

## Week Nine – 11/20: Forms

- We will be creating a simple form that will allow our site visitor to communicate with us through our web page – they will be able to enter their name, email address, some basic information and a message
- Forms generally have two main components to them:
  - the **structure**, or shell that consists of the labels, form fields and buttons
  - the **script** behind the scenes that will allow our form data to be processed
- Your form for this lecture contain:
  - Text Boxes
  - Password Boxes
  - Radio Buttons
  - Check Boxes
  - Select Menus
  - Message Fields
  - Submit & Reset Buttons
- let's start out by using the `form` tag to start our form's structure – I will be using my standard layout I have created for my form page:

```

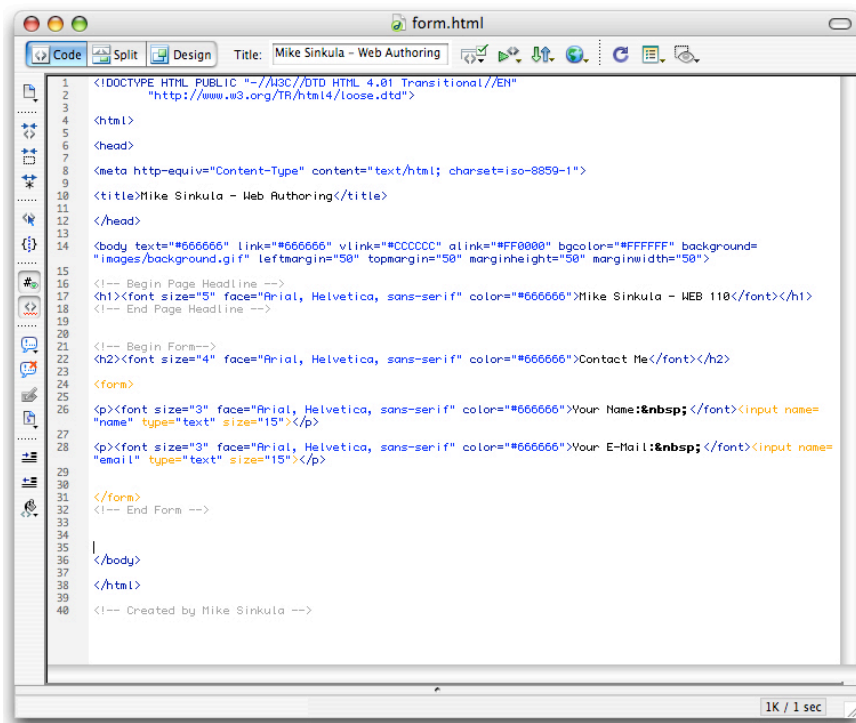
<!-- Begin Form-->
<h2><font size="4" face="Arial, Helvetica, sans-serif" color="#666666">Contact
Me</font></h2>
<form>

</form>
<!-- End Form -->

```

1. I have used my `h2` tag to create a sub-headline for the page
2. I have then used opening and closing `form` tags that will contain the elements of my form

- Let's then put some text fields into our form:

