FEU INSTITUTE OF TECHNOLOGY
COLLEGE OF ENGINEERING • COLLEGE OF COMPUTER STUDIES

Web Design Fundamentals / Basic Web Design

HTML 5

HTML5

A markup language used for structuring and presenting content on the World Wide Web.

It was finalized, and published, on 28 October 2014 by the World Wide Web Consortium (W3C). This is the fifth revision of the HTML standard since the inception of the World Wide Web.

HTML5 Semantic Elements

A semantic element clearly describes its meaning to both the browser and the developer.

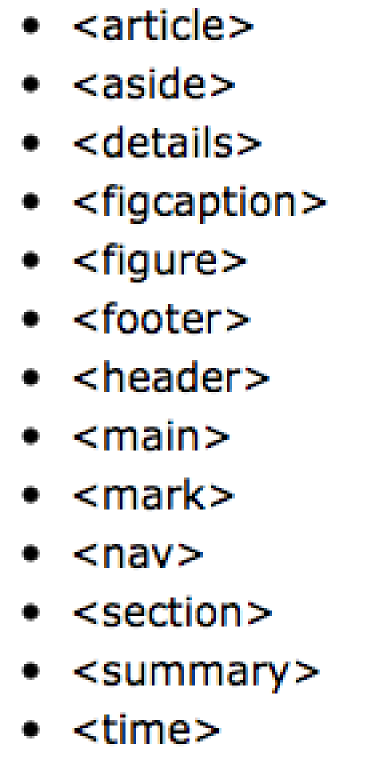
Examples of non-semantic elements:

<div> and <span> - Tells nothing about its content.

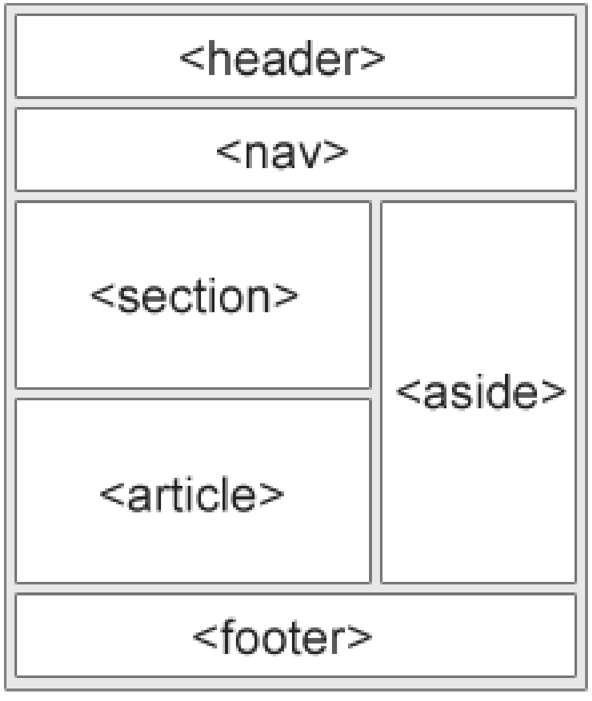
Examples of semantic elements:

<form>, <table>, and <img> - Clearly defines its content.

HTML5 Semantic Elements



- <article>
- <aside>
- <details>
- <figcaption>
- <figure>
- <footer>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <summary>
- <time>



