



Welcome to the DBTS CSV Manager Suite of Extensions!

DB Technosystems is well known for quality, custom made software. The main fields of application are database driven scripts like membership scripts, Shopping Carts, directory scripts and other similar.

The WB CSV Manager suite of extensions is the result of a continuously rising request for a simple CSV File Manager that will allow webmasters to easily manage their grid formatted files and even create nicely formatted Lists and Detail pages from these files. The output is highly customizable and the script will be enriched with additional features in the near future.

Some of the features in the settings dialog box are not yet functional but have been included in the perspective of features that will be added soon. More details will be provided through this manual.

Thank you

DBTechnosystems.com

Requirements

The DBTS CSV Extensions require PHP 4.3 or higher support.

Safe Mode must be OFF and the folder in which they are used should have reading and writing permissions.

Current Version

Current version is 2.4.1.1

What's new

1. Added ability for images and hyperlinks in the "View" object
2. Added ability to "Double quote" records. This allows you to use multiline records or HTML code in the records.
3. Added ability to use the separator (in example semicolon or coma) in the records content
4. Added ability to restrict page access using the WWB Login script tools or the Single Page Protect tool.
5. Added ability to restrict user access to their records only.
6. Code improvements.

Fixes

Fixed default sorting issue.

Installation

The DBTS CSV extensions are installed just as any other WB extension: Through the extension Manager, you can perform either an Automatic installation or a manual installation after you have downloaded the zip with the extension files.

Features

The DBTS CSV List Object allows you to read and display in your web pages custom formatted tables with the file content. Main features are:

- Read and display any grid formatted text file. This includes Excel CSVs which use "," or ";" as column separator, the WB built in Form Processor CSVs and the WB6 Login system created file (usersdb.php) which uses chr(380) "|" as separator, the WB Guestbook which use chr(02) "␣" as separator, and last, our own DBTS Form Processor Excel Type CSVs that uses ";" as separator.
- The entire content is displayed in pages of user defined number of records, in a scrollable div that will take care of the overflow.
- You can select to display all or only certain columns, in your own desired order, you can sort the content, and you can, if you wish, include View / Edit / Delete record links.
- Sorting is made with PHP comparing content as strings.
- If your file includes the column labels, the script will use them as column titles. You can, however, override the titles and use your own. Since DBTS

Form Processor creates files that include the column titles, the two extensions form an ideal combination.

- You can set the look of the table as you wish, set hyperlink styles for links, define fonts, sizes, background colors, use alternate row backgrounds, and select the sorting arrows color among the common colors. Details are going to be provided in the next paragraphs.
- You don't need to retype the visual settings again in the child objects (View, Edit, Delete), as they use the same Visual Configuration File as the List Object.

The DBTS CSV View Object allows you to display a specific record details in a nicely formatted table. The visual settings are those inherited from the List Object. However, the extension allows you to display the table or not. This last option was planned so that you can create your own free layout using the record details.

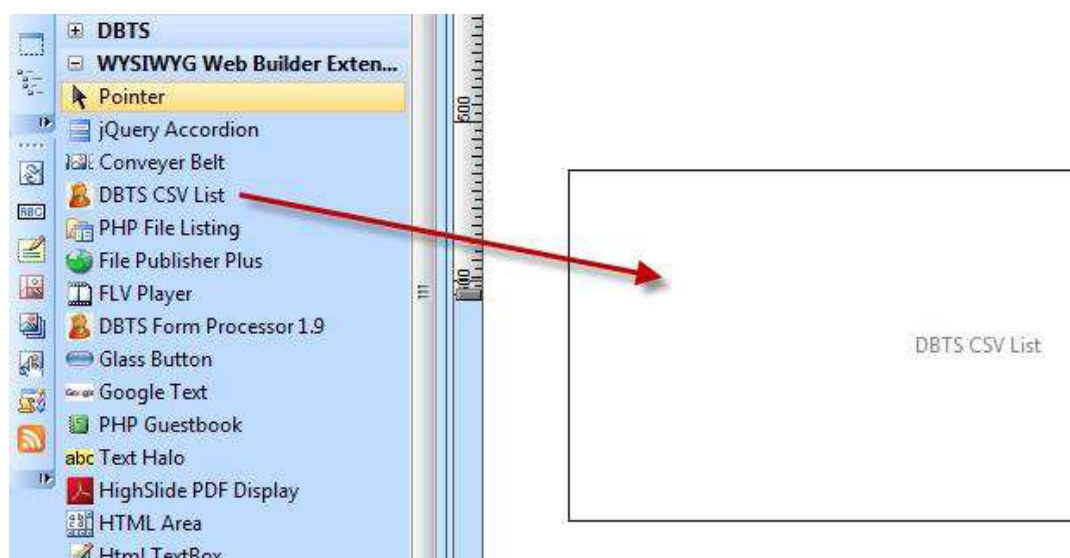
The DBTS CSV Edit Object allows you to edit a specific record details. The visual settings are those inherited from the List Object. The object can automatically create a form that will be used to edit the record details. However, the extension allows you to display the automatically created form, or not. This last option was planned so that you can create your own free layout form.

The DBTS CSV Manager Delete Object allows you to delete a specific record. The visual settings are those inherited from the List Object. It will, however, display the record details in the Delete page, so you can visually check if you are deleting the correct record.

Important: Normally, ONLY the List Object and View Object should be available in public pages. Users should NOT have access to the Edit object or Delete object, UNLESS YOU CAN SET CLEAR ACCESS AND RIGHTS RULES. We have now implemented in the Edit Object a "protection" feature that allows users to edit records that belong to them, in example WWB Login script members can edit their profile.

If the Edit Objects are only for use by the Administrator, they can simply be placed in a password protected folder.

How you can use the DBTS CSV List Object



The DBTS CSV List extension will add an icon in your extensions Toolbar.

