARE ENVIRONMENTS INTO WHICH EditWorks MAY BE USED, NO WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE IS OFFERED. THE USER MUST ASSUME THE ENTIRE RISK OF USING THIS PROGRAM. ANY LIABILITY OF 'EditWorks' WILL BE LIMITED EXCLUSIVELY TO PRODUCT REPLACEMENT OR REFUND OF PURCHASE PRICE. IN NO CASE SHALL 'EditWorks' BE LIABLE FOR ANY INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES OR LOSS, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR THE INABILITY TO USE EQUIPMENT OR ACCESS DATA, WHETHER SUCH DAMAGES ARE BASED UPON A BREACH OF EXPRESS OR IMPLIED WARRANTIES, BREACH OF CONTRACT, NEGLIGENCE, STRICT TORT, OR ANY OTHER LEGAL THEORY. THIS IS TRUE EVEN IF 'EditWorks' IS ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO CASE WILL 'EditWorks' OR LIABILITY EXCEED THE AMOUNT OF THE LICENSE FEE ACTUALLY PAID BY LICENSEE TO 'EditWorks'.

Warning: This program is protected by copyright law. Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under the law.

## **Introduction**

EditWorks is a custom ASP/PHP class that was developed as a replacement for the standard <TextArea> tag in content management systems. It is 100% HTML and JavaScript code on the client, and ASP/PHP code on the server, and can be setup on your web server in just 5 minutes, simply by following this setup guide.

## Web Server Requirements

Before you can install EditWorks, you must make sure that your web server meets the minimum system requirements, which are shown below:

### ASP Version:

- **Operating System:** Windows NT/2000
- **Web Server:** IIS5
- **Scripting Language:** ASP 3.0 and VBScript 5.5

Your IIS web server will need VBScript version 5.5 or above. You can check this by creating a new page called checkversion.asp with the following code:

```
You are running <%= ScriptEngine %> version <%=ScriptEngineMajorVersion %>.<%=ScriptEngineMinorVersion%>
```

Save checkversion.asp in your IIS root directory and run it in your web browser. If VBScript version 5.5 or later is reported, then you can run EditWorks.

### PHP Version:

- **Operating System:** Windows NT/2000 or Linux/Unix
- **Web Server:** Apache Web Server or IIS5
- **Scripting Language:** PHP 4.1 or above
- **Other:** There must be NO restrictions on browser-based file uploading if you want image uploading to work. This includes but is not limited to open\_basedir, file\_upload, etc. If you are unsure, please check with your web host before purchasing

Your Apache/IIS web server will need PHP 4.1 or above. You can check this by creating a new page called checkversion.php with the following code:

```
<?php phpinfo(); ?>
```

Save checkversion.php in your Apache/IIS root directory and run it in your web browser. The first line will tell you PHP's version number. If it reads "PHP Version 4.1.0" or above, then you can run EditWorks.

## Uploading the Files

You should have downloaded the EditWorks zip file. You received this in an email after your payment was accepted. Use an unzip program such as [WinZip](#) to extract the files to a temporary directory on your PC, such as c:\myTemp. Once extracted, move into the c:\myTemp directory using Windows explorer. You should see a directory called "ew" and an ewtest file.

You now need to FTP into your web server and upload both the "ew" directory and the ewtest file into your **root directory**.

After your FTP program has finished uploading the files, EditWorks is ready to test, which is what we will look at next.

## Testing EditWorks

Depending on which version you purchased, you will have uploaded either an ewtest.asp or ewtest.php file along with your "ew" directory. To test EditWorks, create a directory called "myImages" in the root directory of your web server. This will allow you to experiment with image uploading, etc.

Next, run the ewtest.asp or ewtest.php file in Internet Explorer version 5.5 or above:

<http://www.yoursite.com/ewtest.asp>

**OR**

<http://www.yoursite.com/ewtest.php>

The EditWorks control should appear on the page, complete with all buttons on the toolbar and tabs at the bottom of the control.

**Note:** If buttons are not working or you cannot edit the contents of the control, then make sure you are using Internet Explorer v5.5 or above on a Windows PC with JavaScript enabled.

To make sure everything is working, enter a value in the text area and then click on the "Get HTML" button at the bottom of the control. You should see the HTML generated by the EditWorks control appear in the text box at the bottom of the page.

Congratulations, your setup of EditWorks was a success!

## EditWorks Functions

There are many functions that you can use to manipulate both the EditWorks control and its contents. Over the next couple of pages, these functions will be explained, and an example will be provided for each one.

