

Table of contents

Table of contents.....	1
About Advanced CSS Animator.....	2
Features in Detail	3
What people say about Advanced CSS Animator:	10
The Basics: Creating Pupup Effect on Mouse Over	12
Advanced: Creating Slide Out Menu	24
Reference: Basic CSS properties.....	41
Reference: Advanced CSS Properties	44
Video: Creating a Popup Effect on Mouse Over.....	45
Video: Creating Slide Out Menu.....	45
Tutorial: Getting Started with the Advanced CSS Animator	45



About Advanced CSS Animator

Need full power and control of your CSS based animations? Awesome effects combined all together? Full Cross browser support and no Flash at all? The Advanced CSS Animator is the answer to your prayers!

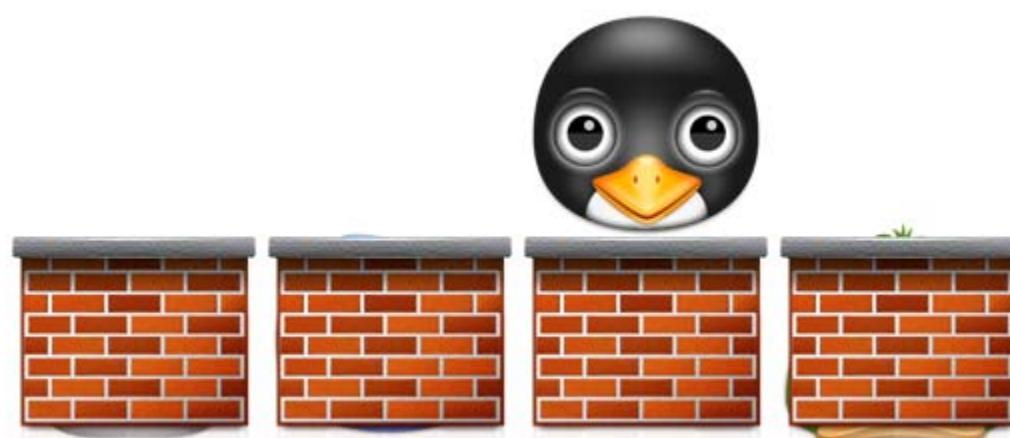
With the Advanced CSS Animator you can animate virtually any CSS Property! Borders, colors, font sizes, position, opacity, apply CSS Transformations like rotate, skew, zoom and more! Use the supplied presets to choose from ready animations like fade in/out, expand, collapse, position in browser, paper spin, shake, blink or save your own CSS animations in a new preset! Need more than one animation – well add as many as you wish! Those will be nicely added to the animation queue. You can run animations simultaneously, apply timing and easing and have a precise control of the animations queue!

Features in Detail

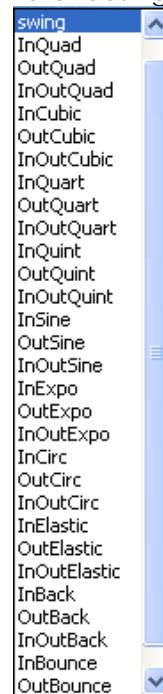
- Pure CSS Based Animation – No Flash required!



- Any CSS Property animation - You can animate all CSS Properties like colors, fonts, sizes, position and more!



- **Full Visual Transformation** – When you animate an object it will visually transform from the old to the new state in the given duration! So it will grow or shrink, rotate or move whatever animation is chosen!
- **Apply great effects and easing** – you can achieve fantastic speed effects, bouncing and elastic movements by giving an effect duration and choosing the easing you want. [See the easing diagram.](#)



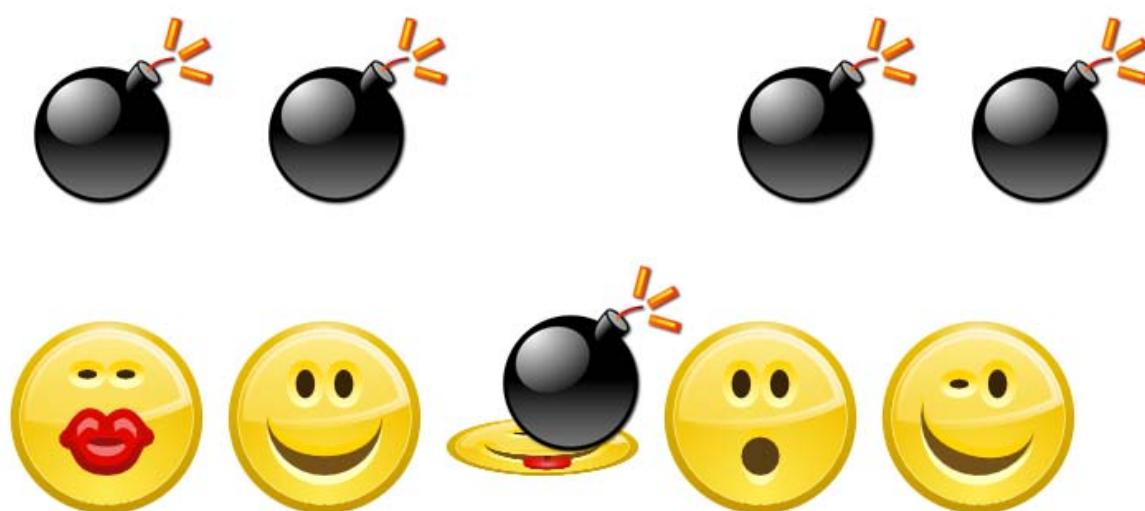
- **Animate Multiple Elements** - animations are fully CSS Selector based, so you can choose to animate one or more objects by just giving the right CSS Selector.
- **Use CSS Transitions** – with CSS transitions you can finally move and/or transform any element as you wish! All cross browser! If some of the CSS Transitions properties are not supported on your browser they will be emulated so you still achieve the same great effect!



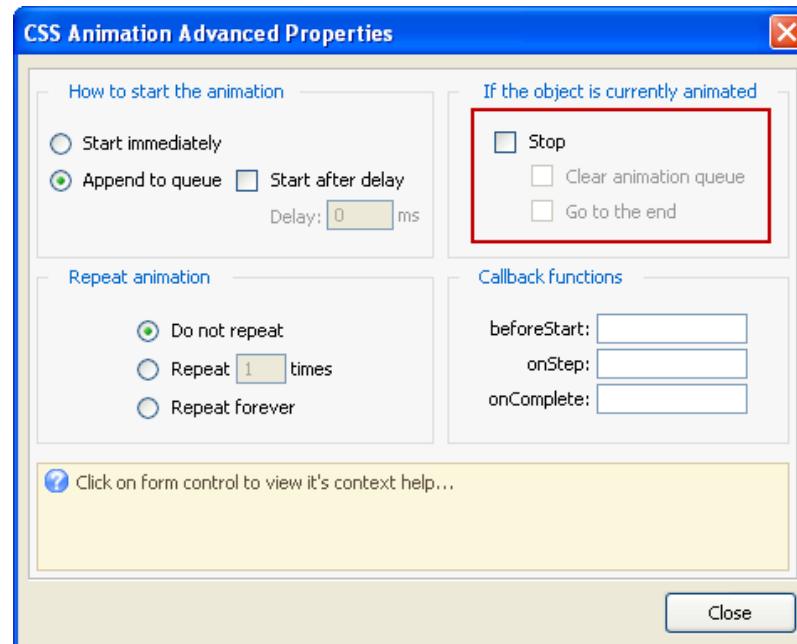
- **Make quick easy animations out of the box** - Select from the many presets we have include or save your own animations as new preset!



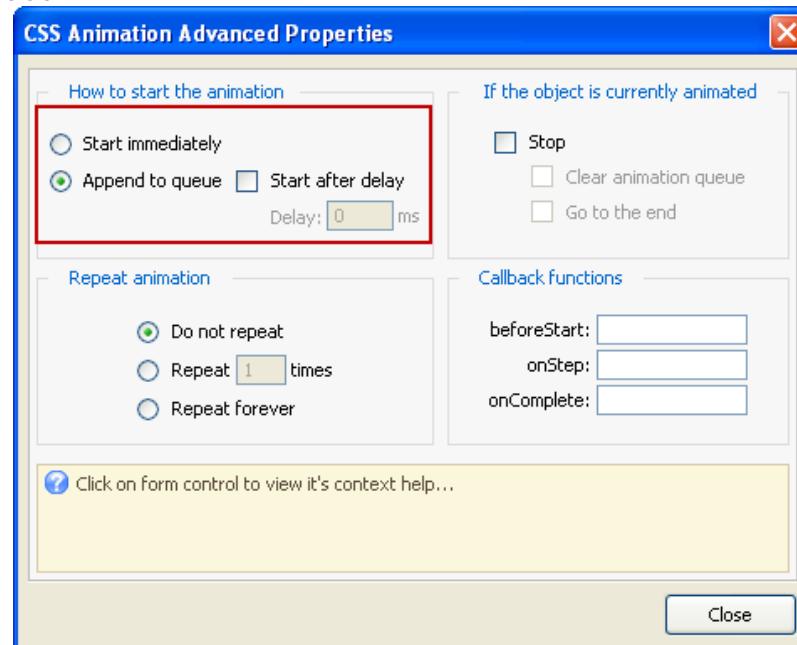
- **Custom Presets** – Produce some great CSS3 effects for your site in no time.



- **Full support for On Mouse Enter and On Mouse Leave events** - Sometimes, On Mouse Over and On Mouse Out events are not enough as they work bad on containers having children. Now we have extended Dreamweaver to offer you support the new special on Mouse Enter and on Mouse Leave events, so now you have full control when animations should start and stop!

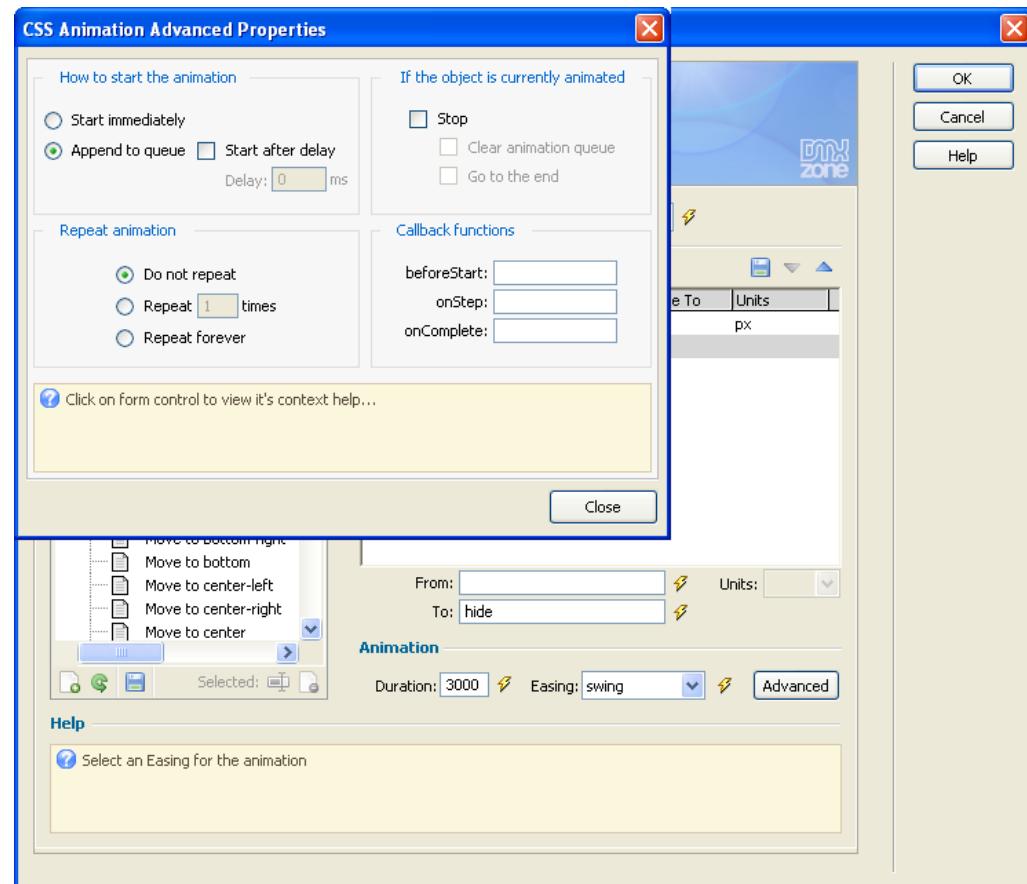


- **Full animation queue support** – run animations simultaneously or after each other! Stop previous animations if needed.

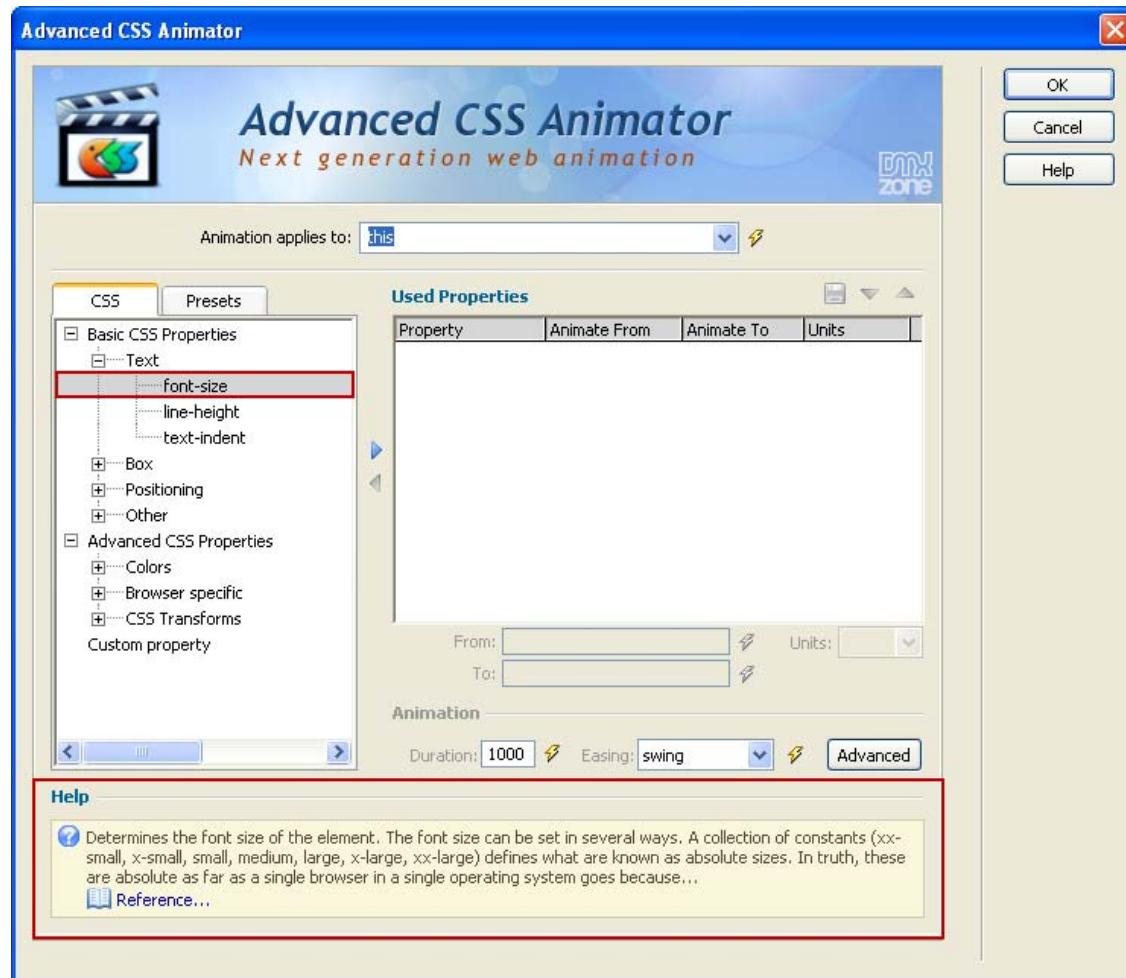


- **Time your animations** – give your animations a duration and start delay if needed.

- **Customizable animation** - Adjust the animation appearance by duration, easing, repeating and more...



- **Full CSS Reference included** - now by just browsing the different CSS Properties shown in the Advanced CSS Animator, you can directly see explanation about them as context help! You can even read more for full explanation of the CSS Property and what it does.



- **Great Dreamweaver Integration** - Interactive dialog in Dreamweaver with all the options you'll need.



- **Use jQuery as power engine** – to achieve all those smoothing animations and CSS support we have used jQuery as industry based standard for building Web 2.0 sites.
- **Search Engine Friendly** - as the Advanced CSS Animator is pure HTML and CSS code it can be nicely indexed by all search engines and web spiders.



- **Fully cross-browser compatible** - Create stunning CSS3 effects even in Internet Explorer thanks to the DMXzone's clever use of JavaScript.

What people say about Advanced CSS Animator:

"Advanced CSS Animator is startling new toy chest for web designers that give you the power to achieve Flash-like results without Flash."

Joseph Lowery

Dreamweaver Bible author and Website Expert (markofthejoe.com)

"Web Design has been a changing medium over the 14 years I have been working in it, but there has rarely been a time when technology was moving so fast and furiously as now. Terms like HTML5 and CSS3 and AJAX are now familiar but take a fair amount of study to learn. Animation and effects are all the rage with CSS3, but browser support is scanty and in some cases, nonexistent of these technologies. That's why I love the Advanced CSS Animator. Not only are animations and effects easy to apply, but they work in whatever browser your user may view them. You can choose from a number of built in effects and presets or you can create your own custom ones and save them for the future. Build up a library of effects and animations that take just seconds to apply!"

Nancy Gill

Adobe Community Professional

"It's simple to use, the interface is clear, and it offers the non-expert a painless way to use the advanced presentational features of both jQuery and CSS."

Murray R. Summers

Adobe Certified Web Site Developer, Dreamweaver Developer, and Community Expert

Before you begin

Make sure to setup your site in Dreamweaver first before creating a page. If you don't know how to do that please [read this](#).

