Side Trip: XHTML

Using XHTML

XHTML

- Extensible Hypertext Markup Language
- Supposedly the replacement for HTML.
- Reformulation of HTML in XML.
  - Still like HTML with the rules of XML (on wellformedness).

Required Elements in XHTML

- <!DOCTYPE>
  - One of three doctypes.
- <html>
  - Root element
- <head>
- <title>
- <body>

Differences from HTML

- Always well-formed.
- Some elements are required elements.
- Some elements in HTML that are formatting, e.g. <b>, <i>, etc. are avoided (Even absent) in XHTML.
  - Presentation rules should be defined in CSS.
- All tags, attributes and even attribute values are in lowercase characters.
- Some attributes of some elements are required
  - OTHERWISE your XHTML document is NOT valid

The DOCTYPES

- XHTML Transitional
  - Compatibility with old browsers that does not support CSS.
- XHTML Strict
  - Free from presentational elements.
- XHTML Frameset
  - When using <frameset> & <frame>

XHTML Template

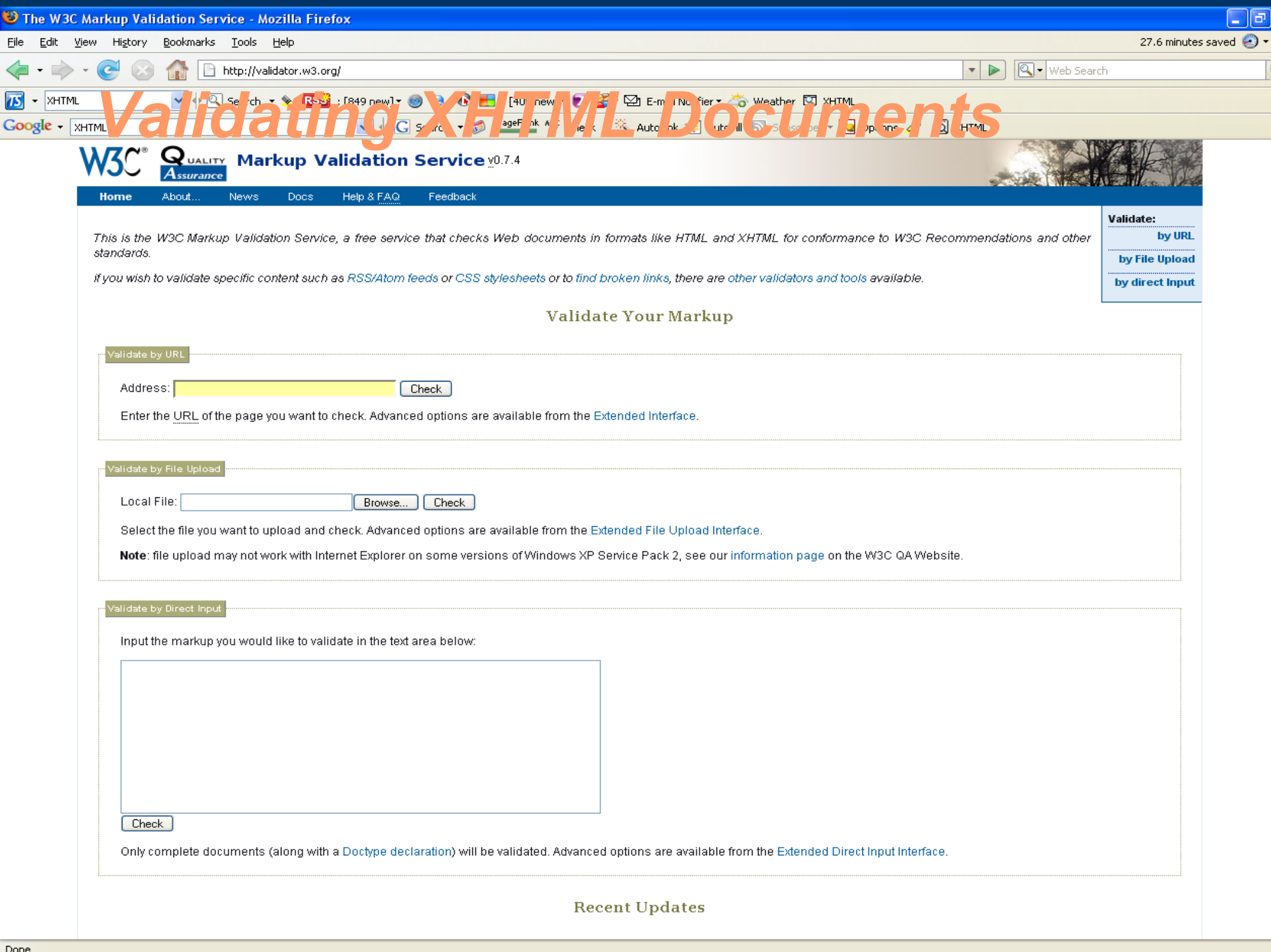
```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
  transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Title goes here</title>
</head>

<body>
</body>

</html>
```

