HTML5

Presentation by Uplatz

Contact Us: https://training.uplatz.com/

Email: info@uplatz.com

Phone:+44 7836 212635

Table Of Contents:

- Using HTML with CSS
- Images
- Image Maps
- Input Control Elements

Uplatz

Directly including JavaScript code:

Instead of linking to an external file, you can also include the JS code as-is in your HTML:

```
<script>
// JavaScript code
</script>
```

Including a JavaScript file executing Asynchronously:

```
<script type="text/javascript" src="URL" async></script>
```

Using HTML with CSS:

- CSS provides styles to HTML elements on the page.
- Inline styling involves usage of the style attribute in tags, and is highly discouraged.
- Internal stylesheets use the <style> tag and are used to declare rules for directed portions of the page.
- External stylesheets may be used through a <link> tag which takes an external file of CSS and applies the

rules to the document.

- This topic covers usage of all three methods of attachment

External Stylesheet Use:

Use the link attribute in the document's head:

```
<head>
 <link rel="stylesheet" type="text/css"
href="stylesheet.css">
</head>
```

- You can also use stylesheets provided from websites via a content delivery network, or CDN for short. (for example,Bootstrap):

```
<head>
 <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap
/3.3.7/css/bootstrap.min.css" integrity="sha384-
BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RY
dg4Va+PmSTsz/K68vbdEjh4u"
crossorigin="anonymous">
</head>
```

Internal Stylesheet:

You can also include CSS elements internally by using the <style> tag:

```
<head>
 <style type="text/css">
 body {
 background-color: gray;
 }
 </style>
</head>
```

- Multiple internal stylesheets can be included in a program as well.

```
<head>
 <style type="text/css">
 body {
 background-color: gray;
 }
 </style>
 <style type="text/css">
 p {
 background-color: blue;
 }
 </style>
</head>
```

Inline Style:

You can style a specific element by using the style attribute:

```
<span style="color: red">This text will appear in red.</span>
```

Note: Try to avoid this -- the point of CSS is to separate content from presentation.

Multiple Stylesheets:

It's possible to load multiple stylesheets:

```
<head>
 <link rel="stylesheet" type="text/css"
href="general.css">
 <link rel="stylesheet" type="text/css"
href="specific.css">
</head>
```

Note that later files and declarations will override earlier ones.

So if general.css contains:

```
body {
 background-color: red;
}
```

and specific.css contains:

```
body {
 background-color: blue;
}
```

if both are used, the background of the document will be blue.

Images:

Parameters	**Details**
src	Specifies the URL of the image
srcset	Images to use in different situations (e.g., high-resolution displays, small monitors, etc)
sizes	Image sizes between breakpoints
crossorigin	How the element handles crossorigin requests
usemap	Name of image map to use
ismap	Whether the image is a server-side image map
alt	Alternative text that should be displayed if for some reason the image could not be displayed
width	Specifies the width of the image (optional)
height	Specifies the height of the image (option

Creating an image:

- To add an image to a page, use the image tag.
- Image tags (img) do not have closing tags.
- The two main attributes you give to the img tag are src, the image source and alt, which is alternative text describing the image.

<img src="images/hello.png" alt="Hello World">

You can also get images from a web URL:

<img src="https://i.stack.imgur.com/ALgZi.jpg?s=48&g=1" alt="StackOverflow user Caleb Kleveter">

- Note: Images are not technically inserted into an HTML page, images are linked to HTML pages.
- The <img> tag creates a holding space for the referenced image.

- It is also possible to embed images directly inside the page using base64:

<img src="data:image/png;base64,iVBOR..." alt="Hello World">

- Tip: To link an image to another document, simply nest the <img> tag inside <a> tags.

Choosing alt text:

- Alt-text is used by screen readers for visually impaired users and by search engines.
- It's therefore important to write good alt-text for your images.
- The text should look correct even if you replace the image with its alt attribute. For example:

```
<!-- Incorrect -->
<img src="anonymous.png" alt="Anonymous user avatar"/>
An anonymous user wrote:
<blockquote>Lorem ipsum dolor sed.</blockquote>
<a href="https://google.com/"><img src="edit.png" alt="Edit icon"/></a> /
<a href="https://google.com/"><img src="delete.png" alt="Delete icon"/></a>
```

To correct this:

- Remove the alt-text for the avatar.
- This image adds information for sighted users (an easily identifiable icon to show that the user is anonymous) but this information is already available in the text.

