



CSS

What is CSS?

- CSS stands for Cascading Style Sheets
 - Styles define how to display HTML elements
 - Styles are normally stored in Style Sheets
 - Styles were added to HTML 4.0 to solve a problem
 - External Style Sheets can save you a lot of work
 - External Style Sheets are stored in CSS files
 - Multiple style definitions will cascade into one

With CSS, your HTML documents can be displayed using different output styles

CSS Syntax

- The CSS syntax is made up of three parts: a selector, a property and a value Styles define how to display HTML elements

```
selector {property: value}
```

The selector is normally the HTML element/tag you wish to define, the property is the attribute you wish to change, and each property can take a value. The property and value are separated by a colon and surrounded by curly braces

CSS Syntax

```
body {color: black}
```

- If the value is multiple words, put quotes around the value

```
p {font-family: "sans serif"}
```

- If you wish to specify more than one property, you must separate each property with a semi-colon. The example below shows how to define a center aligned paragraph, with a red text color

```
p {text-align : center ; color : red}
```

CSS Syntax

```
p {text-align : center ; color : red}
```

- To make the style definitions more readable, you can describe one property on each line, like this

```
p  
{  
text-align: center;  
color: black;  
font-family: arial  
}
```

CSS Grouping

- You can group selectors. Separate each selector with a comma. In the example below we have grouped all the header elements. All header elements will be green

```
h1,h2,h3,h4,h5,h6  
{  
  color: green  
}
```

The class Selector

- With the class selector you can define different styles for the same type of HTML element.

```
p.right {text-align: right}
p.center {text-align: center}
```

- Use the class attribute in HTML document:

```
<p class="right">
  This paragraph will be right-aligned.
</p>
<p class="center">
  This paragraph will be center-aligned.
</p>
```

Note: Only one class attribute can be specified per HTML element!

The class Selector

- You can also omit the tag name in the selector to define a style that will be used by all HTML elements that have a certain class.

```
.center {text-align: center}
```

```
<h1 class="center">
```

```
This heading will be center-aligned
```

```
</h1>
```

```
<p class="center">
```

```
This paragraph will also be center-aligned.
```

```
</p>
```

Do **NOT** start a class name with a number! It will not work in Mozilla/Firefox.

The id Selector

- You can also define styles for HTML elements with the id selector. The id selector is defined as a #.

```
#green {color: green}
```

```
p#green {color: green}
```

```
<p id="green">Some text</p>
```

- The difference between an id and a class is that an id can be used to identify one element, whereas a class can be used to identify more than one.

Do **NOT** start an id name with a number! It will not work in Mozilla/Firefox.

Selector PRI

```
<html>
<head>
<style type="text/css">
  h3 {text-align: left; color: black }
  .right {text-align: right; color: blue }
  #first {text-align: center; color: red }
</style>
</head>
<body>
<h3 class="right" id="first"> How to display </h3>
</body>
</html>
```

How to display?

id Selector > class Selector > tag Selector

CSS Pseudo-class

- CSS pseudo-classes are used to add special effects to some selectors.
- Anchor Pseudo-classes

```
a:link { color: #FF0000 } /* unvisited link */  
a:visited { color: #00FF00 } /* visited link */  
a:hover { color: #FF00FF } /* mouse over link */  
a:active { color: #0000FF } /* selected link */
```

Note: a:hover MUST come after a:link and a:visited in the CSS definition in order to be effective!!

Note: a:active MUST come after a:hover in the CSS definition in order to be effective!!

Note: Pseudo-class names are not case-sensitive.

