

HTML5

IMY 220 ● Lecture 1

Contents

- Introduction
- History
- WHATWG
- Minimal HTML5 document
 - DOCTYPE
 - Character encoding
- Top new features in HTML5
- Forms
- Resources

Introduction

HTML 5

The 5th major revision of the core language of the World Wide Web: the **Hypertext Markup Language** (HTML).

W3C (**World Wide Web Consortium**) and WHATWG (**Web Hypertext Application Technology Working Group**) created the new version of HTML together.

Going forward, the WHATWG just refers to "HTML", without worrying about version numbers.

History

HTML5

HTML was invented by Tim Berners-Lee in 1991.

HTML5 reached recommendation status on 28 October 2014.

The previous version of HTML was standardised in 1999 (HTML 4.01).

<http://www.w3.org/TR/html401/>



WHATWG

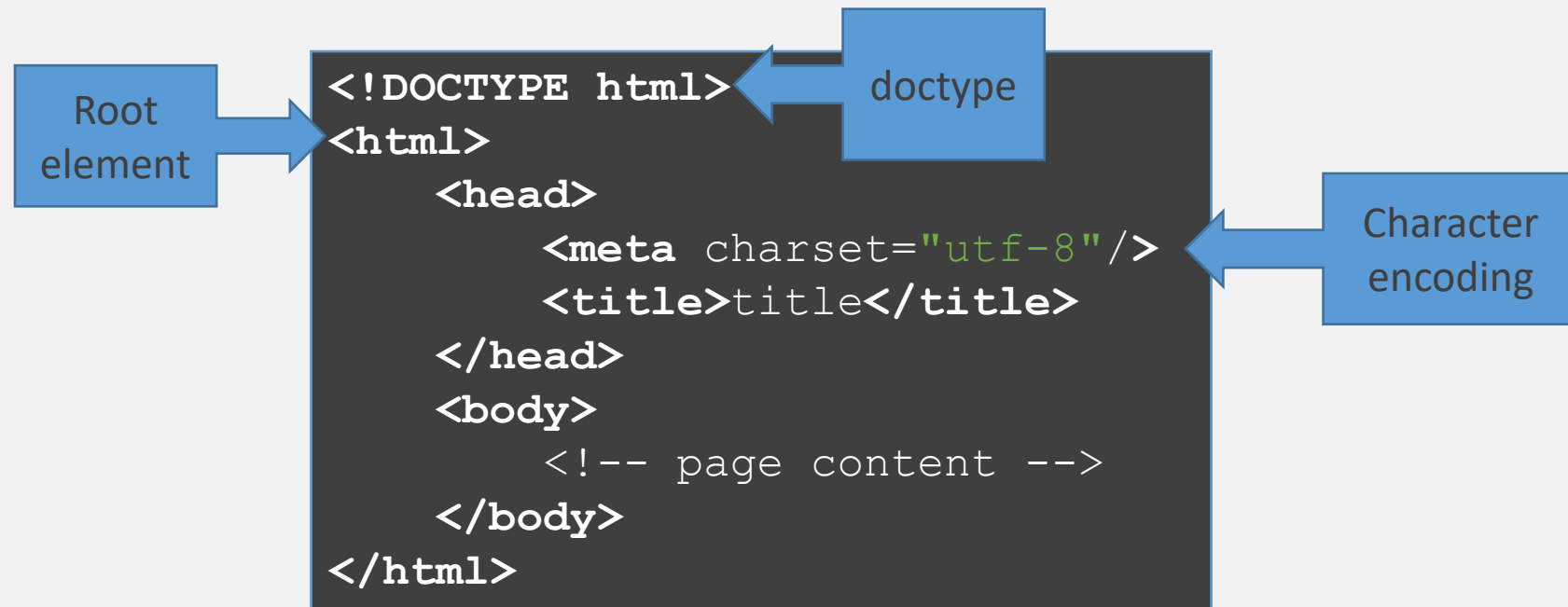
WHATWG is a growing **community of people** interested in evolving the Web.

It focuses primarily on the development of **HTML** and **APIs** needed for **Web applications**.

The WHATWG was founded by individuals of Apple, the Mozilla Foundation, and Opera Software in 2004, after a W3C workshop.