Installing the extension

Read the tutorial at <http://www.dmxzone.com/go?4671>

Login, download the product, and look up your serial at <http://www.dmxzone.com/myZone/purchases>

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

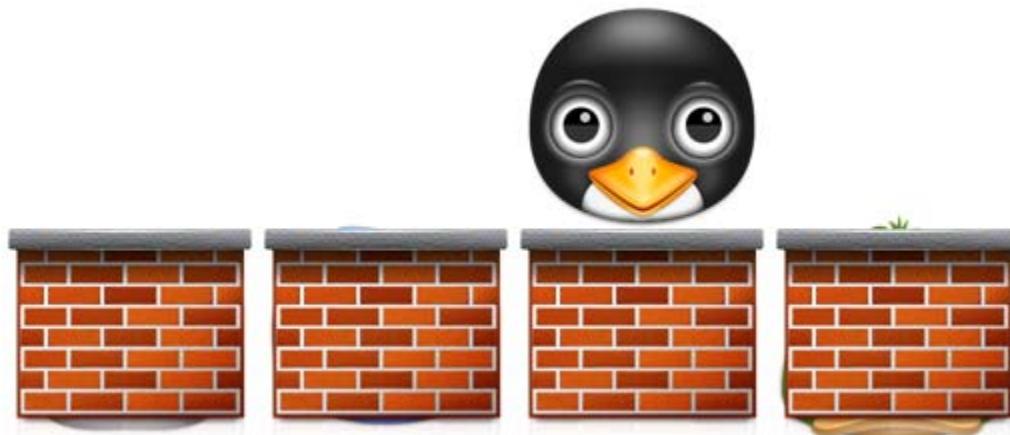
Open the .mxp file (the extension manager should start-up) and install the extension.

If you experience problems contact our Live Support!

The Basics: Creating Popup Effect on Mouse Over

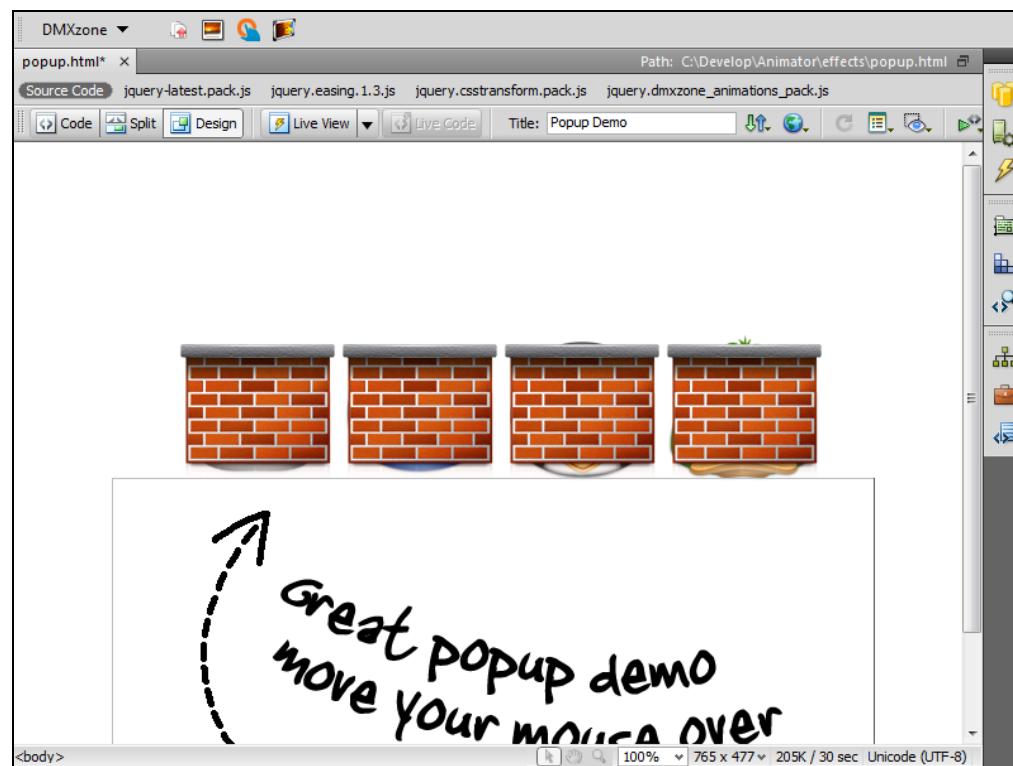
In this tutorial we will show you how to create a great popup effect on mouse over using **Advanced CSS Animator**.

*You can also check out the [video](#).

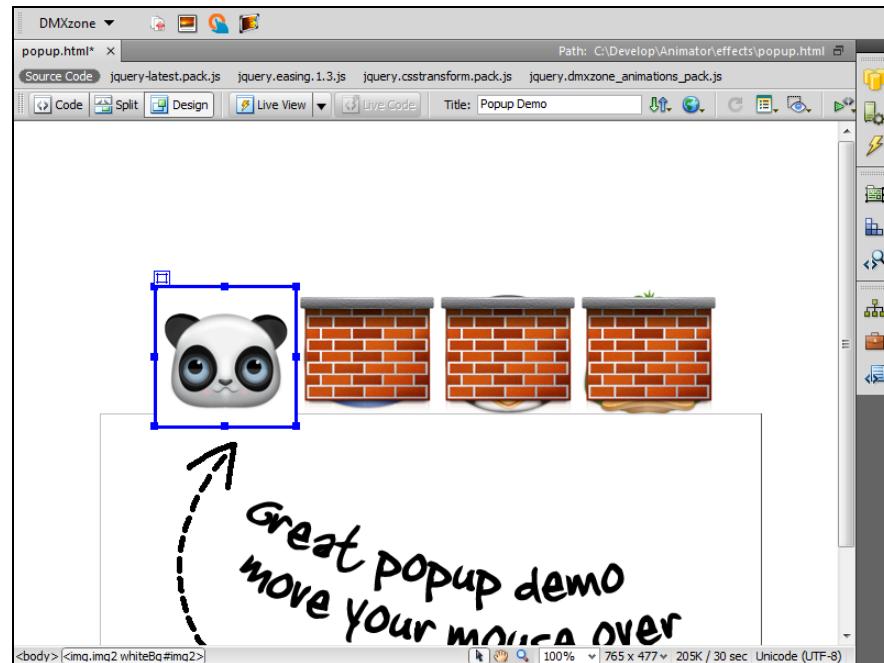


How to do it

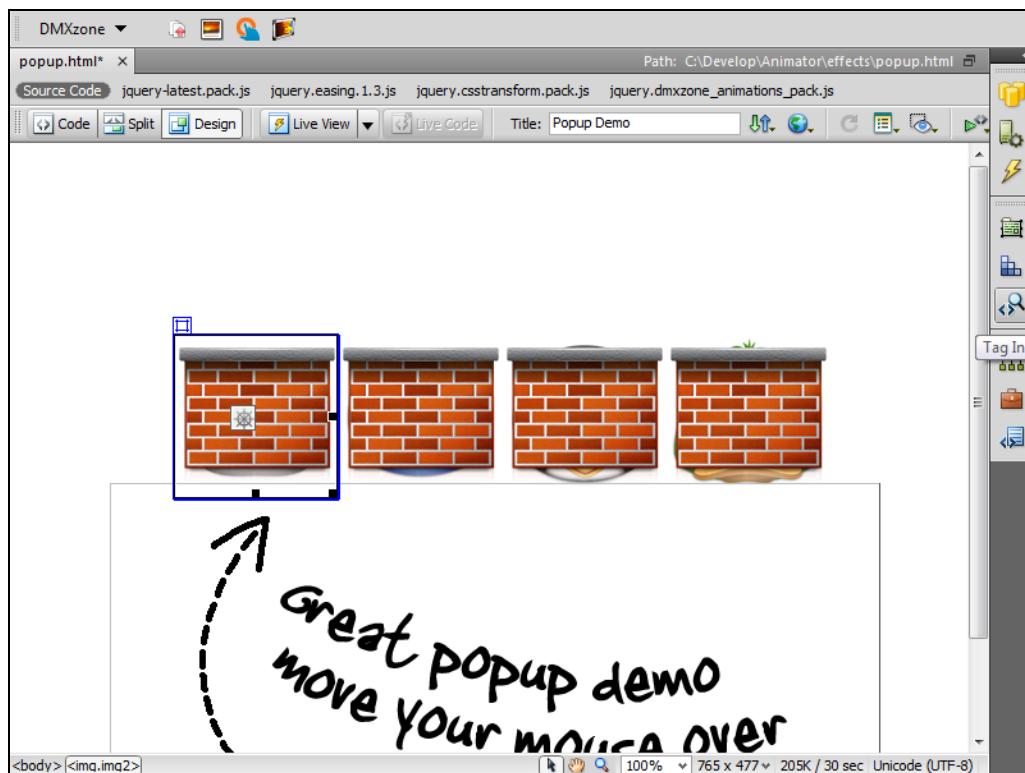
1. Ok, let's start! We've already created our page design. We have four wall images, positioned using CSS and four other images behind them.



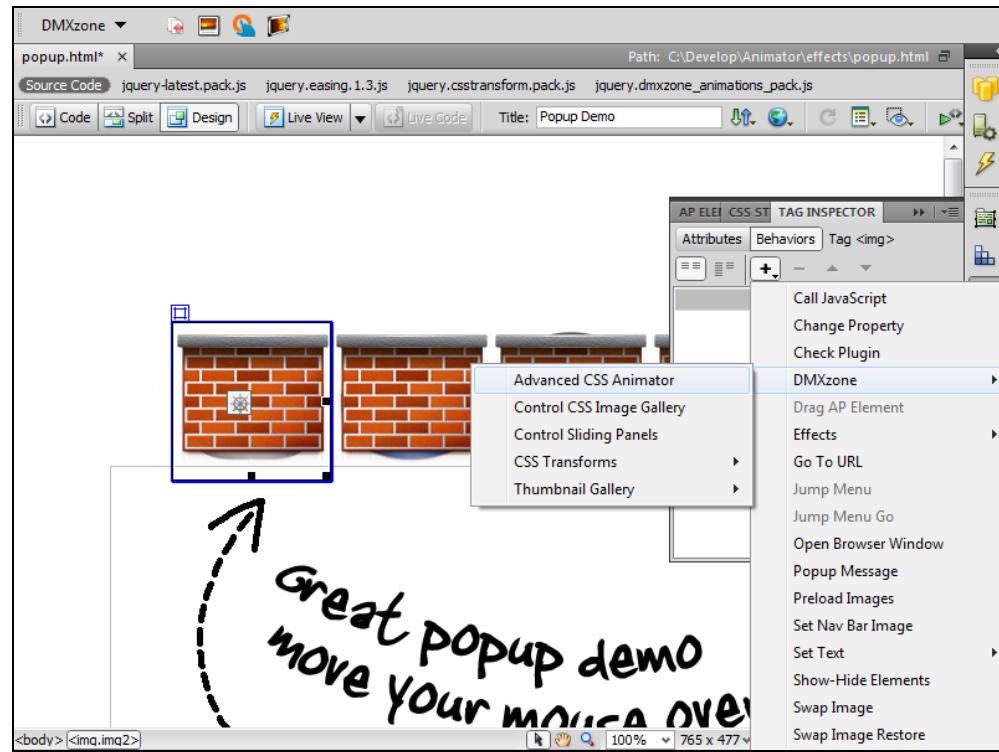
2. Each of the back images has its own **ID**. We are now going to make the back images popup when the mouse goes over the walls in front of them.



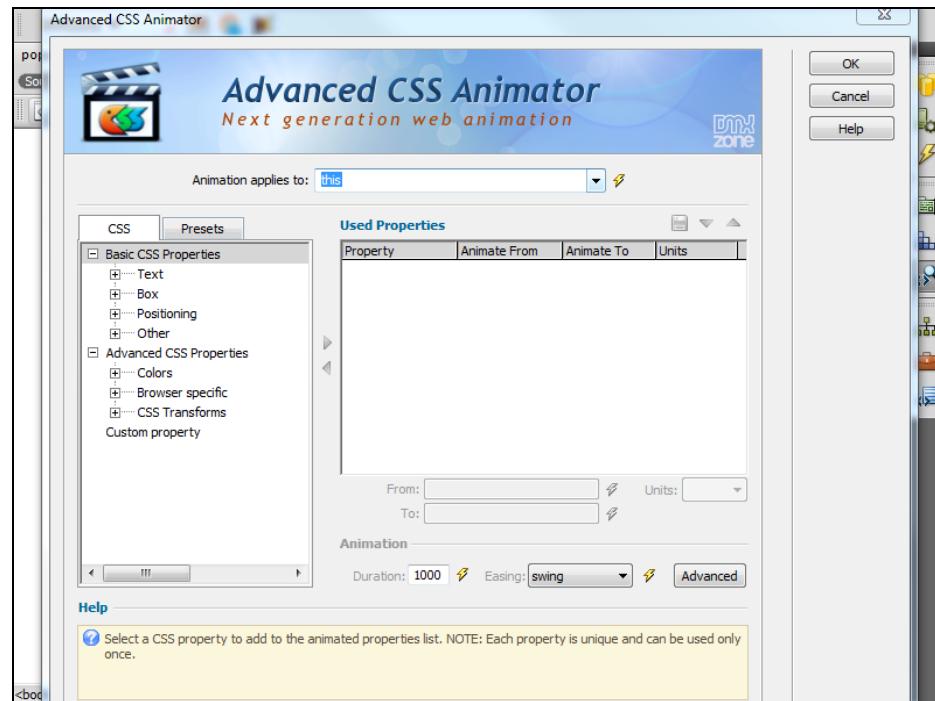
3. Click on the Tag Inspector Icon



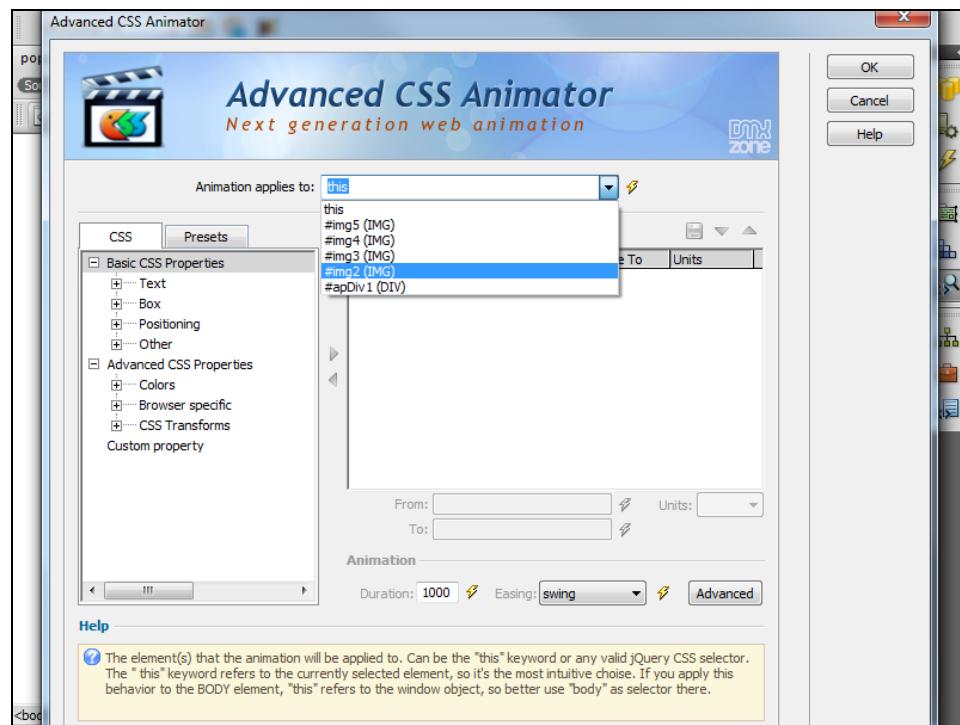
4. Click the Add behavior button



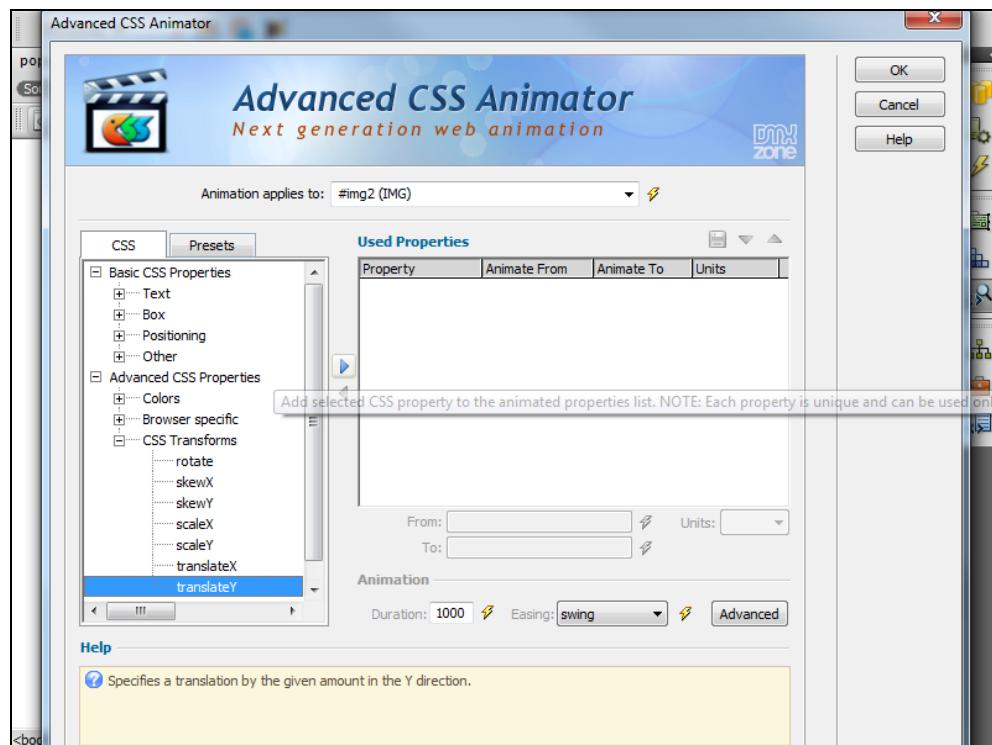
5. The Advanced CSS Animator dialog opens.



