Intro To Forms

The First Step

The first thing you must tell the computer is that you are starting a form, and what you want done with the form data. The command to alert the computer is:

```
<FORM METHOD="POST" ACTION="mailto:your email address">
```

Notice the command did three things:

1. It told the computer a FORM was starting.
2. It stated the METHOD of dealing with the form is to POST it.
3. And the data should be posted to your e-mail address through the "mailto:" ACTION.

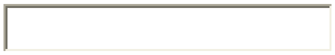
Remember you will need to put your e-mail address immediately after the "mailto:" without a space! This is where the results of the form will be sent.

That's nice and simple. Now that the computer knows a form has begun, it's looking for any one of a number of form styles to deal with. I will go over five here; TEXT, TEXT AREA, the RADIO BUTTON, the CHECKBOX, and the POP-UP BOX. These are by far the most used on the WWW. Also, I will go over the creation of buttons that send the form to the "mailto:" address or clears the form.

Boxes for Entering Text

The Text Box

This is a basic long box that allows for one line of text. When placed on a page, your reader will be able to write in information such as their name or their e-mail address. Here's what a text box looks like:



Go ahead. Write something in it. It works. You may have to use your mouse pointer and click on the box to activate it -- but it works. If this is your first time making forms, you might think that the box above is a .gif or a .jpeg. Not so. The box is placed on the page through an HTML command, not as an image. The command to place it on the page is this:

```
<INPUT TYPE="text" NAME="name" SIZE="30">
```

There are three parts to the command. Here's what they are and what they mean:

- **INPUT TYPE** tells the computer that a form item is going to be placed here. Remember above you placed the command to alert the computer that form items will be placed on this page? Well, here's your first form item. This form type is "text".
- **NAME** is the name you assign to the box. Remember that this is a form that will be sent to you through the mail. When you receive the mail, it won't be just like the page. Only the text will arrive, so you have to denote what each piece of text will be. When the mail arrives from this text box, it will say:

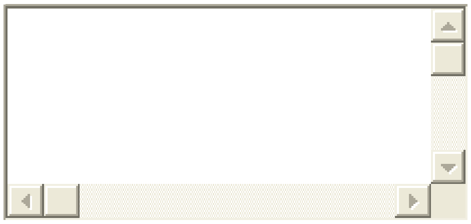
  name=(whatever is written in the box)

  That way you know this information was written in the box marked "name". Also, remember you don't have to call the box "name." Call it whatever you want. It will arrive to you with that name. If you're using the box to get the reader's name, call it "name." If you're using the box to get the reader's e-mail address, call it "e-mail." etc., etc.

- **SIZE** denotes how many characters long this box will be. Make it 60 or 100, if you'd like. I've just found 30 is usually a good size.

The Text Area Box

This is a larger box, like the one above, that allows your reader to write something. The difference between the Text Box (above) and the Text Area is that the Text Box only allows for one line. The Text Area, however, is much larger and will allow for as many words as you want.
Here's a Text Area Box:



Neat, huh? Again, go ahead and write in it. It'll work. You may have to click on the box to activate it. Here's the command that made it appear:

```
<TEXTAREA NAME="comment" ROWS=6 COLS=40>
</TEXTAREA>
```

Please note that the TEXTAREA requires a </TEXTAREA> command, whereas the TEXT BOX command above did not.

Just as before... here are its parts and what they mean:

- **TEXTAREA** (all one word): This yells to the form command above that here will sit another form item. This one will be a text area box.
- **NAME** is the same as before. The information the reader puts in this box will arrive in your e-mail box denoted by whatever name you use. In this case, what is written in this box will arrive in your e-mail box with the words, "comment=".
- **ROWS** tells the computer how many rows of text it will accept, and...
- **COLS** tells the computer how many characters will be in each row. This text box will accept 6 rows of text each being 40 characters long. Go ahead and make the box bigger or smaller. You're in charge here.

Input Buttons

The Radio Button

This is a neat little deal that places a circle on the page. That circle is active and a reader can use the mouse to click on it. When the radio button is chosen, it darkens. Here are three radio buttons:

○ -○ -○

Go ahead. Click on them. I know you're dying to. I have three of them there to prove a point. The point is that radio buttons are a one choice deal only. When you use radio buttons, only one can be checked. When another is checked, the first one gives up its selection. Go ahead -- try it.

Neat, huh? But why are they called **radio** buttons?! The reason is that they act as the radio buttons used in older car radios. When you pushed one, the dial moved. When you pushed another, the first choice was dropped and the dial moved again. You're probably too young to remember. It was back when AM only was a big selling radio, back before dirt. (Man, I'm old.)

