

Modifying an existing form with CSS Form Builder

One of the most useful features of CSS Form Builder is the ability to inspect existing, or custom-made forms to allow you to modify those forms in CSS Form Builder. This feature allows you to take most online forms that you may have created - either manually or using a WebAssist Dreamweaver extension, and enhance the design or apply advanced validation.

This How to shows you the necessary steps for modifying an existing form with CSS Form Builder and also explains the requirements, and limitations of this feature.

What Do You Need to Start?

- Dreamweaver CS3 or CS4
- CSS Form Builder
- Existing form to be modified

Preparing your form

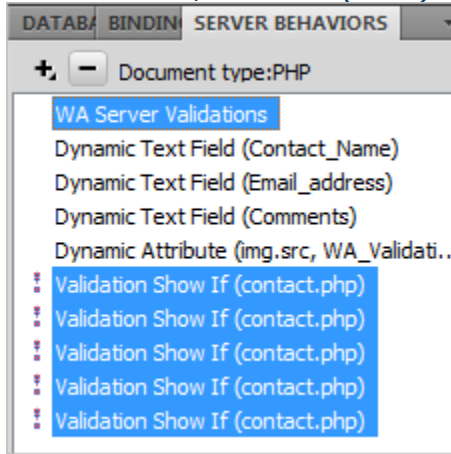
CSS Form Builder will review your form, and display the various form elements within the Form Builder interface. However, it does not have the ability to inspect any validation, or validation show if messages that may already exist on your form. Instead, after customizing your form in CSS Form Builder, all form fields and validation will be replaced. For this reason, the following steps should be followed to clean up your form before entering CSS Form Builder.

Remove Validations

Any preexisting validations should be removed from your form.

1. From the Dreamweaver Server Behaviors panel, select all validations and Validation Show If behaviors.

To save time, use *Ctrl (Cmd) + Click* to select multiple options.



2. Choose the **Delete (-)** option to remove these validations.

Remove any Spry validations

If your page has Spry validation applied you will also want to remove that code from the bottom of your page.

3. Select the **Code** button to enter Code view in Dreamweaver.
4. Scroll to the bottom of your page.

In most cases, Spry validation is added at the end of your page.

5. Highlight all the Spry validations.

```
151 <script type="text/javascript">
152 <!--
153 var sprytextfield1 = new Spry.Widget.ValidationTextField("spr
154 var sprytextfield2 = new Spry.Widget.ValidationTextField("spr
155 var sprytextarea1 = new Spry.Widget.ValidationTextarea("spryt
156 var sprytextfield3 = new Spry.Widget.ValidationTextField("spr
157   $WAGLOBAL_Captcha_Characters; ?>, maxChars:<?php echo $WAGLOBE
158 //-->
159 </script>
```

6. Press **Delete** on your keyboard.

Note: Removing all these validations is not always necessary to ensure your form will work. However, if any of this code is left, it could cause your form to not operate properly.

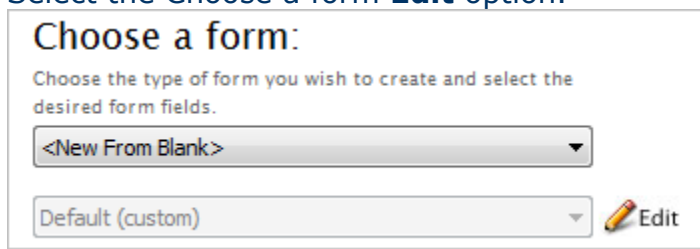
Customize your form elements

Once you have completely removed all validations, you can start using CSS Form Builder to modify your form.

1. Place your cursor inside the form on your page.
2. From the Insert menu, choose **WebAssist > CSS Form Builder > Insert Form**.

When the CSS Form Builder interface opens, the preview displays a page that contains form fields that are named as they are on your form.

3. Select the Choose a form **Edit** option.



CSS Form Builder will display one Fieldset with one group inside of it. All the form elements on your existing page are added to this one group. The next step is to tell CSS Form Builder, the type of form element, and what type of validation you want each element to have.

4. Select **Group 1** from the tree, and choose **Edit**.
5. In the Edit Group Contents interface, select the first field.
6. In the details below, specify the necessary details for the selected field.

Note: CSS Form Builder will be able to read certain types of form elements such as text or text areas. For fields that should be Captcha, Security Question, or something more unique, make sure to choose the appropriate option from the Type menu.

7. Choose the Type of validation you would like for the selected field.
8. Continue defining each form element, until all the form fields on your page are as desired.
9. Click **Finish** when you are done.

Customize your form design

With your form configured properly, you are ready to change the design.

1. From the primary CSS Form Builder interface, choose a preferred design from the Design options.
Although not required, choosing a starting design will save you a lot of time customizing how each form element should look.
2. Choose **Edit** to further modify the design for each aspect of your form.

For more details on customizing the form design, refer to the [Getting Started Guide](#).

3. When you are done customizing your form, click **Finish**.

CSS Form Builder will replace the form on your current page with the new form you have configured.

Note: If the contents of this form are to be inserted into a database, any new fields that you have added will need to have the Insert behavior attached to those new fields.