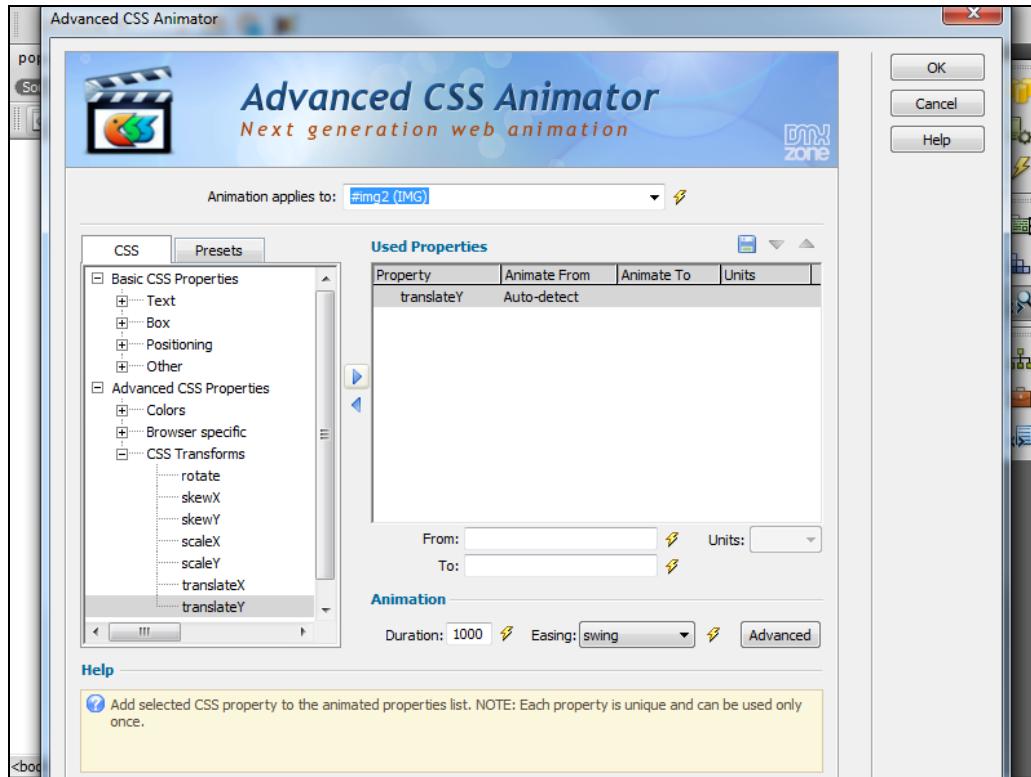
6. We are now going to select the first back image ID in the 'Animation applies to' field. Click the select menu  and select the ID of the first back image 



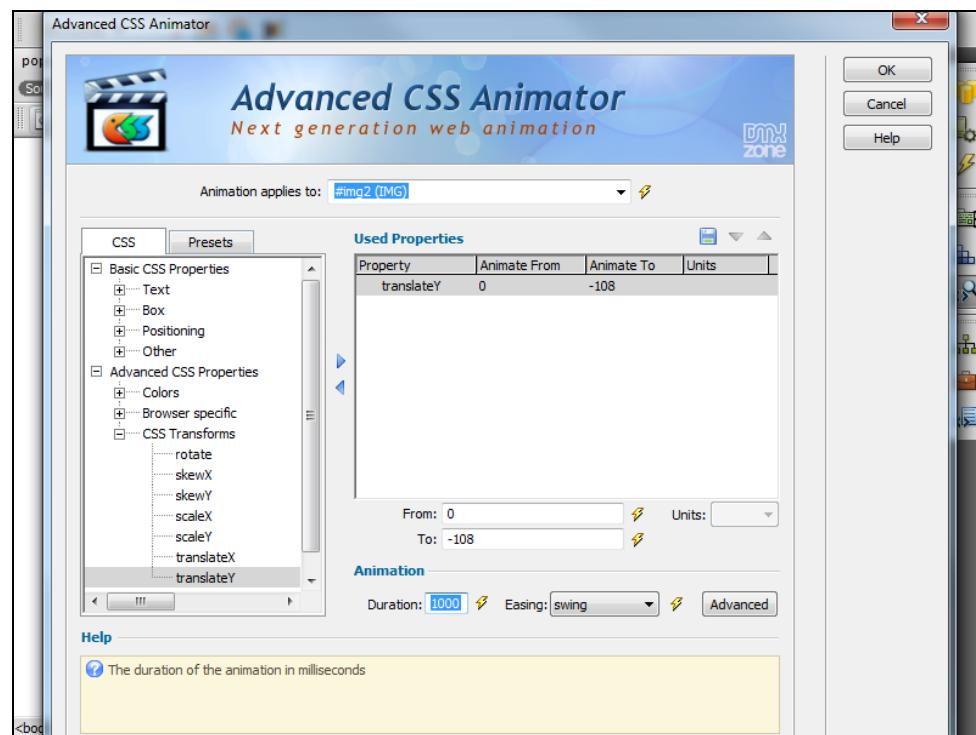
7. Select the TranslateY effect from the CSS Transforms menu 



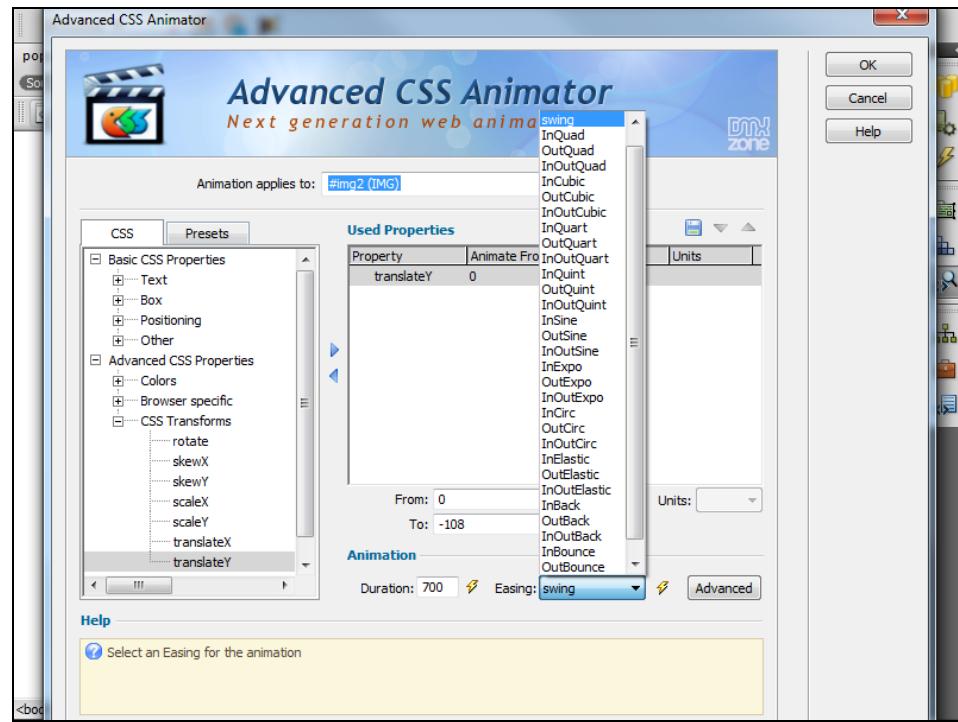
8. Click the blue arrow in order to add the selected effect 



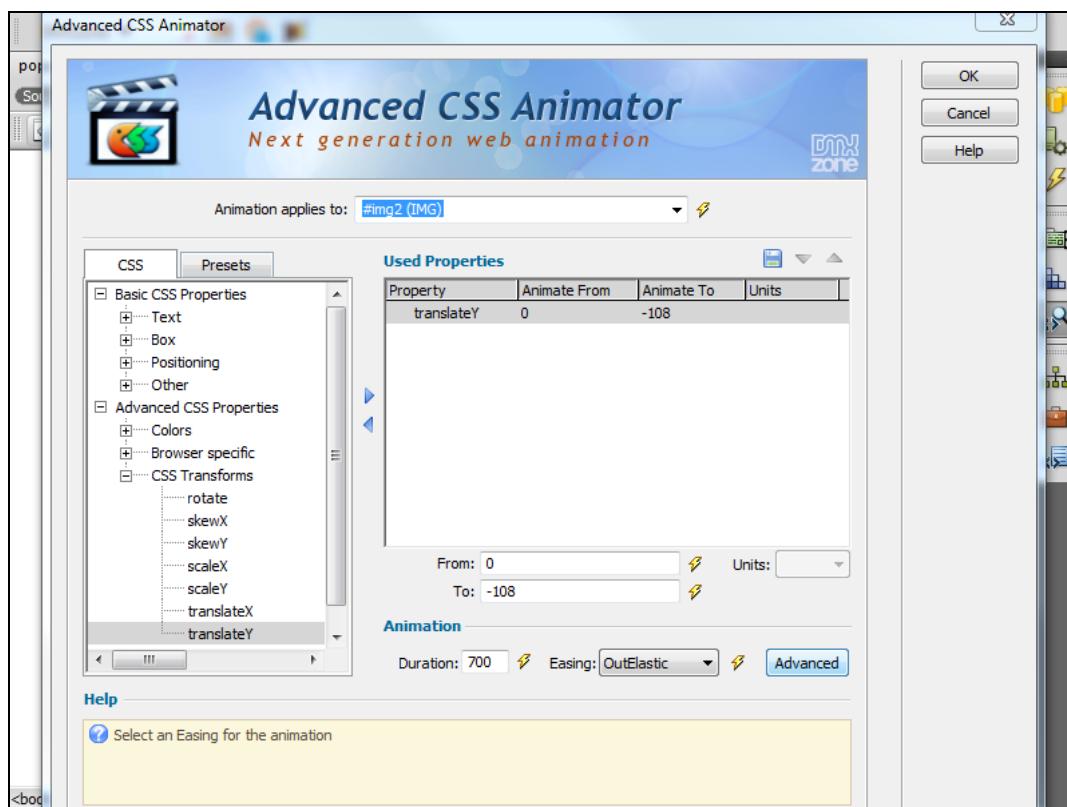
9. Enter 0 as a **start position** of the translation or leave this field blank in order for **Advanced CSS Animator** to auto detect the start position. Enter the **end position** of the translation.

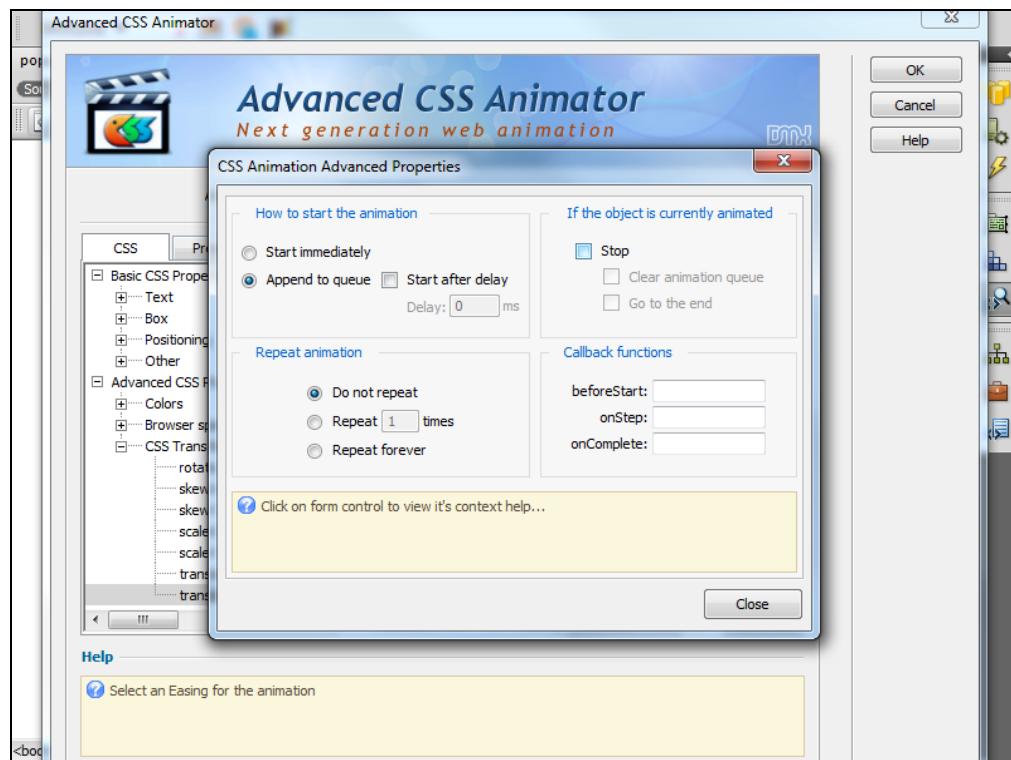


10. Select the Animation duration **1000** and an Easing effect; we will use the **OutElastic** effect **OutElastic**

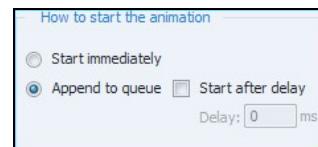


11. Click the Advanced button **Advanced**

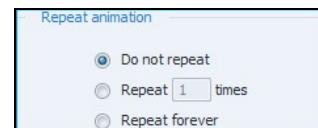




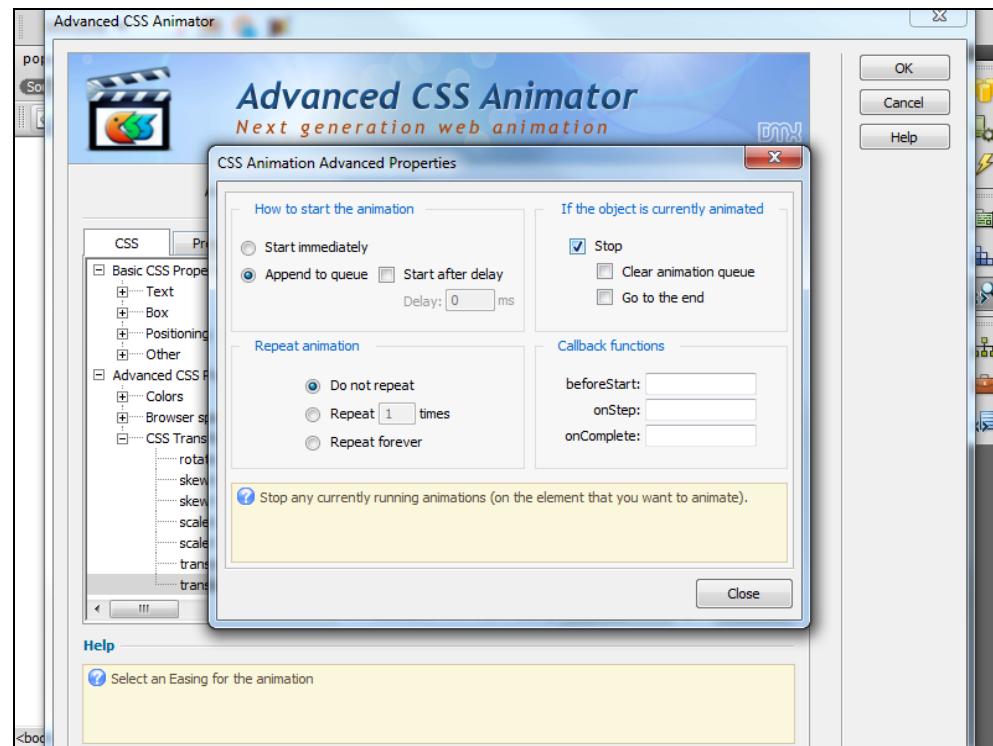
12. Here you can select whether to **Start** the animation immediately or after a delay



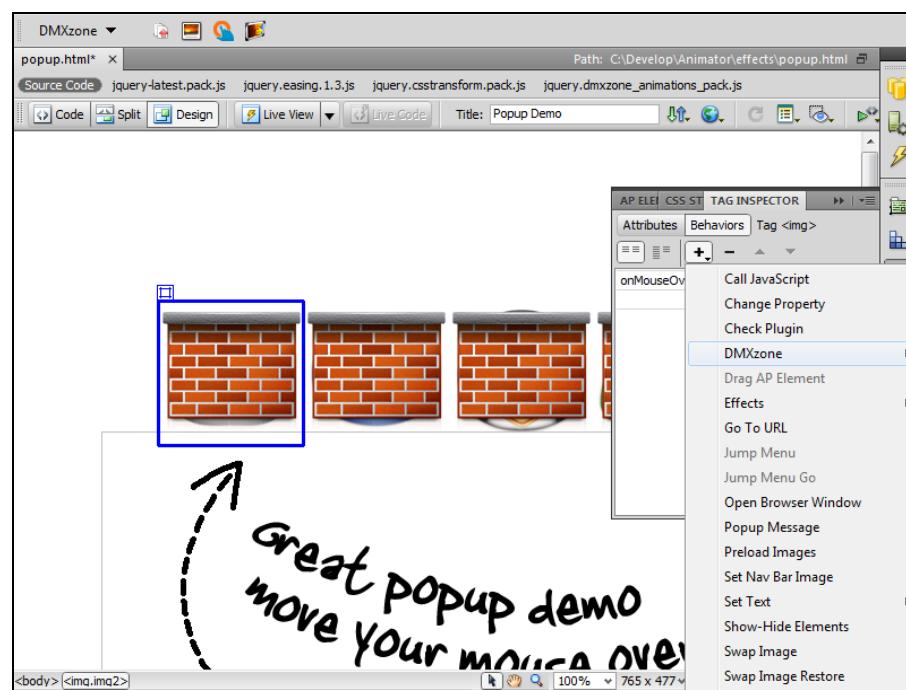
13. Choose whether to **Repeat** the animation or not