Here's the command to place a radio button on your page.

```
<INPUT TYPE="radio" NAME="heading of button" VALUE="button name">
```

You see, the TYPE in the command is "radio." The command is long, but it's not that difficult to understand. Here are its four parts and what they mean:

- **INPUT**: This yells to the computer "HEY! Here's a form item to deal with!"
- **TYPE**: This tells the computer what type of form item it will be. In this case, it's a radio button.
- **NAME**: This names the category the button is in on your form page. Let's say you have six different choices under one heading. Like six ice cream flavors all under the heading, "Favorite Ice Cream." "Ice Cream" would be the category. It's the heading of the group of radio buttons.
- **VALUE** is the name assigned to the button. Like in the ice cream example above, you have six buttons each labeled with six different flavors. Well, you would give one the value of vanilla, one chocolate, etc., etc.

Why on earth would I want to label all those?

Remember that this is going to be sent to you through the mail. You will have to be able to read what the person chose. Say you have a guestbook with a section of radio buttons that ask which page the user is signing in from. Your **"NAME"** in the command might be "signing_in_from." Then each of the radio buttons is assigned the **"VALUE"** of each of your pages. Let's say a person chooses the radio button assigned to your home page. That button's VALUE might be, "home_page."

Thus, when the form arrived to you, the e-mail would read:

"signing_in_from_ home_page"

Pretty darn slick, huh?

The Checkbox

The checkbox is an exact clone of the radio button except for two distinct features:

- The item it places on the page is square and it is marked with a check when chosen.
- You can check as many as you'd like.

Here are a few checkboxes:

□ _□ _□ _□ _□

Go ahead. Click around a bit. You'll note that just one or every one can be chosen. This checkbox is basically a fancy radio button. Here's the command that placed the checkbox on the page:

```
<INPUT TYPE="checkbox" NAME="Signing_from" VALUE="Joes_page">
```

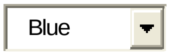
Each of the items mean the same as above so there's no need to go over them again. Please note, however, the TYPE is now "checkbox" instead of "radio."

Remember that when the text from a checkbox arrives at your e-mail box, more than one can show up. With radio buttons, only one item under the NAME heading will arrive. With checkboxes, every item can be checked, thus every item can arrive.

I like radio buttons much more than checkboxes. The reason is that the radio button forces a choice. Checkboxes invite people to just check everything every time. That can waste your time if you have to read through it all. I like to make a one-choice deal. It's easier on me and if people want to leave more info, there's always the TEXT AREA box for that purpose.

The Pop-Up Box

I LOVE these Pop-Up boxes, but I don't use them too often. I like to have all the items people can choose out in the open. The Pop-Up box, unless clicked on, only shows one item. But, this is your form and you can do anything you wish. Here's a Pop-Up box. You have to click on it to see all the choices. This one is for people to choose their favorite color:



Here are the commands that placed the Pop-Up box on the page:

```
<SELECT NAME="Favorite_Color" SIZE="1">
<OPTION SELECTED>Blue
<OPTION>Red
<OPTION>Yellow
<OPTION>Green
<OPTION>Black
<OPTION>Orange
<OPTION>Purple
</SELECT>
```

Although this looks a little bit more involved, it really isn't. It's the same thing again and again. Here are the parts and what they mean:

- **SELECT** tells the computer another form is going here. This time it's a SELECT or Pop-Up form.
- **NAME**: Same as above. This is the heading of the form item. It denotes how the results of the reader will arrive at your e-mail box. In this case it will say; "Favorite_Color=" and then the reader's choice.
- **SIZE** denotes the size of the box. Here, 1 means one line or item is shown. Try putting two there if you'd like to see what it does. I prefer just one. More than one item tends to defeat the purpose of the Pop-Up Box.
- **OPTION SELECTED** denotes which option will appear in the box. Note on the page that "Blue" is visible.

- **OPTION** denotes another choice that will be visible when you click on the item.
- **/SELECT** finishes off the entire deal.

People write and ask if it's possible to make the above pop-up box a series of links for people to choose and then go to. With HTML alone, it isn't possible. Using a Java Applet along with the code -- it is. Go to the Script Tips page and read the series starting with number 38.

Send and Reset Buttons

Well, now that the you have placed all the form items you want on your page, you need a way to have the results sent to your e-mail box (or wherever you said this would go in the original form statement).

This may be the easiest of all the items I've shown on this page. Here are the buttons:

Please do not use the buttons. They are active and will attempt to work -- however, I have put in a false e-mail address so I don't fill my own box with e-mail from this tutorial. You're just going to get scolded!

And here are the commands to put the buttons on the page:

```
<INPUT TYPE="submit">
<INPUT TYPE="reset">
```

Easy, huh? Now when you click on the buttons, the form will enact the ACTION you noted in the original FORM command. Here it would have been mailed to my e-mail box (or not, in this case).

Finally!

Make sure you end your page with this:

</FORM>

That's a beginning on forms, but there is so much more you can do than what I have here. There are forms you can connect to other CGIs (Common Gateway Interfaces), databases, or other data collection devises. All I wanted to do here was give you a very basic, very easy form for you to use on your Web pages.