CS472 WAP

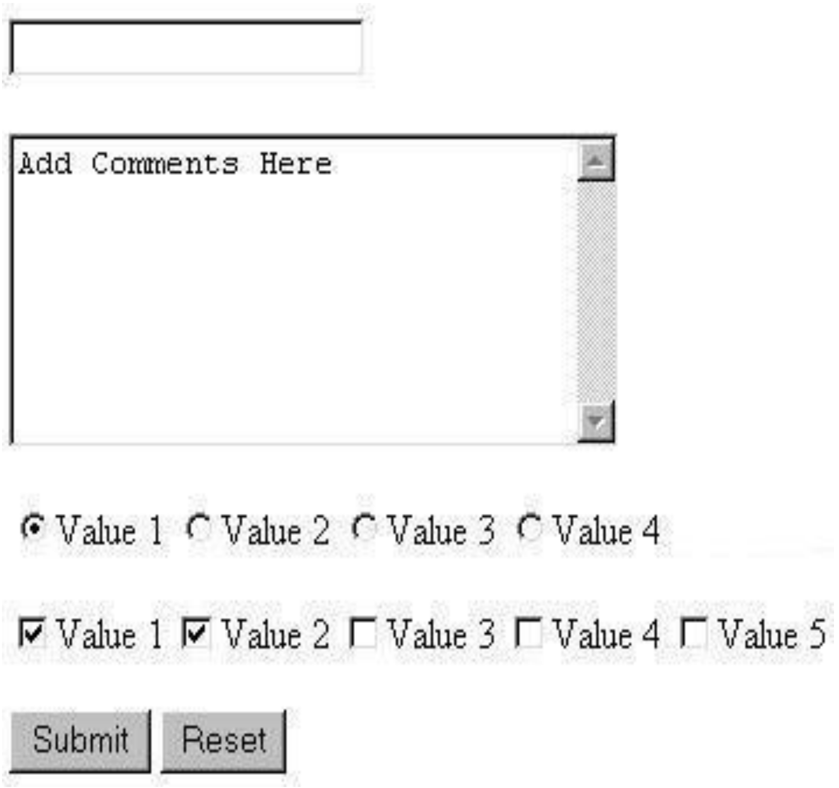
Lecture 1: HTML5 Forms

Maharishi International University - Fairfield, Iowa

HTML forms

Form: group of UI controls that accepts information from the user and sends the information to a web server

- The information is sent to the server as a query string



Add Comments Here

● Value 1 ○ Value 2 ○ Value 3 ○ Value 4

☑ Value 1 ☑ Value 2 □ Value 3 □ Value 4 □ Value 5

HTML form: **<form>**

The <**form**> tag is used to create an HTML form for user input.

The <**form**> element can contain one or more of the following form elements:

<**input**>, <**textarea**>, <**button**>, <**select**>, <**option**>, <**optgroup**>, <**fieldset**>, <**label**>

```
<form action=“sales.html“ method=“get“ novalidate autocomplete=“on”>
         Form controls
</form>
```

Form Attributes

- **action** destination URL
- **method** get, post
- **enctype** application/x-www-form-urlencoded, multipart/form-data, text/plain
- **novalidate** (HTML5) specifies that the form should not be validated when submitted
- **autocomplete** (HTML5) on, off

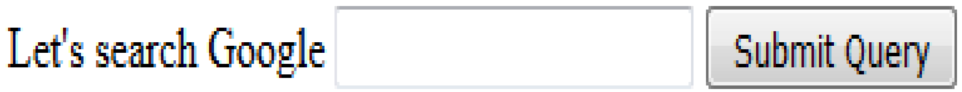
HTTP **GET** vs. **POST** requests

- **GET** : asks a server for a page or data
  - if the request has parameters, they are sent in the URL as a query string (request header)
  - URLs cannot contain special characters without encoding
  - private data in a URL can be seen or modified by users

- **POST** : submits data to a web server (to be saved in DB or file or updates state in server application)
  - parameters are embedded in the HTTP request body, not the URL

Form Example

```
<form action="http://www.google.com/search">
     <div> Let's search Google
          <input name="q" />
          <input type="submit" />
     </div>
</form>
```



Form controls: <input>

```
<input type="text" name="q" value="Colbert Report" />
<input type="submit" value="Booyah!" />
```

- `input` element is used to create many UI controls
  - an inline element that MUST be self-closed

- `name` attribute specifies name/key of query parameter to pass to server
- `type` can be `button, checkbox, file, hidden, password, radio, reset, submit, text, ...`
- `value` attribute specifies control's initial text