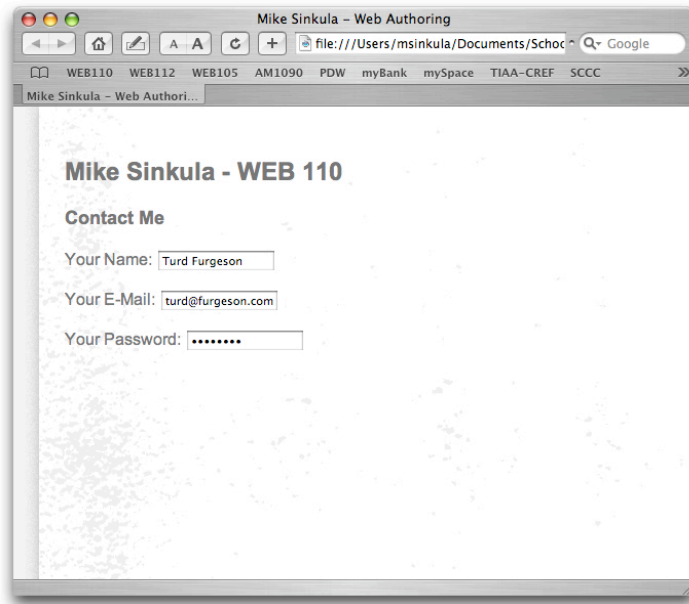


```

<form>
<p><font size="3" face="Arial, Helvetica, sans-serif" color="#666666">Your
Name:&nbsp;</font><input name="name" type="text" size="15"></p>
<p><font size="3" face="Arial, Helvetica, sans-serif" color="#666666">Your E-
Mail:&nbsp;</font><input name="email" type="text" size="15"></p>
</form>

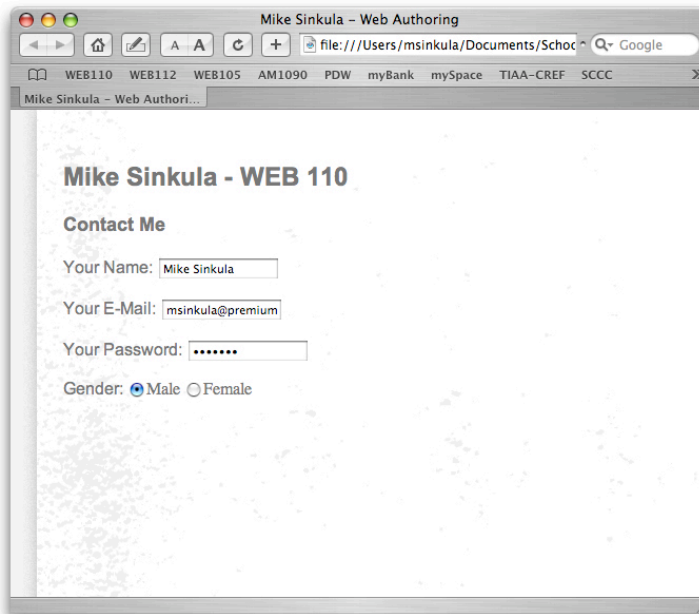
```

1. I am placing my text fields inside my `p` tags to make sure that they stay on the same line as their titles
  2. to create text fields in your form you need to use the `input` tag
  3. we will also be using the `name` attribute again with the appropriate *value* for our field – our name attribute will get sent along when we send our email
  4. for our text field we can also use a `value` attribute with a value of our choosing – in this case I did not use one
  5. once the user fills in the text field that value will also get sent along in the email
  6. we also need to declare the `type` attribute with the *value* of `text` to make sure this will indeed be a text field
  7. lastly we need to declare the `size` attribute with a character number *value* that will declare the width of our text field
- Now, how about a password box just to see what they do:



```
<p><font size="3" face="Arial, Helvetica, sans-serif" color="#666666">Your
Password:&nbsp;</font><input name="password" type="password" size="15"></p>
```

1. to create a password box we still need to use the `input` tag
  2. we also need to give it a `name` off `password`
  3. but this time for the `type attribute` we will use the `value` of `password` to make sure that it will indeed be a password box
  4. notice that the text fields display text inside them and that the password box will display bullets in case someone is looking over your shoulder
- Time for some radio buttons:

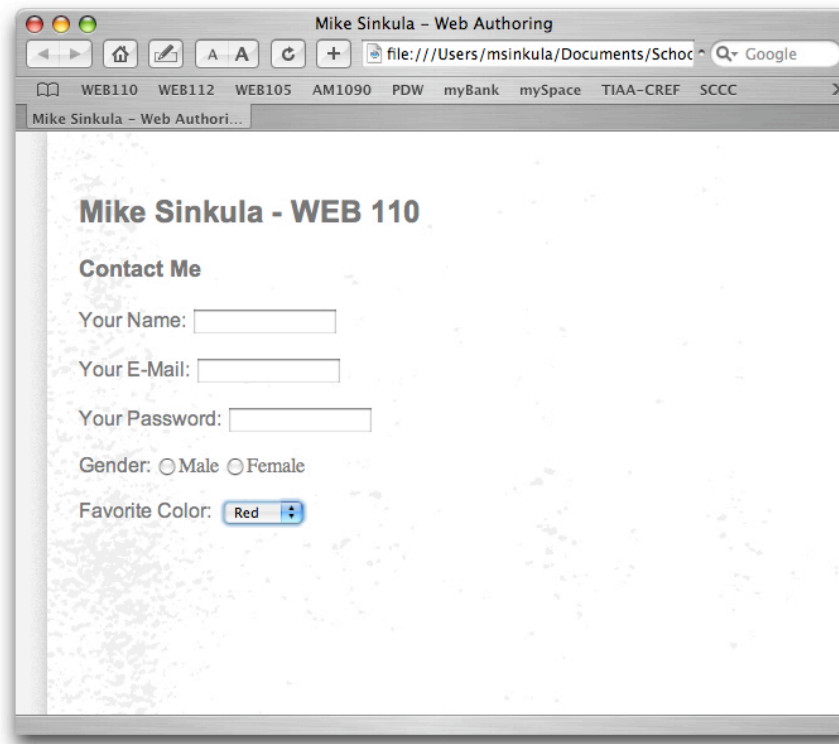


```
<p><font size="3" face="Arial, Helvetica, sans-serif"
color="#666666">Gender:&nbsp;</font><input name="gender" type="radio"
```

```
value="male">Male<input name="gender" type="radio"
value="Female">Female</p>
```