First off, we must learn how to instantiate, or create the EditWorks class. There are 3 steps involved. Firstly, you must include the EditWorks class file. Secondly, you must create a new EditWorks class object. Thirdly, you must assign a **unique** name to your EditWorks control, as shown below:

### ASP Version

```
<!-- #INCLUDE file="ew/class.editworks.asp" -->  
  
<%  
  
' Create a new EditWorks class object  
dim myEW  
set myEW = new ew  
myEW.SetName("myEditWorksControl")  
  
%>
```

### PHP Version

```
<?php  
  
include_once("ew/class.editworks.php");  
  
// Create a new EditWorks class object  
$myEW = new ew;  
$myEW->SetName("myEditWorksControl");  
  
?>
```

Once you've created an EditWorks class object, you call functions against that object, for example:

### ASP Version

```
myEW.ShowControl "90%", "200pt", "/myImages"
```

### PHP Version

```
$myEW->ShowControl("90%", "200pt", "/myImages");
```

### **The SetName Function**

This function assigns a name to your EditWorks control. It must be called before any other function. This name is used to differentiate between multiple EditWorks controls on the same HTML page.

#### **Function Syntax:**

```
void SetName ( string CtrlName )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.SetName("myEditWorksControl")</code>	<code>\$myEW-&gt;SetName("myEditWorksControl");</code>

### **The SetLanguage Function**

This function allows you to set the language that will be used by the spell checker when it returns suggestions for misspelt words.

#### **Function Syntax:**

```
void SetLanguage ( int Language )
```

- Set Language to 1 or EW\_AMERICAN to return spelling suggestions for American English.
- Set Language to 2 or EW\_BRITISH to return spelling suggestions for British English.
- Set Language to 3 or EW\_CANADIAN to return spelling suggestions for Canadian English.

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.SetLanguage(EW_AMERICAN)</code>	<code>\$myEW-&gt;SetLanguage(EW_AMERICAN);</code>

### **The SetValue Function**

This function sets the initial value of the HTML to be displayed in the EditWorks control.

#### **Function Syntax:**

```
void SetValue ( string HTMLValue )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.SetValue("&lt;b&gt;Hello&lt;/b&gt;")</code>	<code>\$myEW-&gt;SetValue("&lt;b&gt;Hello&lt;/b&gt;");</code>

### **The GetValue Function**

This function gets the resultant HTML from the EditWorks control once the control has been submitted as part of a form.

#### **Function Syntax:**

string GetValue ( bool ConvertQuotes )

- If ConvertQuotes is true, then the string returned from GetValue will have single quotes and double quotes escaped so that it can be inserted directly into a database query.

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>HTMLCode = myEW.GetValue(true)</code>	<code>\$HTMLCode = \$myEW-&gt;GetValue(true);</code>

### **The ShowControl Function**

This function outputs all of the necessary HTML and JavaScript to display the EditWorks control as part of a web page.

#### **Function Syntax:**

void ShowControl ( mixed Width, mixed Height, string ImagePath )

- Width specifies the width of the control when it is displayed.
- Height specified the height of the control when it is displayed.
- ImagePath is the absolute path to a directory from where all images will be displayed in the image manager.

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.ShowControl "90%", "200pt", "/myImages"</code>	<code>\$myEW-&gt;ShowControl("90%", "200pt", "/myImages");</code>

### **The SetFontList Function**

This function allows you to change the default list of fonts that are displayed in the font drop down list. Each font name should be separated by a comma. If you do not call this function then the default list will be displayed.

#### **Function Syntax:**

void SetFontList ( string FontList )

#### **Function Example:**

**ASP Version**

```
myEW.SetFontList("Arial,Verdana,Tahoma")
```

**PHP Version**

```
$myEW->SetFontList("Arial,Verdana,Tahoma");
```

**The SetFontSizeList Function**

This function allows you to change the default list of font sizes that are displayed in the font size drop down list. Each font size should be separated by a comma. If you do not call this function then the default list will be displayed.

**Function Syntax:**

```
void SetFontSizeList ( string FontSizeList )
```

**Function Example:****ASP Version**

```
myEW.SetFontSizeList("2,3,4,5")
```

**PHP Version**

```
$myEW->SetFontSizeList("2,3,4,5");
```

**The SetImageDisplayType Function**

This function determines how images will appear in the image manager.

**Function Syntax:**

```
void SetImageDisplayType ( int DisplayType )
```

- If DisplayType is 0 or ew\_IMAGE\_TYPE\_ROW, then images will be displayed in a tabular format without a thumbnail preview.
- If DisplayType is 1 or ew\_IMAGE\_TYPE\_THUMBNAIL, then images will be displayed 4-per-line as thumbnails.

**Function Example:****ASP Version**

```
myEW.SetImageDisplayType  
ew_IMAGE_TYPE_THUMBNAIL
```

**PHP Version**

```
$myEW->SetImageDisplayType(ew_IMAGE_TYPE_THUMBNAIL);
```

**The SetDocumentType Function**

This function is used to specify whether a complete HTML document or only a HTML snippet is being edited.