Text fields: <input>

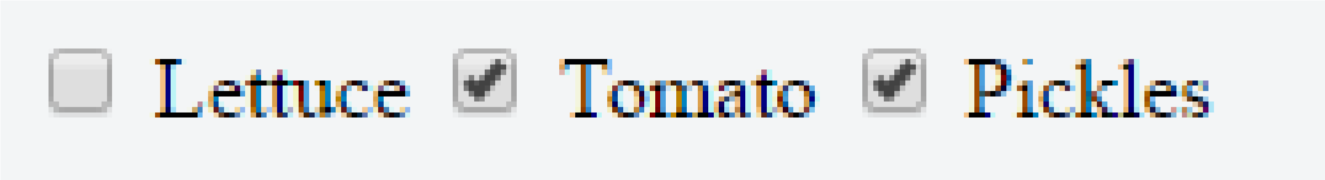
```
<input type="text" name="username" size="10" maxlength="8" />
<input type="text" name="password" size="8" />
```

Attribute	Value(s)	Description
value	text	Initial text to appear in text box
size	integer	Visible length of text box, in characters
maxlength	integer	Maximum number of chars that may be typed into text box
autocomplete	on, off	Whether to offer suggestions of text to auto-complete the field
autofocus	autofocus	Makes control initially receive keyboard focus
novalidate	novalidate	Indicates browser should not check value before submitting
placeholder	text	A hint or example of what the user should type;
pattern	regular expr.	A regular expression indicating what input is valid
required	required	Whether browser should display an error if blank

Checkboxes

- yes/no choices that can be checked and unchecked (**inline**)
  - none, 1, or many checkboxes can be checked at same time
  - Use the **`checked`** attribute in HTML to initially check the box

```
<input  type="checkbox" name="condiment1" /> Lettuce
<input  type="checkbox" name="condiment2" checked /> Tomato
<input  type="checkbox" name="condiment3" checked /> Pickles
```

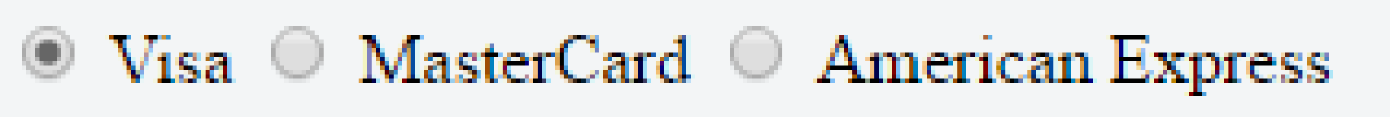


□ Lettuce ☑ Tomato ☑ Pickles

Radio buttons

- Sets of mutually exclusive choices (**inline**)
  - Grouped by **`name`** attribute (only one can be checked at a time)
  - Must specify a **`value`** for each one or else it will be sent as value **`on`**

```
<input type="radio" name="cc" value="visa" checked /> Visa
<input type="radio" name="cc" value="mastercard" /> MasterCard
<input type="radio" name="cc" value="amex" /> American Express
```



● Visa ○ MasterCard ○ American Express

<textarea>

- The <**textarea**> tag defines a multi-line text input control. (inline)
- holds unlimited number of characters
- text renders in a fixed-width font (usually Courier).
- size specified by **cols** and **rows** attributes,
  - Or better, through CSS **height** and **width** properties.
  - textarea { width: 300px; height: 200px }

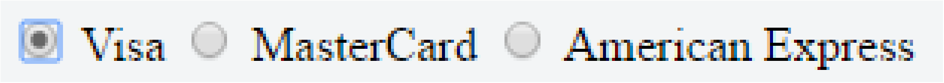
```
<textarea rows="4" cols="20">
    Type your comments here.
</textarea>
```

Text labels: `<label>`

- Associates nearby text with control, so you can **click text to activate control**
- Can be used with **checkboxes** or **radio** buttons
- **`label`** element can be targeted by CSS style rules

```
<label> <input type="radio" name="cc" value="visa" checked="checked" /> Visa</label>
<label> <input type="radio" name="cc" value="mastercard" /> MasterCard</label>

<label> <input type="radio" name="cc" value="amex" /> American Express</label>
```

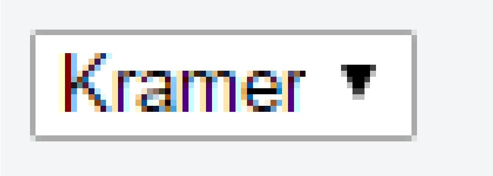


● Visa ○ MasterCard ○ American Express

Drop-down list **<select>** and **<option>**

- Menus of choices that collapse and expand (inline)
  - **option** element represents each choice
  - **select** optional attributes: **disabled**, **multiple**, **size**
  - optional **selected** attribute sets which one is initially chosen

