HTML

Prepared by
Manu Jayaraj K M

What is HTML5?

- HTML5 is the newest version of HTML, only recently gaining partial support by the makers of web browsers.
- It incorporates all features from earlier versions of HTML, including the stricter XHTML.
- It adds a diverse set of new tools for the web developer to use.
- It is still a work in progress. No browsers have full HTML5 support. It will be many years – perhaps not until 2018 or later - before being fully defined and supported.

Goals of HTML5

- Support all existing web pages. With HTML5, there is no requirement to go back and revise older websites.
- Reduce the need for external plugins and scripts to show website content.
- Improve the semantic definition (i.e. meaning and purpose) of page elements.
- Make the rendering of web content universal and independent of the device being used.
- Handle web documents errors in a better and more consistent fashion.

New Elements in HTML5

<article>
<aside>
<audio>
<canvas>
<datalist>
<figure>
<figcaption>
<footer>
<header>
<hgroup>
<mark>
<nav>
<progress>
<section>
<source>
<svg>
<time>
<video>

Other New Features in HTML5

- Built-in audio and video support (without plugins)
- Enhanced form controls and attributes
- The Canvas (a way to draw directly on a web page)
- Drag and Drop functionality
- Support for CSS3 (the newer and more powerful version of CSS)
- More advanced features for web developers, such as data storage and offline applications.

The <html> Element

This is what the <html> element looked like in XHTML:

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
   lang="en">
```

HTML5 simplifies this line:

```
<html lang="en">
```

The **lang** attribute in the <html> element declares which language the page content is in. Though not strictly required, it should always be specified, as it can assist search engines and screen readers.

The <head> Section

Here is a typical XHTML <head> section:

```
<head>
  <meta http-equiv="Content-type" content="text/html; charset=UTF-8" />
  <title>My First XHTML Page</title>
  <link rel="stylesheet" type="text/css" href="style.css" />
</head>
```

And the HTML5 version:

```
<head>
  <meta charset="utf-8">
  <title>My First HTML5 Page</title>
  <link rel="stylesheet" href="style.css">
</head>
```

Basic HTML5 Web Page

Putting the prior sections together, and now adding the <body> section and closing tags, we have our first complete web page in HTML5:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>My First HTML5 Page</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <p>HTML5 is fun!</p>
</body>
</html>
```

5. Audio & video

<audio>

Browser Support

Internet Explorer 9, Firefox, Opera, Chrome, and Safari support the <audio> element.

Until now, there has not been a standard for playing audio files on a web page.

Today, most audio files are played through a plug-in (like flash). However, different browsers may have different plug-ins.

HTML5 defines a new element which specifies a standard way to embed an audio file on a web page: the <audio> element

Currently, there are 3 supported file formats for the <audio> element: MP3, Wav, and Ogg:

Browser	MP3	Wav	Ogg
Internet Explorer 9	YES	NO	NO
Firefox 4.0	NO	YES	YES
Google Chrome 6	YES	YES	YES
Apple Safari 5	YES	YES	NO
Opera 10.6	NO	YES	YES

5. Audio & video

Browser Support

Internet Explorer 9, Firefox, Opera, Chrome, and Safari support the <audio> element.

<audio> : Add Fallback to flash

To be safe, we need to add the fallback to a Flash audio player, in case the browser doesn't support any of the formats we specified. For instance, Firefox 3.5 only supports the audio tag with *Ogg* format, but we might only have the *mp3* file available

```
<audio>
    <source src="test.mp3" type="audio/mpeg" />

  <object class="playerpreview" type="application/x-shockwave-flash"
          data="player_mp3_mini.swf" width="200" height="20">
    <param name="movie" value="player_mp3_mini.swf" />
    <param name="bgcolor" value="#085c68" />
    <param name="FlashVars" value="mp3=test.mp3" />
    <embed src="player_mp3_mini.swf" bgcolor="#085c68" width="200"
           height="20" name="movie" align=""
           type="application/x-shockwave-flash" flashvars="mp3=test.mp3">
    </embed>
  </object>

</audio>

<div id="player_fallback"></div>
<script>
  if (!document.createElement('audio').canPlayType) {
    if (!document.createElement('audio').canPlayType('audio/mpeg')) {
      swfobject.embedSWF(
        "player_mp3_mini.swf",
        "player_fallback",
        "200",
        "20",
        "9.0.0",
        "",
        {"mp3":"test.mp3"},
        {"bgcolor":"#085c68"});
    }
  }
</script>
```

Attribute		Value	Description
autoplay	New	autoplay	Specifies that the audio will start playing as soon as it is ready
controls	New	controls	Specifies that audio controls should be displayed (such as a play/pause button etc).
loop	New	loop	Specifies that the audio will start over again, every time it is finished
preload	New	auto metadata none	Specifies if and how the author thinks the audio should be loaded when the page loads
src	New	URL	Specifies the URL of the audio file

make it easier, we are using the SWFObject library to insert the Flash player via JavaScript. gle AJAX Libraries API inserting these two lines in your header:

5. Audio & video

<video>

Browser Support

Internet Explorer 9, Firefox, Opera, Chrome, and Safari support the <audio> element.