- Remove the "icon" from the alt-text for the icons.
- Knowing that this would be an icon if it were there does not help to convey its actual purpose.

```
<!-- Correct -->
<img src="anonymous.png" alt=""/> An anonymous
user wrote:
<blockquote>Lorem ipsum dolor sed.</blockquote>
<a href="https://google.com/"><img src="edit.png"
alt="Edit"/></a> /
<a href="https://google.com/"><img src="delete.png"
alt="Delete"/></a>
```

Responsive image using the srcset attribute

Using srcset with sizes:

```
<img sizes="(min-width: 1200px) 580px,
 (min-width: 640px) 48vw,
 98vw"
```

```
srcset="img/hello-300.jpg 300w,
 img/hello-600.jpg 600w,
 img/hello-900.jpg 900w,
 img/hello-1200.jpg 1200w"
 src="img/hello-900.jpg" alt="hello">
```

- sizes are like media queries, describing how much space the image takes of the viewport.
- if viewport is larger than 1200px, image is exactly 580px (for example our content is centered in container
- which is max 1200px wide.
- Image takes half of it minus margins).
- if viewport is between 640px and 1200px, image takes 48% of viewport (for example image scales with our page and takes half of viewport width minus margins).

img/hello-300.jpg is 300px wide,
img/hello-600.jpg is 600px wide,
img/hello-900.jpg is 900px wide,
img/hello-1200.jpg is 1200px wide

- src is always mandatory image source.
- In case of using with srcset, src will serve fallback image in case browser is not supporting srcset.
- Using srcset without sizes

```
<img src="img/hello-300.jpg" alt="hello"
 srcset="img/hello-300.jpg 1x,
 img/hello-600.jpg 2x,
 img/hello-1200.jpg 3x">
```

- srcset provides list of available images, with device-pixel ratio x descriptor.

- if device-pixel ratio is 1, use img/hello-300.jpg
- if device-pixel ratio is 2, use img/hello-600.jpg
- if device-pixel ratio is 3, use img/hello-1200.jpg
- src is always mandatory image source.
- In case of using with srcset, src will serve fallback image in case browser is not supporting srcset.

Responsive image using picture element Code:

```
<picture>
 <source media="(min-width: 600px)"
srcset="large_image.jpg">
 <source media="(min-width: 450px)"
srcset="small_image.jpg">
 <img src="default_image.jpg" style="width:auto;">
</picture>
```

Usage

- To display different images under different screen width, you must include all images using the source tag in a picture tag as shown in the above example.

Result

On screens with screen width >600px, it shows large_image.jpg

On screens with screen width >450px, it shows small_image.jpg

On screens with other screen width, it shows default_image.jpg

Image Maps

Tag/Attribute	Value
<img>	Below are the image map-specific attributes to use with <img>. Regular <img> attributes apply.

Usemap The name of the map with a hash symbol prepended to it. For example, for a map with name="map", the image should have usemap="#map".

<map>

name The name of the map to identify it. To be used with the image's usemap attribute.

<area> Below are <area>-specific attributes. When href is specified, making the <area> a link, <area> also supports all of the attributes of the anchor tag (<a>) except ping. See them at the MDN docs.

alt The alternate text to display if images are not supported. This is only necessary if href is also set on the <area>.

Coords The coordinates outlining the selectable area.

When shape="polygon", this should be set to a list of "x, y" pairs separated by commas (i.e., shape="polygon" coords="x1, y1, x2, y2, x3, y3, ...").

When shape="rectangle", this should be set to left, top, right, bottom. When shape="circle", this should be set to centerX, centerY, radius.

href The URL of the hyperlink, if specified. If it is omitted, then the <area> will not represent a hyperlink.

Shape The shape of the <area>. Can be set to default to select the entire image (no coords attribute necessary), circle or circ for a circle, rectangle or rect for a rectangle, and polygon or poly for a polygonal area specified by corner points.

Introduction to Image Maps:

Description:

- An image maps is an image with clickable areas that usually act as hyperlinks.
- The image is defined by the <img> tag, and the map is defined by a <map> tag with <area> tags to denote each clickable area.
- Use the usemap and name attributes to bind the image and the map.

Basic Example:

```
<!DOCTYPE html>
<html lang="en">
<body>
```

```
<img
src="http://jaced.com/blogpix/2007/trisquarecircle/002.gif" usemap="#shapes">
<map name="shapes">
 <area shape="polygon" coords="79,6,5,134,153,134">
 <area shape="rectangle" coords="177,6,306,134">
 <area shape="circle" coords="397,71,65">
 </map>
 </body>
</html>
```

Input Control Elements:

Parameter	Details
class	Indicates the Class of the input
id	Indicates the ID of the input

Type Identifies the type of input control to display. Acceptable values are hidden, text, tel, url, email, password, date, time, number, range, color, checkbox, radio, file, submit, image, reset, and button.Defaults to text if not specified, if the value is invalid, or if the browser does not support the type

specified.

name Indicates the name of the input

disabled Boolean value that indicates the input should be disabled. Disabled controls cannot be edited, are not sent on form submission, and cannot receive focus.

checked When the value of the type attribute is radio or checkbox, the presence of this Boolean attribute indicates that the control is selected by default; otherwise it is ignored.

multiple HTML5 Indicates multiple files or values can be passed (Applies only to file and email type inputs)

placeholder HTML5 A hint to the user of what can be entered in the control . The placeholder text must not contain carriage returns or line-feeds

autocomplete HTML5 Indicates whether the value of the control can be automatically completed by the browser.