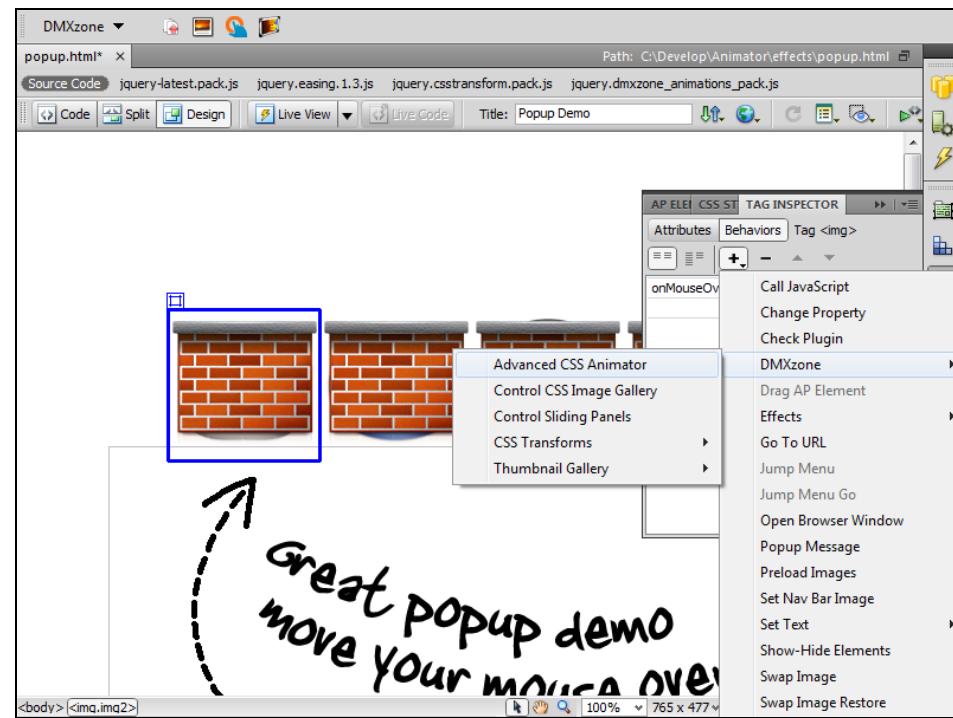
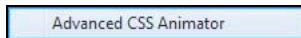
14. Select the **Stop** check box in order to stop any currently running animations on this element and Click the **Close** button when you are done



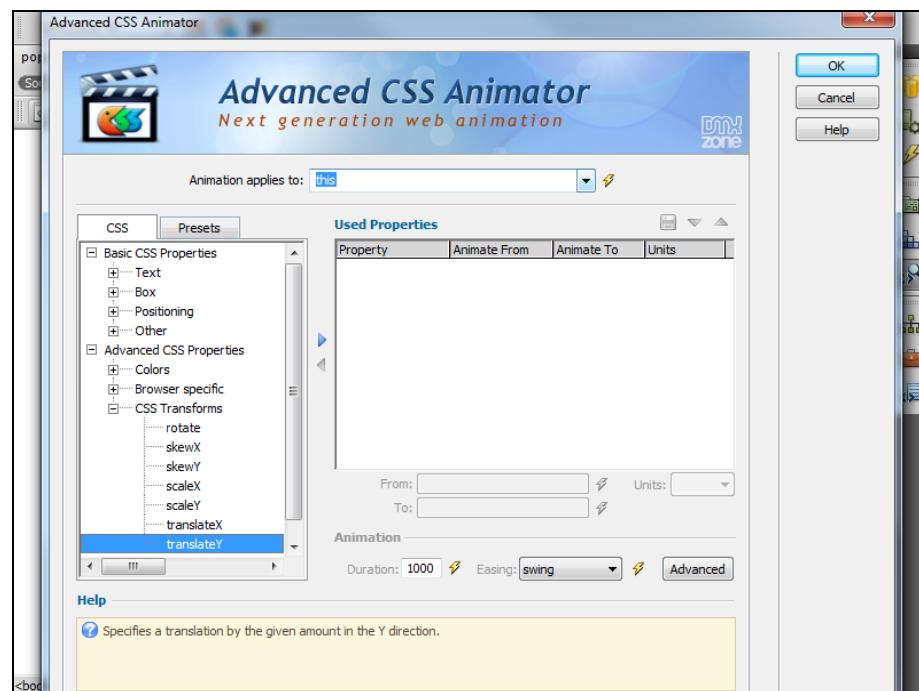
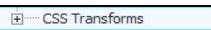
15. After we've added the Popup effect to the picture, we need to hide it behind the wall when the mouse goes out. Click the **Add behavior button** 



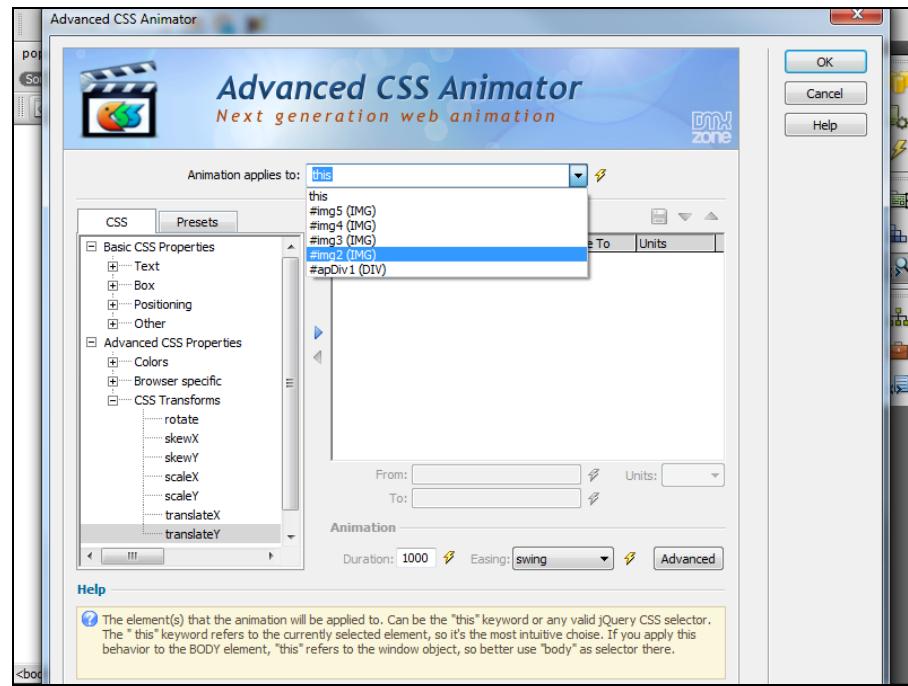
16. Select the **DMXzone** menu item and then the **Advanced CSS Animator**



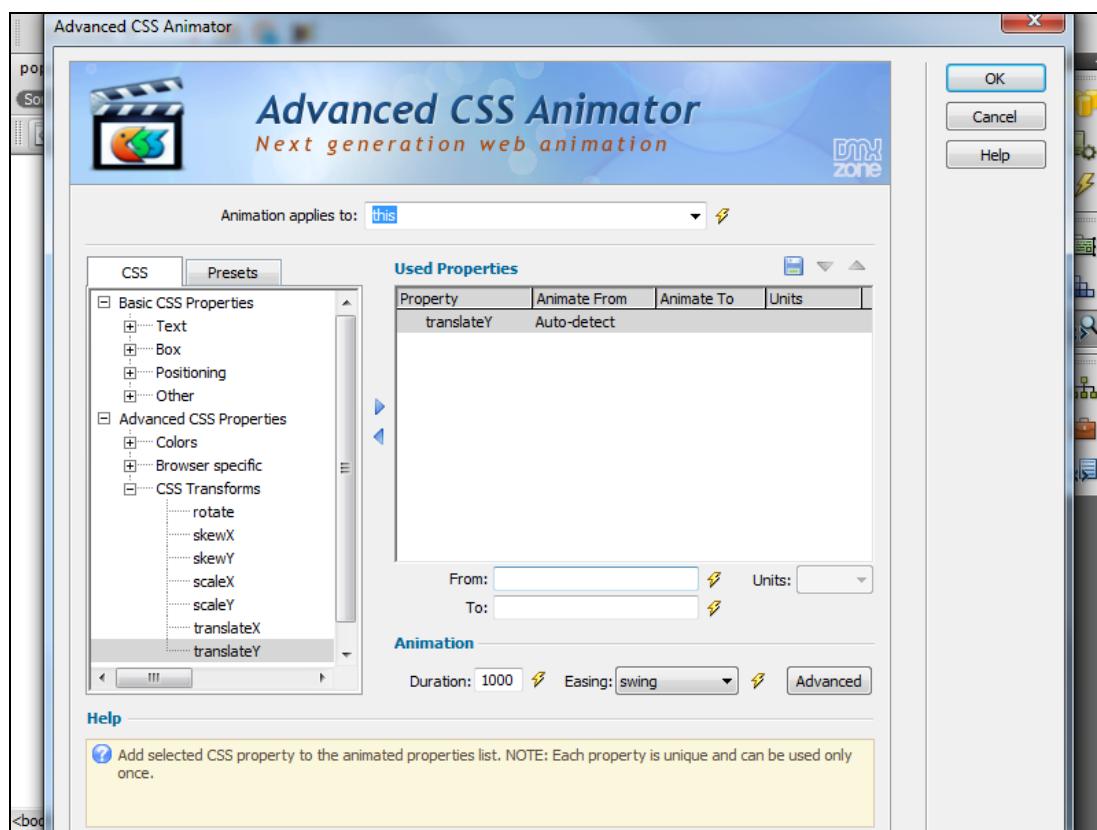
17. The **Advanced CSS Animator** dialog opens. Select **TranslateY** item in the **CSS Transforms** menu



18. Select the ID of the first image again 

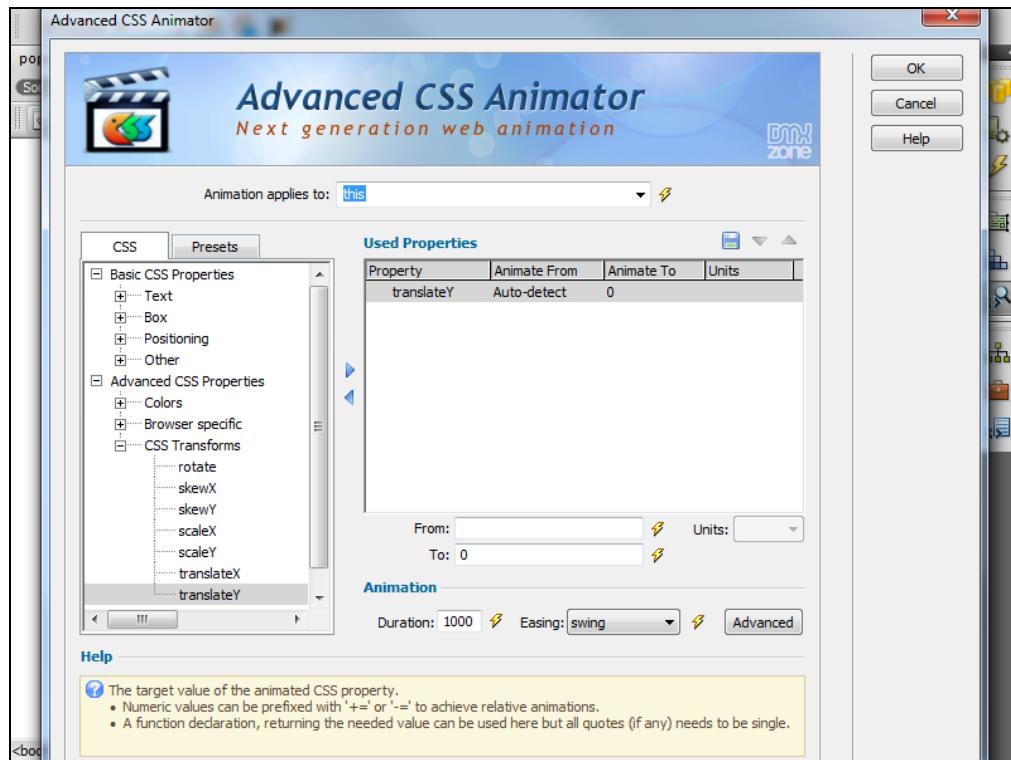


19. Add the translation  and leave the "From:" blank in order for Advanced CSS Animator to auto detect the image position.

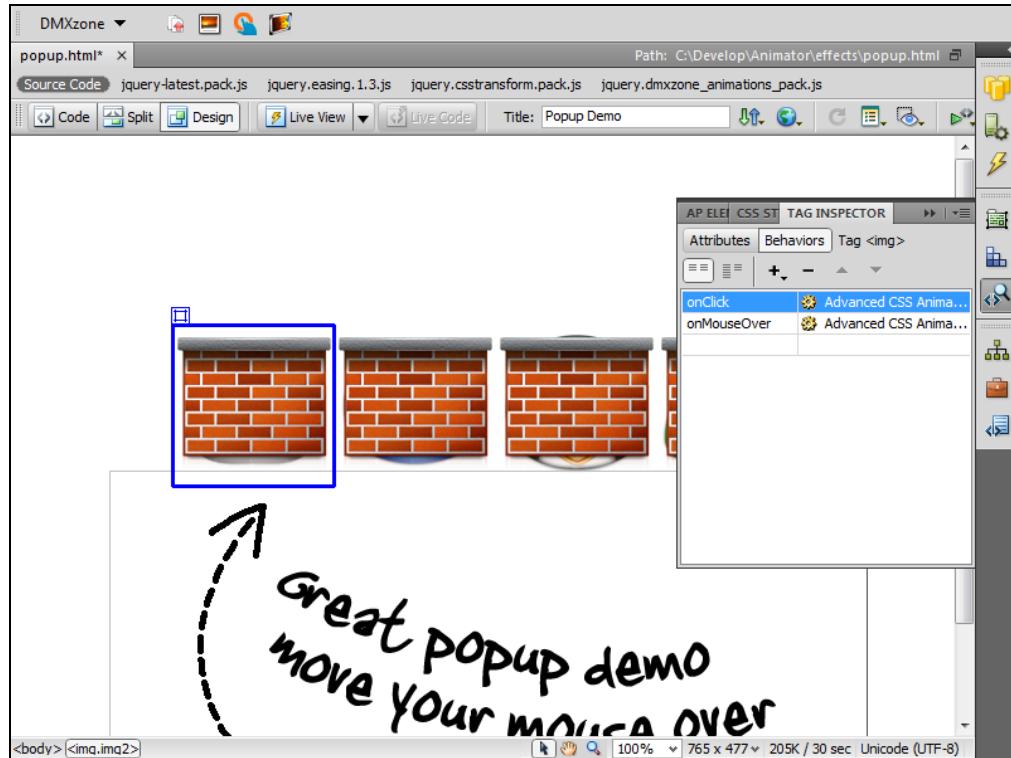


20. Enter **0** as finish position. This will return the image behind the wall picture and Click the **OK** button when you are done

OK



21. We need this animation **onMouseOut** . Do the same with the rest of your images, save your work and preview the result in your browser.



Advanced: Creating Slide Out Menu

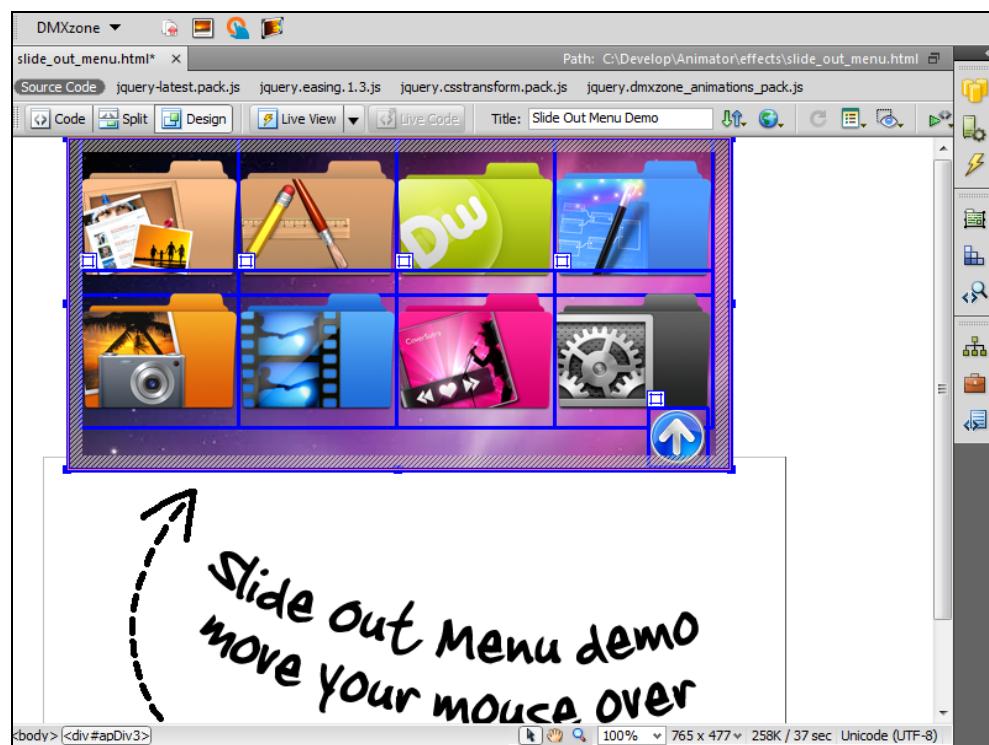
In this tutorial we will show you how to create a **Slide Out Menu** with **Zoom in Menu Items** using **Advanced CSS Animator**

*You can also check out the [video](#).

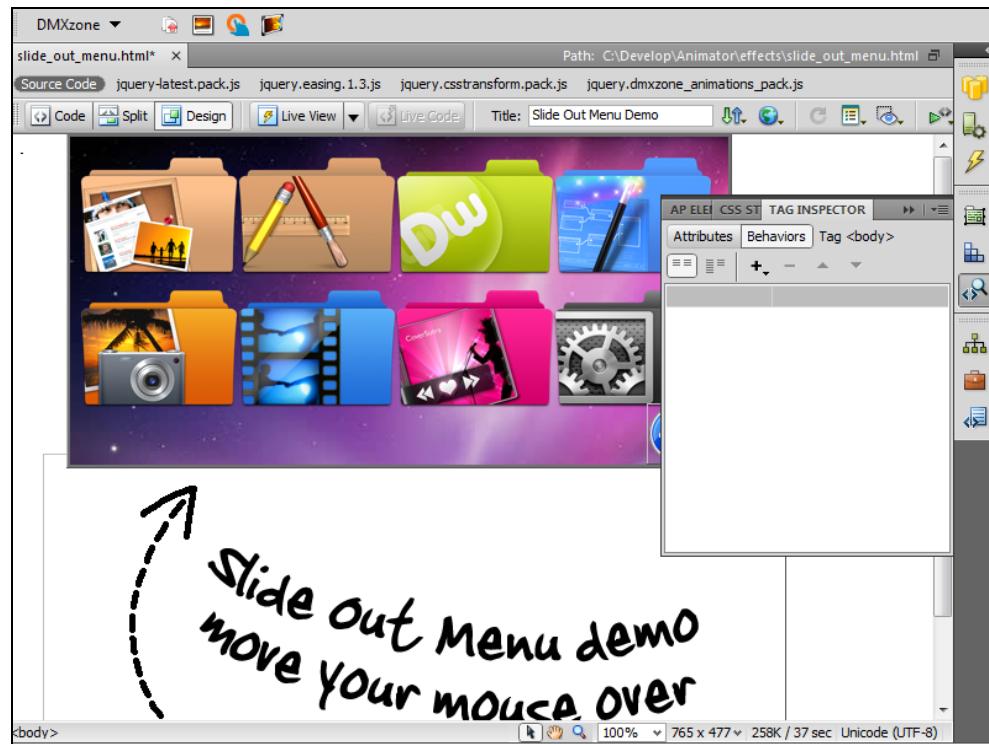


How to do it

1. We've created an absolute positioned **DIV** just below the top side. We've also added some images as menu items inside it.