**Function Syntax:**

```
void SetDocumentType ( int DocType )
```

- Set DocType to 0 or ew\_DOC\_TYPE\_SNIPPET to indicate that you are editing a snippet of HTML code.

- Set DocType to 1 or ew\_DOC\_TYPE\_HTML\_PAGE to indicate that you are editing a complete HTML page.

#### Function Example:

ASP Version	PHP Version
myEW.SetDocumentType ew_DOC_TYPE_SNIPPET	\$myEW->SetDocumentType(ew_DOC_TYPE_SNIPPET);

#### **The SetPathType Function**

This function determines how the src attribute of a link/image will be set.

#### Function Syntax:

```
void SetPathType ( int PathType )
```

- If PathType is 0 or ew\_PATH\_TYPE\_FULL then links/images will have the full path specified in their src attribute, such as http://www.mysite.com/test.html.
- If PathType is 1 or ew\_PATH\_ABSOLUTE then links/images will have the absolute path specified in their src attribute, such as /myimage.gif.

#### Function Example:

ASP Version	PHP Version
myEW.SetPathType ew_PATH_TYPE_FULL	\$myEW->SetPathType(ew_PATH_TYPE_FULL);

#### **The DisableXHTMLFormatting Function**

This function turns of XHTML formatting on all code produced by the EditWorks control.

#### Function Syntax:

```
void DisableXHTMLFormatting ( )
```

#### Function Example:

ASP Version	PHP Version
myEW.DisableXHTMLFormatting	\$myEW->DisableXHTMLFormatting();

#### **The DisableSingleLineReturn Function**

This function converts any hard carriage returns from the default <p> tag to the <br> tag in the code generated by the EditWorks control.

#### Function Syntax:

```
void DisableSingleLineReturn ( )
```

#### Function Example:

**ASP Version**`myEW.DisableSingleLineReturn`**PHP Version**`$myEW->DisableSingleLineReturn();`**The DisableImageDeleting Function**

This function hides the delete link in the image manager.

**Function Syntax:**

```
void DisableImageDeleting ( )
```

**Function Example:****ASP Version**`myEW.DisableImageDeleting`**PHP Version**`$myEW->DisableImageDeleting();`**The DisableImageUploading Function**

This function hides the upload file box in the image manager.

**Function Syntax:**

```
void DisableImageUploading ( )
```

**Function Example:****ASP Version**`myEW.DisableImageUploading`**PHP Version**`$myEW->DisableImageUploading();`**The DisableSourceMode Function**

This function hides the source tab at the bottom of the EditWorks control.

**Function Syntax:**

```
void DisableSourceMode ( )
```

**Function Example:****ASP Version**`myEW.DisableSourceMode`**PHP Version**`$myEW->DisableSourceMode();`**The DisablePreviewMode Function**

This function hides the preview tab at the bottom of the EditWorks control.

**Function Syntax:**

```
void DisablePreviewMode ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.DisablePreviewMode</code>	<code>\$myEW-&gt;DisablePreviewMode();</code>

### **The EnableGuidelines Function**

This function enables dashed guidelines for tables and forms by default.

### Function Syntax:

```
void EnableGuidelines ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.EnableGuidelines</code>	<code>\$myEW-&gt;EnableGuidelines();</code>

### **The LoadHTMLFromMySQLQuery Function (PHP Version Only)**

This function can be used to connect to a MySQL database and retrieve the contents of a field, which will then automatically become the contents of the EditWorks control. You must call this function before you call the ShowControl function.

The DatabaseQuery parameter must be a select query. If it returns more than one row/field, then the first field in the first row will become the value of the EditWorks control.

**Note:**This function will work on the PHP version of EditWorks Professional only.

### Function Syntax:

```
bool LoadHTMLFromMySQLQuery ( string DatabaseServer, string DatabaseName, string DatabaseUser, string DatabasePassword, string DatabaseQuery, reference ErrorDesc )
```

### Function Example:

PHP Version
<pre>\$errDesc = ""; myEW-&gt;SetName("myEditWorksControl");  // Assumes \$myEW is an EditWorks control myEW-&gt;LoadHTMLFromMySQLQuery("localhost", "testdatabase", "admin", "password", "select pName from people where pId = 10", \$errDesc);  // The \$errDesc variable is passed by reference to the function if(\$errDesc != "") {     echo "An error occurred: \$errDesc"; }</pre>

### **The LoadHTMLFromAccessQuery Function (ASP Version Only)**

This function can be used to connect to a Microsoft Access database and retrieve the contents of a field, which will then automatically become the contents of the EditWorks control. You must call this function before you call the ShowControl function.

The DatabaseFile parameter must be the physical path to the database, such as c:\mydatabase.mdb. The DatabaseQuery parameter must be a select query. If it returns more than one row/field, then the first field in the first row will become the value of the EditWorks control.

