HTML useful stuff

# Forms

* <form> tag
  + allows the web browser to submit information from the user to the server
  + action attribute
    - indicates the URL where the form data should be sent
    - Typically the URL uses HTTPS so the form data is encrypted
  + method attribute
    - indicates the HTTP request type the browser will use to communicate with the server
    - The method is either GET or POST
    - Get is the default method if no method is specified
* GET method
  + is a technique used by a web browser to submit information to a web server by altering the URL of the HTTP request
  + When a user clicks the submit button in a form that uses the GET method, the browser performs the following steps
    - collect all data from the form fields into a query string
      * query string is a set of name=value pairs separated by the ‘&’ character
      * each name is specified as an attribute of the HTML field, and the value is the user-entered data
    - create a URL with the server page and name=value pairs
      * The URL is composed of the action attribute specified in the form, the question mark character, and the query string
      * ex. http://example.com/apply?first=Rick&last=Deckard
    - Use the newly created URL to create and send an HTTP GET request
    - Display or update the web page using the HTTP response received from the server
* POST method
  + is a technique used by a web browser to submit information to a web server by sending the information in the HTTP request body
  + When the user clicks the submit button in a form that uses the POST method, the browser performs the following
    - create a HTTP POST request using the URL from the form’s action attribute
    - create a query string from the form data
      * ex. first=Sarah&last=Connor
    - place the query string in the HTTP request message body and send the request
    - display or update the web page using the HTTP response received from the server
  + enctype attribute
    - indicaters the web browser should split a POST request into multiple parts, where each input field is sent as a separate part of the HTTP request message
    - the value is “multipart/form-data”
  + Form widgets
    - is an interactive component (usually graphical) that the browser uses to interact with a user
    - <input>
      * allows the user to enter information into a web page
      * cannot enclose any additional page content, and thus does not have a closing tag
      * creates a text box widget
      * five primary attributes
        + type attribute

indicates the widget type. Common types include text, password, submit, and button.

* + - * + name attribute

names the widget and sends the widget's value when the widget's form is submitted.

* + - * + id attribute

is used to give a widget a unique identifier.

* + - * + placeholder attribute

specifies text that first appears in a text widget, typically for giving the user a hint as to the expected value.

* + - * + value attribute

specifies a default value for a widget.

* + - * text box
        + is an input element with the type attribute of “text” that allows users to enter a single line of text
      * submit
        + an <input> tag with the type attribute of “submit”, which sends the associated form’s data to the server when clicked
        + a submit button uses the value attribute to specify the button’s text
      * password field
        + a widget with the type attribute of “password”
        + allows users to enter a password without the password contents being displayed on-screen
        + size attribute

limits the password field’s width

can be used on text boxes

* + - * + maxlength attribute

limits the maximum number of characters the user can enter

can be used on text boxes

* + - * checkbox
        + is a widget for input elements with the type attribute of “checkbox”
        + allows users to check, or select a value
        + initially appears selected if the checked attribute is set
        + ex. <input type="checkbox" checked>
      * picker
        + an input widget that allows the user to interactively pick a choice using a popup or other guided selection method
        + date picker

a picker that allows the user to enter a date or choose a date from a calendar popup

type attribute “date”

type attribute “month” for month and year

type attribute “week” for week and month

min attribute

earliest date permitted

max attribute

latest date permitted

* + - * + color picker

creates a color selector popup that helps the user explore and choose a color

type attribute “color”

* + - * number input
        + ensures user input is a valid number
        + attribute type “number”
        + min and max attributes commonly used here
      * range input
        + allows the user to select a value by draggin a sliding control along the length of a line
        + attribute type “range”
        + min, max, vaue attributes commonly used here
      * combo box
        + a combination of a text box and drop-down menu into a single widget
        + created within a input element

list=”name”

called inside and outside of the input element

* + - * + has opening and closing brackets
        + created with <datalist>
        + uses <options>
      * Specialized text input
        + four input types exist for entering specific types of text

url – For typing a URL

tel – For typing a telephone number

email – For typing an email address

search – For typing an email address

all of these are attribute types

* + - * required
        + the required attribute can be used to identify what fields are optional and what fields are not optional
    - <label>
      * displays descriptive text associated with a specific widget
      * ***for*** attribute
        + value should match the id attribute for the widget being labeled
    - <textarea>
      * is an input element specified by <textarea> opening and closing tags that allows users to enter multiple lines of text
      * has optional rows and cols attributes to specify the initial size of the text area
    - <select>
      * opening and closing tags
      * create a drop-down menu
        + allows users to select on of several predefined values
        + <option>

opening and closing tags

create a value, or option, the user can select within a drop-down menu

When the user is not interacting with the menu, the drop-down menu usually displays the selected option