<header>

<nav>

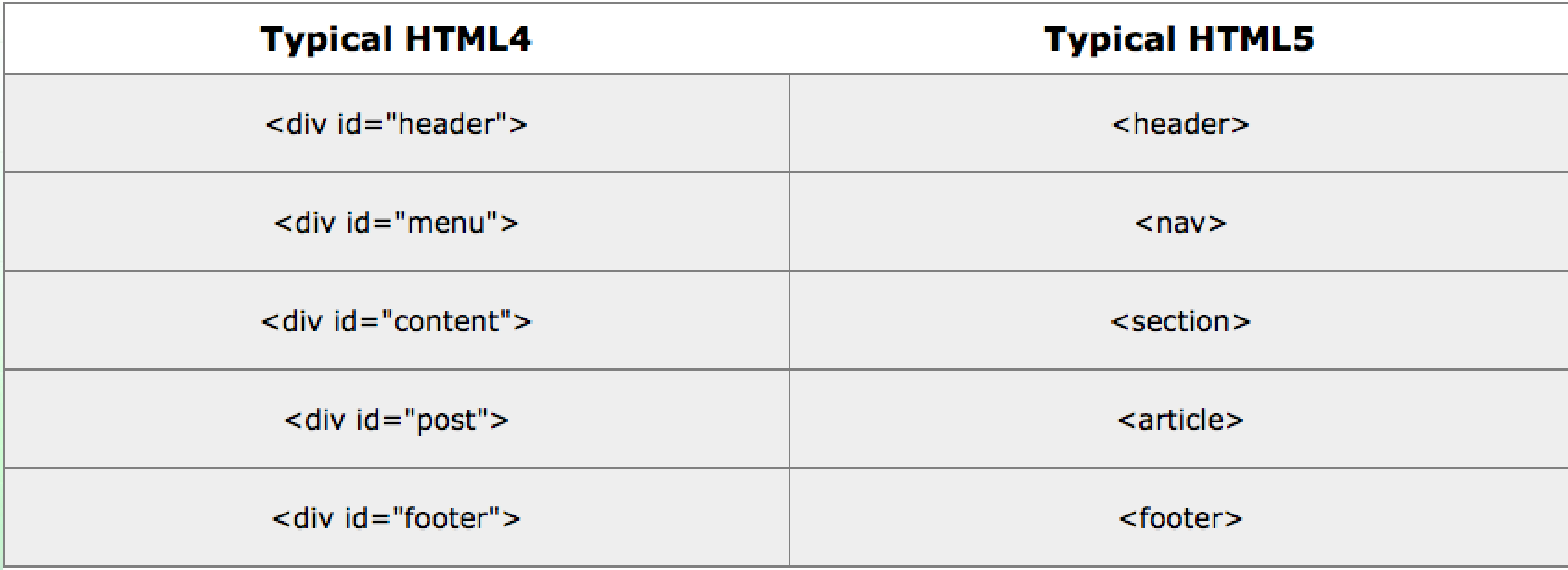
<section>

<article>

<aside>

<footer>

Migration from HTML4 to HTML5



Typical HTML4	Typical HTML5
<div id="header">	<header>
<div id="menu">	<nav>
<div id="content">	<section>
<div id="post">	<article>
<div id="footer">	<footer>

Migration

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

Migration to

<!DOCTYPE html>

<meta http-equiv="Content-Type" content="text/html;charset=utf-8">

Migration to

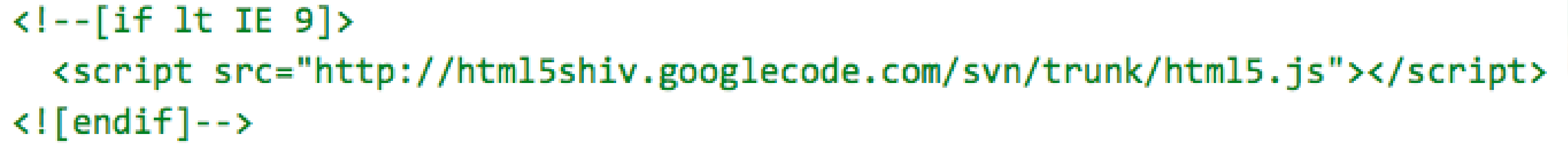
<meta charset="utf-8">

SHIV

HTML5 semantic elements are supported in all modern browsers.

But some older browsers should be taught how to handle "unknown elements”.

Add the **shiv** for Internet Explorer support



```
<!--[if lt IE 9]>
  <script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
<![endif]-->
```

```
<!--[if lt IE 9]>
<script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
<![endif]-->
```

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web.

HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

The new standard incorporates features like video playback and drag-and-drop that have been previously dependent on third-party browser plug-ins such as Adobe Flash, Microsoft Silverlight.