**Note:**This function will work on the ASP version of EditWorks Professional only.

#### **Function Syntax:**

```
bool LoadHTMLFromAccessQuery ( string DatabaseFile, string DatabaseQuery, reference ErrorDesc )
```

#### **Function Example:**

##### **ASP Version**

```
dim errDesc

errDesc = ""
myEW.SetName("myEditWorksControl")

' Assumes myEW is an EditWorks control
myEW.LoadHTMLFromAccessQuery "c:\test.mdb", "select testField from testTable",
errDesc

' The errDesc variable is passed by reference to the function
if errDesc <> "" then
    Response.Write "An error occurred: " & errDesc
end if
```

### **The LoadHTMLFromSQLServerQuery Function (ASP Version Only)**

This function can be used to connect to a Microsoft SQL Server database and retrieve the contents of a field, which will then automatically become the contents of the EditWorks control. You must call this function before you call the ShowControl function.

The DatabaseQuery parameter must be a select query. If it returns more than one row/field, then the first field in the first row will become the value of the EditWorks control.

**Note:**This function will work on the ASP version of EditWorks Professional only.

#### **Function Syntax:**

```
bool LoadHTMLFromSQLServerQuery ( string DatabaseServer, string DatabaseName, string DatabaseUser, string DatabasePassword, string DatabaseQuery, reference ErrorDesc )
```

#### **Function Example:**

### ASP Version

```
dim errDesc

errDesc = ""
myEW.SetName("myEditWorksControl")

' Assumes myEW is an EditWorks control
myEW.LoadHTMLFromSQLServerQuery "localhost", "testDatabase", "testUser",
"testPassword", "select testField from testTable", errDesc

' The errDesc variable is passed by reference to the function
if errDesc <> "" then
    Response.Write "An error occurred: " & errDesc
end if
```

### **The LoadFromFile Function**

This function can be used to load the contents of a HTML file, which will then automatically become the contents of the EditWorks control. You must call this function before you call the ShowControl function.

#### **Function Syntax:**

```
bool LoadFromFile ( string FilePath, reference ErrorDesc )
```

#### **Function Example:**

### ASP Version

```
dim errDesc

errDesc = ""

' Assumes myEW is an EditWorks control
myEW.LoadFromFile "mysite.html", errDesc
myEW.SetName("myEditWorksControl")

' The errDesc variable is passed by reference to the function
if errDesc <> "" then
    Response.Write "An error occurred: " & errDesc
end if
```


### PHP Version

```
$errDesc = "";

// Assumes myEW is an EditWorks control
$myEW->LoadFromFile("mysite.html", $errDesc);
$myEW->SetName("myEditWorksControl");

// The $errDesc variable is passed by reference to the function
if($errDesc != "")
{
    echo "An error occurred: $errDesc";
}
```

### **The AddCustomInsert Function**

This function can be used to add a custom insert snippet. A custom insert is basically a piece of HTML code that you can insert into the EditWorks control quickly. After calling the AddCustomInsert function, the  button will appear on the toolbar. Simply click on it and your list of custom inserts will appear.

### **Function Syntax:**

```
void AddCustomInsert ( string InsertName, string InsertHTMLCode )
```

### **Function Example:**

### ASP Version

```
' Assumes myEW is an EditWorks control  
myEW.SetName("myEditWorksControl")
```

```
myEW.AddCustomInsert "EditWorks Logo", "<img  
src='http://www.editworkspro.com/logo.gif'>"
```

### PHP Version

```
// Assumes myEW is an EditWorks control  
$myEW->AddCustomInsert("EditWorks Logo", "<img  
src='http://www.editworkspro.com/logo.gif'>");
```

### The SetSnippetStyleSheet Function

This function allows you to specify the location of a stylesheet to use when editing code snippets.

**Note:** You must be in snippet editing mode (i.e. SetDocumentType(EW\_DOC\_TYPE\_SNIPPET) to specify a snippet stylesheet. Due to Internet Explorer 6.0 cross-domain security, the stylesheet that you are using must be located on the same domain name as where EditWorks Professional is installed.

### Function Syntax:

```
void SetSnippetStyleSheet ( string StyleSheetURL )
```

### Function Example:

#### ASP Version

```
' Assumes myEW is an EditWorks control  
' MUST be in snippet editing mode not full page editing mode  
myEW.SetName("myEditWorksControl")
```

```
myEW.SetDocumentType EW_DOC_TYPE_SNIPPET  
myEW.SetSnippetStyleSheet("mystyle.css")
```

#### PHP Version

```
// Assumes $myEW is an EditWorks control  
// MUST be in snippet editing mode not full page editing mode  
$myEW->SetName("myEditWorksControl");
```

```
$myEW->SetDocumentType(EW_DOC_TYPE_SNIPPET);  
$myEW->SetSnippetStyleSheet("mystyle.css");
```

### The SetTextAreaDimensions Function

When EditWorks is used in a browser that isn't Internet Explorer v5.0 or above (i.e. Netscape 4), then a <textarea> tag will be shown instead of the EditWorks WYSIWYG editing window. The SetTextAreaDimensions function can be used to set the rows and cols

attributes of this <textarea> tag.