* + - * list box
        + created by specifying a size with the select element’s size attribute
        + Ex: <select size="4">

creates a list box that shows four options at a time

* + - * + if the list box contains more size options, the browser adds a vertical scrollbar so the uuser can scroll through the list of options
        + multiple attribute

allows the usaer to select multiple options

user must hold down ctrl on windows and cmd on mac

* + - <button>
      * opening and closing tags
        + or with <input type=”button”>
      * allows text and images to be displayed in a button
        + <input> button only allows text
      * has a type attribute that can be set to various values like “button” or “submit”
        + “button” is used with JavaScript to perform an action when clicked
        + “submit” creates a submit button for a form
  + <fieldset>
    - opening and closing tags
    - groups related form widgets together and draws a box around the related widgets
    - <legend>
      * opening and closign
      * defines a caption for a fieldset

# Form validation

* required
  + attribute that indicates that the field must have a value (text or selection) prior to submitting the form
* max/min
  + attributes that indicate the maximum and minimum values respectively that can be entered in an input field with ranges, such as a date or number
* maxlength/minlength
  + attributes that indicate the maximum and minimum length of input allowed by an input field
* pattern
  + attribute that provides a regular expression that valid input must match
* title
  + attribute can be used to provide a description of valid input when using the pattern attribute

# Audio element

* plays an audio file in a web page
* the <source> element is used in the <audio> tag to specify an audio file to play
* attributes
  + autoplay
    - Boolean attribute that makes the audio begin playing automatically.
  + controls
    - Boolean attribute that displays audio controls for the user to play, pause, and control the volume.
  + loop
    - Boolean attribute that replays the audio upon reaching the end of the audio.
  + muted
    - Boolean attribute that initially mutes the audio.

# Video element

* displays a video in a web page
* the <source> element is used in the <video> tag to specify an video file to play
* attributes
  + autoplay
    - Boolean attribute that makes the video begin playing automatically.
  + controls
    - Boolean attribute that displays video controls for the user to play, pause, and control the volume.
  + loop
    - Boolean attribute that replays upon reaching the end of the video.
  + muted
    - Boolean attribute that initially mutes the video.
  + width
    - Specifies the pixel width of the video's display area.
* <iframe>
  + allows a web page to be embedded in a rectangular area of the current web page
  + uses the src attribute to specify the URL of the web page to display

# Viewport

* <meta name=“viewport” content=“”>
  + allows developers to specify viewport properties
  + self closing
  + content properties
    - width
    - initial-scale
    - user-scalable
* viewport unit
  + vw and vh
  + is a percentage of the browser viewport’s width or height
  + 1vw = 1% of viewport width and 1vh = 1% of viewport height
  + vmin
    - 1vmin = 1vw or 1vh, whichever is smaller
  + vmax
    - 1vmax = 1vw or 1vh, whichever is larger
* <svg>
  + optimal img file

# More Info

## Table 6.3.1: Input attributes.

|  |  |  |
| --- | --- | --- |
| Attribute | Description | Example |
| maxlength | Sets the maximum number of input characters. | *<!-- Only 4 chars max can be entered -->*  <**input** type="password" maxlength="4"> |
| max | Sets the maximum value that the input can have. | *<!-- Number may not exceed 212 -->*  <**input** type="number" max="212"> |
| min | Sets the minimum value that the input can have. | *<!-- Date may not be earlier than July 4, 1976 -->*  <**input** type="date" min="1976-07-04"> |
| pattern | Provides a pattern (called a regular expression) that the input must match. | *<!-- Value must be A, B, or C followed by single digit -->*  <**input** type="text" pattern="[ABC][0-9]"> |
| required | States that the input is required and must not be left empty. | *<!-- At least one character must be entered -->*  <**input** type="password" required> |
| step | Sets the amount by which the value can change. | *<!-- Number is changed by multiples of 5 -->*  <**input** type="range" step="5"> |