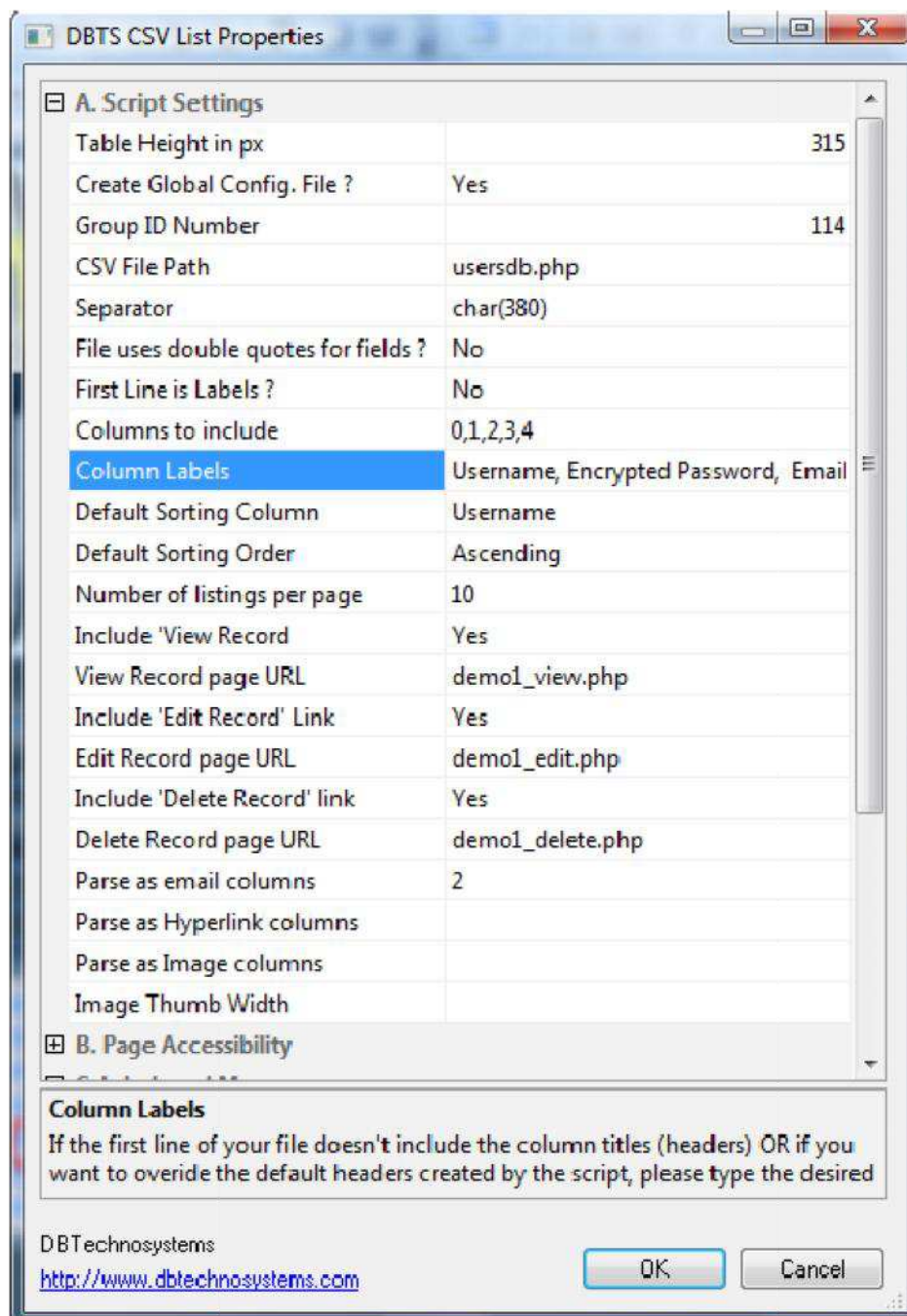
To use the extension simply click and place the CSV List object in your page. Give it the desired dimensions and position it as required.

To setup the processor properties double click the List Object so the properties dialog box will appear.

As you can see there are 10 main categories of properties (A – J):

The last 8 ones only regard visual settings of the presentation: backgrounds, text colors, sizes, styles and decoration, hyperlink styles etc., as well as the text that will appear in the footer bar (navigation system) These are pretty self explanatory. So only the first two will be analyzed here.

A. Script Settings



1. Table Height in px: The script uses a "div" to contain the table that displays the records. This div is set to "overflow" so if the content requires more space than the one you have assigned it, it will display scrollbars (just as an I-frame). Since some browsers (IE mainly) have issues with the div height, it is wise that you assign here the height in pixels of the table that will be created in your page. This height is the same as the CSV List object you have in your page. So simply look at its height and type it here.
2. Create Global Config. File ?: As you may know, from the WB Login system, you need to provide, for each object, both the visual details and the database file details. Though this might be acceptable when you only need to define a few settings, it becomes quite hard to repeat when the settings become many (in our case, more than 30). The solution we gave is this: if you set the above setting to "Yes", it will create a copy both of the main settings AND of the css settings. This allows you to only set local variables, in the other objects, as all the css and the main script settings are retrieved directly from these files. So the View, Edit, and Delete Objects will use the same settings.
3. Group ID number: in the previous paragraph we saw that we can create a two files with the main settings of the script and the the settings of the css file. However, if you have many pages where you use the extension, you need to distinguish the correct setting files. To achieve this, you need to provide a ID number (from 1 to 999), which MUST be unique for each group of objects that share the same file, and visual settings. In the other objects (View, Edit, Delete) you need only to type the Group ID number and only set the local variables.
4. CSV File Path: this is the file path of the file that you want to read and display. Please note that it is not necessary that the file is a CSV file. It can read any text file as long as it is grid formatted, independently from the extension. In fact, it can read the usersdb.php and guestbookdb.php file types, used by the WB6 login system and guestbook. If the file is in the same folder, the path will be simply "filename.csv". If it is in a subfolder, it will be "foldername/filename.csv" .
5. Separator: The separator used in your file. The extension allows for four different separators: ";", ",", chr(02) "␣" and chr(380) "␣". The WB Login system file and the csv created by the WB form processor both use |, the guestbook uses chr(02) "␣" . The DBTS Form Processor extension uses the Excel type csv separator ";".
6. File uses double quotes for fields ? : Excel uses double quotes " to indicate field end / start. This is the only way to allow multiline content in a CSV file. Please note that this is a good practice to set this property to "Yes" even if you are not sure.
If you are using the DBTS Form Processor to populate your form, you can also set it to use double quotes.
7. First Line is Labels? : setting this to "Yes" will tell the script to retrieve and use the first row content as Labels (headers) for the columns.
8. Columns to include: Some files might have a large number of columns. In most cases, the usual practice, is to only display a few ones, and then redirect the visitor to a detailed page for each record. For this purpose, you

can type here the columns that you want to display. Columns must be fed in this field as numbers, with counting starting from 0 for the first column.