Where It's Going

- XHTML 1.1 : Modularized XHTML
- XHTML 2.0 : Still In Development
  - HTML Forms to be replaced by XForms
  - HTML frames to be replaced by XFrames
  - Any element can become a hyperlink
  - Etc..


The W3C Markup Validation Service - Mozilla Firefox
File Edit View History Bookmarks Tools Help
27.6 minutes saved
http://validator.w3.org/
Web Search
Validating XHTML Documents
Markup Validation Service v0.7.4
Home About... News Docs Help & FAQ Feedback
Validate:
by URL
by File Upload
by direct Input
This is the W3C Markup Validation Service, a free service that checks Web documents in formats like HTML and XHTML for conformance to W3C Recommendations and other standards.
if you wish to validate specific content such as RSS/Atom feeds or CSS stylesheets or to find broken links, there are other validators and tools available.
Validate Your Markup
Validate by URL
Address:
Check
Enter the URL of the page you want to check. Advanced options are available from the Extended Interface.
Validate by File Upload
Local File:
Browse...
Check
Select the file you want to upload and check. Advanced options are available from the Extended File Upload Interface.
Note: file upload may not work with Internet Explorer on some versions of Windows XP Service Pack 2, see our information page on the W3C QA Website.
Validate by Direct Input
Input the markup you would like to validate in the text area below:
Check
Only complete documents (along with a Doctype declaration) will be validated. Advanced options are available from the Extended Direct Input Interface.
Recent Updates
Done

Enter the URL of the page you want to check. Advanced options are available from the Extended Interface.

Validate by File Upload

Local File: Browse... Check

Select the file you want to upload and check. Advanced options are available from the Extended File Upload Interface.

Note: file upload may not work with Internet Explorer on some versions of Windows XP Service Pack 2, see our information page on the W3C QA Website.

Validate by Direct Input

Input the markup you would like to validate in the text area below:

```
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Title goes here</title>
</head>

<body>
</body>

</html>
```

Check

Only complete documents (along with a Doctype declaration) will be validated. Advanced options are available from the Extended Direct Input Interface.

Recent Updates

W3C Quality Assurance Markup Validation Service v0.7.4

Result: Passed validation
File: upload://Form Submission
Encoding: utf-8
Doctype: XHTML 1.0 Transitional
Root Namespace: http://www.w3.org/1999/xhtml

Note: The Validator XML support has some limitations.

This Page Is Valid XHTML 1.0 Transitional!

Tip Of The Day: **Don't use "click here" as link text!**

The uploaded document "upload://Form Submission" was checked and found to be valid XHTML 1.0 Transitional. This means that the resource in question identified itself as "XHTML 1.0 Transitional" and that we successfully performed a formal validation using an SGML or XML Parser (depending on the markup language used).

To show your readers that you have taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the HTML you could use to add this icon to your Web page:

```
<p>
  <a href="http://validator.w3.org/check?uri=referer"><img
      src="http://www.w3.org/Icons/valid-xhtml10"
      alt="Valid XHTML 1.0 Transitional" height="31" width="88" /></a>
</p>
```

If you like, you can download a copy of this image (in PNG or GIF format) to keep in your local web directory, and change the HTML fragment above to reference your local image rather than the one on this server.

A full list of icons, with links to alternate formats and colors, is also available.

Markup Validation Service v0.7.4

Jump To:
Results

Result: Failed validation, 1 error
File: upload://Form Submission
Encoding: utf-8
Doctype: XHTML 1.0 Transitional
Root Namespace: http://www.w3.org/1999/xhtml

Note: The Validator XML support has some limitations.