Browser Support:

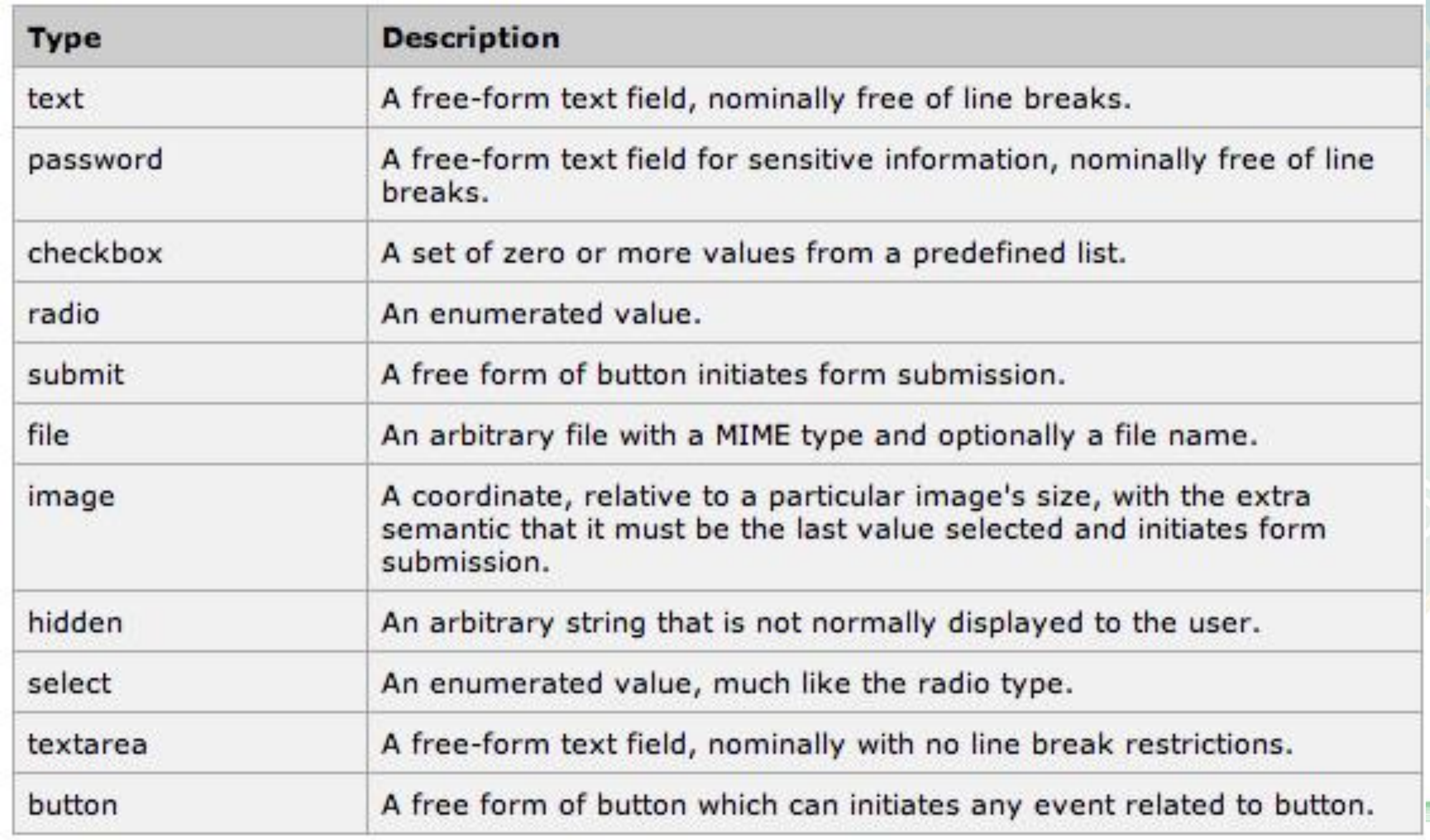
The latest versions of Apple Safari, Google Chrome, Mozilla Firefox, and Opera all support many HTML5 features and Internet Explorer 9.0 will also have support for some HTML5 functionality.

The mobile web browsers that come pre-installed on iPhones, iPads, and Android phones all have excellent support for HTML5.

HTML 5 WEB FORMS 2.0

Web Forms 2.0 is an extension to the forms features found in HTML4. Form elements and attributes in HTML5 provide a greater degree of semantic mark-up than HTML4 and remove a great deal of the need for tedious scripting and styling that was required in HTML4.