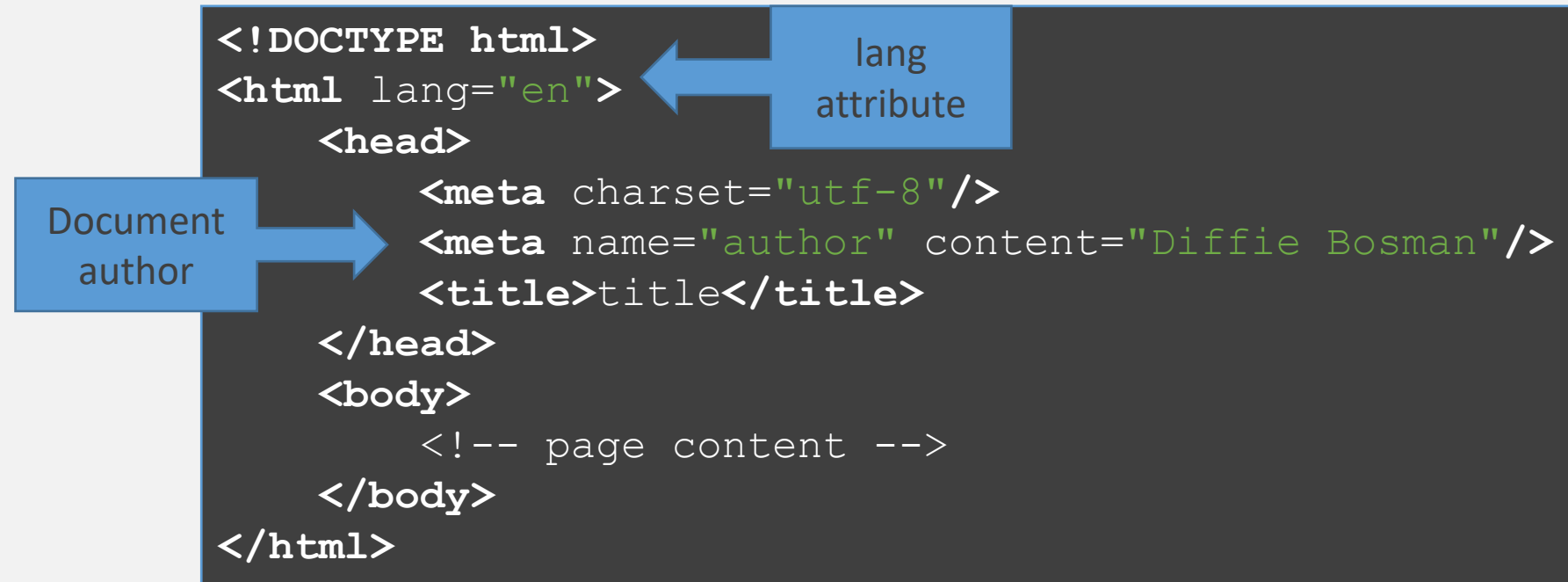
Minimal HTML5 document

Minimum required components for an HTML5 document to be valid:



Minimal HTML5 document

Add the following optional data to make it more complete:



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8"/>
    <meta name="author" content="Diffie Bosman"/>
    <title>title</title>
  </head>
  <body>
    <!-- page content -->
  </body>
</html>
```

The doctype

A doctype is a special instruction that tells the browser what **version of HTML** the document is written in.

The doctype is not an element.

HTML5 doctype:

```
<!DOCTYPE html>
```


The doctype (side note)

Used to look like this

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

and this

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

and other ones for framesets, XHTML, etc.

The doctype

A normal doctype consists of the following parts, in exactly the following order:

- Any case-**ins**sensitive match for the string "<!DOCTYPE".
- One or more space characters.
- Any case-**ins**sensitive match for the string "html".
- Optionally, one or more space characters.
- A ">" character.

Character encoding declaration

Web developers are required to declare the character encoding.

A mechanism for specifying the character encoding used to store or transmit a document.

In HTML5: Use a **meta** element with a **charset** attribute that specifies the encoding within the first 1024 bytes of the document.