This page is **not** Valid XHTML 1.0 Transitional!

Below are the results of checking this document for XML well-formedness and validity.

1. Error *Line 7 column 6*: **end tag for "head" which is not finished.**

```
</head>
```

Most likely, You nested tags and closed them in the wrong order. For example ... is not acceptable, as must be closed before . Acceptable nesting is: ...

Another possibility is that you used an element which requires a child element that you did not include. Hence the parent element is "not finished", not complete. For instance, <head> generally requires a <title>, lists (ul, ol, dl) require list items (li, or dt, dd), and so on.

?

Presentation of (XML) Data for the Web

XML: Separation of Data From Presentation

- XML Documents are data-centric.
  - Markup of data only.
- XML Documents should be free from presentation/styling/display/formatting elements.
  - No elements included as to how other elements should be formatted
  - This would keep any XML portable and display-“platform” independent.

Presentation of XML Data using Stylesheet

- A Stylesheet
  - A specification on how to display the data.
- Cascading Stylesheets (CSS)
  - Added to HTML to solve rendering consistency problems
  - With CSS We can specify how to exactly display HTML elements.
  - CSS can be applied to XML documents as well.

CSS: Attaching to HTML

- External Stylesheets
  - CSS **rules** written in separate file
  - Linked via

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

<!-- styles.css is the URL of the
     CSS file -->
```

CSS: Attaching to HTML

- Internal Stylesheets
  - Style rules written inside <style> inside <head>

```
<html>
   <head>
      <style>

      //style rules here
      </style>
   </head>
. . .
```

CSS: Attaching to HTML

- Inline Style
  - Style rules written inside in style attribute in presentable elements

```
<p style="font:Arial"> ... </p>
. . .
```

CSS Syntax

```
selector { property : value;
           property2 : value2
               }
```

Example:

```
p { font-family : “sans serif” }
```

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

Specifying Selectors

- Element name (e.g. <p>, <a> etc... }

```
a { text-decoration: none }
//represents the <a> elements
```

- Element Class (specified by class attribute)

```
.reddish { color:red }
div.reddish { background-color : red }

Corresponds to the ff:
<a class=”reddish”>
<div class=”reddish”>
```

Specifying Selectors

- Element ID (specified by id attribute

```
#unique { border : 1px solid black }
a#unico { display: block }

Corresponds to
<p id=”unique”>
<a id=”unico”>
```

Specifying Selectors

- Grouped Selector

```
h1,h2,h3,h4,h5,h6,h7 {
   color: blue
}
```

- Heirarchial Selector

```
p a {
   text-decoration: none;
   color red;
}
// applies to: <a> elems inside <p>
```

Specifying Selectors

- Element with Specific Attribute

```
input[type=”text”] {
     border: 1px solid red
}
/* applies to <input type=”text”>
elems, IE sucks on this
dunno about IE 7 though */
```

Specifying Selectors

- Pseudo-classes