CSS Comments

- You can insert comments into CSS to explain your code, which can help you when you edit the source code at a later date. A CSS comment begins with "/*", and ends with "*/", like this:

```
/* This is a comment */  
p  
{  
text-align: center;  
/* This is another comment */  
color: black;  
font-family: arial  
}
```

CSS Background Properties

- The CSS background properties allow you to control the background color of an element, set an image as the background, repeat a background image vertically or horizontally, and position an image on a page.

```
background-color : transparent | color
```

```
background-image : none | url (url)
```

```
code { background-image: url("comet.jpg") }  
body { background-image: none; }
```

CSS Background Properties

```
background-attachment : scroll | fixed
```

```
html { background-image: url("anasazi.tif");  
      background-attachment: fixed }
```

```
background-repeat : repeat | no-repeat | repeat-x | repeat-y
```

```
p { background: url("images/aardvark.gif");  
    background-repeat: no-repeat }
```

background : a shorthand property for setting all background properties in one declaration.

```
background: #0f0 url("smile.gif") no-repeat fixed center;
```

CSS Text Properties

- The CSS text properties allow you to control the appearance of text. It is possible to change the color of a text, increase or decrease the space between characters in a text, align a text, decorate a text, indent the first line in a text, and more.

`color : color`

`letter-spacing : normal | length`

`word-spacing : normal | length`

`text-indent : length | %`

CSS Text Properties

```
text-align : left | right | center
```

```
text-decoration: none | underline | blink |  
                overline | line-through
```

```
text-transform : none | capitalize | uppercase |  
                lowercase
```

```
div { text-decoration : underline overline; }
```


CSS Units

- CSS Units include *Measurements* and Colors

Unit	Description
%	percentage
in	inch
cm	centimeter
mm	millimeter
em	1 em = the current font size of the current element
ex	1 ex = the x-height of a font = half the font-size
pt	point (1 pt is the same as 1/72 inch)
pc	pica (1 pc is the same as 12 points)
px	pixels (a dot on the computer screen)

Units conversion: 1in = 2.54cm = 25.4 mm = 72pt

CSS Font Properties

- The CSS font properties allow you to change the font family, boldness, size, and the style of a text.

line-height : normal | *length*

font-variant : normal | small-caps

font-size : xx-small | x-small | small | medium |
large | x-large | xx-large | *length* | %

font-weight : normal | bold | bolder | lighter
| 100 | 200 | 300 | | 900

Normal = 400

Bold = 700

CSS Font Properties

Note: In CSS1 fonts are identified by a font name. If a browser does not support the specified font, it will use a default font.

```
font-family : family-name | generic-family
```

```
font-style : normal | italic | oblique
```

```
body { font-family : "Lucida Grande", Verdana,  
Lucida, Arial, Helvetica, 宋体, sans-serif; }
```

font : a shorthand property for setting all of the properties for a font in one declaration.

```
p{ font: italic small-caps 900 12px arial; }
```