9. So, in example, if you want to display columns 0, 3, 6 and 8, type these numbers separated by comma. **IMPORTANT:** column numbering starts from 0. **If you desire to change the order columns appear, simply type them in the desired order (in example 3, 8, 0, 6)**
10. Column Labels: You can provide the column labels, if your file doesn't include them in the first row, OR if you wish to override the same if they exist. **Column labels must be in the same number of the columns that you want to display. Also they MUST be in the same order.**
11. Default Sorting Column and default Sorting Order : you can set which column the default sorting will be performed on. Not setting it, will result in the script displaying all records in order or reverse order, depending on the Default Sorting Order (Ascending or Descending) you select. Please note that the Default Sorting Column must be defined as column number, starting from 0 (as we did in point 8.)
12. Number of listings per page: As mentioned, the extension displays records in pages. With this property you can set the number of records to display per page. The navigation system in the footer bar allows navigation between pages.
13. – 17. These settings allow you to add, automatically, links to the View, Edit, Delete pages. This allows you to create a simple Administration interface, where, from the List page, you can go directly in the View/Edit/Delete page to perform required action with the specific page. Since the script can't know where these pages are, you need to provide the URLs of the relevant pages. Paths must be relative to the List page location. In example, if your pages are named "view.php", "edit.php", "delete.php", and they are in the same folder where the list page is, you simply type, as URLs, "view.php", "edit.php" and "delete.php".
18. Parse as Link : You can select one or more columns to parse the content as active hyperlinks. You can control the style of the hyperlink by setting your preferences in the "Links in content Hyperlink Style" category. Please note that the extension will not actually check to see if the record value is actually a URL. It will render as Hyperlink whatever is in the record.
19. Parse as Email columns: You can select one or more columns to parse the content as active <mailto:> links. Just select the column number(s) as explained in point 6. Again, the extension will not check to see if the record is a valid email address.
20. Parse as Image : You can select one or more columns to parse the content as images. This is the ideal solution for creating catalogs, directories etc.
21. Image thumb max width(**Feature still to be implemented**): This setting allows you to set the maximum dimensions of the dynamically created thumbnail. Please note that the script that creates the thumbnail, will maintain the aspect ratio, so, in example, if you set the max dimension to be 100 px, a 600x450 landscape image will result in a 100x75 thumb, whilst a 450x600 portrait image will result in 75x100 thumb.

B. Page Accessibility (New!)

The DBTS CSV Manager extensions now allow you to set access rules to the pages that contain the extension objects.

These settings are:

a. Allowed access

1. All
2. Logged users only
3. Admin only

b. Admin Password: You can set the desired Admin password. This is only required if you have set, in the previous setting, access to be allowed only to the Admin.

c. Login or redirect page path: this is the path (relatively to the present page) of the Login or Error page. This way the user will know that he needs to Login in order to continue. If access is restricted to members only, this should be the members Login page. If access is restricted to the Admin only, this should be the Admin Login page. See section Page Protection for further details.

d. Only display user records: this allows you to restrict member's access ONLY to those records that are related to them. In example, using the extension with the WWB built in Login script, you can allow users to edit their Profile. See "Tips and Tricks" section.

e. Username Column number: to achieve what is described in above point. D, the script needs to know which column in the database CSV file is the user's "username". This is usually (unless you have modified the file) column "0".

Preparing your List page:

Before you can display correctly a text file contents using the CSV List extension, you need to analyze the file structure yourself. Get a pen and a sheet of paper and follow the steps below:

1. Open your page in WB.
2. Drag the CSV List object in your page
3. Set the filename, separator, and the extension object height.
4. Set the "First Line is Labels" to "No"
5. **DON'T** set any "Columns to Include", **NOR** "Column Labels"
6. Publish **All** Files.

Setting, at a first stage the extension like this, will result in the extension displaying all columns, and will display on top of each column its number.

At this point, simply write down the numbers of the columns that you want to display in your page, so you can use them in the extension. **You will also be able to see if your file includes the column labels or not.** If it doesn't, you

can look also the columns content, so you can decide what's the appropriate column header (title) for each column.

At this point you are ready to setup your List.

Go back in your WB page and open the CSV List object Properties window. I advise that you maximize it, so you can display more properties, and you can visualize the entire bottom part, where details about each property are included.

Set the "Columns to include". If, in example, you want to display columns 0, 2,8, and 12, just type these numbers in the field. If you don't type anything here, ALL columns will be displayed.

Next, **if your file doesn't have the column titles included**, type them, as you like them, in the "Column Labels" field. The number of titles must equal the number of columns that you have selected to display. Even if your file has the column headers already included, but you wish to change the wording (titles) you can type them here.


In both cases, if you type anything in the "Column Labels" field, the order and number of column labels **MUST** match the order and number of "Columns to Include".

Select the default Sorting Column. If no selection is made, the default sorting column will be the record number. **The value that you will type in MUST be exactly as the label used for that column.** So if you wish the default sorting order to be a column that contains the "Username" and you have set that column header to be "Username", that's what you will have to type here.

Select the default sorting order (Ascending or Descending).

Continue with the rest of the settings. The visual settings are pretty self explanatory. They regard the text size and colors, the background colors, and the text style for the various parts of the displayed table. The "**D. Header Visual Details**" category, regards the look of both the List object table, AND that of the other objects (View, Edit, Delete). Since the columns titles are linked to the script itself (clicking on them sorts the table content) the color, weight, hyperlink style etc. are instead set through the "**H. Headers Hyperlink Style**" category settings. Please note that the script will use a predefined set of arrows themes to display the sorting order. These are:

Acqua marine 

Blue 


Brown 

Carbon grey 

Green 

Grey 

Orange 

Pink	
Red	
Sky blue	
Violet	
Yellow	

You can set the hyperlink style for the headers in such a way that the active header is displayed in a different color than the rest of the headers.

After setting all variables, you are ready to publish your page again.

