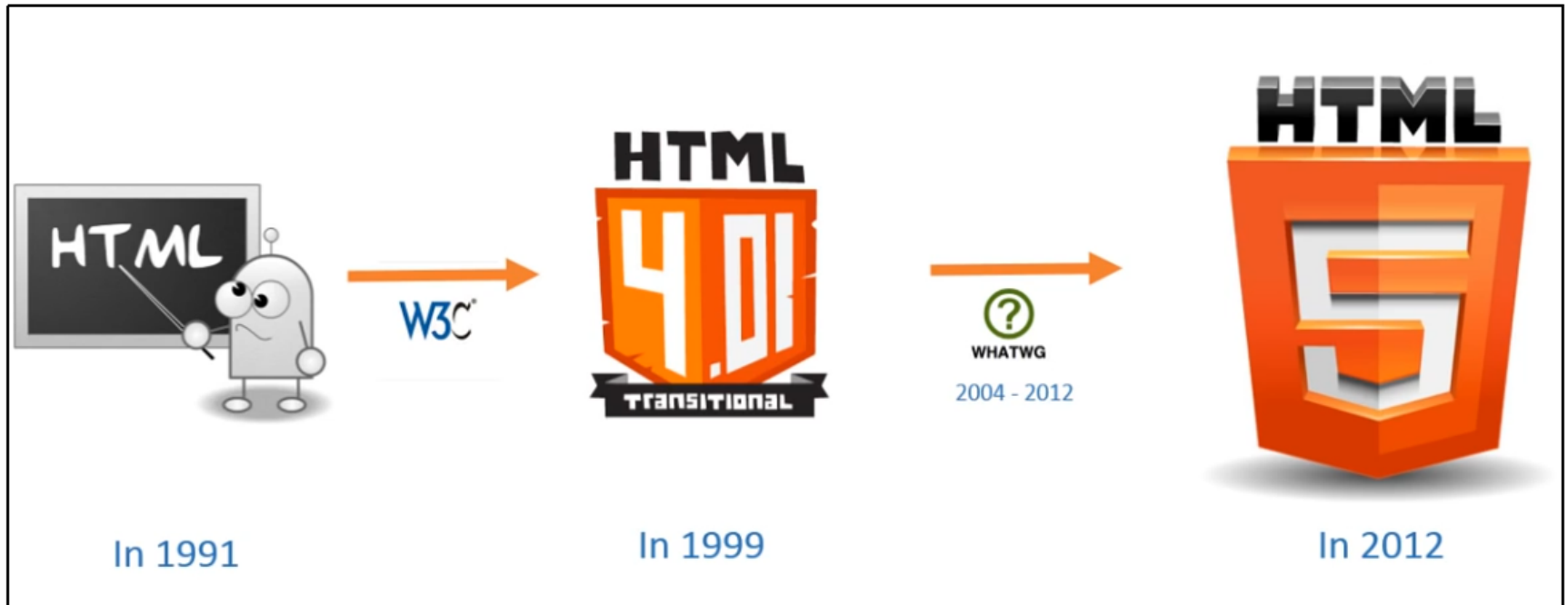


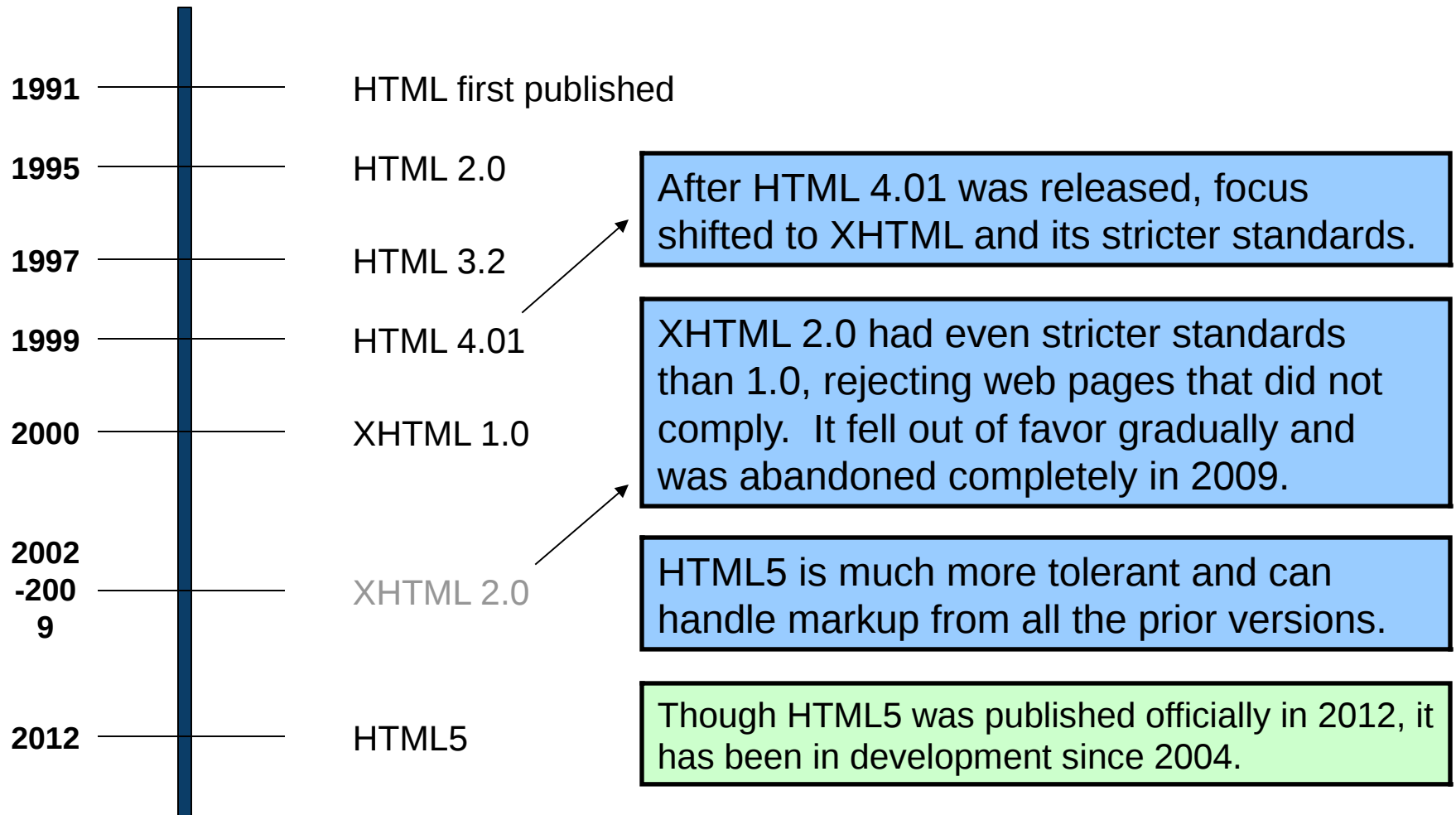
Introduction to HTML5



HTML5 some histroy



History of HTML





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.



New feature of HTML5

- ▣ **New Semantic Elements** – These are like <header>, <footer>, and <section>.
- ▣ **Forms 2.0** – Improvements to HTML web forms where new attributes have been introduced for <input> tag.
- ▣ **Persistent Local Storage** – To achieve without resorting to third-party plugins.
- ▣ **WebSocket** – A next-generation bidirectional communication technology for web applications.
- ▣ **Server-Sent Events** – HTML5 introduces events which flow from web server to the web browsers and they are called Server-Sent Events (SSE).
- ▣ **Canvas** – This supports a two-dimensional drawing surface that you can program with JavaScript.
- ▣ **Audio & Video** – You can embed audio or video on your webpages without resorting to third-party plugins.
- ▣ **Geolocation** – Now visitors can choose to share their physical location with your web application.
- ▣ **Microdata** – This lets you create your own vocabularies beyond HTML5 and extend your web pages with custom semantics.
- ▣ **Drag and drop** – Drag and drop the items from one location to another location on the same webpage.



New semantic elements

HTML5 simplified syntax

- New doctype

```
<!DOCTYPE html>
```

- New Simplified encoding

```
<meta charset = "UTF-8">
```

- New Script tag

```
<script src = "scriptfile.js"></script>
```

- New link tag

```
<link rel = "stylesheet" href = "stylefile.css">
```

New Semantic element in HTML5

- **section** – This tag represents a generic document or application section. It can be used together with h1-h6 to indicate the document structure.
- **article** – This tag represents an independent piece of content of a document, such as a blog entry or newspaper article.
- **aside** – This tag represents a piece of content that is only slightly related to the rest of the page.
- **header** – This tag represents the header of a section.
- **footer** – This tag represents a footer for a section and can contain information about the author, copyright information, et cetera.
- **nav** – This tag represents a section of the document intended for navigation.
- **dialog** – This tag can be used to mark up a conversation.
- **figure** – This tag can be used to associate a caption together with some embedded content, such as a graphic or video.


```
<!DOCTYPE html>

<html>
  <head>
    <meta charset = "utf-8">
    <title>...</title>
  </head>

  <body>
    <header>...</header>
    <nav>...</nav>

    <article>
      <section>
        ...
      </section>
    </article>
    <aside>...</aside>

    <footer>...</footer>
  </body>
</html>
```

HTML5 Document Structure Example

This page should be tried in safari, chrome or Mozilla.