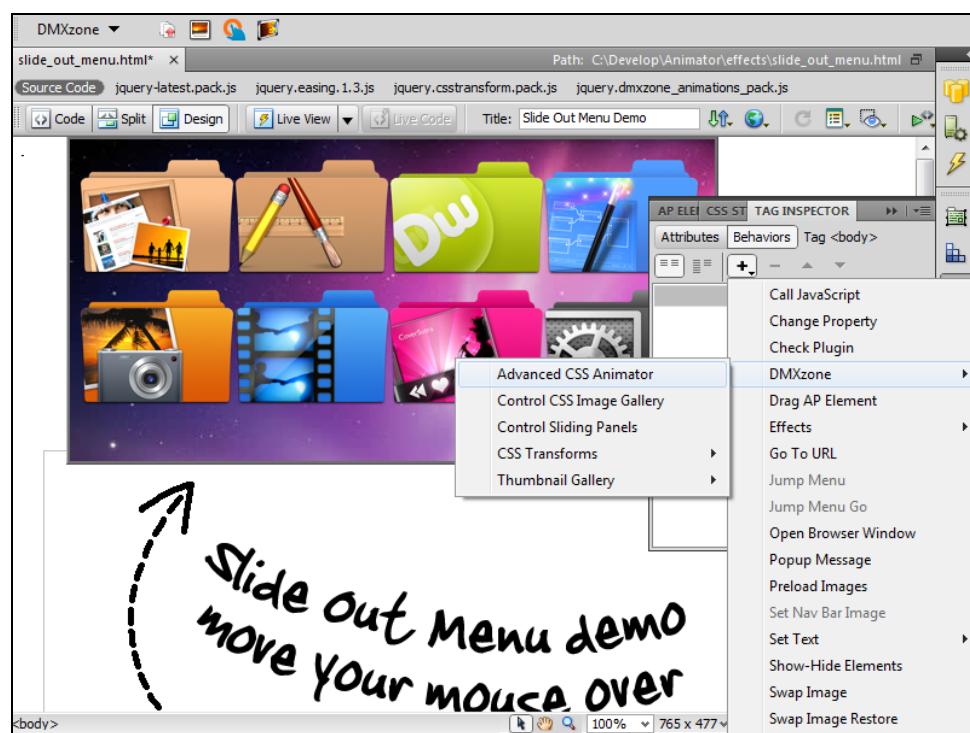


2. First of all we need the **div**, containing our menu items to be pinned up to the top of the page. Click the **Tag Inspector** icon 

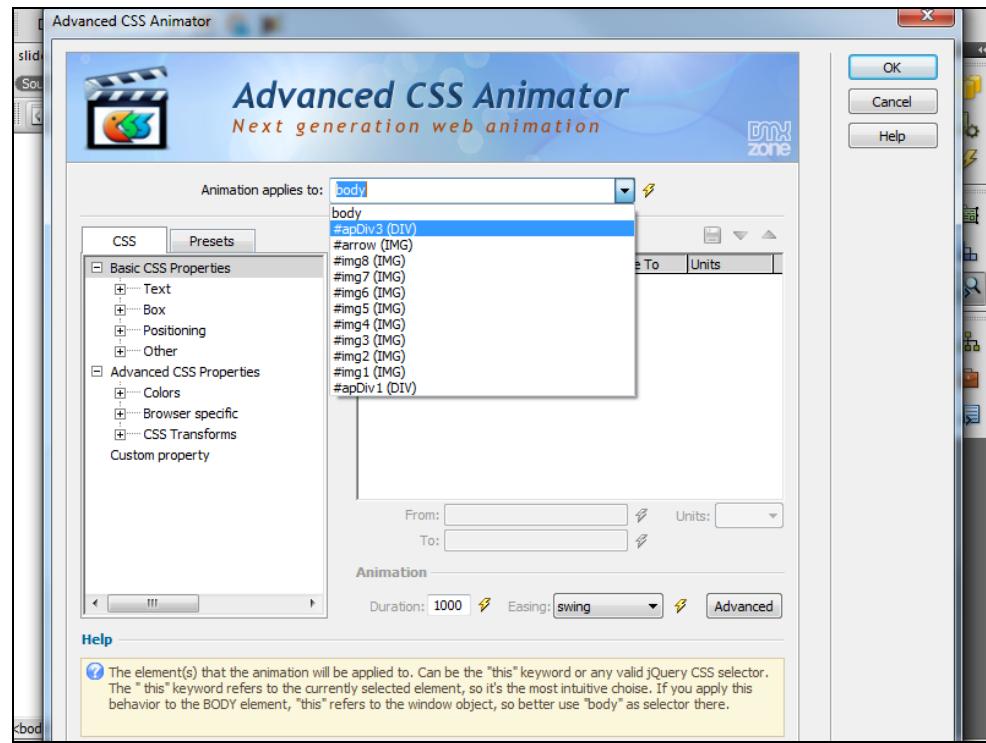


3. Click the **Add behavior** button  and select the **Advanced CSS Animator** menu item

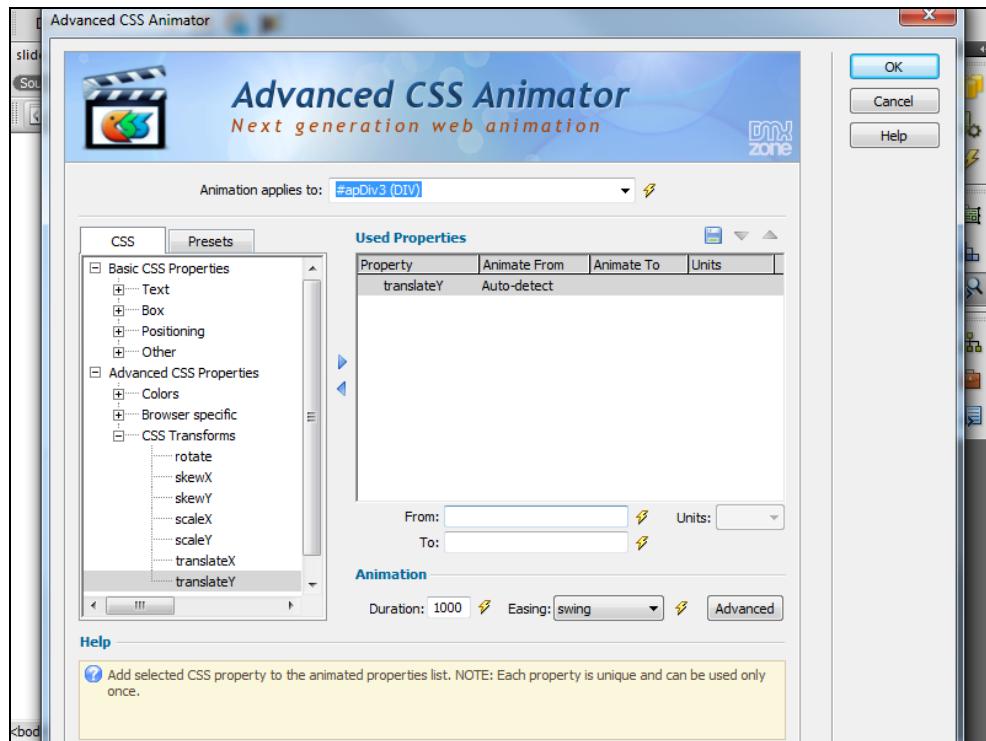
Advanced CSS Animator



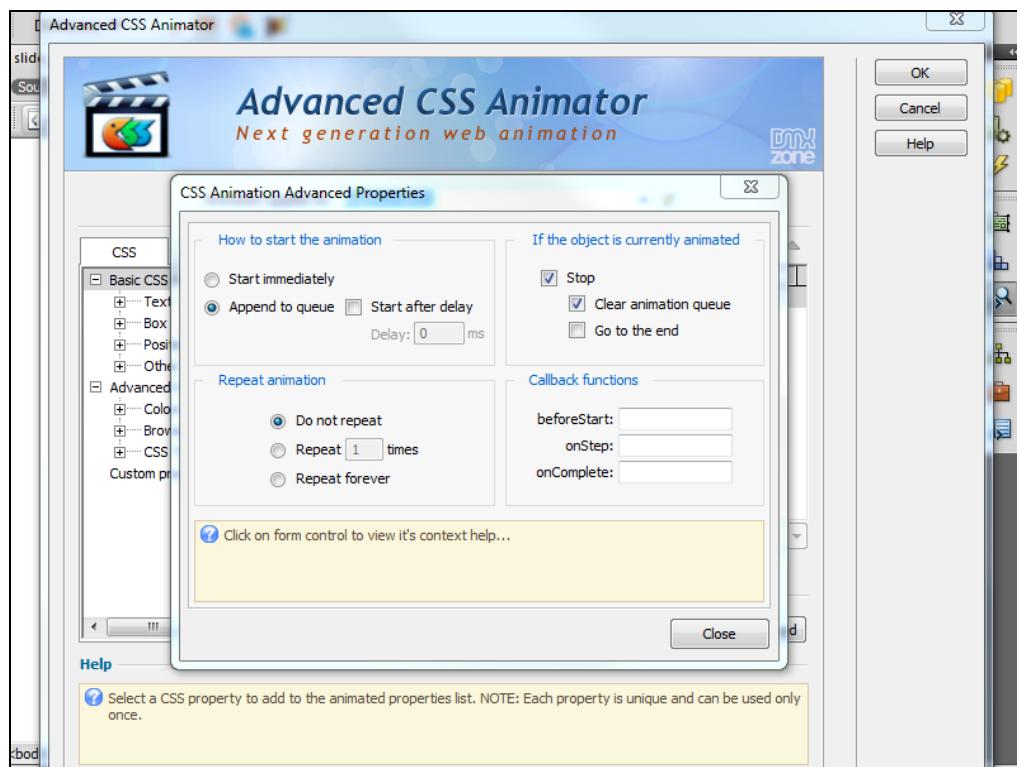
4. The **Advanced CSS Animator** dialog opens. Select the dropdown menu and the ID of the div container



5. Select **TranslateY** item in the **CSS Transforms Menu** and add the translation



6. Enter 0 in the **From:** field, enter the target coordinates of the translation and click the **Advanced** button.



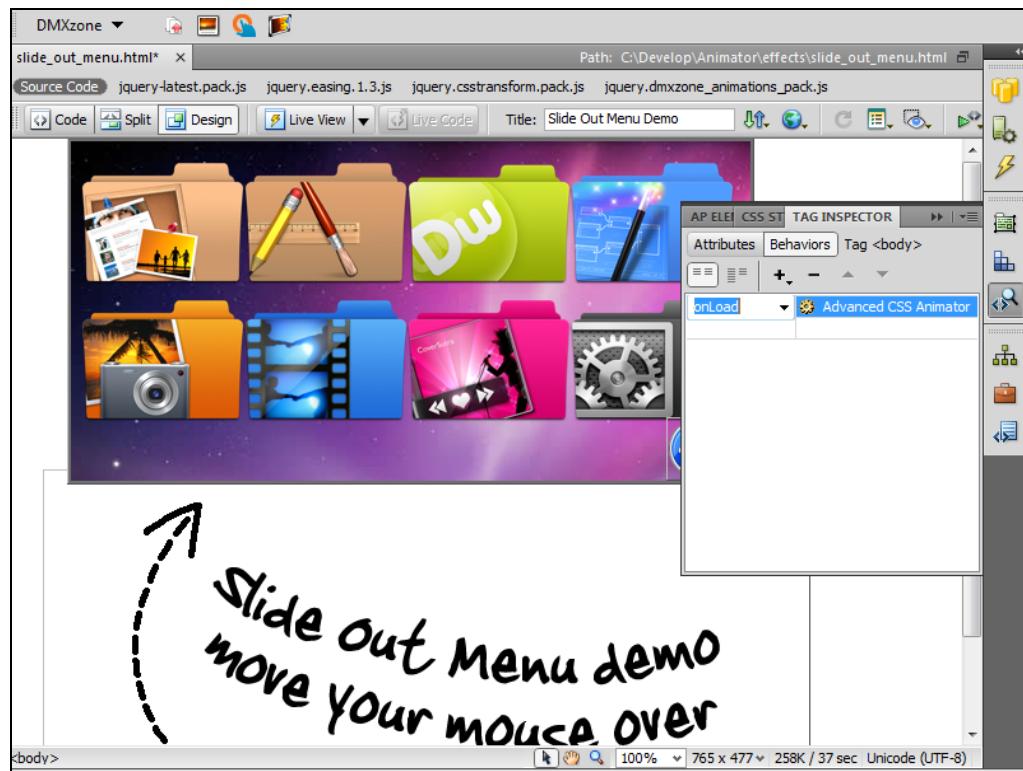
7. Use the following settings

Stop
Clear animation queue
Go to the end

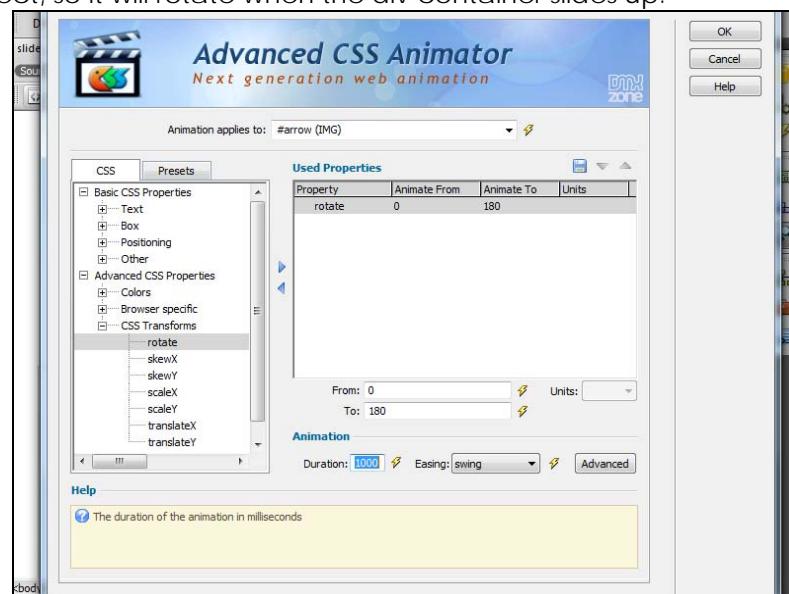
8. We leave the default Animation settings and click the **OK** button

Animation
Duration: 1000 Easing: swing

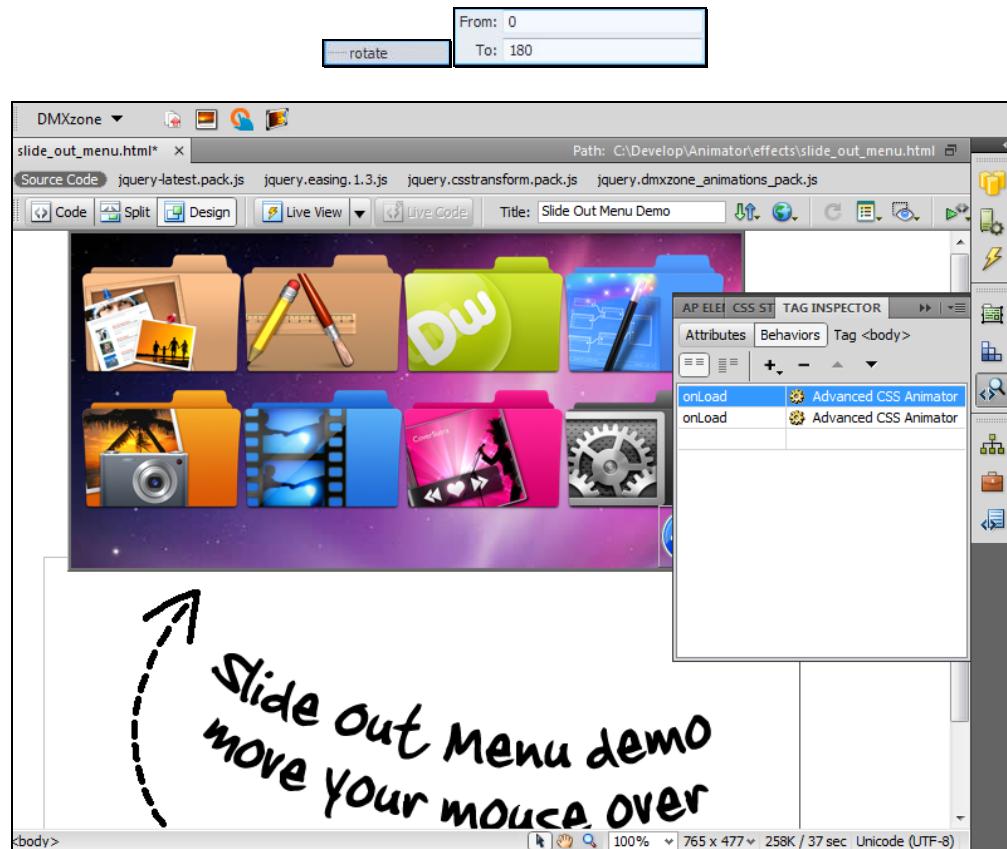
9. Make sure to select **onLoad** event for this animation. Now, when the page loads our div container with menu items will slide up to the top of the page.



10. We do the same with the arrow icon that is located within the div, but instead of translation we use rotation effect, so it will rotate when the div container slides up.

