
Form Intro:

The starting and ending tags are `<form>` and `</form>`. The basic form element is `<input type="" >`
`<html>`
`<body>`
`<form method="post" action="ascript.url">`

Please enter your first name

```
<input type="text" size="10" name="firstname">
```

```
<br>
```

Do you want to be on our mailing list?

```
Yes<input type="radio" name="mailing" value="1" checked>
```

```
No<input type="radio" name="mailing" value="0">
```

```
<br>
```

```
<input type="submit" value="Press Here to Submit">
```

```
<input type="reset" value="Press Here to Reset the Form">
```

```
</form>
```

```
</body>
```

```
</html>
```

There are two ways to send the information gathered in a form to the server: METHOD=GET and METHOD=POST. If you want to make a link to a CGI script, use GET. If you expect there might be a large amount of data, use POST.

In the example above, the ACTION="script.url" tells the server which CGI script to activate when it gets the form data back.

In an INPUT tag, NAME="username1" creates the variable "username1". The data collected will be associated with it when it is collected by the server and passed over to a database program.

Other Inputs:

Checkbox

```
<input type="checkbox" name="OS" value="win">
```

Creating a big box for them to type in:

```
<textarea name="description" rows="4" cols="30">
```

Passwords

```
<input name="password" size="20">
```

Menus

```
<select name="states" size="2">
```

```
<option value="WA" selected>
```

```
<option value="OR">
```

```
<option value="ID">
```

```
<option value="CA">
```

```
</select>
```

Form Design Rules of Thumb (* most important for next 368 assignments)

***1) Position labels consistently.**

Group response options together and maintain spacing.

For example, make it easy to see which radio buttons are related to each question.

***2) Group types of questions or fields into logical groupings.**

Phone and fax would go together.

This adds clarity. For example, if you are asking questions about hardware and software, group all the hardware questions together.

After a space or horizontal rule, group all the software questions together, and so on.

***3) Put these items in a table layout if possible** so that all your logical layout will stay laid out properly, even if the window size is changed.

***4) Make certain zip code area look like zip code areas, and phone number area codes look right.**

Don't let the user decide whether to put in a hyphen. You put it in between the boxes, with just enough space in each box for the right number! Same for zip and state and visa card numbers. Also include `maxlength="n"` to make it impossible someone can type in too many characters.

***5) Forced-response** questions (with limited radio button answer options for instance) make it easy to analyze the data.

Open-ended items (such as text forms or textareas) are much **tougher to analyze** but will give the respondent a chance to mention something important that you had not thought of.

6) Use textareas rather than very long text boxes.

They appear fully in almost any window. Long text boxes may run out of the window!