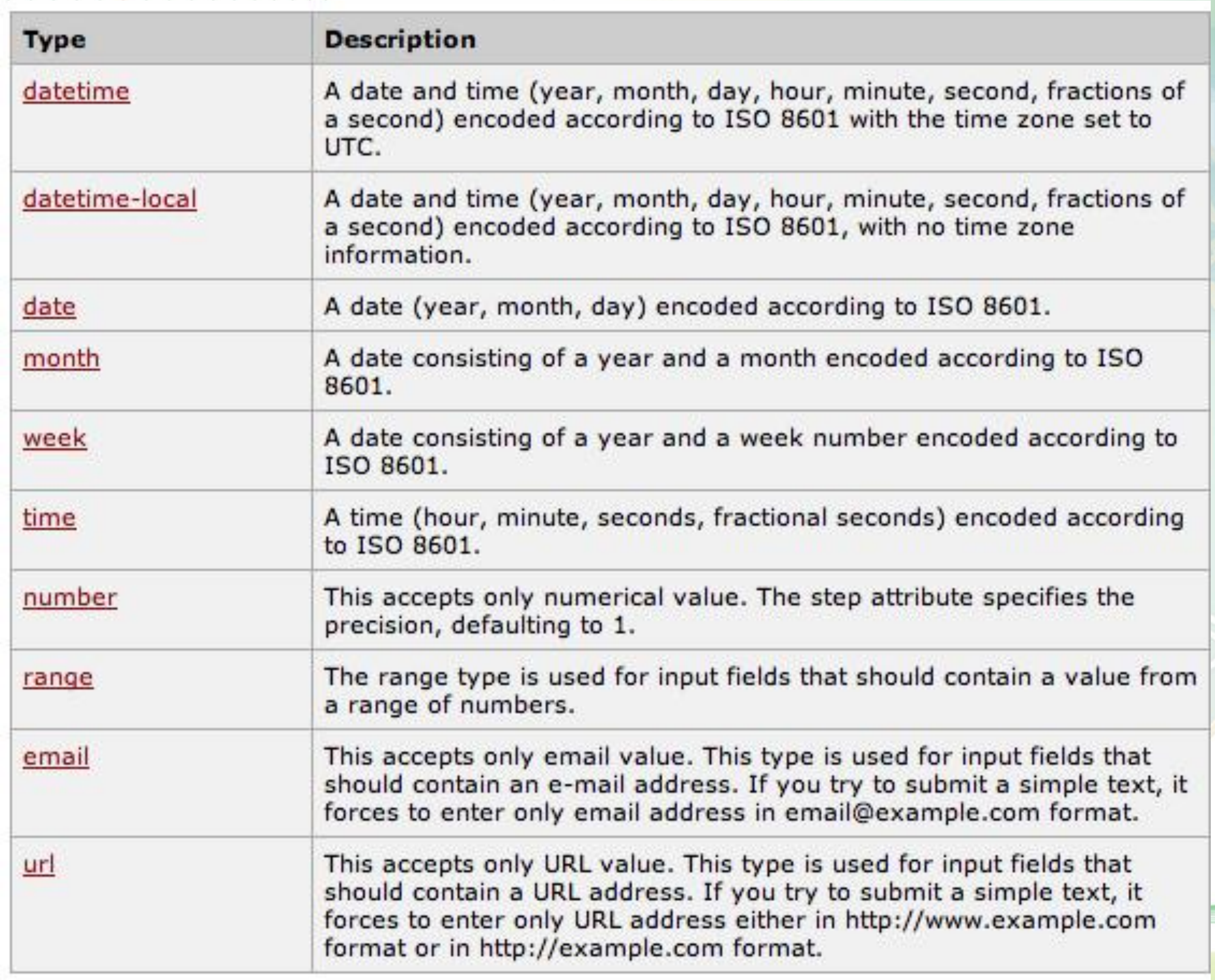
<input> Elements in HTML 4



Type	Description
text	A free-form text field, nominally free of line breaks.
password	A free-form text field for sensitive information, nominally free of line breaks.
checkbox	A set of zero or more values from a predefined list.
radio	An enumerated value.
submit	A free form of button initiates form submission.
file	An arbitrary file with a MIME type and optionally a file name.
image	A coordinate, relative to a particular image's size, with the extra semantic that it must be the last value selected and initiates form submission.
hidden	An arbitrary string that is not normally displayed to the user.
select	An enumerated value, much like the radio type.
textarea	A free-form text field, nominally with no line break restrictions.
button	A free form of button which can initiates any event related to button.

