

A Simple Guide to Putting Evaluation Assessments Onto the Web

Mark J. Kittleson, Ph.D.¹

¹Southern Illinois University

Corresponding author: Mark J. Kittleson, Ph.D., Professor, Health Education, Southern Illinois University, MS 4632, Pulliam 307, Carbondale, IL 62901; fax: 618.453.1829; phone: 618.453.1841; e-mail:

KITTLE@SIU.EDU

Although professionally we are still in its infancy, the use of the Internet to collect data has great potential. With the continued escalation of postal rates, the increasing use of the Internet and email, along with the ease of formatting web-based assessments, there is no doubt that the Internet will serve as one of the primary vehicles to collect data in the very near future. This article is intended to provide the guidelines for novices in web page development on how to create a web-based evaluation assessment. This evaluation could be used in a variety of settings ranging from a classroom assessment, to obtaining opinions from a community, to procuring data from public health professionals.

Although it is true that there are certain professional programs that can make this task simple, most of these programs often have restrictions that may prevent a person from completing this task. For example, with FrontPage by Microsoft, the server that one is working with MUST use Microsoft's IEE server software. If that server is not using that software, FrontPage has difficulty working with other server software (such as Netscape). The following strategy will work with any type of server or its software.

Prior to starting, the following points need to be clarified:

1. One must have access to a world-wide web *server* (a computer that is linked to the world-wide web) to place a web page on.;
2. The following information can only be assured for those using a *windows based* computer. Those individuals using MACs may need to make modifications.
3. One must be able to use a word processing system that can show *hypertext* codes. In lieu of such a word processing system, one can use the Notepad or WordPad program in the Windows 95/98 system.
4. One must have *file transfer protocol* (ftp) capability to send your final file to your web server.

5. This survey can only be utilized with an *Internet browser*. The two most popular are *Netscape Navigator* and *Microsoft Internet Explorer*.

As a brief overview, in order for something to appear on the World-Wide Web, the computer must be connected to the Internet. This computer is oftentimes called a *server*. One connects to a server to access the World-Wide Web. In addition, if one has been given "space" on this server, he/she can distribute information so that others can view and/or download. Typically, people who have a computer in their office will be networked into a server (or one can call in with a phone line). In order to put something onto the server, one must transfer files electronically (called FTP).

The Start

In order to create a web-page evaluation assessment, a series of steps must be taken. As with all web pages, a special formatting code called *hypertext* must be used. Fortunately, both of the two web browsers above read the same codes. Figure 1 shows the hypertext of a 5 item evaluation tool. With this structure, one can literally copy and paste into a web page with very little editing required. Note that each line has a number...those numbers are not to be included in the actual hypertext coding (they are present to assist in the description of the various codes). To use this hypertext program, one merely needs to change the wording of a question or options and/or the number of options available for each question. Figure 2 shows how these hypertext codes appear in a web browser.

There are four basic types of assessment items that one can ask: the *Radio*, *Checkbox*, *Test*, or *Textarea*. Each one has a specific formula that needs to be included within the web page.

The Radio Item

The *Radio* is the type of question that seeks only ONE response. One can liken this to a multiple-choice or true/false question. The following are two examples of a *Radio* type item:

Multiple Choice:

What is the capital of Illinois?

- F Champaign
- F Chicago
- F Lincolnville
- F Peoria
- F Springfield

True or False:

Chicago is the capital of Illinois.

- F True F False

The *Radio* system is formatted in such a way that one cannot check more than one response. Lines 14-19 in Figure 1 shows the hypertext codes for the multiple choice items, and lines 20-22 shows the codes for the true or false item. Lines 14 and 20 contain the probing question. Since an Internet browser does not recognize a “hard return”, one must tell the browser when to “break” the line. One will note that the `
` at the end of the line tells the browser to break to the next line (with the character starting at the left margin).

Looking at line 15 one will see the first option available for the Multiple Choice question. As one can recall from above, the first option is Champaign. Let’s break the code down into specifics:

```
<INPUT TYPE="radio" NAME="capitalmc"
  value="Champaign">Champaign<br>
```

The `<INPUTTYPE="radio">` part tells the browser that this is a radio type of item—which means that only one answer will be accepted. The `NAME="capitalmc"` code asks for the name of this particular question. One can call it anything, but make sure it is called something that helps relates to the question. In our example, it has been named *capitalmc* for the multiple choice question about the capital of Illinois. Another option would be to call it something like “item1”...which indicates that this is the first question of the survey.