### Function Syntax:

```
void SetTextAreaDimensions ( int Cols, int Rows )
```

### Function Example:

#### ASP Version

```
' Assumes myEW is an EditWorks control  
' Set <textarea> cols to 60 and rows to 30  
myEW.SetName("myEditWorksControl")  
myEW.SetTextAreaDimensions 60, 30
```

#### PHP Version

```
// Assumes $myEW is an EditWorks control  
// Set <textarea> cols to 60 and rows to 30  
$myEW->SetName("myEditWorksControl");  
$myEW->SetTextAreaDimensions(60, 30);
```

### The HideFullScreenButton Function

This function hides the fullscreen button on the toolbar.

### Function Syntax:

```
void HideFullScreenButton ( )
```

### Function Example:

#### ASP Version

```
myEW.HideFullScreenButton
```

#### PHP Version

```
$myEW->HideFullScreenButton();
```

### The HideSpellingButton Function

This function hides the spelling button on the toolbar.

### Function Syntax:

```
void HideSpellingButton ( )
```

### Function Example:

#### ASP Version

```
myEW.HideSpellingButton
```

#### PHP Version

```
$myEW->HideSpellingButton();
```

### **The HideRemoveTextFormattingButton Function**

This function hides the "remove text formatting" button on the toolbar.

#### **Function Syntax:**

```
void HideRemoveTextFormattingButton ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.HideRemoveTextFormattingButton</code>	<code>\$myEW-&gt;HideRemoveTextFormattingButton();</code>

### **The HideBoldButton Function**

This function hides the bold button on the toolbar.

#### **Function Syntax:**

```
void HideBoldButton ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.HideBoldButton</code>	<code>\$myEW-&gt;HideBoldButton();</code>

### **The HideUnderlineButton Function**

This function hides the underline button on the toolbar.

#### **Function Syntax:**

```
void HideUnderlineButton ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.HideUnderlineButton</code>	<code>\$myEW-&gt;HideUnderlineButton();</code>

### **The HideItalicButton Function**

This function hides the italic button on the toolbar.

#### **Function Syntax:**

```
void HideItalicButton ( )
```

#### **Function Example:**

**ASP Version**`myEW.HideItalicButton`**PHP Version**`$myEW->HideItalicButton();`**The HideStrikethroughButton Function**

This function hides the strikethrough button on the toolbar.

**Function Syntax:**

```
void HideStrikethroughButton ( )
```

**Function Example:****ASP Version**`myEW.HideStrikethroughButton`**PHP Version**`$myEW->HideStrikethroughButton();`**The HideNumberListButton Function**

This function hides the number list button on the toolbar.

**Function Syntax:**

```
void HideNumberListButton ( )
```

**Function Example:****ASP Version**`myEW.HideNumberListButton`**PHP Version**`$myEW->HideNumberListButton();`**The HideBulletedListButton Function**

This function hides the bullet list button on the toolbar.

**Function Syntax:**

```
void HideBulletedListButton ( )
```

**Function Example:****ASP Version**`myEW.HideBulletedListButton`**PHP Version**`$myEW->HideBulletedListButton();`**The HideDecreaseIndentButton Function**

This function hides the decrease indent button on the toolbar.

**Function Syntax:**

```
void HideDecreaseIndentButton ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.HideDecreaseIndentButton</code>	<code>\$myEW-&gt;HideDecreaseIndentButton();</code>

### **The HideIncreaseIndentButton Function**

This function hides the increase indent button on the toolbar.

### Function Syntax:

```
void HideIncreaseIndentButton ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.HideIncreaseIndentButton</code>	<code>\$myEW-&gt;HideIncreaseIndentButton();</code>

### **The HideLeftAlignButton Function**

This function hides the left align button on the toolbar.

### Function Syntax:

```
void HideLeftAlignButton ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.HideLeftAlignButton</code>	<code>\$myEW-&gt;HideLeftAlignButton();</code>

### **The HideCenterAlignButton Function**

This function hides the center align button on the toolbar.