Until now, there has not been a standard for showing a video/movie on a web page.

Today, most videos are shown through a plug-in (like flash). However, different browsers may have different plug-ins.

HTML5 defines a new element which specifies a standard way to embed a video/movie on a web page: the <video> element

Currently, there are 3 supported video formats for the <video> element: MP4, WebM, and Ogg:

Browser	MP4	WebM	Ogg
Internet Explorer 9	YES	NO	NO
Firefox 4.0	NO	YES	YES
Google Chrome 6	YES	YES	YES
Apple Safari 5	YES	NO	NO
Opera 10.6	NO	YES	YES

5. Audio & video

<video attributes>

Browser Support

Internet Explorer 9, Firefox, Opera, Chrome, and Safari support the <audio> element.

Attribute		Value	Description
autoplay	New	autoplay	Specifies that the video will start playing as soon as it is ready
controls	New	controls	Specifies that video controls should be displayed (such as a play/pause button etc).
height	New	*pixels*	Sets the height of the video player
loop	New	loop	Specifies that the video will start over again, every time it is finished
muted	New	muted	Specifies that the audio output of the video should be muted
poster	New	*URL*	Specifies an image to be shown while the video is downloading, or until the user hits the play button
preload	New	auto metadata none	Specifies if and how the author thinks the video should be loaded when the page loads
src	New	*URL*	Specifies the URL of the video file
width	New	*pixels*	Sets the width of the video player

Canvas

What is Canvas

With HTML5's Canvas API, we're no longer limited to drawing rectangles on our sites. We can draw anything we can imagine, all through JavaScript. This can improve

the performance of our websites by avoiding the need to download images off the network.With canvas, we can draw shapes and lines, arcs and text, gradients and patterns. In addition, canvas gives us the power to manipulate pixels in images and even video.

The Canvas 2D Context spec is supported in:

- Safari 2.0+
- Chrome 3.0+
- Firefox 3.0+
- Internet Explorer 9.0+
- Opera 10.0+
- iOS (Mobile Safari) 1.0+
- Android 1.0+

Canvas

Drawing a canvas

We'll add the function to our `script` element. The first step is to grab hold of the canvas element on our page:

canvas/demo1.html ( )

```
<script>
⋮
function draw() {
  var canvas = document.getElementById("myCanvas");
}
</script>
```

We obtain our drawing context by calling the getContext method and passing it the string "2d", since we'll be drawing in two dimensions:

canvas/demo1.html

```
function draw() {
  var canvas = document.getElementById("myCanvas");
  var context = canvas.getContext("2d");
}
```

Canvas

Creating a canvas Element

The first step to using canvas is to add a canvas element to the page:

canvas/demo1.html

```
<canvas>
Sorry! Your browser doesn't support Canvas.
</canvas>
```

Since drawing on the canvas is done using JavaScript, we'll need a way to grab the element from the DOM. We'll do so by giving our canvas an `id`:

canvas/demo1.html

```
<canvas id="myCanvas">
Sorry! Your browser doesn't support Canvas.
</canvas>
```

Onload of the page we triggering draw(); method

HTML5: Input types

HTML5 has several new input types for forms.

- email
- url
- number
- range
- date pickers (date, month, week, time, datetime, datetime-local)
- search
- color

HTML5: Input types

Input type	IE	Firefox	Opera	Chrome	Safari
email	No	No	9.0	No	No
url	No	No	9.0	No	No
number	No	No	9.0	7.0	No
range	No	No	9.0	4.0	4.0
Date pickers	No	No	9.0	No	No
search	No	No	11.0	No	No
color	No	No	11.0	No	No

COMPARISON WITH OLDER VERSIONS

- HTML5 Is a Work in Progress
- Simplified Syntax
- The New <canvas> Element
- The <header> and <footer> Elements
- New <section> and <article> Elements
- New <menu> and <figure> Elements
- New <audio> and <video> Elements
- New Forms
- No More <frame>, , <big>

THANK YOU