<input> Elements in HTML 5

Apart from the above mentioned attributes, HTML5 input elements introduced several new values for the **type** attribute.



Type	Description
datetime	A date and time (year, month, day, hour, minute, second, fractions of a second) encoded according to ISO 8601 with the time zone set to UTC.
datetime-local	A date and time (year, month, day, hour, minute, second, fractions of a second) encoded according to ISO 8601, with no time zone information.
date	A date (year, month, day) encoded according to ISO 8601.
month	A date consisting of a year and a month encoded according to ISO 8601.
week	A date consisting of a year and a week number encoded according to ISO 8601.
time	A time (hour, minute, seconds, fractional seconds) encoded according to ISO 8601.
number	This accepts only numerical value. The step attribute specifies the precision, defaulting to 1.
range	The range type is used for input fields that should contain a value from a range of numbers.
email	This accepts only email value. This type is used for input fields that should contain an e-mail address. If you try to submit a simple text, it forces to enter only email address in email@example.com format.
url	This accepts only URL value. This type is used for input fields that should contain a URL address. If you try to submit a simple text, it forces to enter only URL address either in http://www.example.com format or in http://example.com format.

The placeholder attribute

HTML5 introduced a new attribute called **placeholder**. This attribute on <input> and <textarea> elements provides a hint to the user of what can be entered in the field. The placeholder text must not contain carriage returns or line-feeds.

```
<input type="text" name="search" placeholder="search the web"/>
```

The autofocus attribute

This is a simple one-step pattern, easily programmed in JavaScript at the time of document load, automatically focus one particular form field.

```
<input type="text" name="search" autofocus/>
```

The required attribute

You do not need to have javascript for client side validations like empty text box would never be submitted because HTML5 introduced a new attribute called **required** .

```
<input type="text" name="search" required/>
```

Embedding Video

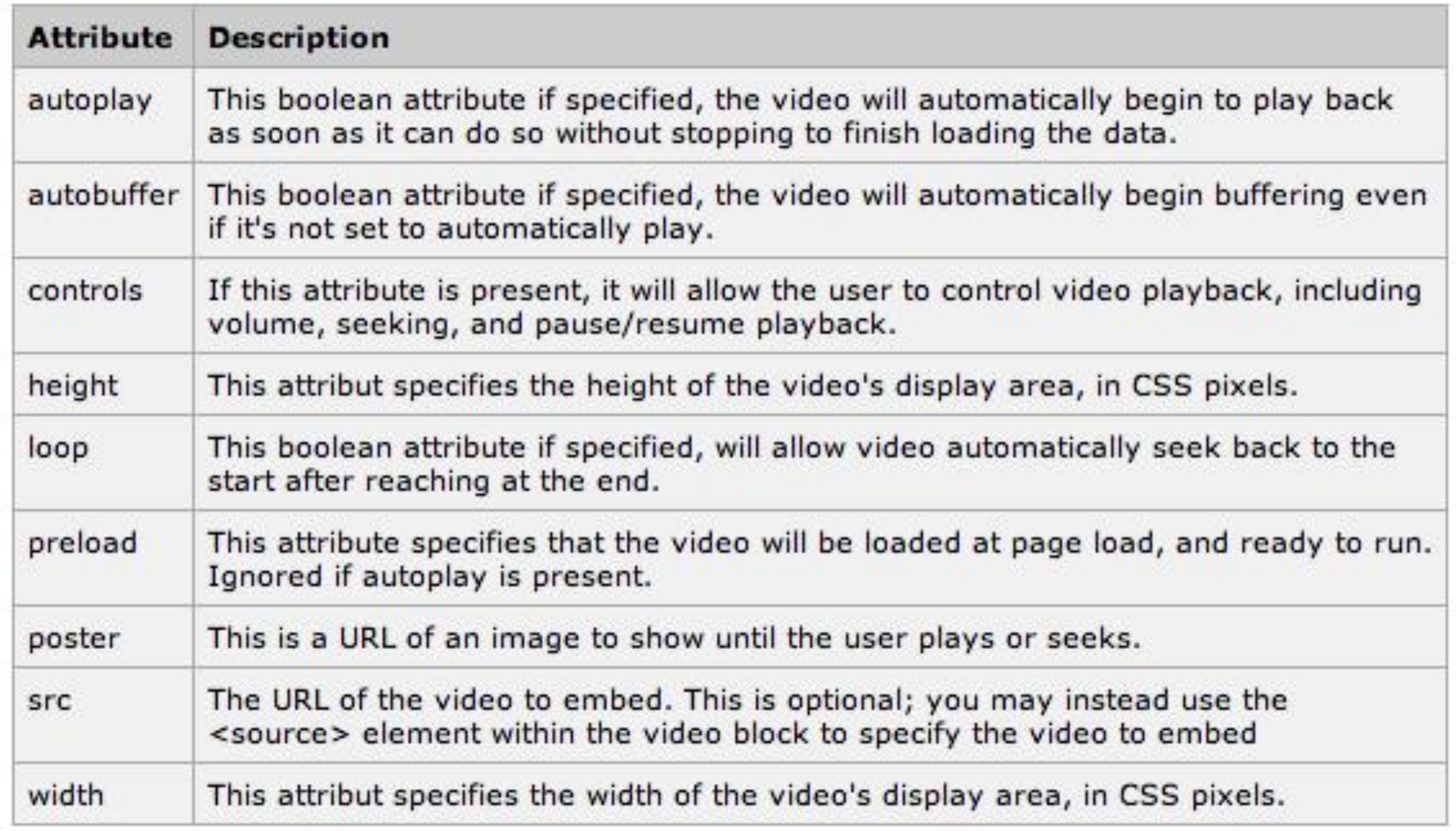
HTML5 features, include native audio and video support without the need for Flash.