- [HTML Tutorial](#)
- [CSS Tutorial](#)
- [JavaScript Tutorial](#)

Once article can have multiple sections

This is aside part of the web page

Created by [rgupta](#)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>sample html5 page</title>
</head>
<body>
  <header role = "banner">
    <h1>HTML5 Document Structure Example</h1>
    <p>This page should be tried in safari, chrome or Mozilla.</p>
  </header>

  <nav>
    <ul>
      <li><a href = "#">HTML Tutorial</a></li>
      <li><a href = "#">CSS Tutorial</a></li>
      <li><a href = "#"> JavaScript Tutorial</a></li>
    </ul>
  </nav>

  <article>
    <section>
      <p>Once article can have multiple sections</p>
    </section>
  </article>

  <aside>
    <p>This is  aside part of the web page</p>
  </aside>






  <footer>
    <p>Created by <a href = "https://rgupta.com/">rgupta</a></p>
  </footer>
</body>
</html>
```








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 free us from a great deal of tedious scripting and styling that was required in HTML4.

Web Form 2.0

| | |
|---|--|
| 1 | <code>datetime</code>  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. |
| 2 | <code>datetime-local</code>  A date and time (year, month, day, hour, minute, second, fractions of a second) encoded according to ISO 8601, with no time zone information. |
| 3 | <code>date</code>  A date (year, month, day) encoded according to ISO 8601. |
| 4 | <code>month</code>  A date consisting of a year and a month encoded according to ISO 8601. |
| 5 | <code>week</code>  A date consisting of a year and a week number encoded according to ISO 8601. |

Web Form 2.0

| | |
|----|---|
| 6 | <p>time </p> <p>A time (hour, minute, seconds, fractional seconds) encoded according to ISO 8601.</p> |
| 7 | <p>number </p> <p>It accepts only numerical value. The step attribute specifies the precision, defaulting to 1.</p> |
| 8 | <p>range </p> <p>The range type is used for input fields that should contain a value from a range of numbers.</p> |
| 9 | <p>email </p> <p>It 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.</p> |
| 10 | <p>url </p> <p>It 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.</p> |

```

<form action = "#" method = "POST">
Enter URL : <input type = "url" name = "newinput" /><br/>
Enter email : <input type = "email" name = "newinput" /><br/>
Select Range : <input type = "range" min = "0" max = "10" step "1"
                value = "5" name = "newinput" /><br/>
Select Number : <input type = "number" min = "0" max = "10" step "1"
                value = "5" name = "newinput" /><br/>
Reg date and time : <input type = "datetime" name = "newinput" /><br/>
Local Date and Time : <input type = "datetime-local" name = "newinput" /><br/>
Date : <input type = "date" name = "newinput" /><br/>
Month : <input type = "month" name = "newinput" /><br/>
Week : <input type = "week" name = "newinput" /><br/>
Time : <input type = "time" name = "newinput" /><br/>
        <input type = "submit" value = "submit" />
</form>

```

Enter URL :

Enter email :

Select Range :

Select Number :

Reg date and time :

Local Date and Time :

Date :

Month :

Week :

Time :



Other attributes

placeholder attribute

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

autofocus attribute

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

required attribute

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



HTML5 - SVG

SVG stand for Scalable Vector Graphics


It is extension to design 2D graphics

Mostly used in diagrams like pie charts and 2D graphs

SVG examples

```
<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">
  <circle id = "redcircle" cx = "50" cy = "50" r = "50" fill = "red" />
</svg>
<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">
  <rect id = "redrect" width = "300" height = "100" fill = "red" />
</svg>
<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">
  <line x1 = "0" y1 = "0" x2 = "200" y2 = "100"
    style = "stroke: ■ red;stroke-width:2"/>
</svg>
```





HTML5 - MathML

- Stands for Mathematical Markup Language
- Used for describing mathematical notations
- Captures the structure and the content
- Helps in integrating mathematical formulae with WWW.