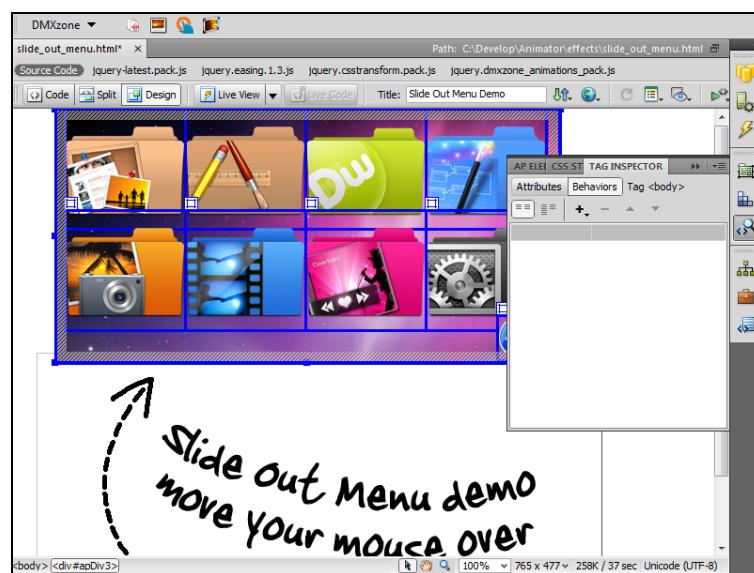


11. Here are the settings for the arrow icon animation and click the **OK** button when you are done

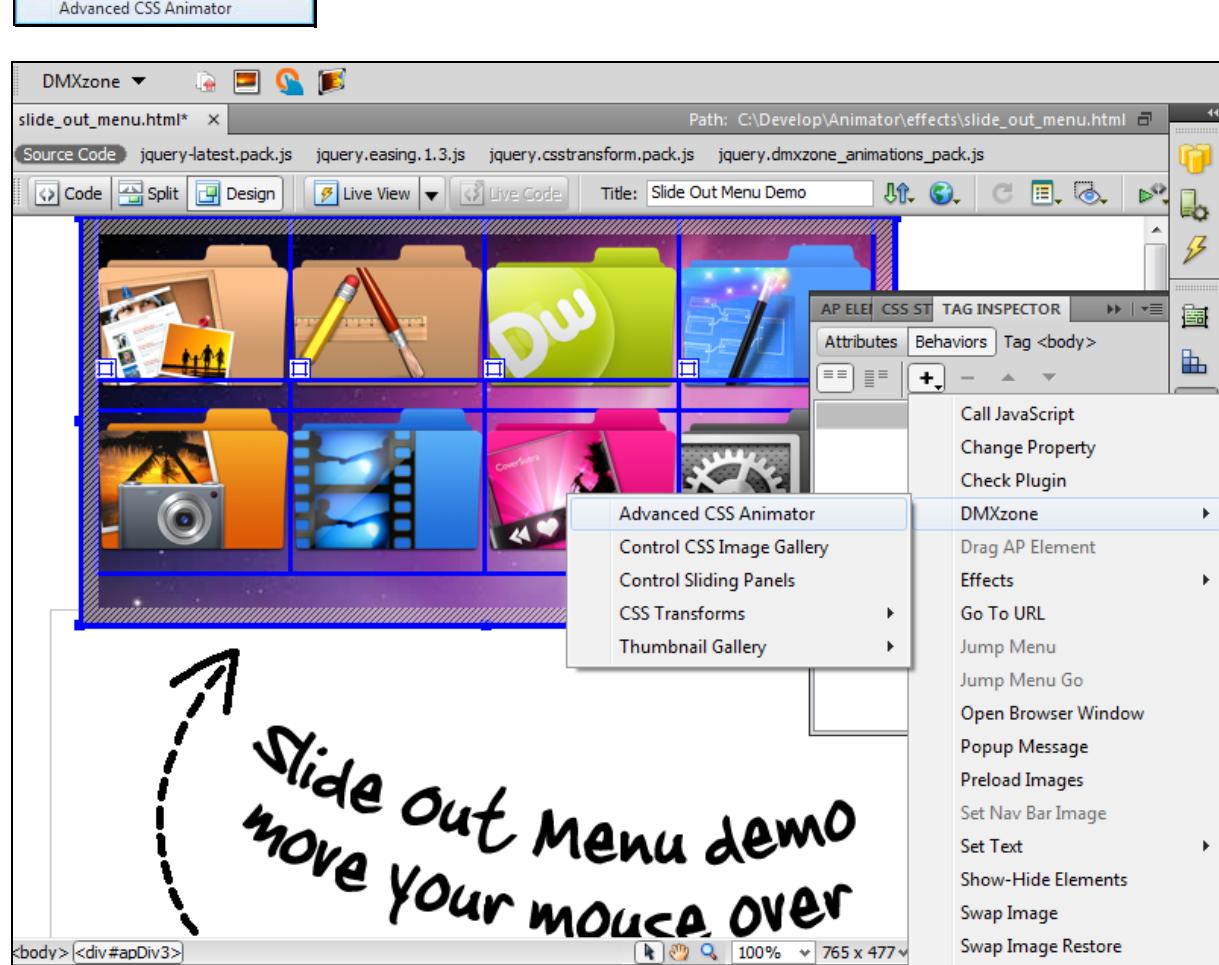


12. We are going to make the div container slide down **onMouseEnter** and slide back up **onMouseLeave**.

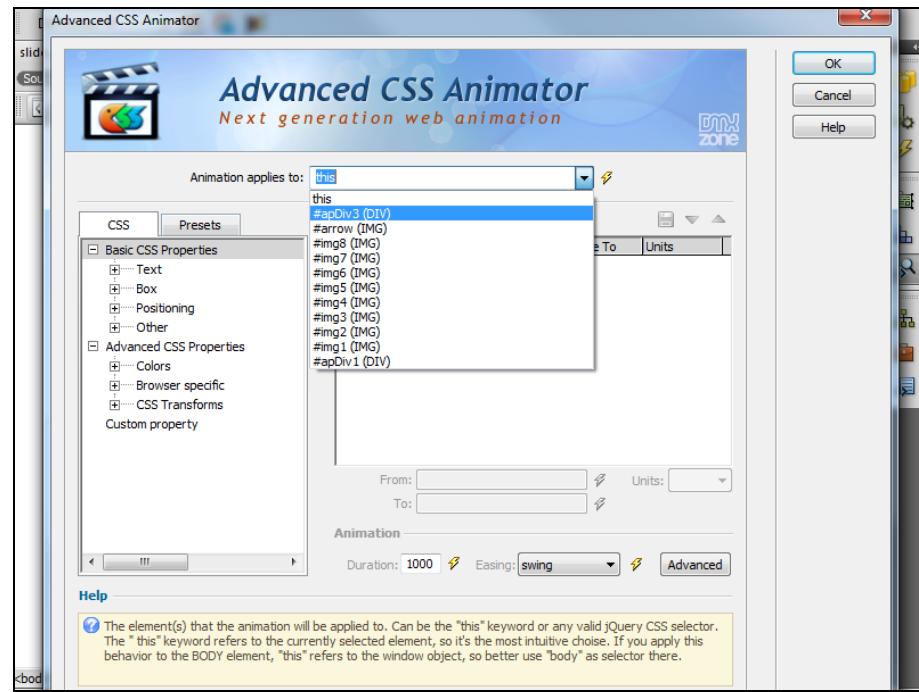
Select the div containing your menu items and click on the **Tag Inspector** Icon



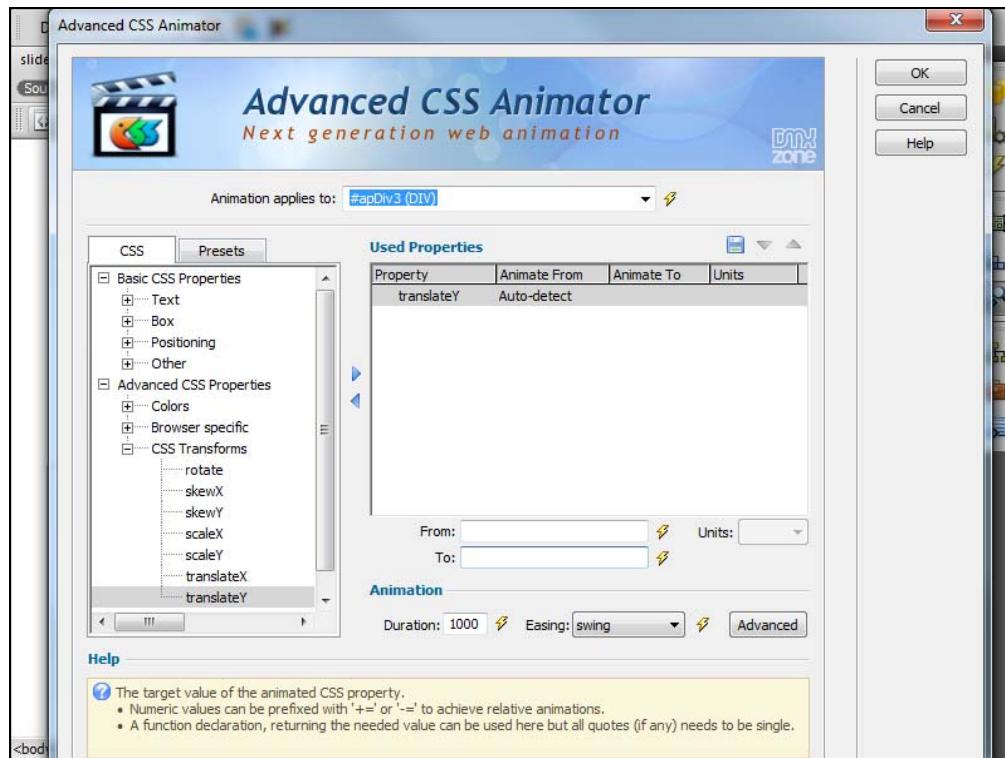
13. Click the **Add behavior** button  and select the **Advanced CSS Animator** menu item



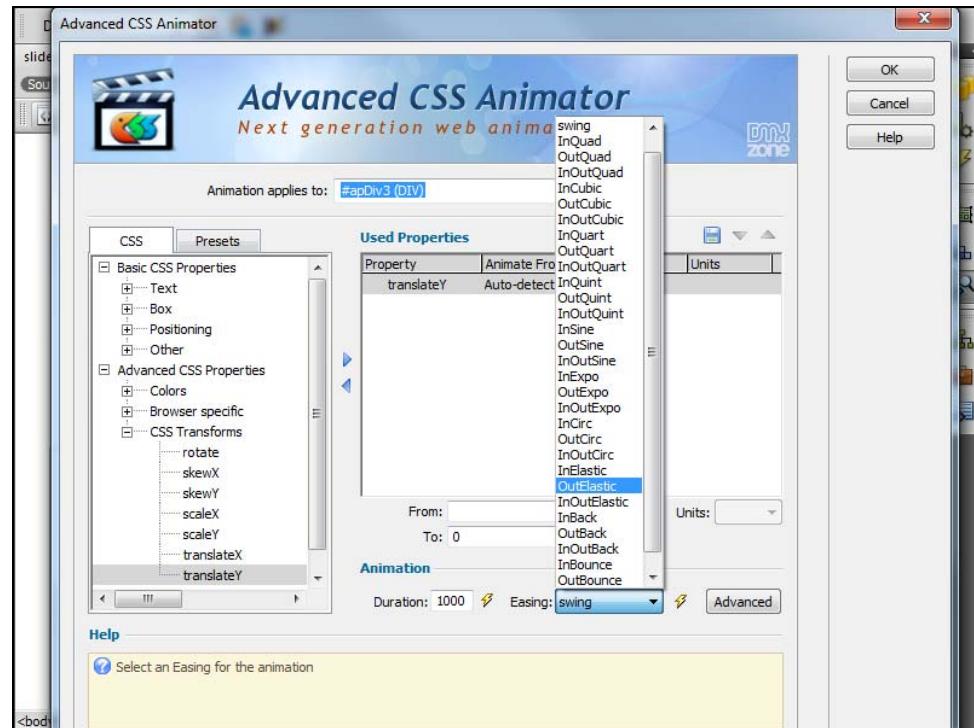
14. The Advanced CSS Animator dialog opens. Select the combo box and choose the ID of the div container **#apDiv3 (DIV)**



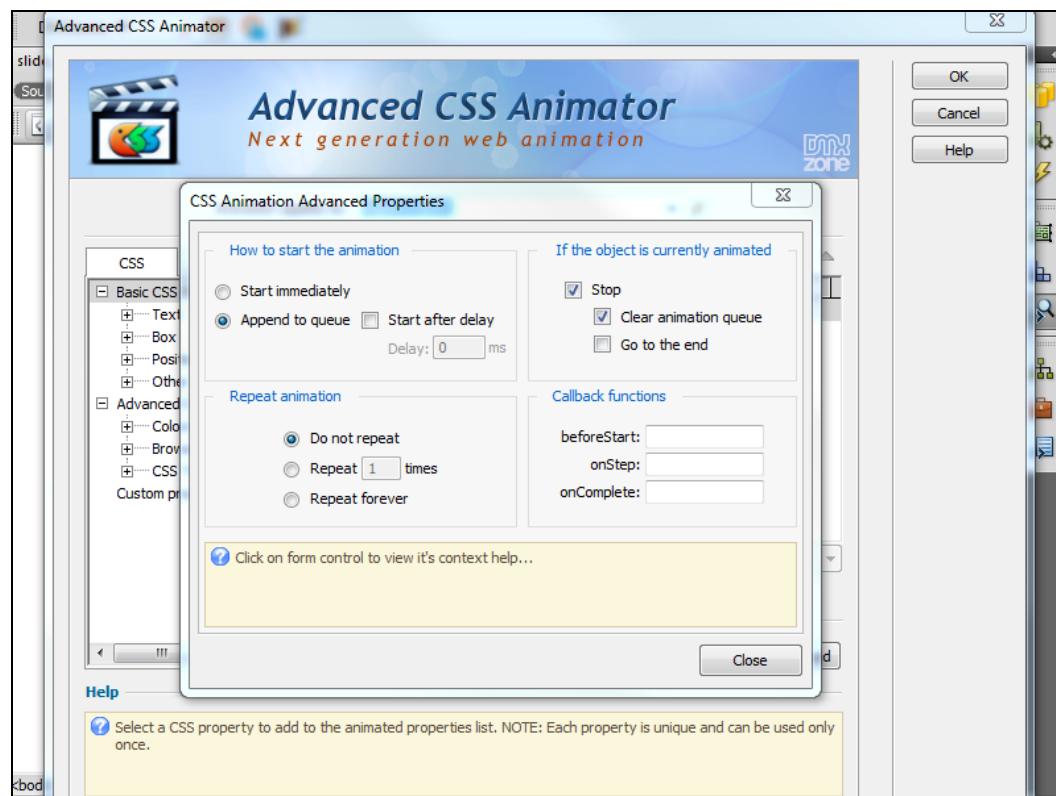
15. Add **TranslateY** effect, leave the **From:** field empty and enter **0** as target position for the translation



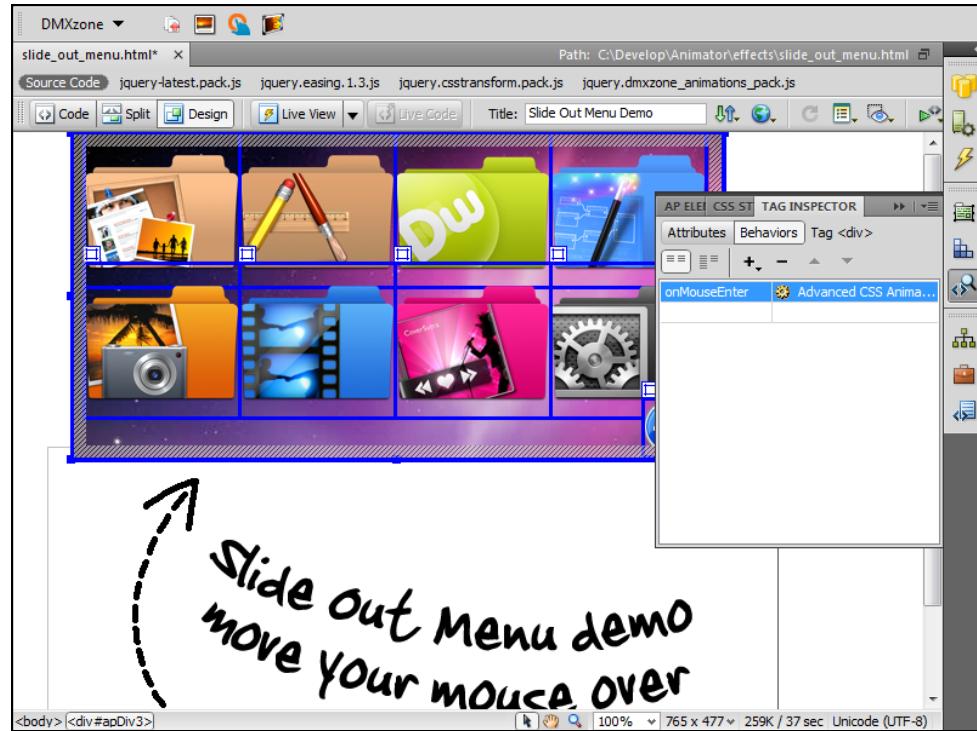
16. Select an Easing effect **OutElastic** and click the **Advanced** button



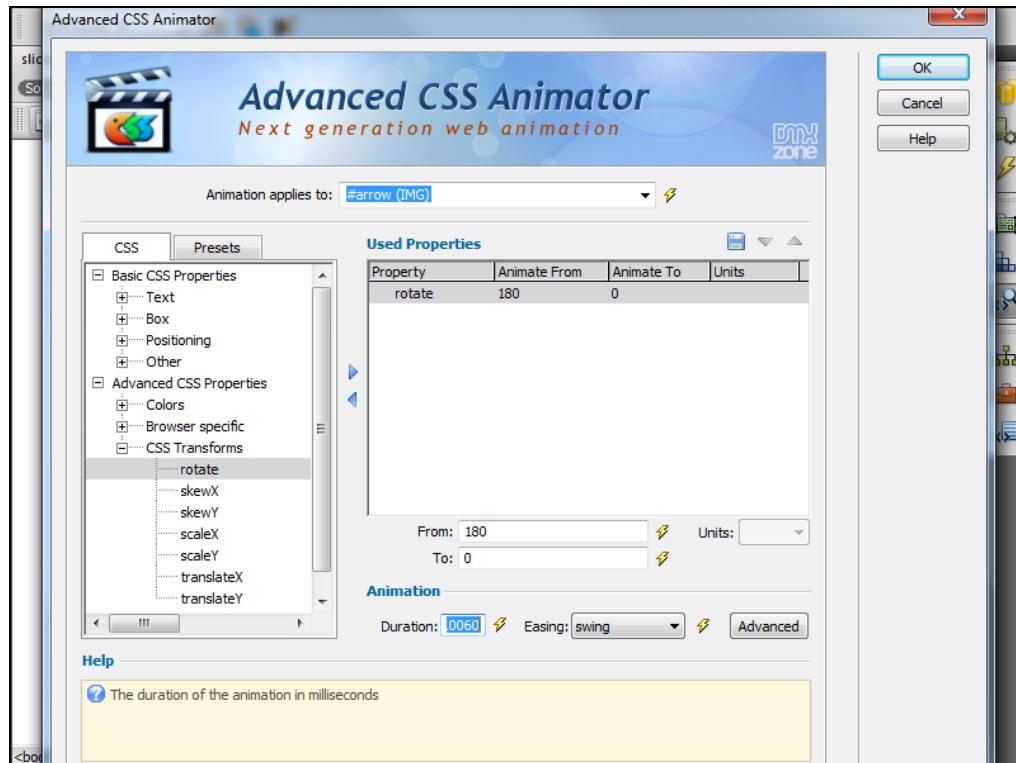
17. Use the following settings and click the **OK** button