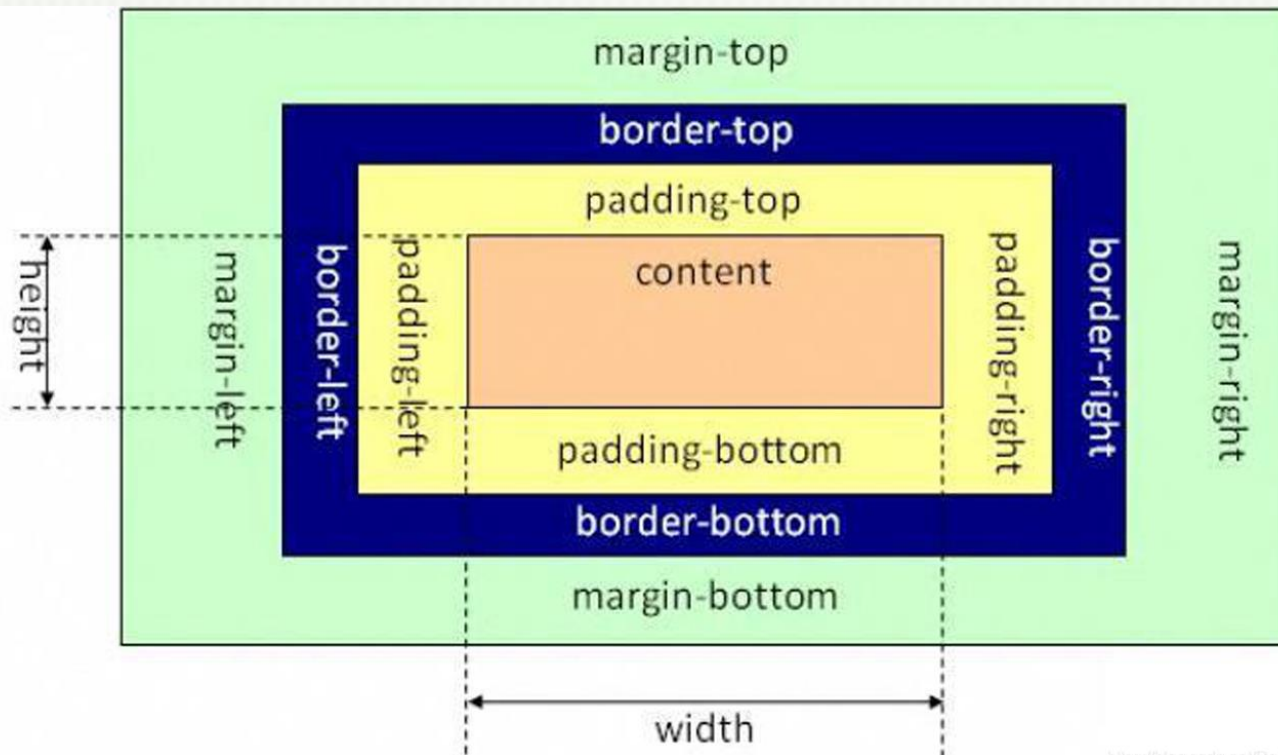
Box Model

Margin box (边界盒)