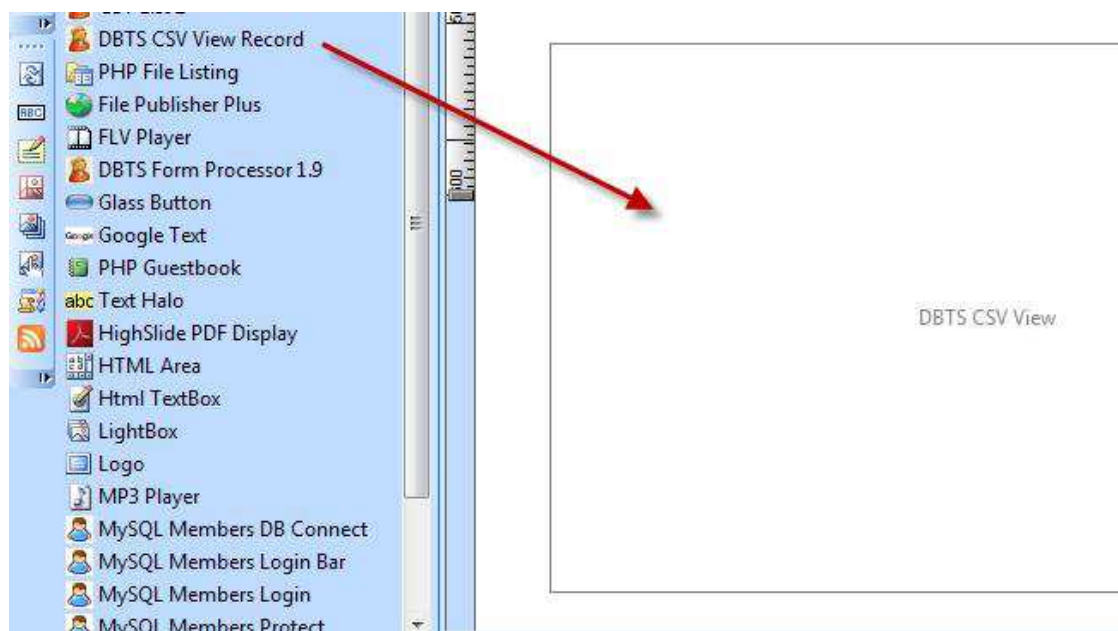
In order to create the Global Configuration file that will be used for the other members of the extension, you must, at this point, run the script in your browser. As soon as you do so, the script will create copies of its own settings and css file, that will be used by the other members of the DBTS CSV Extensions.

If in the future you change any of your visual settings, or the file name or the separator, you must publish ALL files and run the script on the browser right after, again, so the new Global Configuration file will be created

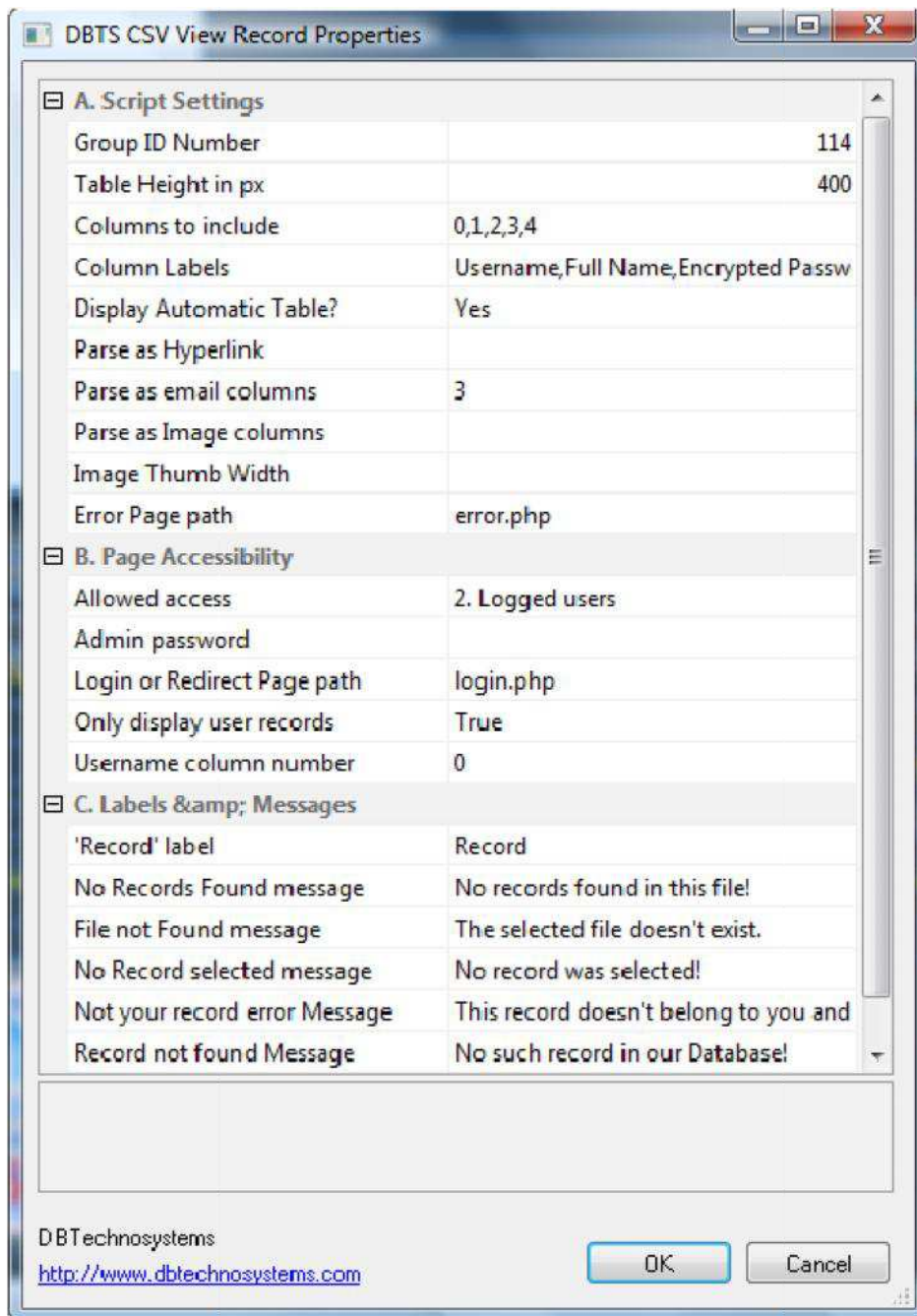
The DBTS CSV View Object.

This extension will create in your page a table that will display the content of a specific row of the file. To reach this page, you must include in the List page, the "View" icon. This is achieved as explained above, in points 11. – 17. When you display the list, you simply click on the "View" icon of a record, to send the visitor to the "view" page.

The extension will create an icon in your WB Extensions toolbar:



Simply drag and drop the object in your page. Give it the desired position and dimensions. Double click it to set its properties:



As you see, the settings are very similar to those of the List object, but all the visual settings are missing. In fact the object will retrieve the visual settings from the Global Configuration file.