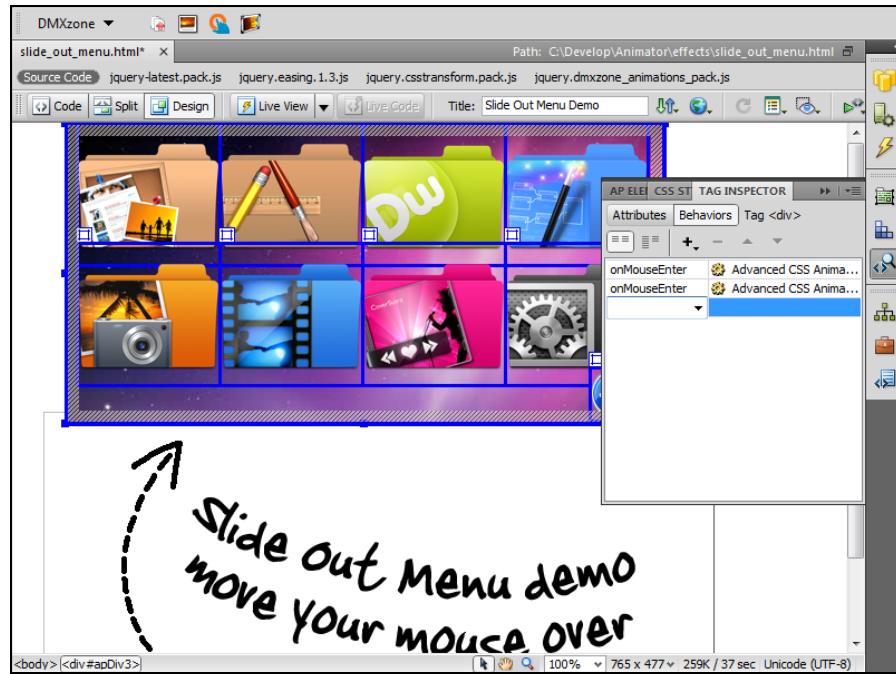
18. Make sure you select **onMouseEnter** event



19. We do the same with the arrow icon, but instead of translation select rotation effect. Here are the rotation settings:

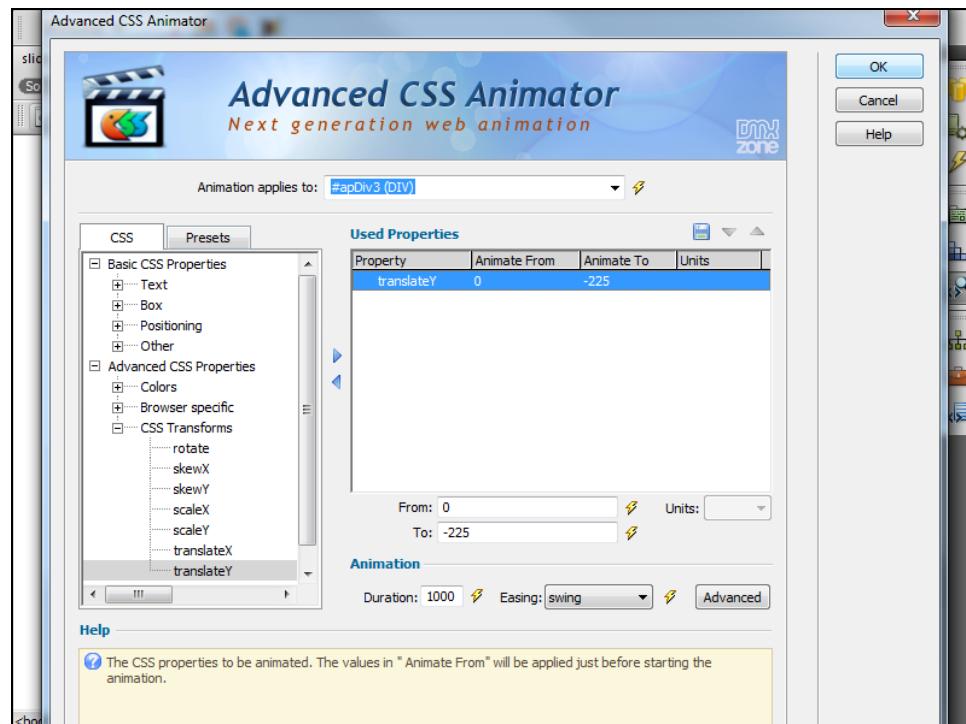


20. We've just set up the div to slide down and the Arrow Icon to rotate **onMouseEnter**. In order for menu to slide back up after the mouse leaves the div, we create two **onMouseLeave** events. The first one is for sliding div up, and the second one is for rotating the arrow icon in opposite direction

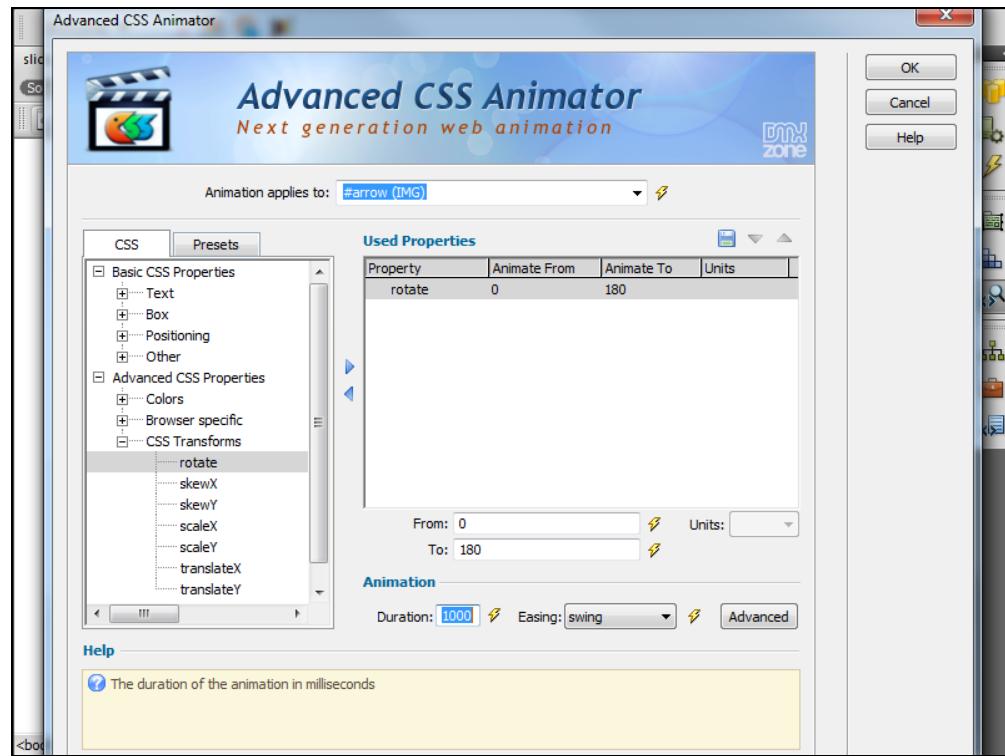


21. Here are the settings needed for sliding the div back up. When you're done click the **OK** button

OK

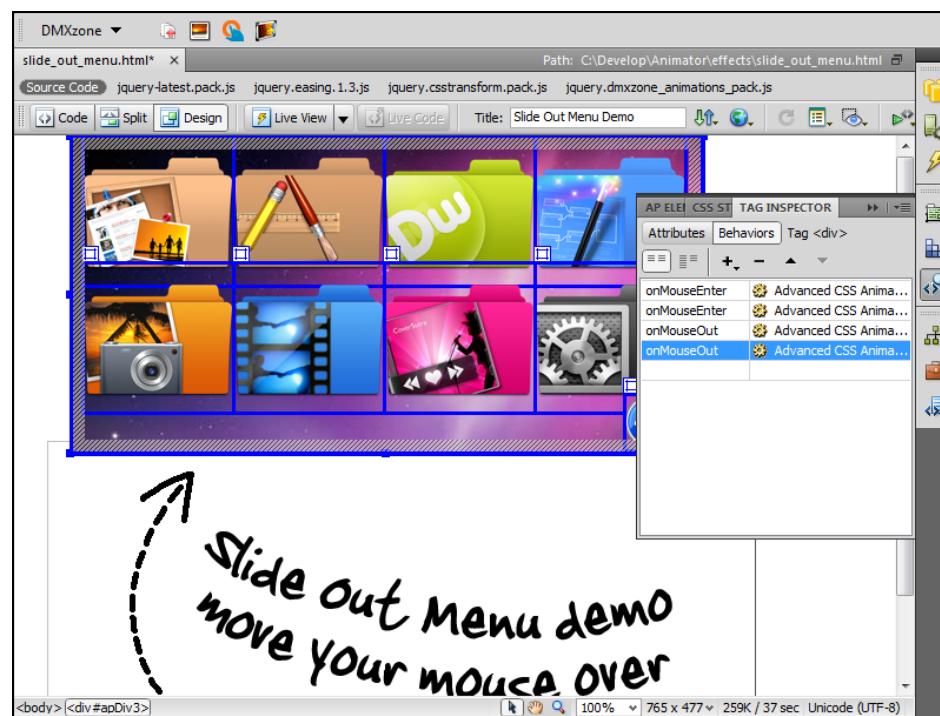


22. These are the settings needed for rotating the arrow



23. Make sure to select **onMouseOut** event for both of these animations

onMouseOut
onMouseOut

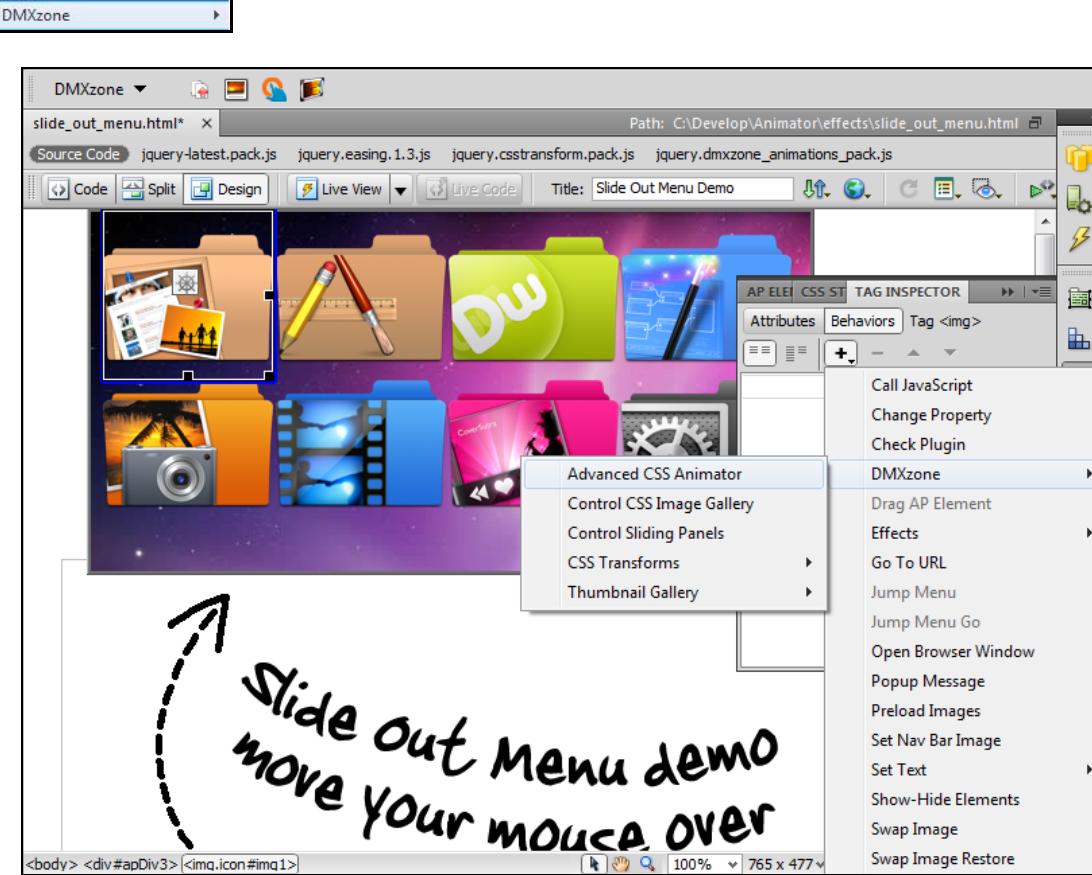


24. We are now going to set up the menu items to zoom in **onMouseOver** and zoom out to normal size

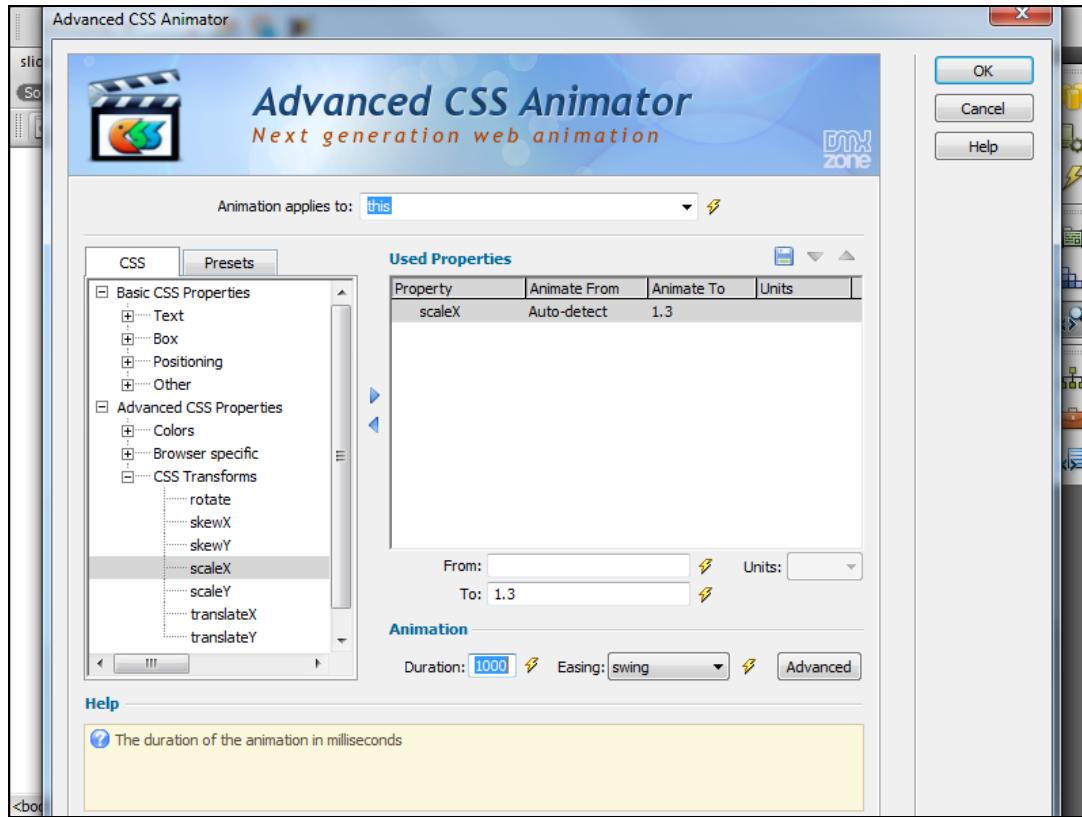


onMouseOut. Click on the first icon

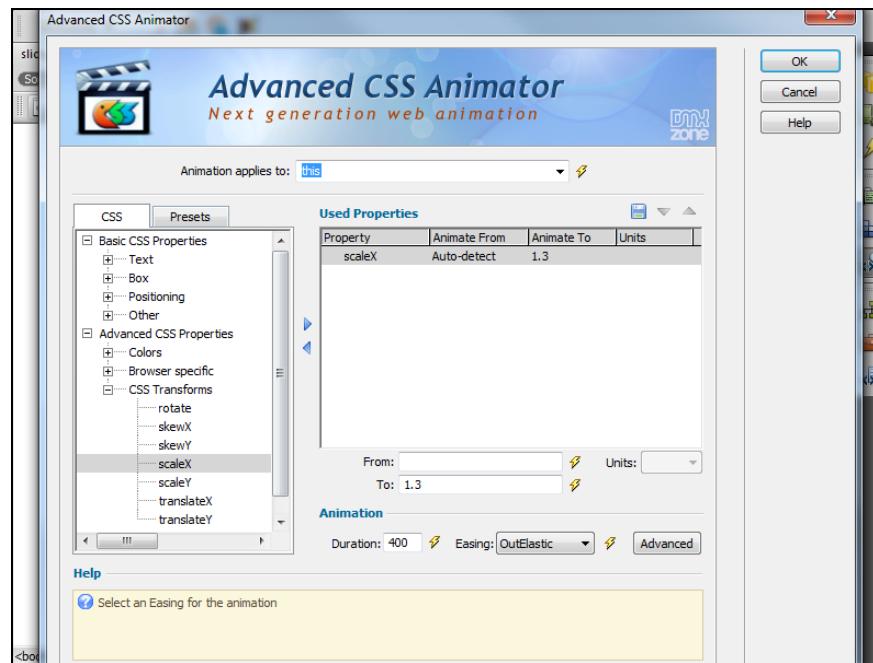
25. Click on the **Tag Inspector** icon , the **Add behavior** button  and select the **DMXzone menu** item