Border box (边框盒)

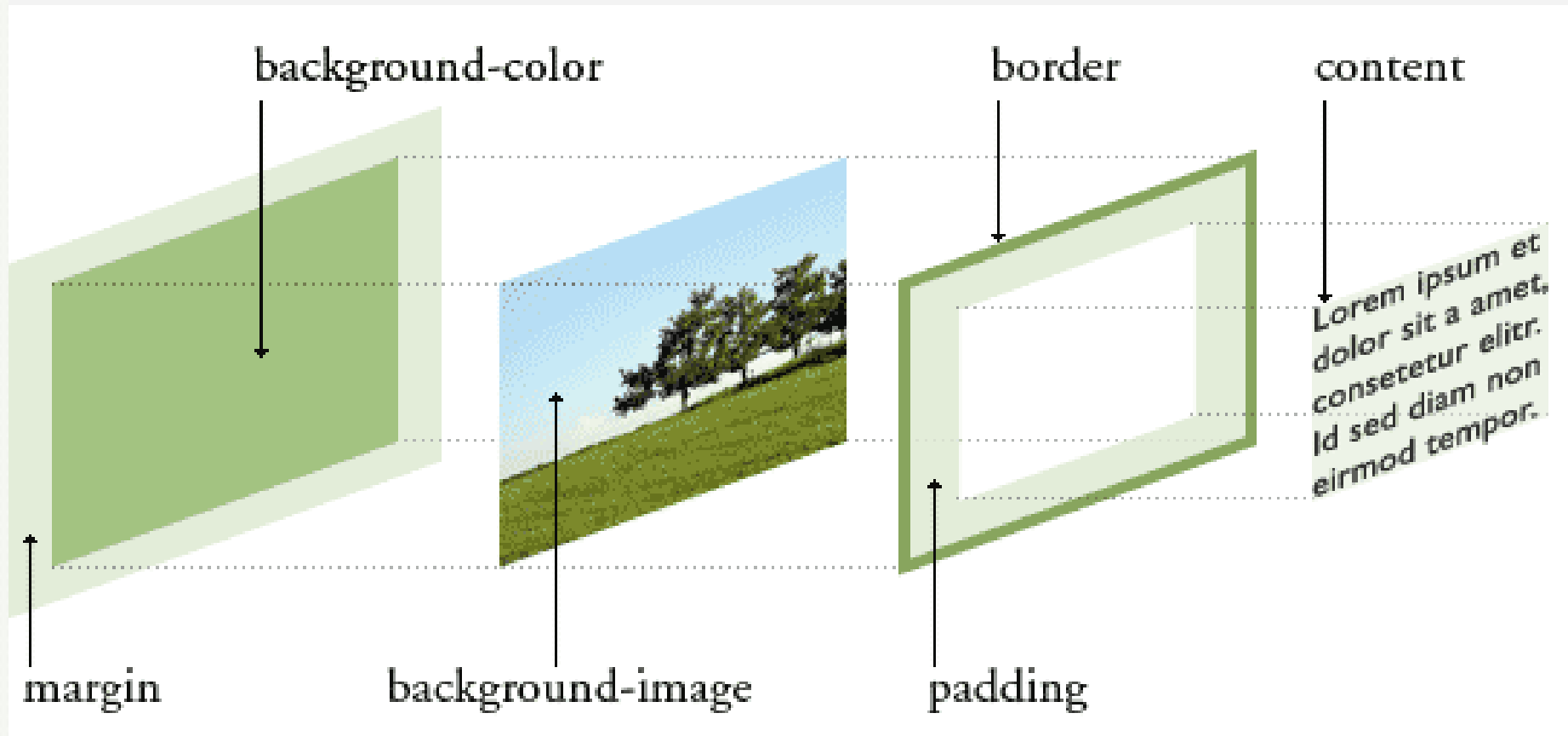
Padding box (补白盒)