1. to create a radio button we still need to use the `input` tag
2. notice here that the `name` attribute for both radio buttons has the same value – this links the set together and will allow for only one radio button to be selected
3. the `name` attribute's value of gender will then get sent in the email with along the `value` attribute's value of either male or female – got that?

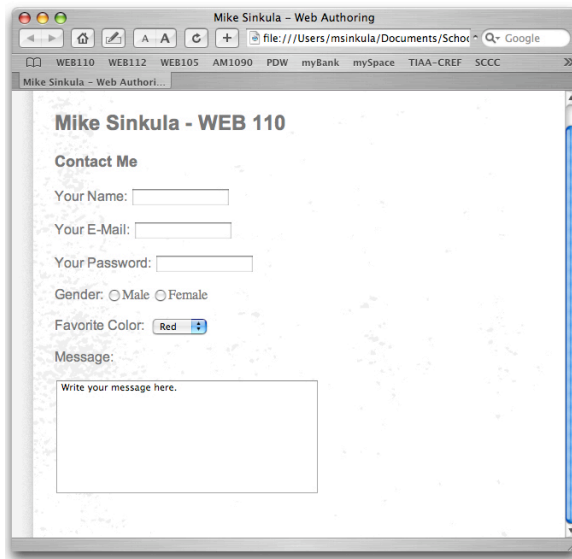
- I think we need a select menu too:



```
<p><font size="3" face="Arial, Helvetica, sans-serif" color="#666666">Favorite
Color:&nbsp;</font>
<select name="color">
  <option value="red">Red</option>
  <option value="green">Green</option>
  <option value="blue">Blue</option>
</select>
</p>
```

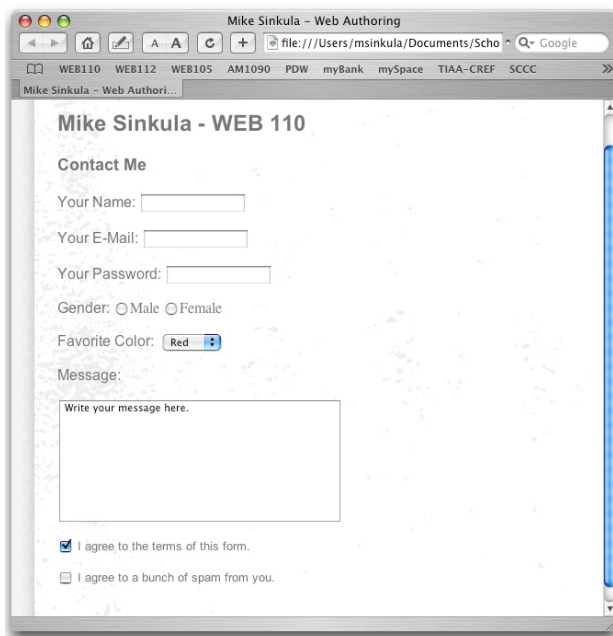
1. here we will need to use parent and child tags of `select` and `option`
2. within the `select` tag we again will use the `value` attribute and give it a value that will get sent in the email
3. in this case I have written three different `option` tags which declare the selections that we can make within this menu
4. each one of these `option` tags will also need a `value` attribute and a value that will get sent in the email
5. between the `option` tags we also need to write the text that will get displayed in the menu on the browser screen

- since we are sending this via email, we need a text area for our visitor to send along a message:



```
<p><font size="3" face="Arial, Helvetica, sans-serif"
color="#666666">Message:&nbsp;</font></p>
<p><textarea name="message" cols="40" rows="10">Write your message
here.</textarea></p>
```