### Function Syntax:

```
void HideCenterAlignButton ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.HideCenterAlignButton</code>	<code>\$myEW-&gt;HideCenterAlignButton();</code>

### **The HideRightAlignButton Function**

This function hides the right align button on the toolbar.

### Function Syntax:

void HideRightAlignButton ( )

**Function Example:**

ASP Version	PHP Version
<code>myEW.HideRightAlignButton</code>	<code>\$myEW-&gt;HideRightAlignButton();</code>

**The HideSuperScriptButton Function**

This function hides the super script button on the toolbar.

**Function Syntax:**

void HideRightSuperScriptButton ( )

**Function Example:**

ASP Version	PHP Version
<code>myEW.HideSuperScriptButton</code>	<code>\$myEW-&gt;HideSuperScriptButton();</code>

**The HideSubScriptButton Function**

This function hides the sub script button on the toolbar.

**Function Syntax:**

void HideRightSubScriptButton ( )

**Function Example:**

ASP Version	PHP Version
<code>myEW.HideSubScriptButton</code>	<code>\$myEW-&gt;HideSubScriptButton();</code>

**The HideJustifyButton Function**

This function hides the justify button on the toolbar.

**Function Syntax:**

void HideJustifyButton ( )

**Function Example:**

ASP Version	PHP Version
<code>myEW.HideJustifyButton</code>	<code>\$myEW-&gt;HideJustifyButton();</code>

**The HideHorizontalRuleButton Function**

This function hides the horizontal rule button on the toolbar.

### Function Syntax:

```
void HideHorizontalRuleButton ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.HideHorizontalRuleButton</code>	<code>\$myEW-&gt;HideHorizontalRuleButton();</code>

### **The HideLinkButton Function**

This function hides the link button on the toolbar.

### Function Syntax:

```
void HideLinkButton ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.HideLinkButton</code>	<code>\$myEW-&gt;HideLinkButton();</code>

### **The HideAnchorButton Function**

This function hides the anchor button on the toolbar.

### Function Syntax:

```
void HideAnchorButton ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.HideAnchorButton</code>	<code>\$myEW-&gt;HideAnchorButton();</code>

### **The HideMailLinkButton Function**

This function hides the mail link button on the toolbar.

### Function Syntax:

```
void HideMailLinkButton ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.HideMailLinkButton</code>	<code>\$myEW-&gt;HideMailLinkButton();</code>

### **The HideHelpButton Function**

This function hides the help button on the toolbar.

#### **Function Syntax:**

```
void HideHelpButton ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.HideHelpButton</code>	<code>\$myEW-&gt;HideHelpButton();</code>

### **The HideFontList Function**

This function hides the font list on the toolbar.

#### **Function Syntax:**

```
void HideFontList ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.HideFontList</code>	<code>\$myEW-&gt;HideFontList();</code>

### **The HideSizeList Function**

This function hides the size list on the toolbar.

#### **Function Syntax:**

```
void HideSizeList ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.HideSizeList</code>	<code>\$myEW-&gt;HideSizeList();</code>

### **The HideFormatList Function**

This function hides the format list on the toolbar.

#### **Function Syntax:**

```
void HideFormatList ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
--------------------	--------------------

<code>myEW.HideFormatList</code>	<code>\$myEW-&gt;HideFormatList();</code>
----------------------------------	---

### **The HideStyleList Function**

This function hides the style list on the toolbar.

#### **Function Syntax:**

```
void HideStyleList ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.HideStyleList</code>	<code>\$myEW-&gt;HideStyleList();</code>

### **The HideForeColorButton Function**

This function hides the foreground color button on the toolbar.

#### **Function Syntax:**

```
void HideForeColorButton ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.HideForeColorButton</code>	<code>\$myEW-&gt;HideForeColorButton();</code>

### **The HideBackColorButton Function**

This function hides the background color button on the toolbar.

#### **Function Syntax:**

```
void HideBackColorButton ( )
```

#### **Function Example:**

<b>ASP Version</b>	<b>PHP Version</b>
<code>myEW.HideBackColorButton</code>	<code>\$myEW-&gt;HideBackColorButton();</code>

### **The HideTableButton Function**

This function hides the table button on the toolbar.

#### **Function Syntax:**

```
void HideTableButton ( )
```

#### **Function Example:**

**ASP Version**`myEW.HideTableButton`**PHP Version**`$myEW->HideTableButton();`**The HideFormButton Function**

This function hides the form button on the toolbar.

**Function Syntax:**

```
void HideFormButton ( )
```

**Function Example:****ASP Version**`myEW.HideFormButton`**PHP Version**`$myEW->HideFormButton();`**The HideImageButton Function**

This function hides the image button on the toolbar.

**Function Syntax:**

```
void HideImageButton ( )
```

**Function Example:****ASP Version**`myEW.HideImageButton`**PHP Version**`$myEW->HideImageButton();`**The HideSymbolButton Function**

This function hides the symbol button on the toolbar.

**Function Syntax:**

```
void HideSymbolButton ( )
```

**Function Example:****ASP Version**`myEW.HideSymbolButton`**PHP Version**`$myEW->HideSymbolButton();`**The HidePropertiesButton Function**

This function hides the properties button on the toolbar.

**Function Syntax:**

