

### CSS3 Introduction

- CSS3 is based on CSS 2.1 specification and is completely backward compatible with CSS2.
- The old specification of CSS2.1 was becoming too large and complex to update as one unit, hence development of CSS3 specification is split up into Modules. Modules can evolve independent of each other.
- It adds new features for developers to solve a number of design problems without the need for non-semantic markup, complex scripting, or extra images.
- Main Features that are included in CSS3 include
  - Support for additional selectors.
  - Drop shadows,
  - Rounded corners,
  - Multiple backgrounds,
  - Gradients,
  - Animation,
  - Transparency and many more...
- The CSS3 specification is still under development by W3C. However, many of the new CSS3 properties have been implemented in modern browsers.
  - New Selectors
  - Backgrounds and Borders
  - Gradient Effects
  - Text Effects and Fonts
  - Transformations
  - Transitions
  - Animations
  - Multiple Column Layout.

### CSS 3 New Selectors

In CSS, pattern matching rules determine which style rules apply to elements in the document tree. These patterns, called selectors, may range from simple element names to rich contextual patterns.

Using selectors Styles can be applied selectively.

Selector	Example	Description
<b>[attribute^=value]</b>	a[href^="https"]	Selects every <a> element whose href attribute value begins with "https"
<b>[attribute\$=value]</b>	a[href\$=".pdf"]	Selects every <a> element whose href attribute value ends with ".pdf"
<b>[attribute*=value]</b>	a[href*="deccansoft"]	Selects every <a> element whose href attribute value contains the substring "deccansoft"
<b>E1~E2</b>	p~ul	Selects every <ul> element that are preceded by a <p> element
<b>E:checked</b>	input:checked	Selects every checked <input> element

<b>E:disabled</b>	input:disabled	Selects every disabled <input> element
<b>E:enabled</b>	input:enabled	Selects every enabled <input> element
<b>E:empty</b>	p:empty	Selects every <p> element that has no children (including text nodes)
<b>E:last-child</b>	p:last-child	Selects every <p> element that is the last child of its parent
<b>E:only-child</b>	p:only-child	Selects every <p> element that is the only child of its parent
<b>E:nth-child(n)</b>	p:nth-child(2)	Selects every <p> element that is the second child of its parent
<b>E:nth-last-child(n)</b>	p:nth-last-child(2)	Selects every <p> element that is the second last child of its parent.
<b>E:first-of-type</b>	p:first-of-type	Selects every <p> element that is the first <p> element of its parent
<b>E:last-of-type</b>	p:last-of-type	Selects every <p> element that is the last <p> element of its parent
<b>E:only-of-type</b>	p:only-of-type	Selects every <p> element that is the only <p> element of its parent
<b>E:nth-of-type(n)</b>	p:nth-of-type(2)	Selects every <p> element that is the second <p> element of its parent
<b>E:nth-last-of-type(n)</b>	p:nth-last-of-type(2)	Selects every <p> element that is the second last <p> element of its parent.
<b>E:in-range</b>	input:in-range	Selects input elements with a value within a specified range
<b>E:out-of-range</b>	input:out-of-range	Selects input elements with a value outside a specified range
<b>E:invalid</b>	input:invalid	Selects all input elements with an invalid value
<b>E:valid</b>	input:valid	Selects all input elements with a valid value
<b>E:not(selector)</b>	:not(p)	Selects every element that is not a <p> element
<b>E:optional</b>	input:optional	Selects input elements with no "required" attribute
<b>E:read-only</b>	input:read-only	Selects input elements with the "readonly" attribute specified
<b>E:read-write</b>	input:read-write	Selects input elements with the "readonly" attribute NOT specified
<b>E:required</b>	input:required	Selects input elements with the "required" attribute specified
<b>E:root</b>	:root	Selects the document's root element
<b>E:target</b>	#news:target	Selects the current active #news element (clicked on a URL containing that anchor name)