```
<meta charset="utf-8" />
```

This replaces the need for the following declaration, although that syntax is still allowed.

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

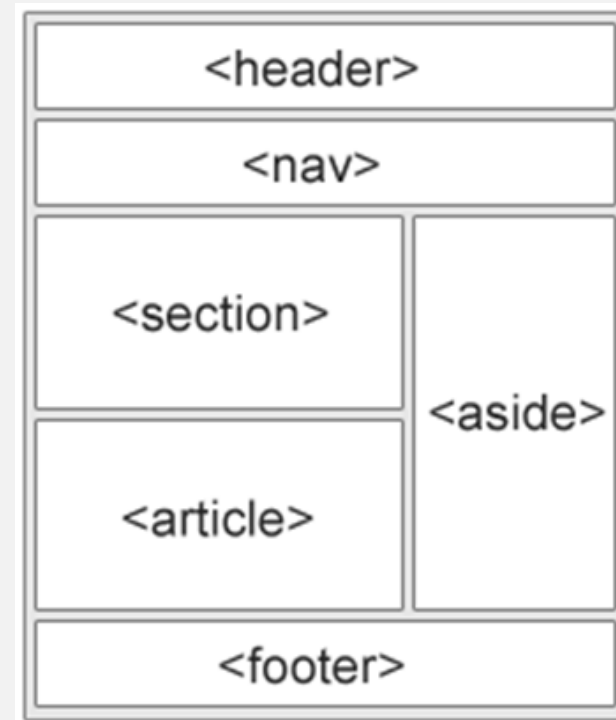
6 new features in HTML5

- New **structural elements**
- **SVG elements** for drawing in HTML
- New **form controls & attributes** for better form handling
- **Web storage API** for storing data on the client

New Structural Elements

The following **elements** have been introduced for better structure:

- section
- article
- main
- aside
- header
- footer
- nav
- figure
- figcaption
- template



New Structural Elements

section

represents a generic document or application section. It should be used together with the h1, h2, h3, h4, h5, and h6 elements to indicate the document structure.

article

represents an independent piece of content of a document, such as a blog entry or newspaper article.

main

represents the main content of the body of a document or application.

New Structural Elements

aside

represents a piece of content that is only slightly related to the rest of the page.

header

represents a group of introductory or navigational aids.

footer

represents a footer for a section and can contain information about the author, copyright information, etc.

New Structural Elements

`nav`

represents a section of the document intended for navigation.

`template`

can be used to declare fragments of HTML that can be cloned and inserted in the document by script.

New Structural Elements

figure

represents a piece of self-contained flow content, typically referenced as a single unit from the main flow of the document.

figcaption

can be used as caption (it is optional).

```
<figure>  
  <video src="example.webm" controls></video>  
  <figcaption> Example </figcaption>  
</figure>
```

New Structural Elements

What do these new elements actually do?

New Structural Elements

These new elements provide a logical way to describe documents in a standardised way

This helps developers create self-describing markup, in other words, the structure and meaning of different pieces of content is self-evident from the use of the HTML elements themselves

(As opposed to, for example, using divs with classes for everything)

They also help with search engine optimisation (SEO) and assistive technologies

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element>

https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Using_HTML_sections_and_outlines

SVG

HTML5 syntax allows for SVG (Scalable Vector Graphics) elements to be embedded directly inside an HTML document.

Unlike JPEG or GIF, SVG does not lose quality when scaled, but is not as standard a format.

SVG allows for inside-element animation.

SVG

An XML-based format of vector-based images that can directly be embedded in the HTML.

```
<p>  
  A green circle:  
  <svg> <circle r="50" cx="50" cy="50" fill="green"/> </svg>  
</p>
```

New form controls (input types)

The **input element's type attribute** now has the following new values:

- tel
- search
- url
- email
- date
- time
- number
- range
- color

```
<form method="post" action="profile.php">  
  
  <label for="email1"> Email </label>  
  <input type="email" name="email1" id="email1"/>  
  
  <input type="submit" value="Log In"/>  
  
</form>
```

New form controls (input types)

These new features allow for **better input control** and **validation**.

The idea of these new types is that the user agent can use the **user interface**, such as a calendar date picker, to submit a **defined format** to the server.

It gives the user a better experience as his input is **checked** before sending it to the server meaning there is less time to wait for feedback.

New form attributes

- autofocus
- placeholder
- required

New form attributes

autofocus

This attribute can be specified on the **input** (except when the type attribute is hidden), **select**, **textarea** and **button** elements.

It provides a declarative way to **focus a form control during page load**.

Using this feature should enhance the user experience compared to focusing the element with script as the user can turn it off if the user does not like it, for instance.

New form attributes

placeholder

This attribute can be specified on the **input** and **textarea** elements.

It represents a hint intended to aid the user with data entry.