F i n a l l y t h e c o d e `value="Champaign">Champaign
` asks what to call the actual choice for this particular multiple choice item. Note that the value lists Champaign in quotes. Also note that the same word is used after “>” and before the “
”. The latter is what will appear in the actual text of the evaluation assessment and can include any amount of text. As with the “name”, one

can call the “value” anything they want. For example, since Champaign was the first choice of five options, we could have given this value “1a” (representing option A of question 1).

Lines 20-22 deal with the true or false component. Line 20 shows the statement of the problem... “Chicago is the capital of Illinois”. Lines 21 and 22 are the codes for the “true” or “false” response respectively. Again, let’s look at the specifics of the codes:

```
<INPUT TYPE="radio" NAME="capitaltf"
  value="True">True
<INPUT TYPE="radio" NAME="capitaltf"
  value="False">False<p>
```

The `<INPUT TYPE="radio">` indicates the type of response this item will have (true or false...only one choice). The `NAME="capitaltf"` will identify which question this is. In this case we have combined the key words of capital, true and false to represent that this is the true or false item about the capital (again, using the alternate example, we could name this “item2”). The `value="True">True` tells the browser which answer one wants to place here. In this case, it is the true response. Note that these codes are the same except for the “value” component.

Observe that there is no `
` code after the last true in line 21. This means that the next item will appear next to it and not below (as in the case of the lines 16-20). If one desires to place the “false” below the true, then a `
` would need to be positioned after the “true.” At the end of line 22 one will note the command `<p>`. This tells the browser to break this line and to skip a space.

To summarize, *Radio* items are designed for questions in which the evaluator only wants one option chosen. A respondent can change their mind by simply clicking another option. The original checkmark will be removed and the new one will appear where the individual has pointed and clicked.

The Checkbox

A second type of item is referred to as a *Checkbox*. The *Checkbox* allows a respondent to check as many responses as one desires. In the example, one is asked which of the following five items causes stress in their life. One could theoretically check none or all of the following. Unlike the *Radio*, once the box is checked, the checkmark will stay there if another box is checked. To undo an existing checkmark one has to physically click the box that is to be removed.

Figure 1. Hypertext Codes for Evaluation Assessment

```

1 <HTML>
2 <HEAD>
3 <TITLE>Evaluation Form</TITLE>
4 </HEAD>
5 <BODY TEXT="#000000" BGCOLOR="#FFE0C0" LINK="#0000FF" VLINK="#551A8B"
6 ALINK="#FF0000">
7 <H2>Evaluation Form</H2>
8 <P>
9 <FORM METHOD="POST" ACTION="mailto:youreemail@why.edu">
10 Please answer the questions to the best of your ability. <br>
11 Please note that when you are through, hit the "Submit" button at the end. <br>
12 If you want to start over, click the "Clear All" button that is at the end of the page.
13 <p>
14 1. What is the capital of Illinois?<br>
15 <INPUT TYPE="radio" NAME="capitalmc" value="Champaign">Champaign<br>
16 <INPUT TYPE="radio" NAME="capitalmc" value="Chicago">Chicago<br>
17 <INPUT TYPE="radio" NAME="capitalmc" value="Lincolnville">Lincolnville<br>
18 <INPUT TYPE="radio" NAME="capitalmc" value="Peoria">Peoria<br>
19 <INPUT TYPE="radio" NAME="capitalmc" value="Springfield">Springfield<p>
20 2. Chicago is the capital of Illinois.<br>
21 <INPUT TYPE="radio" NAME="capitaltf" value="True">True
22 <INPUT TYPE="radio" NAME="capitaltf" value="False">False<p>
23 3. Which of the following causes you stress?<br>
24 <INPUT type="checkbox" Name="stress" value="money">Money<br>
25 <INPUT type="checkbox" Name="stress" value="relationships">Relationships<br>
26 <INPUT type="checkbox" Name="stress" value="work">Work<br>
27 <INPUT type="checkbox" Name="stress" value="children">Your Children<br>
28 <INPUT type="checkbox" Name="stress" value="internet">The Internet<br>
29 <p>
30 4. What is one word to best describe how you react to stress? <br>
31 <INPUT type="text" name="word" size="10" maxlength="10">
32 <p>
33 5. Describe how you deal with stress.<br>
34 <TEXTAREA name="dealwithstress" Rows="3" cols="60"></TEXTAREA>
35 <p>
36 <INPUT type="reset" value="Clear all"><INPUT type="submit" value="Submit">
37 </FORM>
38 </BODY>
39 </html>

```

The example of the *Checkbox* in Figure 1 asks the following:

3. Which of the following causes you stress?
 - G Money
 - G Relationships
 - G Work
 - G Your Children
 - G The Internet

Line 23 to 29 of Figure 1 will reveal the coding needed to use a *Checkbox*. Line 23 asks the question “Which of the following causes you stress?”. One will note the `
` after the question mark. That code tells the browser to break that line and start a new one. Lines 24-28 reveal the codes for the five options for this particular question.