Readonly Boolean value that indicates the input is not editable. Readonly controls are still sent on form submission, but will not receive focus. HTML5: This attribute is ignored when the value of type attribute is either set to hidden, range, color, checkbox, radio, file or button.

required HTML5 Indicates a value must be present or the element must be checked in order for the form to be submitted. alt An alternative text for images, in they are not displayed.

autofocus The <input> element should get the focus when page loads.

value Specifies the value of <input> element.

step The step attribute specifies the legal number intervals. It works with the following input types:

number, range, date, datetime-local, month, time and week.

- A key component of interactive web systems, input tags are HTML elements designed to take a specific form of input from users.
- Different types of input elements can regulate the data entered to fit a specified format and provide security to password entry

Text:

- The most basic input type and the default input if no type is specified.
- This input type defines a single-line text field with line-breaks automatically removed from the input value.
- All other characters can be entered into this.
- <input> elements are used within a <form> element to declare input controls that allow users to input data.

Syntax

<input type="text">

or (without specifying a type, using the default attribute):

<input>

- The default width of a text field input is 20 characters.
- This can be changed by specifying a value for the size attribute like this:

<input type="text" size="50">

- The size attribute is distinctly different than setting a width with CSS.
- Using a width defines a specific value (in number of pixel, percentage of the parent element, etc.) that the input must always be wide.
- Using the size calculates the amount of width to allocate based on the font being used and how wide the characters normally are.
- Note: Using the size attribute does not inherently limit the number of characters which can be entered into the box, only how wide the box is displayed.

- For limiting the length, see Input Validation.
- An input field only allows one line of text.
- If you need a multi-line text input for substantial amount of text, use a <textarea> element instead.

Checkbox and Radio Buttons:

Overview

- Checkboxes and radio buttons are written with the HTML tag <input>, and their behavior is defined in the HTML specification.
- The simplest checkbox or radio button is an <input> element with a type attribute of checkbox or radio,

respectively:

<input type="checkbox">

<input type="radio">

- A single stand-alone checkbox element is used for a single binary option such as a yes-or-no question.
- Checkboxes are independent, meaning the user may select as many choices as they would like in a group of checkboxes.
- In other words, checking one checkbox does not uncheck the other checkboxes in checkbox group.
- Radio buttons usually come in groups (if it's not grouped with another radio button, you probably meant to use a checkbox instead) identified by using the same name attribute on all buttons within that group.
- The selection of radio buttons are mutually exclusive, meaning the user may only select one choice from a group of radio buttons.

- When a radio button is checked, any other radio button with the same name that was previously checked becomes unchecked.

Example:

<input type="radio" name="color" id="red" value="#F00">

<input type="radio" name="color" id="green" value="#0F0">

<input type="radio" name="color" id="blue" value="#00F">

- When viewed, radio buttons appear as a circle (unchecked) or a filled circle (checked).
- Checkboxes appear as a square (unchecked) or a filled square (checked).
- Depending on the browser and operating system, the square sometimes has rounded corners.

Attributes

- checkboxes and radio buttons have a number of attributes to control their behavior

value

- Like any other input element, the value attribute specifies the string value to associate with the button in the event of form submission.
- However, checkboxes and radio buttons are special in that when the value is omitted, it defaults to on when submitted, rather than sending a blank value.
- The value attribute is not reflected in the button's appearance.

Checked:

- The checked attribute specifies the initial state of a checkbox or radio button.
- This is a boolean attribute and may be omitted.

- Each of these are valid, equivalent ways to define a checked radio button:

```
<input checked>
<input checked="">
<input checked="checked">
<input checked="ChEcKeD">
```

- The absence of the checked attribute is the only valid syntax for an unchecked button:

```
<input type="radio">
<input type="checkbox">
```

- When resetting a <form>, checkboxes and radio buttons revert to the state of their checked attribute.

Accessibility

Labels

To give context to the buttons and show users what each button is for, each of them should have a label.

- This can be done using a <label> element to wrap the button.
- Also, this makes the label clickable, so you select the corresponding button.

Example:

```
<label>
 <input type="radio" name="color" value="#F00">
 Red
</label>
```

or with a <label> element with a for attribute set to the id attribute of the button:

```
<input type="checkbox" name="color" value="#F00"
id="red">
<label for="red">Red</label>
```

Button Groups

- Since each radio button affects the others in the group, it is common to provide a label or context for the entire group of radio buttons.
- To provide a label for the entire group, the radio buttons should be included in a <fieldset> element with a <legend> element within it.