```
void HidePropertiesButton ( )
```

### Function Example:

ASP Version	PHP Version
<code>myEW.HidePropertiesButton</code>	<code>\$myEW-&gt;HidePropertiesButton();</code>

### **The HideCleanHTMLButton Function**

This function hides the clean HTML button on the toolbar.

### Function Syntax:

void HideCleanHTMLButton ( )

### Function Example:

ASP Version	PHP Version
<code>myEW.HideCleanHTMLButton</code>	<code>\$myEW-&gt;HideCleanHTMLButton();</code>

### **The HidePositionAbsoluteButton Function**

This function hides the absolute position button on the toolbar.

### Function Syntax:

void HidePositionAbsoluteButton ( )

### Function Example:

ASP Version	PHP Version
<code>myEW.HidePositionAbsoluteButton</code>	<code>\$myEW-&gt;HidePositionAbsoluteButton();</code>

### **The HideGuidelinesButton Function**

This function hides the guidelines button on the toolbar.

### Function Syntax:

void HideGuidelinesButton ( )

### Function Example:


ASP Version	PHP Version
<code>myEW.HideGuidelinesButton</code>	<code>\$myEW-&gt;HideGuidelinesButton();</code>

## EditWorks Image Manager

EditWorks includes an extremely powerful image manager which lets you easily insert or upload an image directly into your HTML content with just one click.

To load the image manager, you need to do 2 things. Firstly, when you call the ShowControl function, you need to specify an absolute path to your image directory as the third parameter:

ASP Version	PHP Version
<code>MyEW.SetName("myEditWorksControl")</code>	<code>\$myEW-&gt;SetName("myEditWorksControl");</code>
<code>myEW.ShowControl "90%", "200pt", "/myImages"</code>	<code>\$myEW-&gt;ShowControl("90%", "200pt", "/myImages");</code>

Secondly, once the EditWorks control is displayed in a page, you need to click on the  button on the EditWorks toolbar. We will now examine each part of the image manager.

### Using an Image From the Web

The "external image" box allows you to specify the location of an image on the Internet. To do this, just enter the URL of the image, such as <http://www.mysite.com/image.gif>.

Inserting an image from the web can be disabled by calling the DisableInsertImageFromWeb method of your EditWorks class, like this:

ASP Version	PHP Version
<code>myEW.DisableInsertImageFromWeb</code>	<code>\$myEW-&gt;DisableInsertImageFromWeb();</code>

### Uploading an Image

Under the "external image" box you will see a file upload box. This is where you can upload an image (GIF, JPEG or PNG). To upload an image, simply click on the "Browse..." button to select an image. Next, click on the "Upload" button. EditWorks will upload the image into the directory specified in the ShowControl function, and it will be displayed in the image manager.

Image uploading can be disabled by calling the DisableImageUploading method of your EditWorks class, like this:


ASP Version	PHP Version
<code>myEW.DisableImageUploading</code>	<code>\$myEW-&gt;DisableImageUploading();</code>

### Viewing an Image

Each image in the image manager has a "view" link. This will open the image in a new preview window, so that you can see the image in its entirety before inserting it into your content.

### Inserting an Image

To insert an image from the image manager into your HTML content, click on the "insert" link next to that image. The image manager window will close and the selected image will appear in your HTML content.

Once an image has been inserted, you can modify the properties of that image including its alignment and alt text. To do this, just select the image in your HTML content and click on the  button on the toolbar.

### Set Image as Page Background (Professional Version Only)

If the "backgd" link next to an image is clicked, then that image will be set as the background image in your HTML content. For this to work, you need to make sure EditWorks knows that you are editing a complete HTML page, by calling the SetDocumentType function with ew\_DOC\_TYPE\_HTML\_PAGE, like this:

ASP Version	PHP Version
<code>myEW.SetDocumentType ew_DOC_TYPE_HTML_PAGE</code>	<code>\$myEW-&gt;SetDocumentType(ew_DOC_TYPE_HTML_PAGE);</code>

### Deleting an Image (Professional Version Only)

Each image has a "delete" link. Clicking on this link will delete the image from your web server, and thus it will no longer display in the image manager.

You can disable the deletion of images by calling the DisableImageDeleting function, like this:

ASP Version	PHP Version
<code>myEW.DisableImageDeleting</code>	<code>\$myEW-&gt;DisableImageDeleting();</code>

### Disabling the Image Manager

If you would like to completely remove the image manager from EditWorks, then you need to call the HideImageButton function, like this:

ASP Version	PHP Version
<code>myEW.HideImageButton</code>	<code>\$myEW-&gt;HideImageButton();</code>