Element box (元素盒)

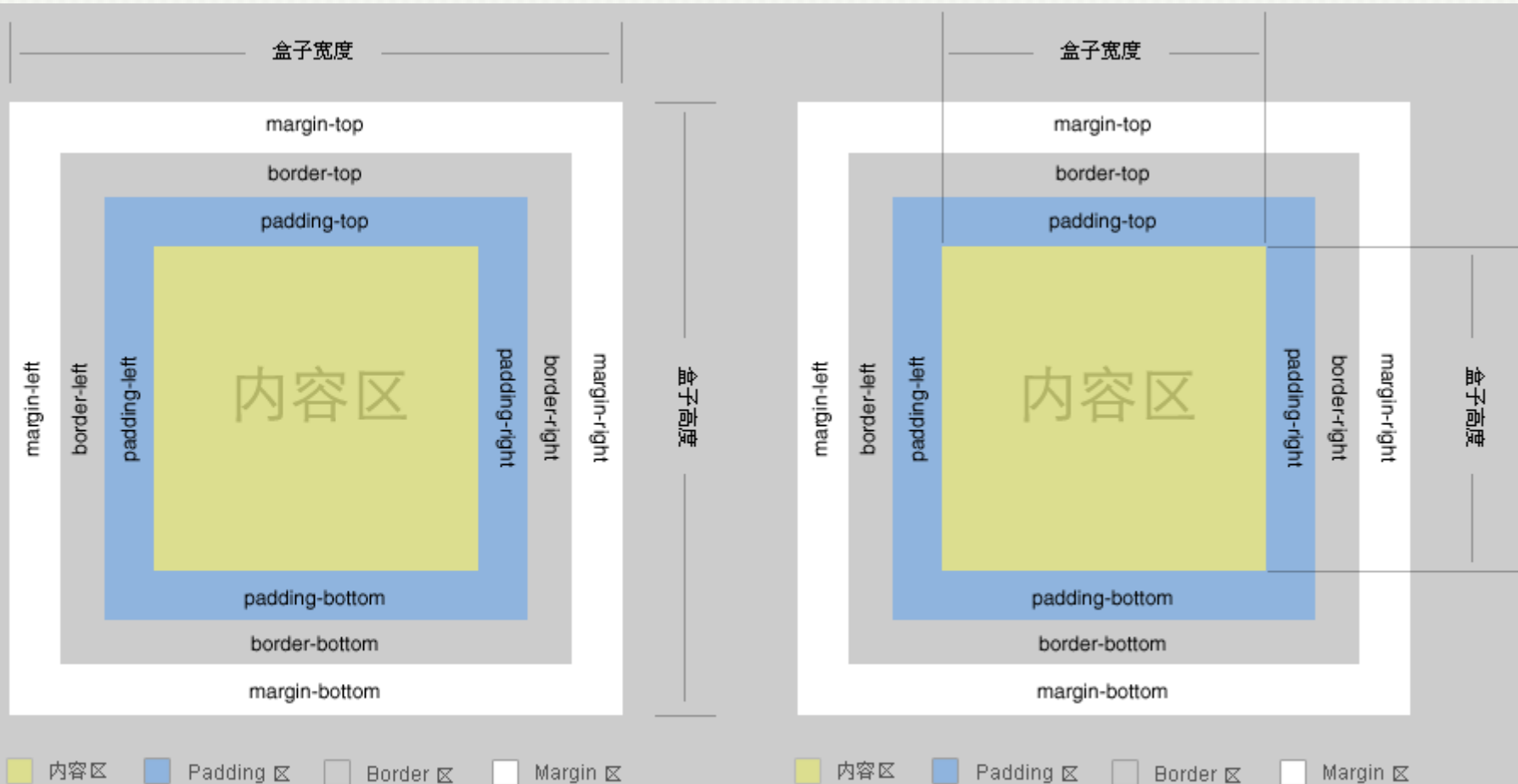


Box Model

THE BOX MODEL



IE6 vs W3C (FireFox, Chrome, Opera)



CSS Margin Properties

- The CSS margin properties define the space around elements. It is possible to use negative values to overlap content. The top, right, bottom, and left margin can be changed independently using separate properties. A shorthand margin property can also be used to change all of the margins at once.

```
margin-bottom : auto | % | length
```

```
margin : margin-top | margin-right | margin-bottom  
| margin-left
```

CSS Border Properties

- The CSS border properties allow you to specify the style and color of an element's border. In HTML we use tables to create borders around a text, but with the CSS border properties we can create borders with nice effects, and it can be applied to any element. (top right bottom left)

```
border-bottom-color : border-color
```

```
border-bottom-style : border-style
```

```
border-bottom-width : thin | medium | thick | length
```


CSS Border Properties

`border-bottom` : a shorthand property for setting all of the properties for the bottom border in one declaration

```
border-bottom : border-bottom-width | border-style | border-color
```

```
border-color : color
```

```
border-style : none / hidden / dotted / dashed / solid / double /  
groove / ridge / inset / outset
```

```
div { border-bottom: medium solid #ff0000 }
```

CSS Border Properties

`border` : a shorthand property for setting all of the properties for the four borders in one declaration

```
border : border-width | border-style | border-color
```

```
span { border : medium double rgb(250,0,255) }
```

```
div {border : thick double yellow; }
```

CSS Padding Properties

- The CSS padding properties define the space between the element border and the element content. Negative values are not allowed. The top, right, bottom, and left padding can be changed independently using separate properties. A shorthand padding property is also created to control multiple sides at once.

```
padding -bottom : auto | % | length
```

```
padding : padding-top | padding-right | padding-bottom | padding-left
```

CSS Positioning Properties

- The CSS positioning properties allow you to specify the left, right, top, and bottom position of an element. It also allows you to set the shape of an element, place an element behind another, and to specify what should happen when an element's content is too big to fit in a specified area.

`bottom : auto | % | length`

`position : static | relative | absolute | fixed`

`z-index : auto | number`

CSS Classification Properties

- The CSS classification properties allow you to control how to display an element, set where an image will appear in another element, position an element relative to its normal position, position an element using an absolute value, and how to control the visibility of an element.