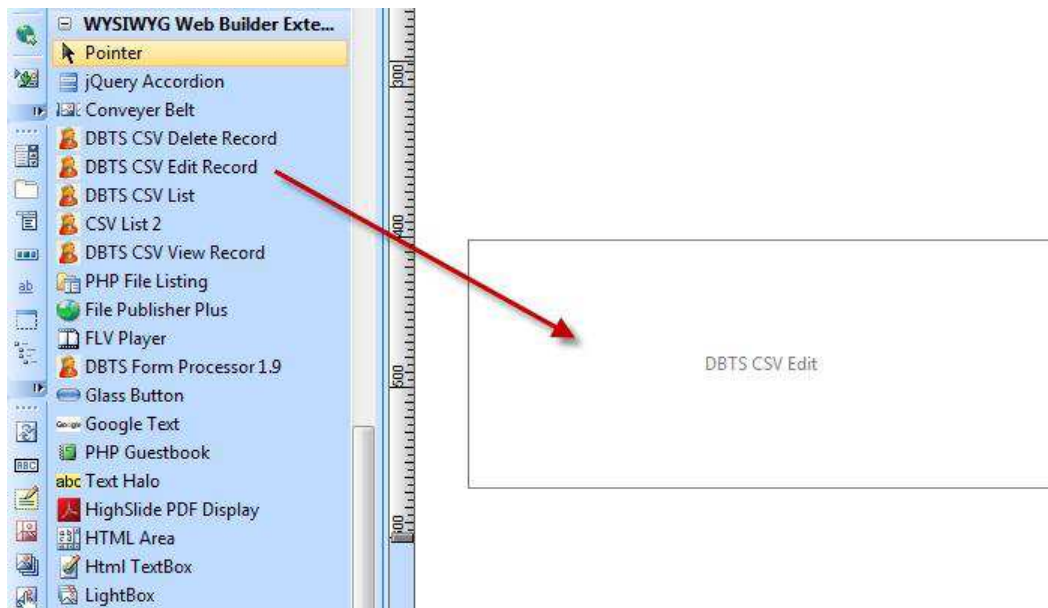
It is mandatory that you fill in the Group ID Number, which MUST be the same as that of the relevant List object.

The "Columns to include" and "Column Labels" properties work exactly the same way as those of the List Object.

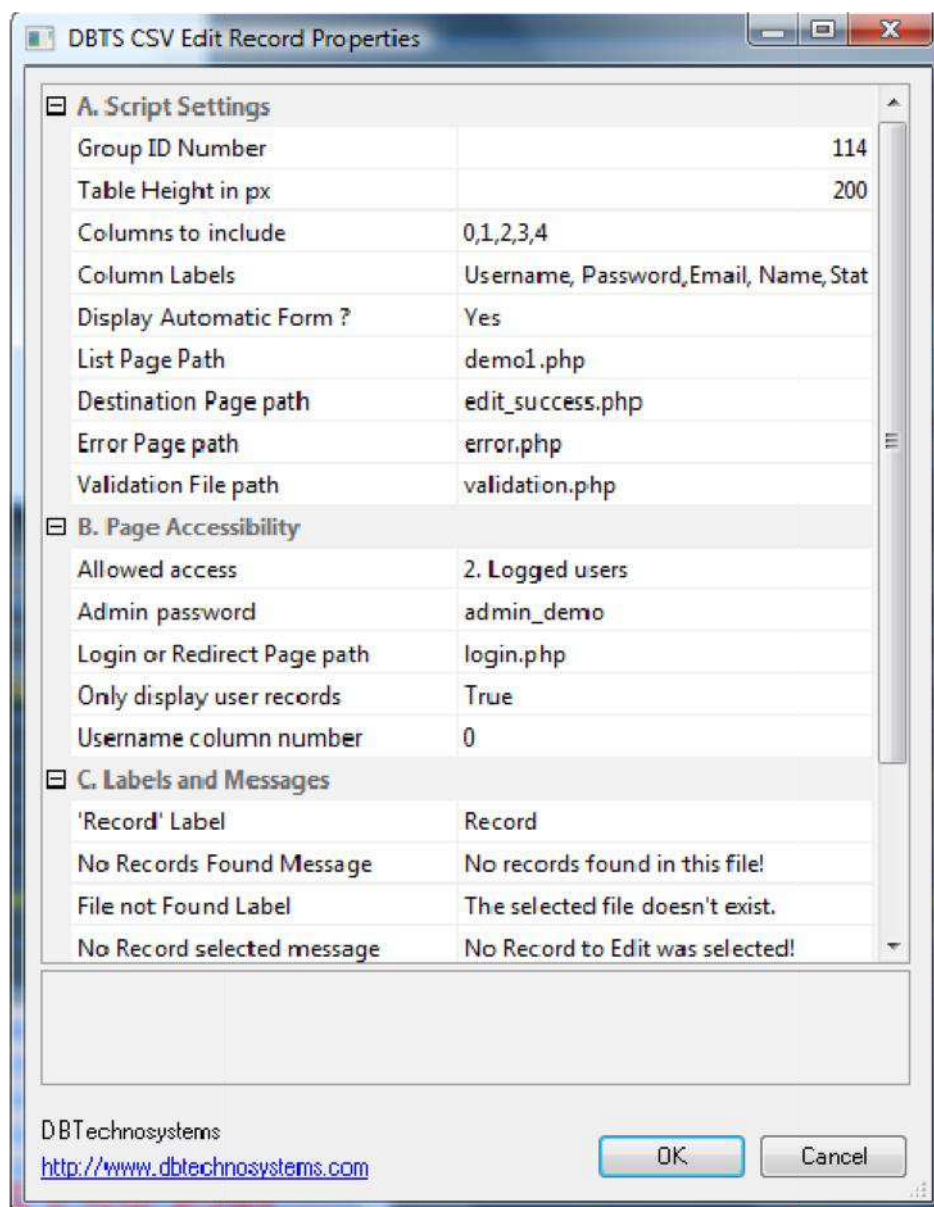
The DBTS CSV Manager Edit Object.

This extension will create in your page a table like form that will display all or part of the content of a specific row of the file. To reach this page, you must include in the List page, the "Edit" icon. This is achieved as explained in the List Object Features, in points 11. – 17. When you display the list, you simply click on the "Edit" icon of a record, to send the visitor to the "Edit" page.

The extension will create an icon in your WB Extensions toolbar:



Simply drag and drop the object in your page. Give it the desired position and dimensions. Double click it to set its properties:



As you see, the settings are very similar to those of the View object. In fact the object will retrieve the visual settings from the Global Configuration file just as the "View" object does.

It is mandatory that you fill in the Group ID Number, which MUST be the same as that of the relevant List object.

The "Columns to include" and "Column Labels" properties work exactly the same way as those of the List Object.

The extension can automatically create a form containing the selected columns in your page. The look of it is like the following image (of course, yours will have your own settings):

Name	George 1
Username	demo1
Email	test1@test.com
Status	1

However, sometimes it can be more convenient to use a form created directly in WB . This can be quite useful if you want to have a customized look for your form, or if you wish to have dropdowns with option values that use different wording.