This will remove the image button from the EditWorks toolbar.

## EditWorks Spell Checker

The most popular feature of EditWorks is its integrated spell checking capabilities. You are not required to install any additional libraries or files on your web server to use the spell checker, and it will work straight away.

Shown below is a list of common frequently asked questions about the integrated spell checker in EditWorks.

**Q: How does the spell checker work?**

A: Basically, it is a hosted solution that we manage on one of our servers. It uses proprietary server-side functions that we have written to interact with a spelling library.

When the spell check button is clicked, the EditWorks control queries its contents for a list of words. These words are then passed through DHTML code to our spell-checking server, which returns a list of suggestions for each word.

When these suggestions are returned, EditWorks will step through each word and list the suggestions, giving you an option to change the misspelled word as required.

**Q: Does it cost anything extra to use?**

A: No. Spell checking is included for free with every purchase of EditWorks, so please feel free to use it as much as you like!

**Q: I want to sell EditWorks using a redistributable licence, and I want spelling enabled. What if my clients look at the URL of the spell check page? Will they know that I am using EditWorks?**

A: No. Our spell checker sits on a domain called spellcheckme.com. This domain is used exclusively for EditWorks spelling operations, and it is not branded in any way.

**Q: How fast is the EditWorks spell checker?**

A: We've tested the spell checker when there have been several dozen misspelled words in a document, and it takes no longer than 5-10 seconds for the spell checker to return a list of suggestions.

**Q: I don't want to use the spell checker. How can I remove it from the toolbar?**

A: Simple. Just call the HideSpellingButton function against your EditWorks control, like this:

ASP Version	PHP Version
<code>myEW.HideSpellingButton</code>	<code>\$myEW-&gt;HideSpellingButton();</code>

## Using Multiple EditWorks Controls

Along with the release of EditWorks version 4 comes the ability to use more than 1 EditWorks control on your web page. In fact, you can use as many as you like. Shown below are examples of how to use 2 EditWorks controls in 1 page:

**To display the controls on your web page:**

ASP Version	PHP Version
<pre>&lt;!-- #INCLUDE file="ew/class.editworks.asp" --&gt;</pre>	<pre>&lt;?php</pre>
<pre>&lt;%</pre>	<pre>include_once("ew/class.editworks.php");</pre>
<pre>' First EditWorks control dim myEW1 set myEW1 = new ew myEW1.SetName("ew1") myEW1.SetValue("&lt;b&gt;Control1&lt;/b&gt;")</pre>	<pre>// First EditWorks control \$myEW1 = new ew; \$myEW1-&gt;SetName("ew1"); \$myEW1-&gt;SetValue("&lt;b&gt;Control1&lt;/b&gt;");</pre>
<pre>' Second EditWorks control dim myEW2 set myEW2 = new ew myEW2.SetName("ew2") myEW1.SetValue("&lt;i&gt;Control2&lt;/i&gt;")</pre>	<pre>// Second EditWorks control \$myEW2 = new ew; \$myEW2-&gt;SetName("ew2"); \$myEW2-&gt;SetValue("&lt;i&gt;Control2&lt;/i&gt;");</pre>
<pre>' Show both controls myEW1.ShowControl "90%", "200pt", "/myImages"</pre>	<pre>// Show both controls \$myEW1-&gt;ShowControl("90%", "200pt", "/myImages");</pre>
<pre>myEW2.ShowControl "100%", "100%", "/myImages"</pre>	<pre>\$myEW2-&gt;ShowControl("100%", "100%", "/myImages");</pre>
<pre>%&gt;</pre>	<pre>?&gt;</pre>

**To get the HTML content from each control on the form-submitted page:**

ASP Version	PHP Version
<pre>&lt;!-- #INCLUDE file="ew/class.editworks.asp" --&gt;</pre>	<pre>&lt;?php</pre>
<pre>&lt;%</pre>	<pre>include_once("ew/class.editworks.php");</pre>
<pre>' First EditWorks control dim myEW1 set myEW1 = new ew myEW1.SetName("ew1")</pre>	<pre>// First EditWorks control \$myEW1 = new ew; \$myEW1-&gt;SetName("ew1");</pre>
<pre>' Second EditWorks control dim myEW2</pre>	<pre>// Second EditWorks control \$myEW2 = new ew; \$myEW2-&gt;SetName("ew2");</pre>

```
set myEW2 = new ew
myEW2.SetName("ew2")

' Show the first controls HTML
Response.Write("Value from first control
is:<br><br>" & myEW1.GetValue(false))

' Show the second controls HTML
Response.Write("Value from second control
is:<br><br>" & myEW2.GetValue(false))

%>

// Show the first controls HTML
echo "Value from first control
is:<br><br>" . $myEW1-
>GetValue(false);

// Show the second controls HTML
echo "Value from second control
is:<br><br>" . $myEW2-
>GetValue(false);

?>
```

## Conclusion

Thank you for purchasing EditWorks. Hopefully this setup guide has gotten you up and running in the shortest amount of time possible. If you are still having difficulties, please visit our [support page](#) for helpful tips, tricks and advice in our FAQ's and knowledge base.