About AJ AX AutoComplete

How many times have you struggled with entering long lists, while you had to lookup a product or country? AJ AX AutoComplete will you make your life easier. Your user can get a filtered dropdown with choices after entering a few characters, enabling quick search and selection without having to enter long text.
Features in Detail

Web standards based
AJAX Autocomplete complies with the latest web standards of the W3C

Customization
The search box and the result list can easily be customized to fit in your site. You can apply any style using CSS. Choose in and outgoing transitions for your completion box such as slide, or fade and set it’s opacity.

Dynamic Result Lists
Generate your search suggestions using a database. The result list loads mighty fast as the extension takes full advantage of the latest AJAX technologies so only the data that has to be displayed is loaded.

Predefined lists
You don’t want to use a database? No problem, we have included a great collection of standard lists with useful search lists, such as countries, European capitals, states (US) and zip codes (US).

Generate your Autocomplete manually
If you don’t want to use a database or the predefined templates, you can generate your result list manually, which allows you to generate any completion suggestion you like.

Server Side Compatibility
The extension is fully compatible with PHP or ASP, it also supports the Dreamweaver Database model so you can define your connections in Dreamweaver and load it from there.

JavaScript Unobtrusive
Even if your users don’t use JavaScript, they can still use the entry field as JavaScript is only used to display the entry suggestions.

Integrated in Dreamweaver
Interactive dialog in Dreamweaver with all the options you’ll need;
Ajax Autocomplete

Main options
- Delay: 100 ms.
- Autoselect first suggestion: false
- Autocompletion sensitivity: 1 chars.

Effects
- Completion box opacity: 10%
- Show autocomplete using: slide effect
- Hide autocomplete using: slide effect

Autocomplete data provider
- Generate autocomplete manually
- Generate autocomplete from database
- Generate autocomplete from predefined templates

Provider properties
<table>
<thead>
<tr>
<th>Word</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSS Gallery</td>
<td>CSS Gallery</td>
</tr>
<tr>
<td>Advanced Layer Popup</td>
<td>Advanced Layer Popup</td>
</tr>
<tr>
<td>Flash Album Generator</td>
<td>Flash Album Generator</td>
</tr>
</tbody>
</table>

Add word | Remove selected

OK | Cancel
Before you begin
Make sure to setup your site in Dreamweaver first before creating a page, if you don't know how to do this please read this.

Installing the extension
Read the tutorial at http://www.dmxzone.com/ShowDetail.asp?NewsId=4671

Login, download the product, and lookup your serial at http://www.dmxzone.com/myZone.asp?Section=purchases

Get the latest extension manager from Adobe at http://www.adobe.com/exchange/em_download/

Open the .mxp file (the extension manager should start-up) and install the extension.

If you experience problems please read the FAQ at http://www.dmxzone.com/index.asp?TypeId=7&CatId=687 before contacting support.
Creating a Basic AJAX AutoComplete Field

Introduction
In this tutorial we’re going to show you how easy it is to insert an AJAX AutoComplete field on to your page. We’ll generate the search suggestions (the fields below the entry field), using manual input, a database and finally we’ll use one of the supplied AutoComplete templates.

- California
- California
- Colorado
- Colorado
- Connecticut
- Connecticut

Generating an AutoComplete Field manually

1. Create a new HTML page
   Create a new or open an existing HTML page and save the page.

2. Applying the extension
   Choose the icon from the DMXzone object bar to apply the extension, the following popup appears:
3. **Setting the Main Options**

   The **Delay** determines how long it takes before your search suggestions are displayed, the higher the value the longer the delay, we’ll set this value to 75ms.

<table>
<thead>
<tr>
<th>Main options</th>
<th>Effects</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Delay:</strong> 75 ms.</td>
<td><strong>Completion box opacity:</strong> 10 %</td>
</tr>
<tr>
<td>Autoselect first suggestion: False</td>
<td>Show autocomplete using: slide effect</td>
</tr>
<tr>
<td>Autocompletion sensitivity: 1 chars.</td>
<td>Hide autocomplete using: slide effect</td>
</tr>
</tbody>
</table>

   When the **Autoselect first suggestion** option is set to **true**, the extension will complete the word after the first character(s) are entered, we’ll set it to true.

<table>
<thead>
<tr>
<th>Main options</th>
<th>Effects</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Delay:</strong> 100 ms.</td>
<td><strong>Completion box opacity:</strong> 10 %</td>
</tr>
<tr>
<td><strong>Autoselect first suggestion:</strong> True</td>
<td>Show autocomplete using: slide effect</td>
</tr>
<tr>
<td>Autocompletion sensitivity: 1 chars.</td>
<td>Hide autocomplete using: slide effect</td>
</tr>
</tbody>
</table>

   **Autocompletion sensitivity** determines how any characters must be entered before the suggestions are displayed, we’ll set it to 1.