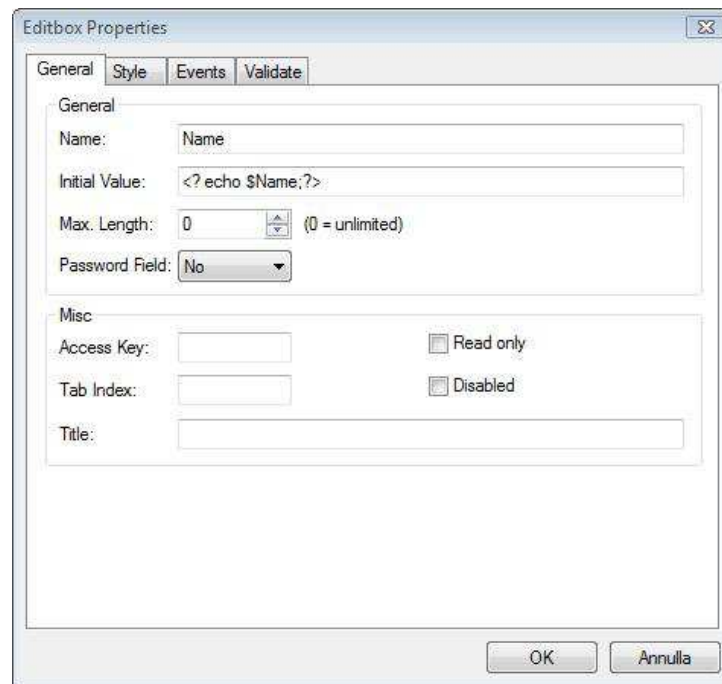
Example of Manually created Edit Form

Name	<input type="text" value="George 1"/>
Username	<input type="text" value="demo1"/>
Email adress	<input type="text" value="test1@test.com"/>
Status	<input type="text" value="Pending"/>

In the above example, we have used the file that is created by WB's Login Tools. This script uses the number 0 for Status = Pending and the number 1 for status = active. Obviously, it is easier to have a dropdown with the actual wording. So we add two options: Pending (with value = 0) and Active (with value = 1)

To make your form display the values relevant to the specific field you need to act on your form fields as follows:

1. Text Fields and Text Areas: Just fill in the "Initial Value" field, the code: `<? echo $Name;?>` where Name is the field name



2. Radio Buttons and checkboxes: Here we have a bit of a more complicated situation. Originally, the radio Button or Checkbox must NOT be selected. If you make a Radio button or a checkbox selected right from the start, it will not reflect the record value, as the "selected" status you have originally set will override the file value.

Let's make the example of a column, in our file, that is named "Newsletter". And let's suppose that this column might have the values "Yes", "No", or it can be blank.

Let's also suppose that you want to use two Radio button: One for "Yes" and one for "No".

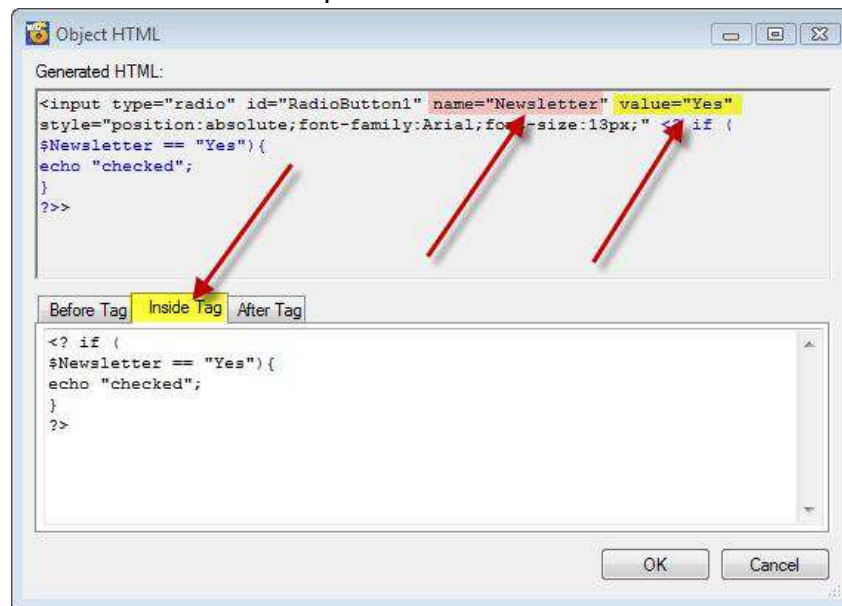
With this said, you must right click the "Yes" radio button, and paste the following code in the Inside Tag of its HTML:

```
<? if (  
$Newsletter == "Yes"){  
echo "checked";  
}  
?>
```

For the "No" Radio button the code will be:

```
<? if (  
$Newsletter == "No"){  
echo "checked";  
}  
?>
```

As we said, "Newsletter" is the group name of the radio buttons, and "Yes" and "No" are the value that the specific Radio will have if selected. Look at the image below that better explains this:



3. Drop Down lists (listboxes - combo boxes): this is also another complicated task. First of all, let me say that there is nothing you can do with WB6's listboxes if you have multiple selections enabled, unless you use case-specific code, but this is not what we are doing here. Since listboxes have multiple values to chose from, there is actually no way to make a specific value (or values) selected. So we will do a trick: we will ADD, before all hardcoded values, the one that has been previously selected by the user. To achieve this, right click the list box and add the following code in the Inside tag of its HTML:

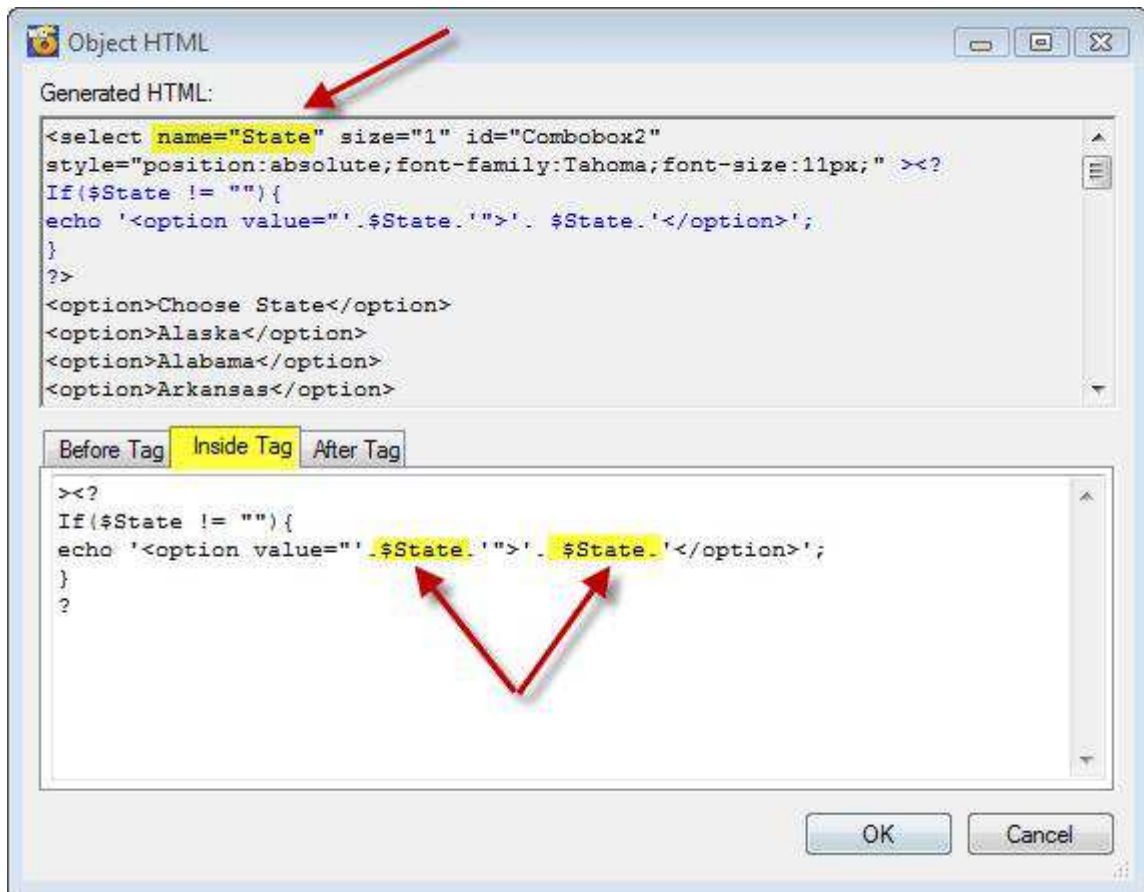
4.

```
><?
If($State != ""){
echo '<option value="' . $State . '>'. $State . '</option>';
}
?
```

Of course, you must replace the field name used here, **State** with the actual name of your field.

Please note that since we add the submitted value on top of the list, you should NOT set, in the field validation, the "Disallow First Choice" option.

Please refer to the following image so the above result clear:



If any of your form fields is composed with more than one words, the field name must have underscores instead of white spaces among words. In example, if for a column you have set the name "Email address", the relevant form field MUST be "Email_address".

Since most probably the CSV file is created through a form submission, using a form on your site (can be a contact form, a Login Tool Signup Form, or whatever other form) I also advise that you use ALWAYS in your original forms, field names that contain no whitespaces but use underscores instead. It will make your life much easier. Don't worry about what you will be receiving in your email: both the WB6 built in form processor and our own DBTS Form Processor Extension, replace underscores with whitespaces and capitalize the first letter of each word, in the mails that are sent out.

When using your own, manually created form, it is important that you use the following settings:

- The form action MUST be `<? echo basename(__FILE__);?>`
- The form Encoding Type MUST be `multipart/form data`
- The form fields MUST have exactly the same name as your file columns, if the file includes the column names, or exactly the same as you have provided them in the script.
- You MUST include in the form a hidden field named "record" and its value MUST be `<? echo $record;?>`.
- You MUST include in the form a hidden field named "formid" and its value MUST be `csveditform`
-

- The form will only have those fields for which you have selected the relevant columns in the "Columns to include" or, ALL fields (columns) if you wish so.

I advise that, at least at a first time, you use the automatically created form. After you have seen how this works, and you have analyzed the form code (looking at the form HTML) you can proceed with creating your own form.

In the current version a new feature has been added: Page Accessibility settings.

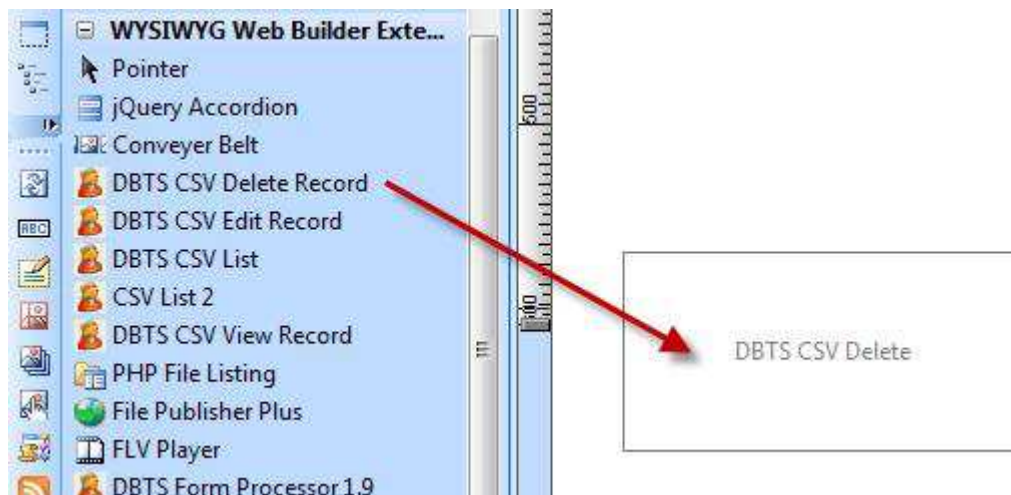
There are 4 settings in this section:

1. Only allow users to edit their own profile : with this property set to True, the script will automatically recognize the user (he must be already logged in) and will only allow him to edit his own record. This is useful to allow you creating a "Edit Profile" page for the WWB Login script based on flat file database.
2. Username Column number: since the WWB Login script recognizes users thanks to their "username", the script needs to know which column contains the username. This is column 0 for the standard WWB Login script.
3. - 4. These settings contain the paths to the "login" and the "error" pages. They are necessary to redirect users to the Login Page, if they are not logged in, and to display the necessary error messages if any errors are encountered.

The DBTS CSV Manager Delete Object

This extension will create in your page a table like form that will display all or part of the content of a specific row of the file. To reach this page, you must include in the List page, the "Edit" icon. This is achieved as explained in the List Object features, in points 11. - 17. When you display the list, you simply click on the "Edit" icon of a record, to send the visitor to the "Edit" page.

The extension will create an icon in your WB Extensions toolbar:



The object will create a table AND a form in the page, looking like the following image:

Submission Date	07/08/2009 16.36
First Name	George
Last Name	Naval 36
Country	United States
Email	info@dbtechnosystems.com

Are you sure you want to delete this record?
Please note that this can't be undone!

Delete Record!