`display : none | block | inline |`

`visibility : hidden | visible | collapse`

Note: the different between “display : none” and “visibility : hidden”.

Example

```
margin: 0px 10px 0px 10px ;  
margin: 1em 10em;  
margin : 10px
```

```
#navigation  
{ position: absolute;  
  top: 0px;  
  left: 0px;  
  width: 10em;  
}
```

Example

```
#sample
{
margin: 10px 10px 10px 10px;
padding:20px 10px 10px 20px;
border-right: #CCC 2px solid;
border-bottom: #CCC 2px solid;
border-left: #CCC 2px solid;
border-top: #CCC 2px solid;
background: url(images/bg_poem.jpg) #fff no-repeat
right bottom;
color: #666;
text-align: center;
line-height:150%; width:60%;
}
```


How to Insert a Style Sheet

- **External Style Sheet**

An external style sheet is ideal when the style is applied to many pages. With an external style sheet, you can change the look of an entire Web site by changing one file. Each page must link to the style sheet using the `<link>` tag. The `<link>` tag goes inside the head section

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

Do **NOT** leave spaces between the property value and the units! If you use "margin-left: 20 px" instead of "margin-left: 20px" it will only work properly in IE6 but it will not work in Mozilla/Firefox or Netscape.

How to Insert a Style Sheet

- **Internal Style Sheet**

An internal style sheet should be used when a single document has a unique style. You define internal styles in the head section by using the `<style>` tag

```
<head>
  <style type="text/css">
    p {margin-left: 20px}
    body {background-image: url("images/back40.gif")}
  </style>
</head>
```

How to Insert a Style Sheet

Note: A browser normally ignores unknown tags. This means that an old browser that does not support styles, will ignore the <style> tag, but the content of the <style> tag will be displayed on the page. It is possible to prevent an old browser from displaying the content by hiding it in the HTML comment element

```
<head>
  <style type="text/css">
    <!--
    p {margin-left: 20px}
    body {background-image: url("images/back40.gif")}
    -->
  </style>
</head>
```

How to Insert a Style Sheet

- **Inline Styles**

An inline style loses many of the advantages of style sheets by mixing content with presentation. Use this method sparingly, such as when a style is to be applied to a single occurrence of an element.

```
<p style="color: sienna; margin-left: 20px">  
  This is a paragraph  
</p>
```

- To use inline styles you use the style attribute in the relevant tag. The style attribute can contain any CSS property.

- **Multiple Style Sheets**

If some properties have been set for the same selector in different style sheets, the values will be inherited from the more specific style sheet.

```
mystyle.css
```

```
h3 {color: red; text-align: right; font-size: 8pt }
```

```
<head>
```

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

```
  <style type="text/css">
```

```
    h3 {text-align: left; font-size: 20pt }
```

```
  </style>
```

```
</head>
```

```
<body>
```

```
<h3> How to display the heading? </h3>
```

```
</body>
```

How to display?

Cascading Order

Generally speaking we can say that all the styles will "cascade" into a new "virtual" style sheet by the following rules, where number four has the highest priority:

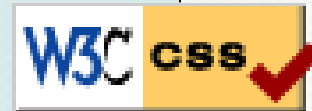
- Browser default
- External style sheet
- Internal style sheet (inside the <head> tag)
- Inline style (inside an HTML element)

The Web Site Was Validated

- After that, all pages were validated against the official W3C DTD with this link:

<http://jigsaw.w3.org/css-validator/validator-uri.html>

```
<p>  
<a href="http://jigsaw.w3.org/css-validator/">  
</a>  
</p>
```



Homework 2

- 1.what is XHTML?
- 2.what is color?
- 3.what is Website Development Process?
- Translate
- Design is a process that involves creating and communicating ideas or concepts through form. Designers' primary role is to create perception in their viewer's mind through form. This perception is based on the gestalt theory, which states that the whole is greater than the sum of its parts, meaning that viewers perceive the entire design before they notice the details that make up the whole.



The End