```
<input type="search" name="q" placeholder="Enter search phrase..." />
<button> Search </button>
<label>Email
    <input type="email" name="email" placeholder="john@example.com" />
</label>
```

New form attributes

placeholder

The placeholder attribute should not be used as a replacement for the label element.

```
<!-- Do not do this: -->  
<input type="email" name="email" placeholder="Email" />
```

New form attributes

Disappearing placeholder text strains users' short-term memory.

Users often multitask while using the web, which means they're likely to have multiple tabs open and be working on different tasks

Users are thus likely to forget what each input box requires, which is why the label should always clearly indicate what each input box is for

("Recognition rather than recall")

New form attributes

When error messages occur, people don't know how to fix the problem

If there are error messages, users would have to delete text, check the placeholder, and re-enter text

New form attributes

Placeholder text that disappears when the cursor is placed in a form field is irritating for users navigating with the keyboard

If there are error messages, users would have to delete text, check the placeholder, and re-enter text

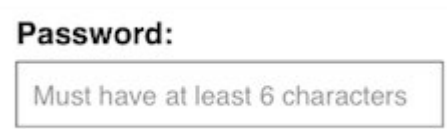
New form attributes

Bad (placeholder only)



Password

Better (label + placeholder)



Password:

Must have at least 6 characters

<https://www.nngroup.com/articles/form-design-placeholders/>

New form attributes

required

This attribute applies to **input** (except when the type attribute is hidden, image or some button type such as submit), **select** and **textarea**.

It indicates that the user has to fill in a value in order to submit the form. For select, the first option element has to be a placeholder with an empty value.

```
<label>Color:
  <select name="color" required>
    <option> Red </option>
    <option> Green </option>
    <option> Blue </option>
  </select>
</label>
```

Note: In this example, the form control is nested inside the label element.
Don't do this.

Web storage API

HTML5 offers two new methods for storing data on the client:

- `localStorage` - stores data with no time limit
- `sessionStorage` - stores data for one session

Earlier, this was done with cookies.

Cookies are not suitable for large amounts of data, because they are passed on by EVERY request to the server, making it very slow and ineffective.

Web storage API

In HTML5, the data is NOT passed on by every server request, but used ONLY when asked for.

It is possible to store large amounts of data without affecting the website's performance.

The data is stored in different areas for different websites

Website can only access data stored by itself.

HTML5 uses JavaScript to store and access the data.

Web storage API

The `localStorage` Method

The `localStorage` method stores the data with no time limit. The data will be available the next day, week, or year.

```
<script>
  localStorage.lastname = "Sanchez";
  document.write(localStorage.lastname);
</script>
```

Web storage API

The `sessionStorage` Method

The `sessionStorage` method stores the data for one session. The data is deleted when the user closes the browser window.

```
<script>
  sessionStorage.lastname = "Sanchez";
  document.write(sessionStorage.lastname);
</script>
```

child who does not master this stage feels inferior, as if he or she were incapable of reaching positive solutions and unable to achieve what peers are accomplishing.

IDENTITY VERSUS ROLE CONFUSION

LISTEN TO THE AUDIO:



The fifth ego crisis (corresponding roughly to Freud's genital stage) is the most famous and influential of Erikson's stages: "Identity Versus Role Confusion." At this stage, the adolescent experiments with different roles, while trying to integrate identities from previous stages. For instance, the child is son (or daughter), student, friend, and possibly sibling. How do these fit together into a cohesive whole?

The
End

Resources

- <http://www.w3.org/TR/html5-diff/>
- <http://www.w3.org/TR/html-markup/syntax.html>
- <http://www.w3.org/TR/html5/syntax.html>
- <https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5>
- <https://developer.mozilla.org/en-US/docs/Web/HTML/Element>