The following is the coding required to use a *Checkbox*:

```
<INPUT type="checkbox" Name="stress"
      value="money">Money<br>
```

Similar to the *Radio* examples, the first part of the code tells the browser what type of item this is to be— `<INPUT type="checkbox">`. Since this is an item that multiple answers are acceptable, the word *Checkbox* is used instead of *Radio*. The `Name="stress"` allows one to identify which question this is. In this case it is being called stress (although the alternative option would be to call this “item3”). Note that the codes `value="money">Money
` uses the word money twice...one in its value and once between the `>` and `
` codes. This is the first option of this particular question (as with the multiple-choice *Radio* example, one could identify the value as “3a”, “3b”, “3c”, “3d”, and “3e” with the number 3 representing the item number and each letter representing the five options. As one can see, the `value="money">Money
` segment is the only difference among the five options for this line. One would place the words money, relationships, work, children, and internet in their respective order. One can put a much more extensive statement between the `>` and `
`, as one can see with the *children* and *Internet* examples. In both of these cases the `value` states “children” or “Internet,” but the appearance of the phrase includes the word “the”. Finally, note that line 29 includes the code `<p>` to tell the browser to break the line and to skip a line.

The Test and Textarea Items:

The third and fourth type of item can be an essay or open-ended response. Figure 1 in Lines 30-32 gives an example of a single word response whereas lines 33-35

gives the respondent an opportunity to type several words and/or sentences.

Again, lines 30 and 33 are the statements or questions. In both of these examples, there is a `
` code to tell the browser to move the space for the respondent to type back to the left margin.

Let’s look at the differences between the two. The *Test* item is used for a single word responses. Here are the codes:

```
<INPUT type="test" name="word"
      size="10" maxlength="10">
```

The `<INPUT type="test">` tells the browser that one is only expecting a very brief response. The code `name="word"`, like the other items, gives a description of the question one is asking. In this case, it is called “word” because the statement asks “What is one word to best describe how you react to stress.” If one were to follow the alternative strategy, the “name” for this would be “item4”. Finally, the codes `size="10" maxlength="10">` tells the browser how many characters long one wants to allow. The `size` code tells how physically long the box should appear on the screen...whereas the `maxlength` code is how many actual characters that can be typed. In this example, both are ten so one cannot type more than ten characters. If one would increase the `maxlength` to some other number (i.e., 20) it will allow one to type up to that number of characters. Only ten characters will be visible in the box while the remaining letters will be scrolled to the left. Whatever one sets as a `maxlength`, the entire comment will be forwarded. Finally, please note that line 32 has a `<p>` code which tells the browser to break to the next line and to skip a space.

The last type of item is the *textarea* allows one to type several words and/or sentences. The coding for this appears in lines 34:

```
<TEXTAREA name="dealwithstress"
      Rows="3" cols="60"></TEXTAREA>
```

This coding is slightly different than the others. First, the `<TEXTAREA>` tells the browser that this is to have a response of several words and/or sentences. The code `name="dealwithstress"` is similar to the previous items where one wants to be able to recognize which question the responses are related to (the alternative option would be to call this “item5”). Codes `Rows="3" cols="60">` tells the browser the size of this box. In this

example the box will appear on the screen as 3 rows high and 60 characters wide (although one does have unlimited writing capability). One could just use one line and keep typing (the words will scroll off the box if it is longer than 60 characters). Or, one can hit the return button and it will go to the next line. Again, if one keeps hitting the return the *Textarea* will scroll down and the earlier lines will move out of the box. As with the *Test* item, the entire message will be forwarded, regardless of whether it all fits on the screen.

Unlike any of the other items, after one uses a *textarea*, they must include the `</TEXTAREA>` code. This tells the browser that you are through with this particular *Textarea* item.

