

Actions Tech. Note 12

Draft 1.0

Adding CSS Style Definitions

Freeway 3.1 introduced a new method `fwAddCSSStyles` to the class `FWTag` which will find or add a CSS style enclosure of the form:

```
<style type="text/css">
<!--
-->
</style>
```

It is suggested that if you wish to define CSS styles within your action that you use this method as it will allow you to append to any existing CSS style definition rather than defining a new one each time that you want one. Additionally the enclosure automatically includes the commenting structure that will allow your style definitions to be ignored by early browsers.

The method `fwAddCSSStyles` is completely analogous to the method `fwAddJavaScript` that is used to find or add JavaScript enclosures.

If you want to define the style `myStyle` in your output

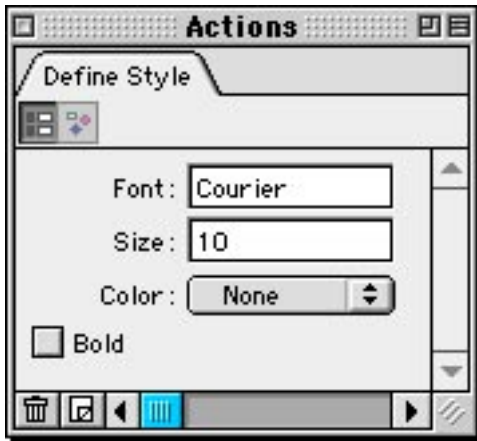
```
.myStyle { font-weight: bold; }
```

You could do so as follows:

```
function fwBeforeEndBody()
{
    // find the head tag
    var headTag = fwDocument.fwTags.fwFind("head");
    if (headTag)
    {
        // find or create a style enclosure
        var cssStyles = headTag.fwAddCSSStyles();

        // define the style
        cssStyles.fwAddRawln(".myStyle { font-weight: bold; }");
    }
}
```

Naturally it is possible to build up the style definition using parameters defined in the action. The following action will define a style that allows the user to text styling characteristics.



```

<page-action name="Define Style">
<action-text name="Font" default="Courier New,Courier">
<action-number name="Size" minimum=1 maximum=100 default=10>
<action-color name="Color">
<action-checkbox name="Bold" default="No">
<action-javascript>

function AddStyle(cssStyles, styleName)
{
    // get the text colour
    var textColor;
    if (fwParameters["Color"].fwValue != null &&
        fwParameters["Color"].fwValue.fwHasValue)
        textColor = fwParameters["Color"].fwValue.fwHex;

    // style name
    cssStyles.fwAddRaw(".",styleName," { ");

    //font-size
    cssStyles.fwAddRaw("font-size: ",fwParameters["Size"],"px; ");

    //font-family
    if (fwParameters["Font"] != "")
        cssStyles.fwAddRaw("font-family: ",fwParameters["Font"],"; ");

    //font-weight
    if (fwParameters["Bold"].fwBoolValue)
        cssStyles.fwAddRaw("font-weight: bold; ");

    //color
    if (textColor)
        cssStyles.fwAddRaw("color: #",textColor, "; ");

    // close the style
    cssStyles.fwAddRawln("}");
}

function fwBeforeEndBody()
{
    // find the head tag
    var headTag = fwDocument.fwTags.fwFind("head");
    if (headTag)
    {
        // find or create a style enclosure
        var cssStyles = headTag.fwAddCSSStyles();

        // define the style

```

```

        AddStyle(cssStyles, "myStyle");
    }
}

</action-javascript>
</page-action>

```

You should be careful to ensure that the style that are created by your action are unique and do not conflict with other styles that have been defined. If you are creation a style for a page action might like to consider using a CSS friendly version of the action name as the style name. If you are creating style that are specific to a particular item you can use the item name as a part of your style name - all item names are guaranteed to be unique (within a page) and are valid for use as CSS styles (i.e. they contain no invalid characters). To ensure that the name is unique across actions you might also consider appending a CSS friendly version of the action name to this to make a unique style name. The following code is taken from a "Ticker Tape" action and will create a unique style based on the name of the item that it is applied to and the action name "ticker".

```

// add a style whose name is "ticker"+itemname
cssStyles.fwAddRaw(".ticker",fwItem," { ");

// add the styles
if (textColor)
    cssStyles.fwAddRaw("color: #",fwParameters["text color"].fwValue.fwHex, "; ");
if (boxColor)
    cssStyles.fwAddRaw("background-color: #",fwParameters["box color"].fwValue.fwHex, "; ");

// close the style
cssStyles.fwAddRawln("}");

```