Commonly used video formats are:

Ogg: Ogg files with Thedora video codec and Vorbis audio codec.

mpeg4: MPEG4 files with H.264 video codec and AAC audio codec.

Video Attribute Specification:



Attribute	Description
autoplay	This boolean attribute if specified, the video will automatically begin to play back as soon as it can do so without stopping to finish loading the data.
autobuffer	This boolean attribute if specified, the video will automatically begin buffering even if it's not set to automatically play.
controls	If this attribute is present, it will allow the user to control video playback, including volume, seeking, and pause/resume playback.
height	This attribut specifies the height of the video's display area, in CSS pixels.
loop	This boolean attribute if specified, will allow video automatically seek back to the start after reaching at the end.
preload	This attribute specifies that the video will be loaded at page load, and ready to run. Ignored if autoplay is present.
poster	This is a URL of an image to show until the user plays or seeks.
src	The URL of the video to embed. This is optional; you may instead use the <source> element within the video block to specify the video to embed
width	This attribut specifies the width of the video's display area, in CSS pixels.

New HTML5 Form `input` Types

```html
<!DOCTYPE html>

<!-- Fig. 3.1: newforminputtypes.html -->
<!-- New HTML5 form input types and attributes. -->
<html>
   <head>
      <meta charset="utf-8">
      <title>New HTML5 Input Types</title>
   </head>

   <body>
      <h1>New HTML5 Input Types Demo</h1>
      <p>This form demonstrates the new HTML5 input types
         and the placeholder, required and autofocus attributes.
      </p>

      <form method = "post" action = "http://www.deitel.com">
         <p>
            <label>Color:
               <input type = "color" autofocus />
                  (Hexadecimal code such as #ADD8E6)
            </label>
         </p>
```

Fig. 3.1 | New HTML5 form input types and attributes. (Part 1 of 5.)

```
<p>
   <label>Date:
      <input type = "date" />
         (yyyy-mm-dd)
   </label>
</p>
<p>
   <label>Datetime:
      <input type = "datetime" />
         (yyyy-mm-ddThh:mm+ff:gg, such as 2012-01-27T03:15)
   </label>
</p>
<p>
   <label>Datetime-local:
      <input type = "datetime-local" />
         (yyyy-mm-ddThh:mm, such as 2012-01-27T03:15)
   </label>
</p>
<p>
   <label>Email:
      <input type = "email" placeholder = "name@domain.com"
         required /> (name@domain.com)
   </label>
</p>
```