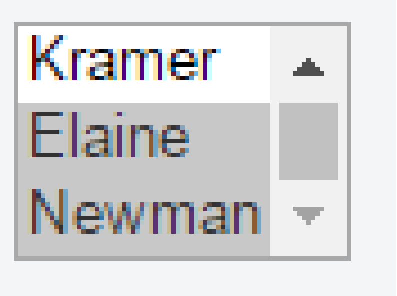
```
<select name="favoritecharacter">
  <option>Jerry</option>
  <option>George</option>
  <option selected>Kramer</option>
  <option>Elaine</option>
</select>
```



Multiple `<select>`

- optional **`multiple`** attribute allows selecting multiple items with shift- or ctrl- click
- **`option`** tags can be set to be initially **`selected`**

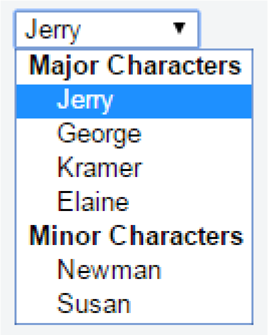
```
<select name="favoritecharacter"   size="3" multiple>
  <option>Jerry</option>
  <option>George</option>
  <option>Kramer</option>
  <option>Elaine</option>
  <option selected>Newman</option>
</select>
```



Kramer
Elaine
Newman

Option groups: <optgroup>

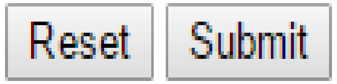
```
<select name="favoritecharacter">
 <optgroup label="Major Characters">
   <option>Jerry</option>
   <option>George</option>
   <option>Kramer</option>
   <option>Elaine</option>
 </optgroup>
 <optgroup label="Minor Characters">
   <option>Newman</option>
   <option>Susan</option>
 </optgroup>
</select>
```



Reset and Submit buttons

- When we click **`reset`** button, it returns all form controls to their initial values
- When we click **`submit`** buttons, it sends all data with the specified **`method`** (Get/Post) to the **`action`** page in the form
- Specify custom text on the button by setting its **`value`** attribute

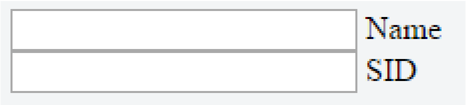
```
<input type="reset"  />
<input type="submit" />
```



Hidden input parameters

- An invisible parameter that is still passed to the server when form is submitted, it's useful for passing on additional state that isn't modified by the user

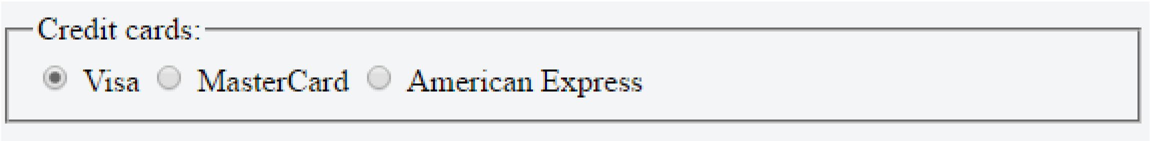
```
<input type="text" name="username" /> Name
<br />
<input type="text" name="sid" /> SID
<br />
<input type="hidden" name="school" value="MUM" />
<input type="hidden" name="year" value="2048" />
```



Grouping `<fieldset>`, `<legend>`

- Groups of input fields with optional caption (legend)

```
<fieldset>
 <legend>Credit cards:</legend>
 <input type="radio" name="cc" value="visa" checked="checked" /> Visa
 <input type="radio" name="cc" value="mastercard" /> MasterCard
 <input type="radio" name="cc" value="amex" /> American Express
</fieldset>
```



Credit cards:

● Visa ○ MasterCard ○ American Express

Main Point

HTML provides many different types of input widgets, including text fields, text areas, check boxes, radio buttons, and dropdown lists, this is also an area HTML 5 is expanding to make form filling more efficient and effortless.

Nature supports the growth of things that are efficient. Do less and accomplish more.

CONNECTING THE PARTS OF KNOWLEDGE WITH THE WHOLENESS OF KNOWLEDGE

HTML Forms: Connecting with the Source

1. Forms let us submit data to the web server, which can then generate a custom response based on server-side information.
2. GET requests are intended to only retrieve information and should be idempotent. POST requests are intended to submit data and not request a direct response.

3. Transcendental consciousness is the experience of the source of thought.

4. Impulses within the Transcendental field: inputs and perceptions are appreciated in their full value at the quiet levels of the mind.

5. Wholeness moving within itself: In Unity Consciousness one appreciates the full value of all inputs and perceptions. Everything is appreciated as some expression of Totality.