Finally, the part of the assessment tool to allow the browser to e-mail the responses are included in line 36 of Figure 1. The codes are:

```
<INPUT type="reset" value="Clear all"><INPUT
      type="submit" value="Submit">
```

The first code, `<INPUT type="reset" value="Clear all">` is a button to allow the person to clear all of their responses and to type them over. The second code, `<INPUT type="submit" value="Submit">` is the button that one will press to send the responses to the designated e-mail address listed in line 9:

```
<FORM METHOD="POST"
      ACTION="mailto:youremail@why.edu">
```

One must use these codes exactly as they appear. The only item to be changed would be one's e-mail address. In addition, this code must precede all *Radio*, *Checkbox*, *Test*, or *Textarea* commands. Also note that on line 37 the command `</form>` must be included. These items will only work if they are between the commands given in lines 9 and 37.

Getting Results

OK, one has now created a web page with an evaluation assessment. People will enter your web page and complete the survey. They will then hit the "send" button and the evaluation will be sent to one's designated e-mail. Now one will need to interpret these results. The raw data received may appear somewhat confusing, but with a little time, each item can be clearly identified. Let's show how to interpret such data.

For the sake of arguments, let's create some responses (shown in Figure 2). For the first multiple choice item, let's say the person clicked "**Chicago**". For the true or false statement, they clicked "**false**". In the *checkbox* the individual clicked the following items that causes them stress: "**money**," "**relationships**," and "**children**." The

one word used to describe how they dealt with stress was "**sleep**". Finally, in the open-ended statement, the person typed the following response:

"There are a number of ways that I deal with stress. First, I tend to drink a lot; second, I tend to sleep a lot; third, I tend to be very moody."

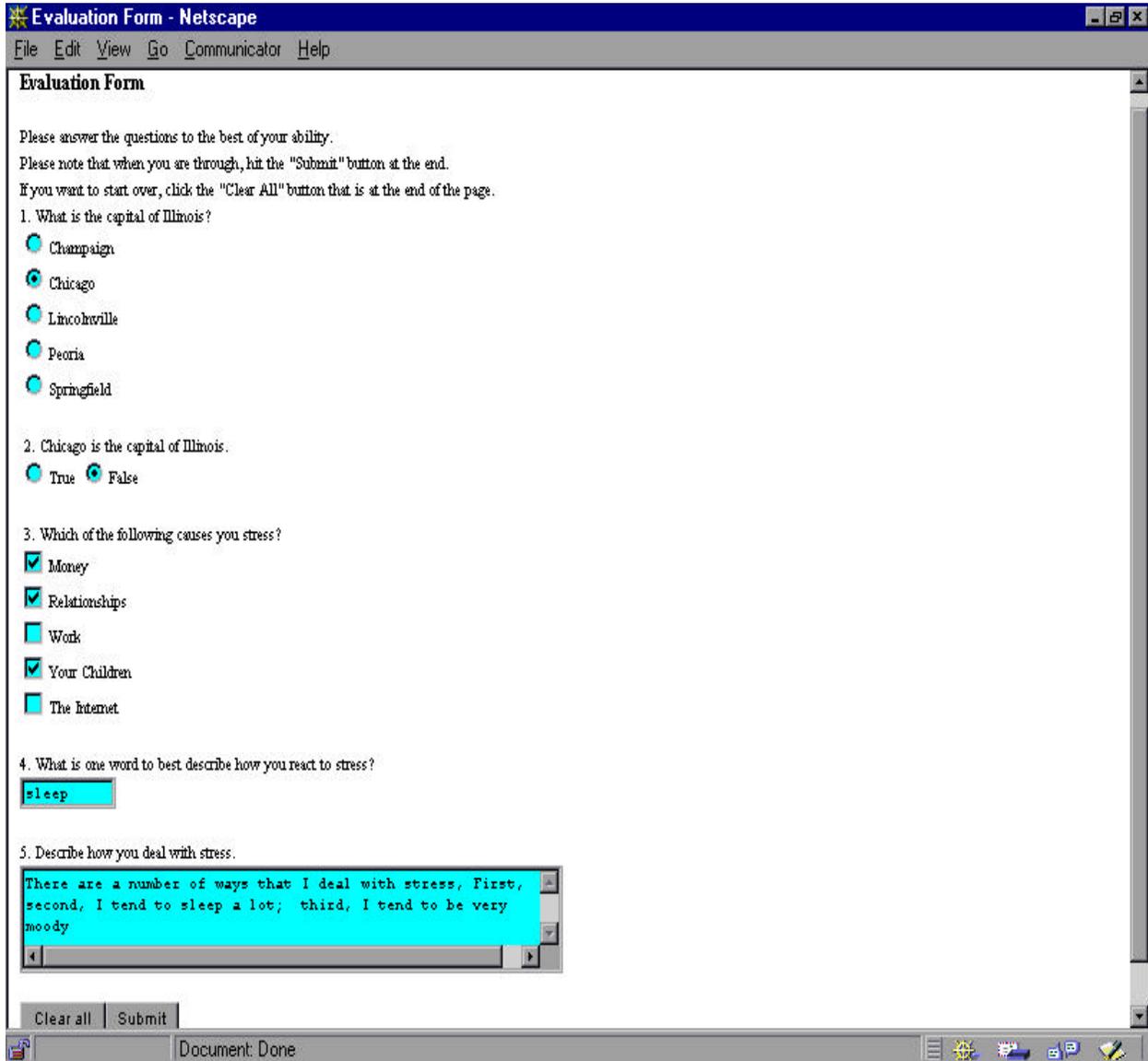
(Please note that the respondent hit the enter button after the first use of "lot" and after "very" and , some of the typed response cannot be seen in the box in Figure 2.)