Fig. 3.1 | New HTML5 form input types and attributes. (Part 2 of 5.)

```
<p>
   <label>Month:
      <input type = "month" /> (yyyy-mm)
   </label>
</p>
<p>
   <label>Number:
      <input type = "number"
         min = "0"
         max = "7"
         step = "1"
         value = "4" />
   </label> (Enter a number between 0 and 7)
</p>
<p>
   <label>Range:
      0 <input type = "range"
         min = "0"
         max = "20"
         value = "10" /> 20
   </label>
</p>
```

Fig. 3.1 | New HTML5 form input types and attributes. (Part 3 of 5.)

```
<p>
   <label>Search:
      <input type = "search" placeholder = "search query" />
   </label> (Enter your search query here.)
</p>
<p>
   <label>Tel:
      <input type = "tel" placeholder = "(###) ###-####"
         pattern = "\(\d{3}\) +\d{3}-\d{4}" required />
         (###) ###-####
   </label>
</p>
<p>
   <label>Time:
      <input type = "time" /> (hh:mm:ss.ff)
   </label>
</p>
<p>
   <label>URL:
      <input type = "url"
         placeholder = "http://www.domainname.com" />
         (http://www.domainname.com)
   </label>
</p>
```

Fig. 3.1 | New HTML5 form input types and attributes. (Part 4 of 5.)

```
         <p>
            <label>Week:
               <input type = "week" />
                  (yyyy-Wnn, such as 2012-W01)
            </label>
         </p>
         <p>
            <input type = "submit" value = "Submit" />
            <input type = "reset" value = "Clear" />
         </p>
      </form>
   </body>
</html>
```

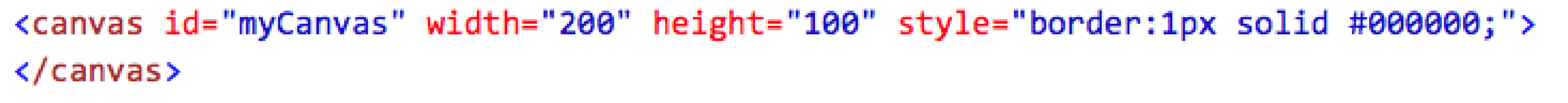
Fig. 3.1 | New HTML5 form input types and attributes. (Part 5 of 5.)

HTML5 CANVAS

The HTML <canvas> element is used to draw graphics on a web page, via scripting (usually JavaScript).

Canvas has several methods for drawing paths, boxes, circles, text, and adding images.

A canvas is a rectangular area on an HTML page. By default, a canvas has no border and no content.



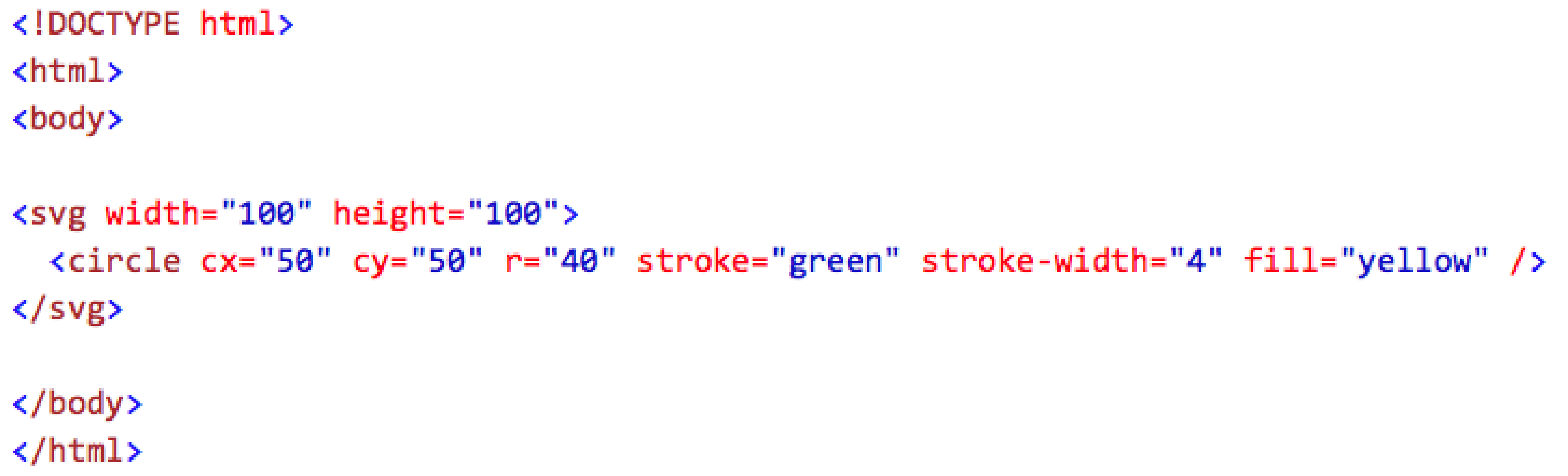
```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
</canvas>
```

