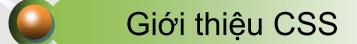


Trình bày: Nguyễn Phú Trường



NỘI DUNG







Cách dùng CSS

Các CSS cơ bản

Các CSS nâng cao



GROUPING SELECTORS



In style sheets there are often elements with the same style.

```
h1
  color:green;
  h2
  color:green;
  color:green;
```



GROUPING SELECTORS



- To minimize the code, you can group selectors.
- Separate each selector with a comma.
- In the example below we have grouped the selectors from the code above:

```
h1,h2,p
{
  color:green;
}
```



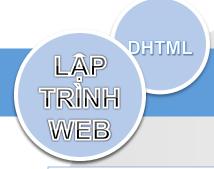
NESTING SELECTORS



- It is possible to apply a style for a selector within a selector.
- In the example below, one style is specified for all p elements, and a separate style is specified for p elements nested within the "important" class:

Example

```
color:blue;
text-align:center;
.important
background-color:blue;
.important p
color:white;
```



CSS DIMENSION



- The CSS dimension properties allow you to control the height and width of an element.
 - Set the height of elements
 This example demonstrates how to set the height of different elements.
 - Set the height of an image using percent
 This example demonstrates how to set the height of an element using a percent value.
 - Set the width of an element using a pixel value
 This example demonstrates how to set the width of an element using a pixel value.
 - Set the maximum height of an element
 This example demonstrates how to set the maximum height of an element.
 - Set the maximum width of an element using percent
 This example demonstrates how to set the maximum width of an element using a percent value.
 - Set the minimum height of an element
 This example demonstrates how to set the minimum height of an element.
 - Set the minimum width of an element using a pixel value
 This example demonstrates how to set the minimum width of an element using a pixel value.



ALL CSS DIMENSION PROPERTIES



Property	Description	Values CSS
<u>height</u>	Sets the height of an element	auto 1 <i>length</i> % inherit
max-height	Sets the maximum height of an element	none 2 <i>length</i> % inherit
max-width	Sets the maximum width of an element	none 2 <i>length</i> % inherit
min-height	Sets the minimum height of an element	length 2 % inherit
<u>min-width</u>	Sets the minimum width of an element	<i>length</i> 2 % inherit
<u>width</u>	Sets the width of an element	auto 1 <i>length</i> % inherit



CSS DISPLAY AND VISIBILITY



- The CSS classification properties specifies if/how an element is to be displayed, and to control the visibility of an element.
- Hiding an Element display:none or visibility:hidden
 - Hiding an element can be done by setting the display property to "none" or the visibility property to "hidden". However, notice that these two methods produce different results:
 - visibility:hidden hides an element, but it will still take up the same space as before. The element will be hidden, but still affect the layout.



CSS DISPLAY AND VISIBILITY



Example

- h1.hidden {visibility:hidden}
- display:none hides an element, and it will not take up any space. The element will be hidden, and the page will be displayed as the element is not there:

Example

h1.hidden {display:none}



CSS DISPLAY - BLOCK AND INLINE ELEMENTS



- A block element is an element that takes up the full width available, and has a line break before and after it.
- Examples of block elements:
 - <h1>

 - <div>
- An inline element only takes up as much width as necessary, and does not force line breaks.
- Examples of inline elements:
 -
 - <a>>



CHANGING HOW AN ELEMENT IS DISPLAYED



- Changing an inline element to a block element, or vice versa, can be useful for making the page look a specific way, and still follow web standards.
- The following example displays list items as inline elements:

Example

- li {display:inline}
- The following example displays span elements as block elements:

Example

span {display:block}