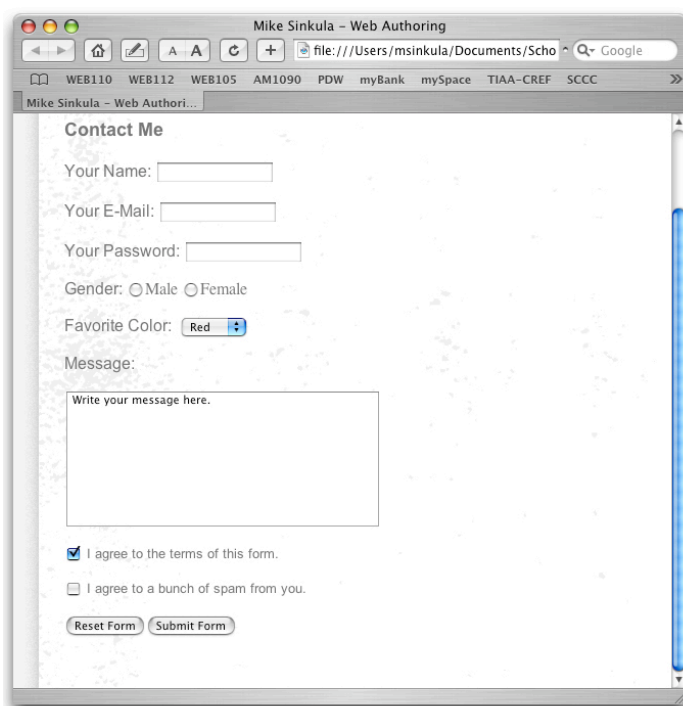
1. for our message box we will use an opening and closing `textarea` tag
  2. again we need to use the `name` attribute with a value
  3. in order to make the text are big enough, we will use the `cols` and `rows` attributes to control the width and height
  4. I've written text that will appear in the text area by placing it in between the `textarea` tags
- What form would be complete without check boxes:



```
<p><input name="agree" type="checkbox" value="yes" checked="checked"><font
size="2" face="Arial, Helvetica, sans-serif" color="#666666">&nbsp;I agree to
the terms of this form.</font></p>
```

```
<p><input name="spam" type="checkbox" value="yes"><font size="2" face="Arial,
Helvetica, sans-serif" color="#666666">&nbsp;I agree to a bunch of spam from
you.</font></p>
```

1. again, we use the `input` tag with a `name` attribute
  2. for the `type` attribute we need to use the value of `checkbox` – makes sense
  3. we also need to give our check boxes `value` attributes
  4. notice, however that the first check box has a `checked` attribute with a value of `checked` – this will make the check box default to having a check inside of it
- no form will work without buttons to control what they are doing – so let's make sure we add those:



```
<p><input name="reset" type="reset" value="Reset Form"><input name="submit"
type="button" value="Submit Form"></p>
```

1. I am adding a "Reset Form" button, which will clear your form, and a "Submit Form" button which will send the email to you
2. both use the `input` tag
3. the reset button needs the `name` attribute with the value of `reset` and the `type` attribute with the value of `reset`
4. the submit button needs the `name` attribute with the value of `submit` and the `type` attribute with the value of `button`
5. to display the proper text in your button you will need to use the `value` attribute with the value of what you want your button to say

- Since the form doesn't automatically know where to send the email to, we need to declare that by using hidden fields – fields that do not show up in the browser screen:

```
<input type="hidden" name="recipient" value="msinkula@premiumdw.com">
<input type="hidden" name="subject" value="Contacting Mike!">
<input type="hidden" name="redirect"
value="http://www.premiumdw.com/WEB110/examples/form.html">
```

1. we will be using three hidden field that all use the `input` tag
  2. we declare them as hidden by using the `type` attribute with the value of `hidden`
  3. the first hidden field is declaring the form to send the email to a specific address using the `name` attribute with a value of `recipient` and a `value` attribute with my email address
  4. the second hidden field is declaring the form to send the email with a specific subject using the `name` attribute with a value of `subject` and the `value` attribute containing a subject for the email
  5. the third hidden field will redirect the page back to a page of your choosing by using the `name` attribute with the value of `redirect` and the `value` attribute with a specific (absolute path) URL
- almost done – now that we have our form written we need to use a script that does the actual “sending” of the email:

```
<form action="cgi-bin/formmail.pl" method="post">
```

1. in the opening `form` tag we need to use the `action` attribute with the value of the script that we will be using
2. we also need to use the `method` attribute with the value of `post` in order to send the email

**NOTE: unfortunately the Edison server does not allow to send email via CGI scripts to email addresses other than school addresses at this time – I am trying to overturn this.**

Assignment: Form  
Reading: Chapter 16