The text circled in yellow is the Warning Message Text. The button text can also be set by you.

You can opt NOT to display the automatically created form. The script makes available to you all the record values, with the same names that you use for the columns. In example, if you have used the title "Username" for a column, the script will make available to you a variable "\$Username" that you can use to create a free layout to display the record details. However, in order to complete the record deletion, you will need to add your own form and warning message.

When using your own, manually created form, it is important that you use the following settings:

- The form action MUST be `<? echo basename(__FILE__);?>`
- The form Encoding Type MUST be `multipart/form data`
- The form will have no visible fields at all.
- You MUST include in the form a hidden field named "record" and its value MUST be `<? echo $record;?>`.
- You MUST include in the form a hidden field named "formid" and its value MUST be `"csvdeleteform"`

Logical Procedures:

Normally, the core application is the page where the List object is. From there, you can View, Edit or Delete specific records. Add in each of these pages links to the others so you can navigate through the pages without issues.

However, you can also display directly the View, Edit and Delete pages if you know the record number. Now this can be extremely difficult in certain cases, as it might be extremely easy in others.

With the next update we will provide (see next paragraph) a feature to allow a user based access to the pages. When this is released, you will be able to set the Edit and Delete Objects to ONLY display the records that are related to the specific user. This way, in example, if you send the user directly to the Edit page, it will automatically display the his own details record.

Page Protection:

When using the List, View, Edit and Delete Objects, you might want to restrict access to these pages. Otherwise anyone would be able to edit / delete records without your permission, and as you understand it would be possible that they edit / delete records that they should not be able to. Also, in many cases, you might want users to be able to list / view / edit / delete records related to them only. The following paragraphs explain how you can allow access to the Edit / Delete pages to only authorized users (including you, as site Administrator).

1. Allowing access only to the Administrator

Set, in the object's properties, section "Page Accessibility", the "**Allow access**" property to "**2. Admin only**". Set a password for the Administrator.

Create a separate, auxiliary Administrator Login Page and use the WWB Single Page Protect in that page. Then add a link from the Login confirmation page to the List, Edit and Delete pages.

2. Allowing access only to Registered, Logged in Users

This property allows you to protect these pages. To do so, simply set the "**Allow access**" property to "**2. Logged users**". You will need to create separately the login / logout / reset password pages using WWB's standard Login tools, but, of course, if you want this, you already must have these pages!

3. Allowing conditional access to logged in user : sometimes you need to allow

Listing / viewing / Editing / Deleting only to the record owner. A typical example is allowing a user to edit his own Profile details. To allow this, the DBTS CSV Manager uses sessions. It also needs to know which column in the database file contains the "identification" value and the SESSION variable that holds the related value.

To make clear what I mean in the previous point 3, let's see how the WWB Login script works.

When a registered user logs in, a SESSION is started.

Session related values are stored in the so called \$_SESSION superglobal array.

An array can be considered simply as a table where the first column is the "key" and the second column is the value.

So, in the case of the WWB login script, considering a user with username "demo" and full name "George 1" the \$_SESSION array holds the following info:

Key	\$_SESSION variable	Value
username	\$_SESSION['username']	demo
fullname	\$_SESSION['fullname']	George 1

The Database file that holds the user's data, has the following structure:

	username	password	email	name	active
Column nr.	0	1	2	3	4
Record 1	demo1	1066726e7160bd9c987c9968e0cc275a	test@test.com	George 1	1
Record 2	demo2	1066726e7160bd9c987c9968e0cc275a	test2@test.com	George 2	1
Record 3	demo3	297e430d45e7bf6f5f5dc929d6b072b	test3@test.com	George 3	1

At this point, the script will use the session variable `$_SESSION['username']`, which value is "demo" and will compare it (for each record) with the "username" column, which is column 0. If it finds the SESSION stored value equal to the value of column 0 of a record, it "understands" that this record belongs to user "demo" and will allow him to List / View / Edit / Delete this record.

To enable this feature in the extension, set the **"Only display user records"** to True. Of course, access MUST be allowed to "2. Logged users".

How you can create a "Edit Profile" page for the WWB Login script:

1. Create a "List" page in WWB. This is necessary even if you can delete this page later. Add the List Object. Set its properties as desired. Save and Publish.
Open the page in your Browser so that the necessary css and settings files are created. After doing so, you can (if you wish) delete this page from the server).
2. Now create the "Edit Profile" page. You can use the automatically created form or a manually created (using WWB's form tools) form. See related instructions in previous section of the manual. Set the page to be accessible by "2. Logged users", and also set the "Only display user records" to True; Set the "Session Key" property to "username"; Set the File Column number to 0 ;

That's all that is required.

Sometimes, the same user may have more than one records on the database file. In example, imagine an "ad" system where each user can post more than one ads.

In such systems, there is usually a two file structure. The first file is a "members" file like the one described above. The second file, contains the ads. To distinguish which user has made a ad post, the ad record contains, among other info, the userID or username of the user that has made the post.

In this case, you will need to allow the user to edit both his profile, AND his ad posts. We have already seen how to create the "Edit Profile" page.

To allow the user to edit his ads, you should first allow him to select which record he needs to edit / delete. This is very simple:

Create a "List" Page for the Ads. Set in Page Accessibility, "Only display user records". Of course, set the rest of properties as required.

This will allow any registered user to "filter" the ads and only display those that belong to him. By adding "Edit" "View" and "Delete" links in the List page, you can allow the user to perform desired actions on a specific record of his own.

Support:

Contact us at info@dbtechnosystems.com . But please first read this manual carefully, the answer to your question is probably already contained in this manual.

Support that goes BEYOND simple email help and requires entering your hosting account to solve any issues, can only be provided on an agreed upon, paid for, basis.

License:

The PHP code used in this extension has been developed by DB Technosystems.com . The extension is freely available to all WB users, and can be embedded in any work, personal or commercial. You are not allowed to redistribute or sell this extension. Please address to our download page anyone interested in using this extension:

http://www.dbtechnosystems.com/wb6/DBTS_CSV_Manager/

This software is delivered AS IS. By installing and using it, you agree that Pablo Software Solutions or DB Technosystems will not be held liable for ANY damage, of ANY type, caused by the use of this software. Installing and using this software implies that you fully and unconditionally accept these terms. USE AT YOUR OWN RISK. Please fully test before using it.

Credits:

1. Credits go to Pablo Software Solutions for making (and further improving) their Extension Builder! And for continuously improving their WYSIWYG WB. Cudos!
2. Many thanks to the WB Forum Users that have provided feedback! Without this feedback it wouldn't have been possible taking the script to this level.

Have fun!

George (Forum nickname Navaldesign)