```
a:link { color: #FFFF00 }
a:hover { color: rgb(123,234,54) }
a:visited { color: salmon }
```

CSS Properties and Their Values

- Background Properties
- Text
- Font
- Border
- Margin
- Padding
- List
- Dimension
- Classification
- Positioning
- Print (CSS2 Media)
- Aural (CSS2 Media)

CSS Properties and Their Values

- Consult a CSS Reference for the different property names and their possible values.

CSS Inheritance

- Most CSS properties are inheritable from more general selectors to more specific selectors
- When two properties conflict, the more specific rule is applied, otherwise the rules combine.

Observe:

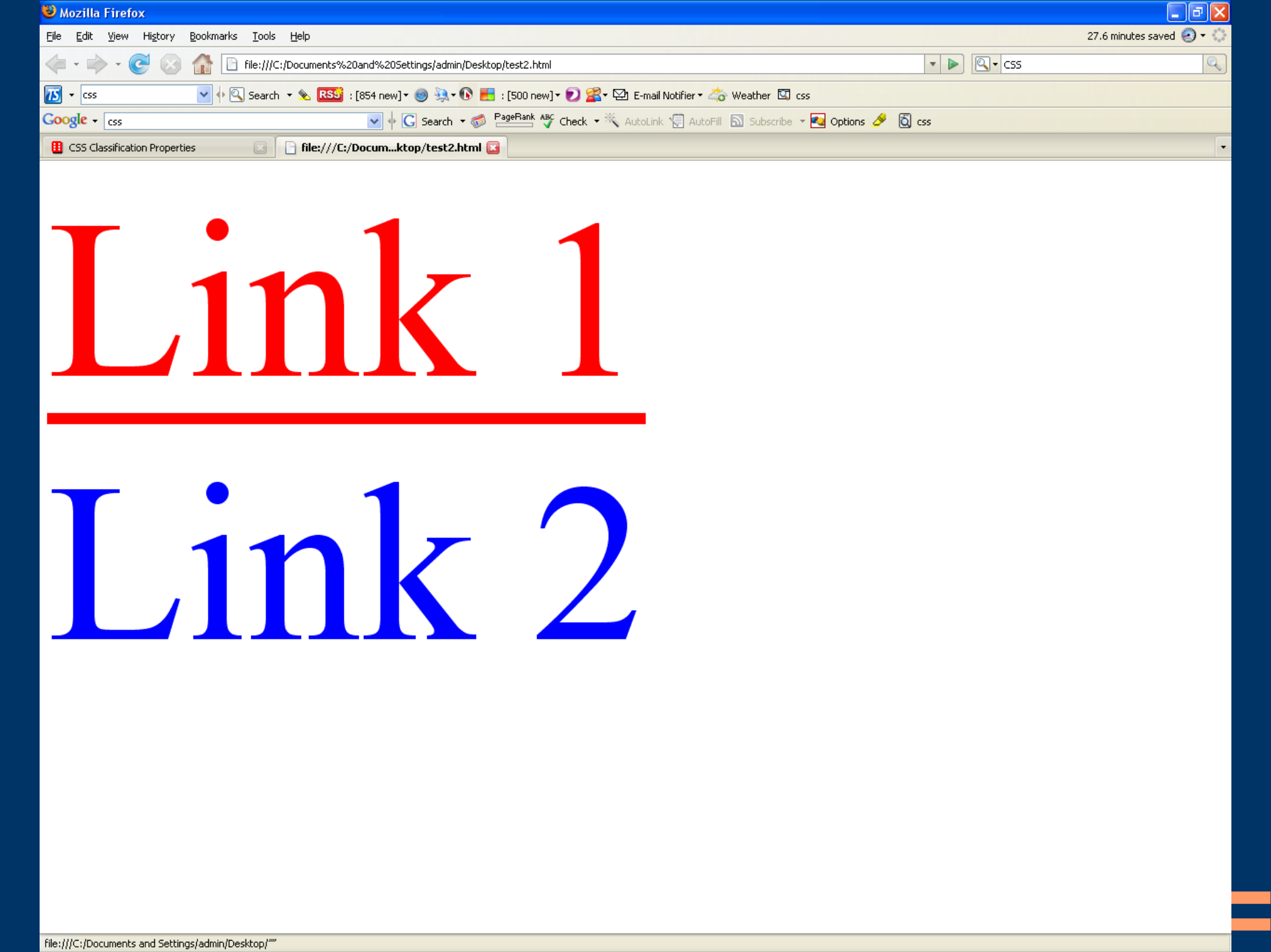
```
/* CSS Rules */

a {
    font-family: “sans
       serif”;
    color:red
    }
div a {
    color:blue;
    text-decoration:none;
}
```

```
<!-- HTML Markup-->

<a href=””>Link 1</a>

<div>
<a href=””>Link 2</a>
</div>
```


Mozilla Firefox
File Edit View History Bookmarks Tools Help
27.6 minutes saved
file:///C:/Documents%20and%20Settings/admin/Desktop/test2.html
CSS Classification Properties
file:///C:/Docum...ktop/test2.html
Link 1
Link 2
file:///C:/Documents and Settings/admin/Desktop/""

CSS Rule application:

- Browser Default
- External CSS
- Internal CSS
- Inline

In terms of selector, the rules of the most specific selector that applies to an element overwrites any less specific declarations.

XML + CSS = Web Page

- With any XML document, we can apply CSS to display it as web page.
- The selectors will be the elements of the XML documents.