<table>
<thead>
<tr>
<th>Main options</th>
<th>Effects</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Delay:</strong> 100 ms.</td>
<td><strong>Completion box opacity:</strong> 10 %</td>
</tr>
<tr>
<td><strong>Autoselect first suggestion:</strong> True</td>
<td>Show autocomplete using: slide effect</td>
</tr>
<tr>
<td><strong>Autocompletion sensitivity:</strong> 1 chars.</td>
<td>Hide autocomplete using: slide effect</td>
</tr>
</tbody>
</table>

4. **Configuring Effects**

   **Completion box opacity** determines the opacity of the background of the search suggestion box, we’ll set it to 60%.
5. **Choosing the Autocomplete data provider**

We’ll select **Generate autocomplete manually** so we can make our own search suggestions, we’ll explore the other options later on.

**Autocomplete data provider**

- Generate autocomplete manually.
- Generate autocomplete from database.
- Generate autocomplete from predefined templates.

6. **Setting the Provider properties**

Press the **Add word** button to add your search suggestions, we’ve added three items:

<table>
<thead>
<tr>
<th>Word</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;name&gt;</td>
<td>&lt;name&gt;</td>
</tr>
<tr>
<td>&lt;name&gt;</td>
<td>&lt;name&gt;</td>
</tr>
<tr>
<td>&lt;name&gt;</td>
<td>&lt;name&gt;</td>
</tr>
</tbody>
</table>

Press **Add word** or **Remove selected**
Double click on the `<name>` or `<value>` field to edit the search term and the descriptive text.

### Provider properties

<table>
<thead>
<tr>
<th>Word</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSS Image Gallery</td>
<td>Create cool pure CSS galleries and slideshows</td>
</tr>
<tr>
<td>Flash MP3 Player</td>
<td>A great MP3 player on your website within a f…</td>
</tr>
</tbody>
</table>

You can remove items by pressing the **Remove selected** button.

### Provider properties

<table>
<thead>
<tr>
<th>Word</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSS Image Gallery</td>
<td>Create cool pure CSS galleries and slideshows</td>
</tr>
<tr>
<td>Flash MP3 Player</td>
<td>A great MP3 player on your website within a f…</td>
</tr>
<tr>
<td>Advanced Layer Popup</td>
<td>Create amazing inline Layer Popups</td>
</tr>
</tbody>
</table>

---

7. **Final Steps**

Press **OK** to generate your AJAX AutoComplete field.

Be sure to copy the images folder and the styles folder to your server.

Press **OK** and press **F12** to preview and **save**. That’s it you’ve created your **own AutoComplete field**!
Generating an AutoComplete Field using a Database

1. **First steps**
   Make sure you have followed steps one to four of the Generating an AutoComplete Field manually tutorial.

2. **Define a database connection**
   We’ll select Generate autocomplete from database so we can get our search suggestions from one of the supplied XML files.

   **Autocomplete data provider**
   - Generate autocomplete manually.
   - Generate autocomplete from database.
   - Generate autocomplete from predefined templates.

3. **Setting the Provider properties**
   Select the database you want to use in the Connection dropdown, the Table where you want to retrieve your fields from and choose your search suggestions in the Autocomplete for drop down. If you like you can also display additional fields by choosing the Optional fields list. Hold Ctrl on your keyboard to select multiple fields.

   **Provider properties**
   - Connection: phpdatabase
   - Table: emails
   - Autocomplete for: id
   - Optional Fields: naam, email

   ![autocomplete example](image-url)
4. **Final Steps**  
Press **OK** to generate your AJAX AutoComplete field.

Be sure to copy the images folder and the styles folder to your server.

Press **OK** and press **F12** to preview and **save**. That’s it you’ve created your own dynamic AutoComplete field!
Generating an AutoComplete from Predefined Templates

1. **First steps**
   Make sure you have followed steps one to four of the *Generating an AutoComplete Field manually* tutorial.

2. **Choosing the Autocomplete data provider**
   We’ll select *Generate autocomplete from predefined templates* so we can select one of the supplied template files.

3. **Setting the Provider properties**
   Select the template you want to use in the **Provider properties** list.

   **Provider properties**
   Select template from the list below:

<table>
<thead>
<tr>
<th>Countries</th>
</tr>
</thead>
<tbody>
<tr>
<td>European Capitals</td>
</tr>
<tr>
<td>United States</td>
</tr>
<tr>
<td>USA ZIP codes</td>
</tr>
</tbody>
</table>

4. **Final Steps**
   Press **OK** to generate your AJ AX AutoComplete field.

   Be sure to copy the images folder and the styles folder to your server.

   Press **OK** and **press f12** to preview and **save**. That’s it you’ve created your own **AutoComplete field**!
Editing the Field with the Property Inspector

Introduction

In this chapter we’re going to edit the properties of the AutoComplete field using the property inspector of Dreamweaver. Click on the preview of the AutoComplete field in the design view to display its properties. Below is a screenshot of the property inspector.

The Property Inspector

1. **Setting the basic options**

   The *Field name* is used to identify the field, make sure it is unique for every field you use, we’ll leave it unchanged as we’ll only use one field, there is no need to identify it.

   ![Property Inspector Screen](image)

   **Char width** sets the width of the entry field and **Max chars** determines the maximum number of characters that can be entered, we’ll set them both to twelve.