One will receive the results of this survey via an e-mail message, generally as an attachment. The appearance of this message will look similar to what one sees in Figure 3. Once this file is brought into a word processing system one should be able to make some universal commands to replace some of these codes with some other commands. For example, all spaces are represented by a "+". Thus, once this data is brought into a word processing program, one can request a universal replacement of all "+"s" with just a blank space.

The first major code in Figure 3 is the "names" of the various questions that are included in this survey. The first question was a multiple choice question that asked the capital of Illinois. The respondent indicated that Chicago was the capital. On line 1 of Figure 3 one will see the word *capitalmc*...one should recall that this is the multiple choice question about the capital of Illinois (Figure 1, lines 15-19). Continuing with Figure 3, one will notice that after that *name* (and all other *names*) one will see an equal sign (=). After each "=" comes the answer given by the respondent. In this case, the response (or the *value*) was Chicago.

Still in Figure 3, line 1, after Chicago one will see the code "&". This is what separates *names* from each other. Again, one could do a universal replacement of all "&'s" with a hard return...one would then have each item lined up under each other in a column format. According to this data, the respondent indicated false for the second item that asked "Chicago is the capital of Illinois."

Figure 2. A Copy of the Web Page with Fictitious Answers



Also on line 1, and continuing onto line 2 are the results of the *Checkbox*. Remember the survey asked which of the following five items were causes of stress. The respondent checked off money, relationships and children as their sources of stress. Again, looking at line 1 after the word “false” is the “value” *stress=...*the items shown are money, relationships, and children. Had either of the other two options been checked (the internet or work), those two would have also been shown as *stress=work&stress=internet*.

To see the response to the item that asked for a one word description on how the respondent deals with stress, on line 2 after the code *word=* is what the respondent typed as their way of dealing with stress. In this example, the response was *sleep*. Finally, starting on line 2 and ending on line 5 is the response to the essay question “*Describe how you deal with stress?*” As noted earlier, the spaces are shown with “+’s”. All commas are represented by “%2C”. There is a comma after the word “First” (line 3); “second” (line 3), and “third” (line 4). Semi-colons are represented as “%3B”. Semi-colons are shown in Figure 3 on lines 3 and 4 after the words “lot”. Finally, all hard returns (those returns that are caused by the hitting of the enter/return button on one’s keyboard) are represented by “%0D%0A”. Returns can be noted on lines 3 (after the

word “lot”) and on line 5 right before the word “moody”. As with the previous examples, one could perform a universal replacement of these letters and substitute the appropriate comma, semi-colon, or hard returns. (There are ways to configure one’s server so that such codes are not sent, but one would need to contact their local computer expert to help program the system accordingly.)

Conclusion

In a nutshell this is how one can create a web page evaluation. All one needs to do is copy all of the codes (not the line numbers) from Figure 1 into a word processing file. Save that file in an “ascii” or text format. Once saved, go into the Windows Explorer to rename the file with an extension of either .htm or .html file (these are the only two extensions that the Internet browsers accept for interactivity on the world-wide web). After renaming the file, one will need to transfer the file to your designated server (one should discuss with their information technology personnel on your institution’s prefer mode of “ftping”). If one follows these commands as they are shown, one’s evaluation web page should work. More information on how to create web-based evaluations can be found on the author’s home page. Click the “tutorial” link.