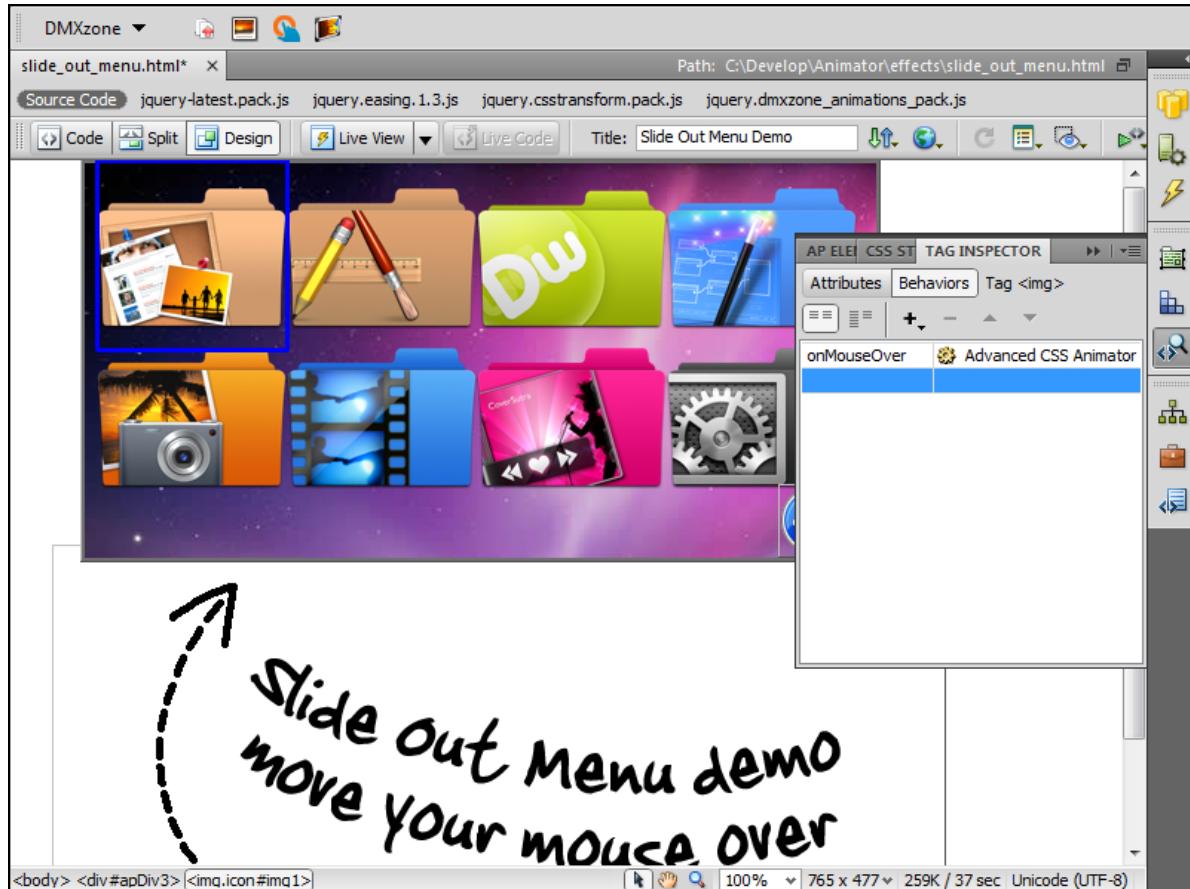
26. Add the **ScaleX** effect , leave the **From:** field blank, enter 1.3 in **To:** field and 400 as duration.



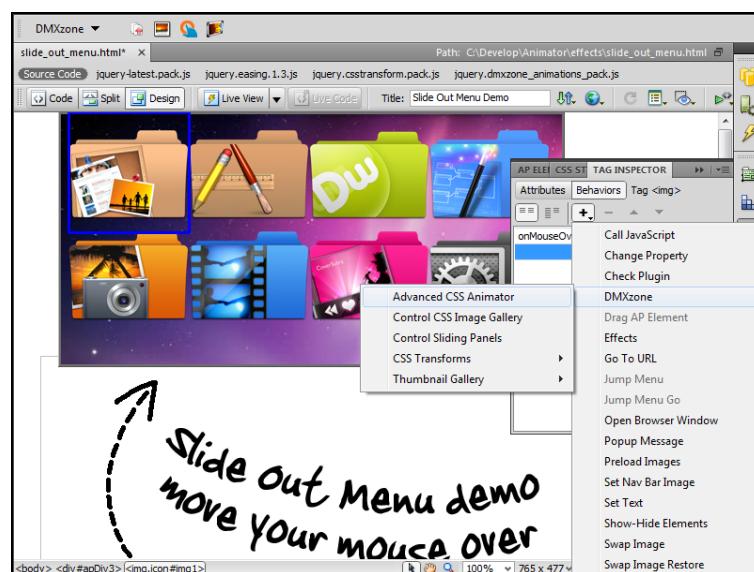
27. Select the **OutElastic** easing  and add **ScaleY** effect using the same settings.



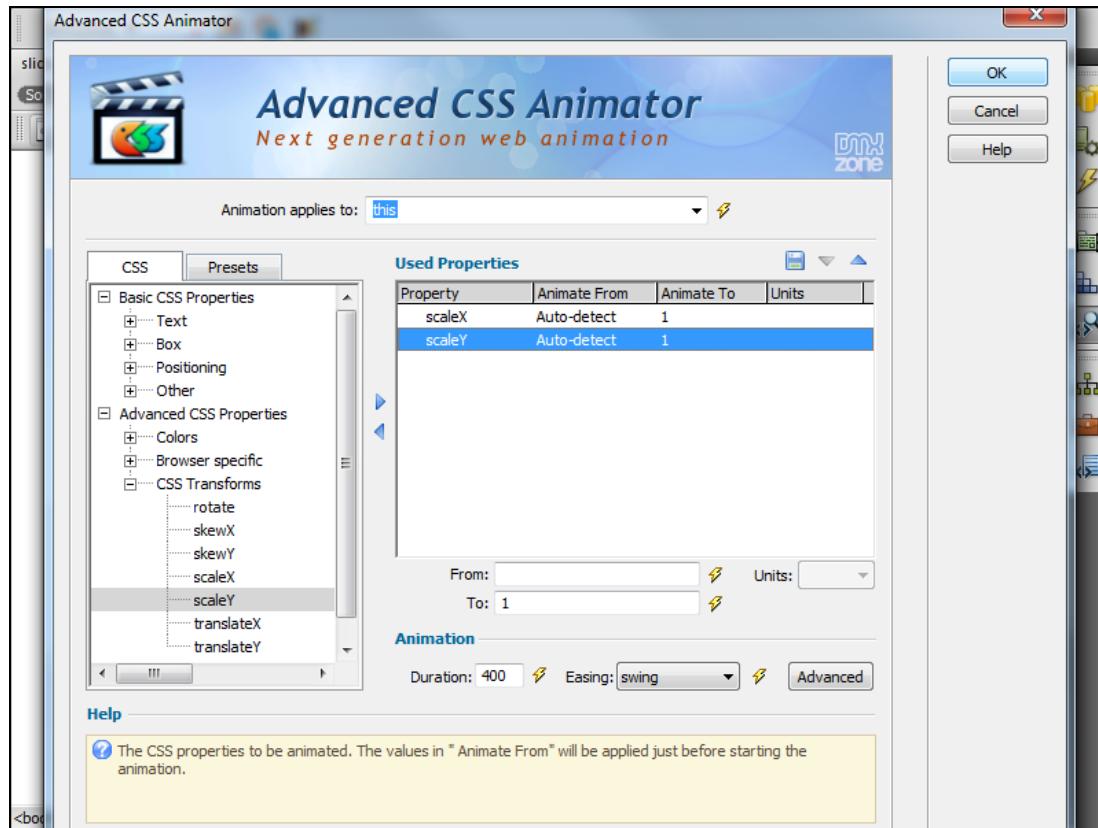
28. Make sure to select **onMouseOver** event



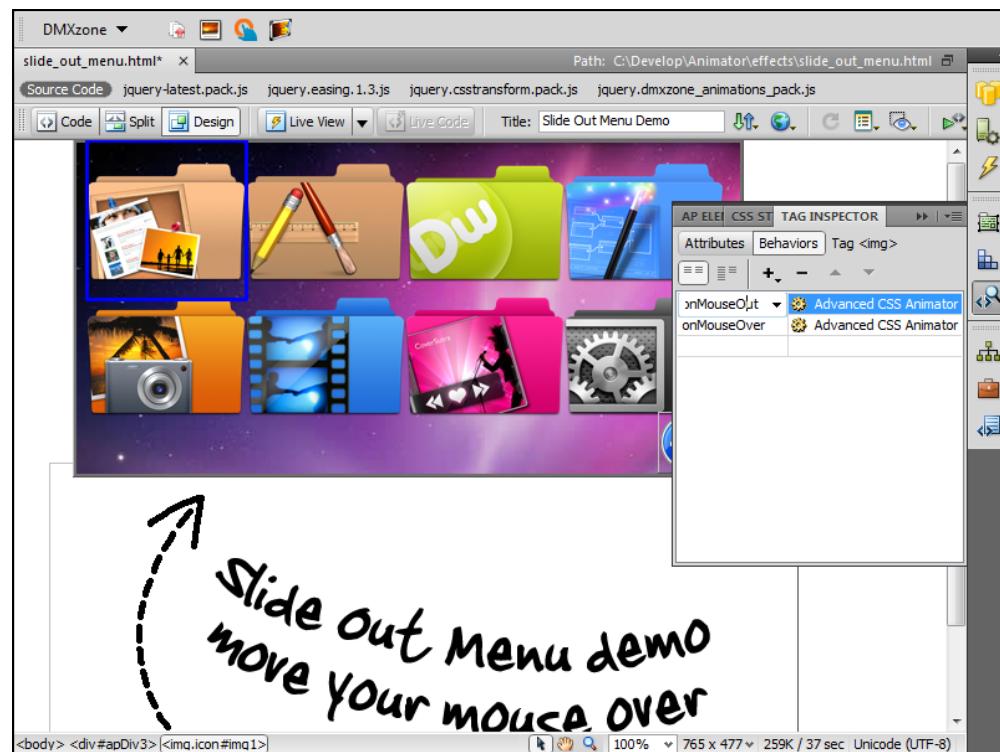
29. We are now going to add an **onMouseOut** event which will zoom the menu items to their default size.
Click the **Add behavior** button and select the **Advanced CSS Animator** menu item



30. Add **ScaleX** and **ScaleY** effects with the settings shown on the screenshot



31. You can now save your work and preview the result in your browser. That's how you can create a Slide Out Menu with Zoom in Menu Items



Reference: Basic CSS properties

Text

Font-size - Determines the font size of the element. The font size can be set in several ways. A collection of constants (xx-small, x-small, small, medium, large, x-large, xx-large) defines what are known as absolute sizes. In truth, these are absolute as far as a single browser in a single operating system goes because the reference point for these sizes varies with browser and operating system (analogous to the old HTML font sizes of 1 through 7).

Line-height - Sets the height of the inline box (the box holding one physical line of content). Under normal circumstances, the line-height of the tallest font in a line of text or the tallest object governs the line height for that content line. Mainstream browsers have come a long way since the Version 4 wrinkles that frequently made a visual mess out of mixed font sizes and line heights in the same block-level element.

Text-indent - Sets the size of indenting of the first line of a block of inline text (such as a p element). Only the first line is affected by this setting. A negative value can be used to outdent the first line, but be sure the text does not run beyond the left edge of the browser window or frame.

Box

Border-width - This is a shortcut attribute that lets you set multiple border edges to the same or different widths. You may supply one to four space-delimited width length values. The number of values determines which sides receive the assigned widths.

Border-top-width, **Border-right-width**, **Border-bottom-width**, **Border-left-width** - Each attribute sets the width of a single border edge of an element. Note that Navigator 4's initial value is zero, which means that you must set the width for all border attribute settings if you expect to see the border in that browser. See also the border-width attribute for setting the width of multiple edges in one statement.

Width - Sets the width of a block-level, replaced, and positioned element's content width (exclusive of borders, padding, and margins).

IE for Windows counts left and right margins, padding, and borders when calculating the width of an element until you reach IE 6 in standards compatibility mode. When observing the CSS standards, the width applies to only the content portion of an element, irrespective of borders, padding, or margins. This change may have significant impact on legacy code that you are bringing up to W3C compatibility.

Height - Sets the height of a block-level, replaced, and positioned element's content height (exclusive of borders, padding, and margins).

IE for Windows counts top and bottom margins, padding, and borders when calculating the height of an element until you reach IE 6 in standards-compatibility mode. When observing the CSS standards, the height applies to only the content portion of an element, irrespective of borders, padding, or margins. This change may have significant impact on legacy code that you are bringing up to W3C compatibility.

Padding, Padding-top, Padding-right, Padding-bottom, Padding-left – This is a shortcut attribute that can set the padding widths of up to four edges of an element with one statement. Padding is space that extends around the content box of an element up to but not including any border that may be specified for the element. Padding picks up the background image or color of its element. As you add padding to an element, you increase the size of the visible rectangle of the element without affecting the content block size. You may supply one to four space-delimited padding values. The number of values determines which sides receive the assigned padding.

Min-width/Max-width – These attributes let you establish a minimum and/or maximum width for an element. You can bracket the permissible width of an element regardless of the width caused by the natural flow of the content within a parent container.

Min-height/Max-height - These attributes let you establish a maximum and/or minimum height for an element. You can bracket the permissible height of an element regardless of the height caused by the natural flow of the content.

When you set the max-height attribute of an element that has content that may extend beyond that maximum, you should also set the overflow style attribute to hidden so that excess content is cropped. Failure to do so causes the overflowing content to bleed into the succeeding elements' content. Netscape 6 supports both attributes for block-level and positioned elements.

Internet Explorer 6 for Windows supports only the min-height attribute, and is limited to td, th, and tr elements inside a table with its table-layout style attribute set to fixed. This conflicts with the CSS2 specification, which explicitly excludes table-related elements from being influenced by these attributes. As such, min-height settings you make for td, th, or tr elements in IE 6 do not affect Netscape 6. IE 5 for Macintosh supports neither attribute.

Positioning

Left – For positionable elements, defines the offset position of the left edge of an element's box (content plus left padding, border, and/or margin) relative to the left edge of the next outermost block content container. When the element is relative-positioned, the offset is based on the left edge of the inline location of where the element would normally appear in the content.

Top – For positioned elements, this defines the position of the top edge of an element box (content plus top padding, border, and/or margin) relative to the top edge of the next outermost block content container.

Right – For positionable elements, this defines the position of the right edge of an element box (content plus padding, border, and/or margin) relative to the right edge of the next outermost block content container. When the element is relative-positioned, the offset is based on the right edge of the inline location of where the element would normally appear in the content.

Bottom – The CSS specification calls for this attribute to define the position of the bottom edge of a positioned element's content (exclusive of borders and margins) relative to the bottom edge of the next outermost block content container. Of mainstream browsers, only IE 5 for Macintosh behaves this way.

IE for Windows and Netscape 6 do something unexpected when the positioned element uses the root positioning context: instead of using the bottom of the document as the comparative edge, these browsers use the bottom of the browser window space (the viewport in CSS terminology).

As a result, the precise bottom position of the element varies with the user's browser window size. This discrepancy makes it more practical to use the bottom property for a positioned element nested inside

another positioned element. When the element is relative-positioned, the offset is based on the bottom edge of the inline location of where the element would normally appear in the content.

Margin, Margin-top, Margin-right, Margin-bottom, Margin-left – All four attributes set the width of a single margin edge of an element. A margin is space that extends beyond the element's border and is not calculated as part of the element's width or height.

Other

Opacity – The opacity property defines how transparent the object on the screen should be to objects below it.

Z-index - For a positioned element, this sets the stacking order relative to other elements within the same parent container.

Reference: Advanced CSS Properties

Colors

Color – Defines the foreground text color of the element. For some graphically oriented elements, such as form controls, the color attribute may also be applied to element edges or other features. Such extracurricular behavior is browser-specific and may not be the same across browsers.

Background-color – Sets the background color for the element. Although it may appear as though a nested element's background-color attribute is inherited, in truth the initial value is transparent, which lets the next-outermost colored element show through whitespace of the current element.

Border-color, Border-top-color, Border-right-color, Border-bottom-color,

Border-left-color – This is a shortcut attribute that lets you set multiple border edges to the same or different colors. Navigator 4 allows only a single value, which applies to all four edges. For Internet Explorer and Netscape 6, you may supply one to four space-delimited color values. The number of values determines which sides receive the assigned colors.

Outline-color – Controls the color of an outline around an element.

Browser Specific

BorderRadius – The border-radius property will round all four corners of the element.

Zoom - Controls the magnification of rendered content. This is particularly useful for output that might be displayed on monitors with very high pixel density.

CSS Transforms

Rotate – Specifies a 2D rotation by the angle specified in the parameter about the origin of the element, as defined by the transform-origin property.

SkewX - Specifies a skew transformation along the X axis by the given angle.

SkewY – Specifies a skew transformation along the Y axis by the given angle.

ScaleX – Specifies a scale operation using the [sx,1] scaling vector, where sx is given as the parameter.

ScaleY – Specifies a scale operation using the [1,sy] scaling vector, where sy is given as the parameter.

TranslateX – Specifies a translation by the given amount in the X direction.

TranslateY – Specifies a translation by the given amount in the Y direction.

Video: Creating a Popup Effect on Mouse Over



With this [movie](#) will show you how to create a great popup effect on mouse over using the Advanced CSS Animator.

Video: Creating Slide Out Menu



With this [movie](#) we will show you how to create Slide Out menu with Zoom In menu items.

Tutorial: Getting Started with the Advanced CSS Animator

In this [article](#), Nancy will introduce you to the DMXzone's latest extension, the [Advanced CSS Animator](#). She will introduce you to the extension and take you through 3 very simple examples of how you can use it to create some wonderful CSS Animation. And, unlike CSS animation that only works in a few browsers, the Advanced CSS Animator has seamlessly integrated jQuery functions so they work in all major browsers.