```

<body>
  <math xmlns = "http://www.w3.org/1998/Math/MathML">
    <mrow>
      <mi>A</mi>
      <mo>=</mo>

      <mfenced open = "[" close="]">
        <table>
          <mtr>
            <mtd><mi>x</mi></mtd>
            <mtd><mi>y</mi></mtd>
          </mtr>
          <mtr>
            <mtd><mi>z</mi></mtd>
            <mtd><mi>w</mi></mtd>
          </mtr>
        </table>
      </mfenced>
    </mrow>
  </math>
</body>

```

mi: integral value
mo: operator

$$A = \begin{bmatrix} x & y \\ z & w \end{bmatrix}$$



HTML5 – web storage



HTML5 – web storage

HTML5 introduces two mechanisms, similar to HTTP session cookies, for storing structured data on the client side and to overcome following drawbacks.

Cookies are included with every HTTP request, thereby slowing down your web application by transmitting the same data.

Cookies are included with every HTTP request, thereby sending data unencrypted over the internet.

Cookies are limited to about 4 KB of data. Not enough to store required data.

The two storages are :

session storage

local storage



HTML5 – Session Storage

The `sessionStorage` is a read only property of window object. It stores data in a web browser specifically to the domain and protocol for a particular session. It doesn't get sent to the server. Data stored in `sessionStorage` gets cleared when the page session ends. A page session lasts for as long as the browser is open and survives over page reloads and restores.

Methods

- `setItem (key, value)` – It allows to add a key/value pair to the storage object. If the key already exists, the name value will overwrite the old value.
- `getItem(key)` – It returns the value of the item that is set with the given key.
- `key(n)` – It returns the key of the item in the storage object at the nth index which can be useful for looping.
- `removeItem(key)` – It removes the item in the storage object with the given key.

```
<script type = "text/javascript">  
    sessionStorage.setItem('username', "raja");  
    sessionStorage.setItem('perference', "java book");  
    console.log(sessionStorage.getItem('username'));  
    sessionStorage.removeItem('username');  
    sessionStorage.clear();  
</script>
```



HTML5 – Local Storage

Local Storage

The localStorage is a read only property of window object. It stores data in a web browser specifically to the domain and protocol. It doesn't get sent to the server as it is stored locally in the web browser with no expiration date. The data will not be deleted when the browser is closed and reopened.

Methods

- `setItem(key, value)` – It allows to add a key/value pair to the storage object. If the key already exists, the name value will overwrite the old value.
- `getItem(key)` – It returns the value of the item that is set with the given key.
- `key(n)` – It returns the key of the item in the storage object at the nth index which can be useful for looping.
- `removeItem(key)` – It removes the item in the storage object with the given key.

```
window.localStorage.setItem('username', "raja");  
window.localStorage.setItem('perference', "java book");  
console.log(localStorage.getItem('username'));  
localStorage.removeItem('username');  
localStorage.clear();
```

`</script>`



HTML5 - Audio & Video

HTML5 – Audio & Video

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

Embedding Video

Here is the simplest form of embedding a video file in your webpage –

```
<video src = "foo.mp4"  width = "300" height = "200" controls>  
  Your browser does not support the <video> element.  
</video>
```

The current HTML5 draft specification does not specify which video formats browsers should support in the video tag. But most commonly used video formats are –

- **Ogg** – Ogg files with Theora video codec and Vorbis audio codec.
- **mpeg4** – MPEG4 files with H.264 video codec and AAC audio codec.

| Sr.No. | Attribute & Description |
|--------|--|
| 1 | 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. |
| 2 | autobuffer This Boolean attribute if specified, the video will automatically begin buffering even if it's not set to automatically play. |
| 3 | controls If this attribute is present, it will allow the user to control video playback, including volume, seeking, and pause/resume playback. |
| 4 | height This attribute specifies the height of the video's display area, in CSS pixels. |

| | |
|---|---|
| 5 | loop This Boolean attribute if specified, will allow video automatically seek back to the start after reaching at the end. |
| 6 | preload This attribute specifies that the video will be loaded at page load, and ready to run. Ignored if autoplay is present. |
| 7 | poster This is a URL of an image to show until the user plays or seeks. |
| 8 | 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. |
| 9 | width This attribute specifies the width of the video's display area, in CSS pixels. |



HTML5 - Geolocation

HTML5 Geolocation API lets you share your location with your favourite web sites. A Javascript can capture your latitude and longitude and can be sent to backed web server and do fancy location-aware things like finding local businesses or showing your location on a map.