Example:

```
<fieldset>
 <legend>Theme color:</legend>
 <p>
 <input type="radio" name="color" id="red"
value="#F00">
 <label for="red">Red</label>
 </p>
```

```
<p>
 <input type="radio" name="color" id="green"
value="#0F0">
 <label for="green">Green</label>
 </p>
<p>
 <input type="radio" name="color" id="blue"
value="#00F">
 <label for="blue">Blue</label>
 </p>
</fieldset>
```

Input Validation

- HTML input validation is done automatically by the browser based on special attributes on the input element.

- It could partially or completely replace JavaScript input validation.
- This kind of validation can be circumvented by the

user via specially crafted HTTP requests, so it does not replace server-side input validation.

- The validation only occurs when attempting to submit the form, so all restricted inputs must be inside a form in order for validation to occur (unless you're using JavaScript).
- Keep in mind that inputs which are disabled or read-only will not trigger validation.
- Some newer input types (like email, url, tel, date and many more) are automatically validated and do not require your own validation constraints

Required:

Use the required attribute to indicate that a field must be completed in order to pass validation.

<input required>

Minimum / Maximum Length:

- Use the minlength and maxlength attributes to indicate length requirements.
- Most browsers will prevent the user from typing more than max characters into the box, preventing them from making their entry invalid even before they attempt submission.

<input minlength="3">

<input maxlength="15">

<input minlength="3" maxlength="15">

Specifying a range:

- Use min and max attributes to restrict the range of numbers a user can input into an input of type number or range

Marks:

```
<input type="number" size="6" name="marks" min="0" max="100" />
```

Subject Feedback:

```
<input type="range" size="2" name="feedback" min="1" max="5" />
```

Match a Pattern:

- For more control, use the pattern attribute to specify any regular expression that must be matched in order to pass validation.
- You can also specify a title, which is included in the validation message if the field doesn't pass.

<input pattern="\d*" title="Numbers only, please.">

Accept File Type:

- For input fields of type file, it is possible to accept only certain types of files, such as videos, images, audios,specific file extensions, or certain media types.
- For example:

<input type="file" accept="image/*" title="Only images are allowed">

Multiple values can be specified with a comma, e.g.:

<input type="file" accept="image/*,.rar,application/zip">

Note: Adding novalidate attribute to the form element or formnovalidate attribute to the submit button, prevents validation on form elements. For example:

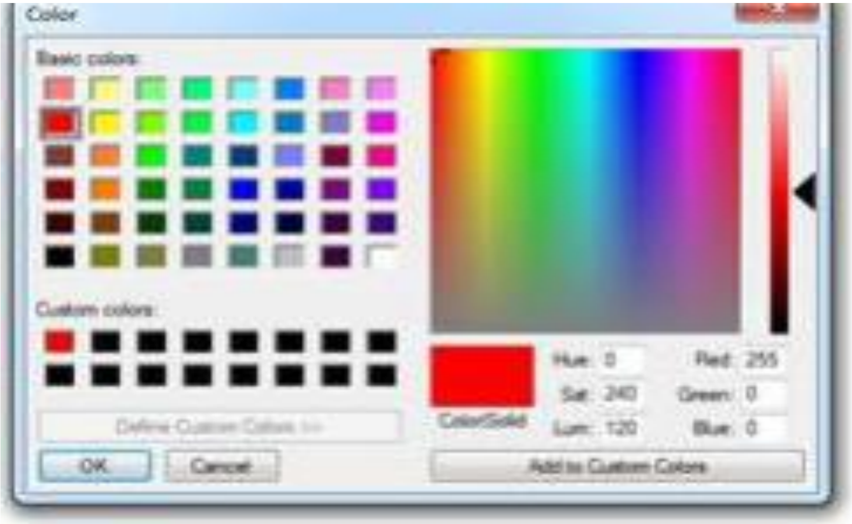
```
<form>
 <input type="text" name="name" required>
 <input type="email" name="email" required>
 <input pattern="\d*" name="number" required>
 <input type="submit" value="Publish"> <!-- form will be validated -->
 <input type="submit" value="Save" formnovalidate> <!-- form will NOT be validated -->
</form>
```

- The form has fields that are required for "publishing" the draft but aren’t required for "saving" the draft.

Color:

```
<input type="color" name="favcolor" value="#ff0000">
```

- In supporting browsers, the input element with a type attribute whose value is color creates a button-like control, with a color equal to the value of color attribute (defaults to black if value is not specified or is an invalid hexadecimal format).
- Clicking this button opens the operating system's color widget, which allows user to select a color



Hue: 0
Sat: 240
Lum: 120
Red: 255
Green: 0
Blue: 0

Summary:

- Using HTML with CSS
- Images
- Image Maps
- Input Control Elements

Thank You……….

If you have any quries please write to   info@uplatz.com".