


Forms

Forms are an important part of websites because they allow you to ask questions of your users. A simple *mailto* email submission does allow them to contact you, but doesn't give you any control on what information you receive from them. We see these most often as Contact forms (collecting contact info: name, email, address, phone number, etc.), but they can also be used for questionnaires, voting, etc.

We need to look at the form field setup (the text areas, checkboxes, etc.) and how to make the forms function (actually send you the form content). That functionality is controlled by scripts on the web server, so it varies from web host to web host (and is not a function of XHTML).


Text Fields

To get started it's a good idea to switch to the **Forms tab** of the Insert Bar. That way we'll have easy access to our insert buttons.

Our first step is to insert the <form> tags using the **Insert Form button**  (leftmost button). All pieces of the form must sit within the <form> tags for them to function. In Dreamweaver you will see a red dashed rectangle that we need to work within.


Now that we have the <form> tags in place, make sure that your cursor is clicked in that area and insert a table (**Insert menu > Table**). This isn't necessary for the form to work, but it will make it look much better. Let's set it to have *10 rows* and *2 columns* to start; set **Table Width** to *600 pixels* and **Border Thickness, Cell Padding and Cell Spacing** each to *0 pixels*. Also make sure that the **Header** is set to *None*. This gives us aligned cells in which to put our form fields.

Let's insert a text field now:

- Click in the top, right cell of your table, then click on the **Insert Text Field** button  in the Insert Bar
- This brings up the **Input Tag Accessibility Attributes** dialog where we enter info about this form field
- Let's give it the **Label** of *Name* and select *Attach label tag using 'for' attribute*. The label displays as text, but is a specific tag associated with your form field. There is additional functionality in some browsers by using the <label> tag, but more importantly it is more accessible for screen readers. We now have the label and text field showing up in our table.
- Click on the **TextField** and in the Properties Inspector give it a name; let's use *Name* (no spaces or special characters).
- Now click on the label, then click on the <label> tag in the bottom border of the window (selecting the entire tag). Now cut it (⌘-X) and paste it (⌘-V) into the table cell to the left (or you can drag-and-drop it there). This is going to allow us to align our labels and form fields.
- The only disadvantage to moving our <label> to the other cell is that it breaks the association between the label and the text field. While you have your <label> highlighted, switch to Split view and change the line of code

```
from <label for="label">Name</label>
```

```
to <label for="Name">Name</label>
```

Now they are associated with each other again. (You can also make this adjustment in the **Quick Tag Editor** by clicking -T while you have the <label> tag selected.)

- Repeat the steps above, this time inserting a text field one row down with the **Label** of *Email Address* and the **TextField** named *Email*

Now add a Text Area field by clicking on the **Insert Text Area** button  (have your cursor in the 3rd row down, 2nd column)

- give this a **Label** of *Comments* and **TextField** of *Comments* (also move the <label> and check its association to the TextField)

You can select your various text fields and check out the options in the Properties Inspector. You'll notice that the **Type** being set to **Single Line** and **Multi Line** is what determines if it's a Text Field or a Text Area.

Char Width sets the width of the text input area

Max Chars limits the number of characters a user can input in the text field

Input Val puts text in the text field that will display before the user inputs their info

Num Lines sets the number of text rows that will display in a Text Area

Wrap determines if text in a Text Area wraps around or just keeps pushing to the right

Check Boxes and Radio Buttons

Check boxes can be used to select multiple options while radio buttons only allow users to select one answer.

Lets start by inserting a checkbox (**Insert Checkbox** button  in the Insert Bar)


- set the **Label** to *Send me the newsletter* and select the **Style** of *Wrap with label tag*
- also rename the **Checkbox** to *Newsletter* and set **Checked Value** to *yes* in the Properties Inspector
- I would leave **Initial State** set to *Unchecked*. It's usually considered bad form to autocheck boxes for your users (you should default to not signing them up).

Now lets insert a group of radio buttons (**Insert Radio Group** button  in the Insert Bar)

- set the **Name** to *Referrals* and then you can enter your list of ways people found your website. The **Label** is what shows up in the browser and the **Value** is what is sent to the server (often identical to your label, or an abbreviated version)
- click on each label and value to set the list as follows (use the plus button to add more fields as necessary):

| | |
|---------------|-------------|
| Newspaper Ad | newspaper |
| Radio Ad | radio |
| Television Ad | television |
| Word of Mouth | wordofmouth |
- leave **Lay Out Using** set to *Line Breaks*
- this didn't give us a viewable label, so click in the cell to the left and type *How did you hear about us?*


Lists & Menus

Lists and Menus are typically used when there is a large number of items the users can select their answer from. They are pretty similar to the form fields we've already worked with. You insert one with the **Insert Lists/Menu** button 

- this brings up the standard **Input Tag Accessibility Attributes** dialog where you can enter your **Label** and set your **Style** to *Attach label tag using 'for' attribute*
- then you can name the **List/Menu** field to *FavoriteColor* in the Properties Inspector (don't forget to move your `<label>` to the left column and check its association to your List/Menu field)
- now you can click on the **List Values...** button in the Properties Inspector and create your list of **Labels** and **Values** (similar to the ones we just did for radio buttons). Lets set our list like this:

| | |
|--------|--------|
| Red | red |
| Blue | blue |
| Green | green |
| Yellow | yellow |
- now if you would like you can set the **Initially Selected** value in your list (I created an additional item in the list, *Select One*, to show when they first view the page)
- now insert a List to see the difference (it only affects how it views on the page) (Menu displays as a popup menu, List as a scrollable list)
- you can check *Allow Multiple* under **Selections** if you want to let them pick multiple answers. If you do, it's a good idea to tell them how to by typing *Hold Ctrl (Windows) or Cmd (Mac) and click to select multiple shapes*.

Reset and Submit Buttons

To finish our form we need to add two important buttons. We'll add a **Clear/Reset** button as a courtesy and a **Submit** button to send the data. To do this click on the **Insert Button** button  in the Insert Bar. We don't need an **ID** or **Label** and you can select *No Label Tag* as the **Style**. This gives you a Submit button by default, but you should change the **Button Name** to *reset* in the Properties Inspector and **Value** to *Clear Form*. We also need to change the **Action** to *Reset Form*.

Following that we should insert one more button (put one space between them), this time setting the **Button Name** to *submit*, **Value** to *Submit Form* and **Action** to *Submit Form*.

That's how simple it is to add the buttons. However, there is nothing telling the server where to send the form. To do that we select the `<form>` tag in the bottom border of the window and now in the Properties Inspector we see the form variables. This is where we change the **Action** to the specific line of code needed for your web host. It often looks something like this

```
http://www.yourdomain.com/cgi-bin/formmail.cgi
```

but varies from host to host. You'll also need to verify with your webhost whether to use *GET* or *POST* as the **Method**.

Styling Forms

Forms vary in appearance from browser to browser. However, you can control that to a great deal through the use of CSS. You usually do this through the use of Classes, though you can make adjustments to tags like `<select>`. Then make adjustments just as we did back in the tutorials on CSS.