   ![Property Inspector Screen with Char width and Max chars set to 12](image)

   **Min. chars** determines the minimum number of characters that needs to be entered, we’ll leave it to 2.
Check the **Autofill** option to set the AutoCompletion of words when typing in the entry field (it is the equivalent of the **Autoselect first suggestion** option in the main menu).

**The Initial value** sets the default value that is entered in your field, we’ll leave it empty.

---

2. **Setting the Animation options**

   - **Show** sets the incoming transition and **Hide** sets the outgoing transition of the completion box.

   *Delay* sets the time it takes before the search suggestions field is displayed, we’ll set it back to 100.
### Properties

#### Ajax Autocomplete

<table>
<thead>
<tr>
<th>Field name</th>
<th>Char width</th>
<th>Min.chars</th>
<th>Max.chars</th>
<th>Initial value</th>
<th>Autocomplete</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>12</td>
<td>2</td>
<td>12</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

#### Animation

- **Show**: Fade
- **Delay**: 100 ms
- **Hide**: Fade
- **Opacity**: 60%

---

**Opacity** sets the transparency of the search suggestions field.

### Properties

#### Ajax Autocomplete

<table>
<thead>
<tr>
<th>Field name</th>
<th>Char width</th>
<th>Min.chars</th>
<th>Max.chars</th>
<th>Initial value</th>
<th>Autocomplete</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>12</td>
<td>2</td>
<td>12</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

#### Animation

- **Show**: Fade
- **Delay**: 100 ms
- **Hide**: Fade
- **Opacity**: 60%
Press the Advanced settings... button if you want to edit the main options, such as the Autocomplete data provider.

Advanced Customization

Introduction
In this chapter we are going to show you the layout and design flexibility our extension offers. By editing the autocomplete.css file you can set a huge range of options. We’ll discuss the most important ones.

You can open the CSS file in Dreamweaver by double clicking on it in the CSS panel:
Changing the Completion Box Properties

This table explains how to change the colors and borders of your completion box.

<table>
<thead>
<tr>
<th>Item to change</th>
<th>Relevant CSS rule</th>
<th>Property and value to add or change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background color</td>
<td></td>
<td>background-color</td>
</tr>
<tr>
<td></td>
<td></td>
<td>border</td>
</tr>
<tr>
<td>Border Thickness and Color</td>
<td>.ac_results</td>
<td></td>
</tr>
<tr>
<td>The color of highlighted characters of the completion box that match with the entry field</td>
<td>.ac_highlight</td>
<td>color</td>
</tr>
<tr>
<td>Background color of the selected item in the completion box</td>
<td>.ac_over</td>
<td>background-color</td>
</tr>
</tbody>
</table>
# Setting Margins and Paddings

This box explains how you change margins and padding of your completion box items.

<table>
<thead>
<tr>
<th>Item to change</th>
<th>Relevant CSS rule</th>
<th>Property and value to add or change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Padding around the results</td>
<td>.ac_results</td>
<td>padding</td>
</tr>
<tr>
<td>The margin around the individual list items</td>
<td>.ac_results li</td>
<td>margin</td>
</tr>
<tr>
<td>The margin around the list items as a group</td>
<td>.ac_results ul</td>
<td>margin</td>
</tr>
<tr>
<td>The left padding around the individual list items</td>
<td>.ac_results li</td>
<td>padding</td>
</tr>
<tr>
<td>The left padding on the list items as a group</td>
<td>.ac_results ul</td>
<td>padding</td>
</tr>
</tbody>
</table>

# Setting the Font Properties

This box explains how you change the font and its properties.

<table>
<thead>
<tr>
<th>Item to change</th>
<th>Relevant CSS rule</th>
<th>Property and value to add or change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Font of the completion list items</td>
<td>.ac_results li</td>
<td>font</td>
</tr>
<tr>
<td>Font size of the completion list items</td>
<td>.ac_results li</td>
<td>font-size</td>
</tr>
</tbody>
</table>
FAQ

Q: **My AutoComplete does not work**
A: Make sure you have uploaded all the files and folders to your server and that JavaScript is enabled in your browser.

Make sure you’ve uploaded the following files in your ScriptLibrary folder:
- `autocomplete.css`
- `jquery-latest.pack.js`
- `jquery.autocomplete.js`
- `jquery.bgiframe.min.js`
- `jquery.dimensions.pack.js`

Also, do not forget to upload the files that list the text `autocomplete` in their filenames (their names are generated dynamically so they may differ from yours) such as:
- `autocomplete-pagename-xxx-1.txt`
- `autocomplete-tpl-pagename-xxx-Countries-1.txt`
- `autocomplete-pagename-xxx-1.asp`
- `autocomplete-pagename-xxx-1.php`

Q: **Can I use a database to provide the AutoComplete information?**
A: Yes, the extension will detect all the connections defined on your site and provides the opportunity to select a particular table and various fields for its auto-complete information.

**Note:** This feature only works for sites that use PHP or ASP VBScript.

Q: **Which server-side technologies are supported?**
A: The extension could be used with all server-side technologies supported by Dreamweaver. But the database interface integration can only be used with PHP or ASP VBScript.