Drawing with Javascript

```
<!DOCTYPE html>
<html>
<body>

<canvas id="myCanvas" width="200" height="100" style="border:1px solid #c3c3c3;">
Your browser does not support the HTML5 canvas tag.
</canvas>

<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.fillStyle = "#FF0000";
ctx.fillRect(0,0,150,75);
</script>

</body>
</html>
```

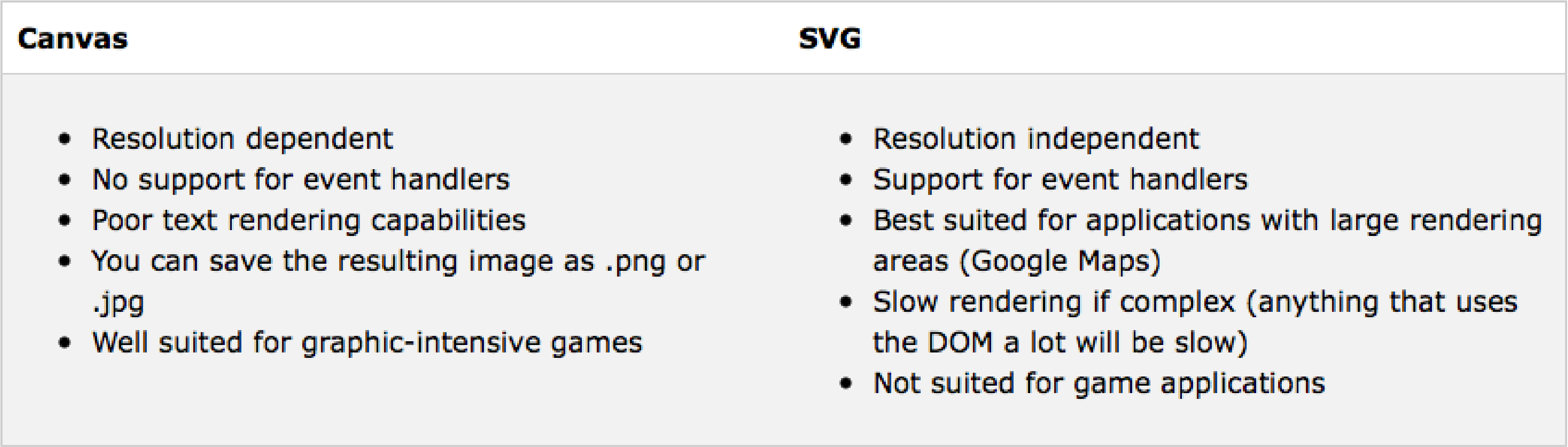
HTML5 SVG

- ✓ SVG stands for Scalable Vector Graphics
- ✓ SVG is used to define graphics for the Web
- ✓ SVG is a W3C recommendation
- ✓ The HTML <svg> element (introduced in HTML5) is a container for SVG graphics.
- ✓ SVG has several methods for drawing paths, boxes, circles, text, and graphic images.



```
<!DOCTYPE html>
<html>
<body>

<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow" />
</svg>

</body>
</html>
```

CANVAS VS SVG



Canvas	SVG
• Resolution dependent • No support for event handlers • Poor text rendering capabilities • You can save the resulting image as .png or .jpg • Well suited for graphic-intensive games	• Resolution independent • Support for event handlers • Best suited for applications with large rendering areas (Google Maps) • Slow rendering if complex (anything that uses the DOM a lot will be slow) • Not suited for game applications