

# INFO 151

# Web Systems and Services

Week 1 (T2)

Dr Philip Moore

Dr Zhili Zhao

# The Core Course Technologies



- **HTML 5**
  - Hypertext Markup Language
- **JavaScript**
  - Client-side scripting language
- **PHP**
  - Hypertext preprocessor – a server-side scripting language
- **Database (MySQL server)**
  - Back-end relational database management system

# Introduction to Web-Systems Programming

## Introduction to Web Technologies

### History of the HTML and the Internet

### Introduction to HTML

### Overview of the DOM

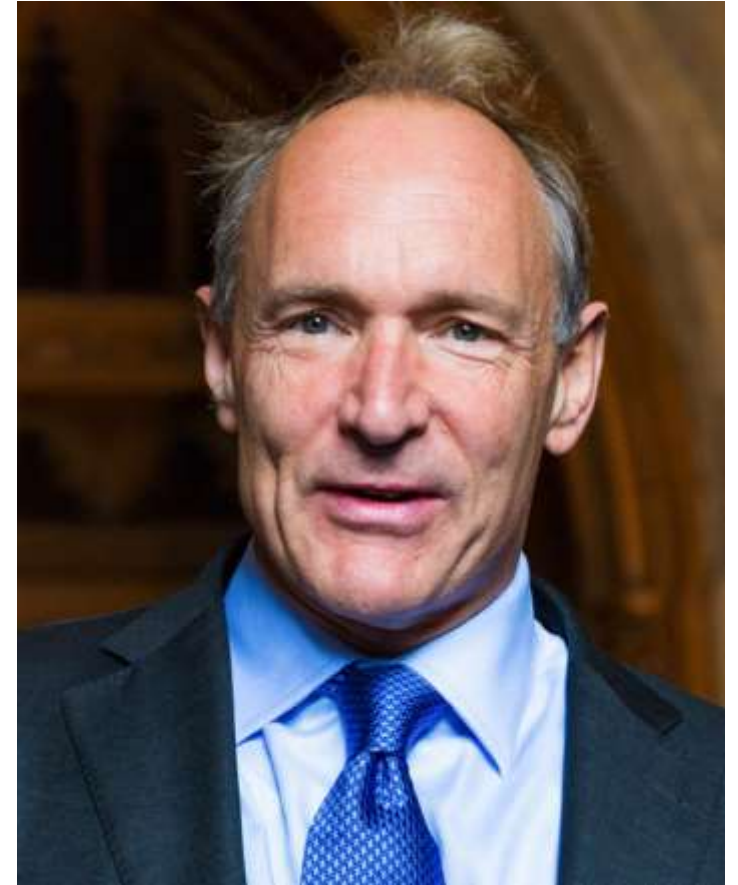
# The Internet

## Hypertext

### HTML 5

# Tim Berners-Lee

- While working at CERN:
  - *Prof Dr Tim Berners-Lee* developed a program for his own use
  - The program was called *Enquire*
  - The aim was to store information in files that contained connections (or *links*) both within and among separate files
- The technique built into *Enquire* became known as *hypertext*



Prof Dr Tim Berners-Lee

# Hypertext and the Internet

- He proposed the creation of a global *hypertext* document system that would make use of a system termed the *Internet*.
- The design goal was to provide researchers with the ability to share their work and results without having to rely on email communications
- Hypertext links enabled: researchers to:
  - Researchers to place such information *online*
  - Information *online* could be retrieve it *anytime* and *anywhere* (with a computer and an *Internet* or *intranet connection*)

# Tim Berners Lee and the Internet

- He wrote and created the software for the first Web server
  - The central repository for the files to be shared
- He wrote and created the first *Web client* (now known as a *web browser*) between October 1990 and the summer of 1991
  - The program to *access* and *display* files retrieved from the *server*
- The first *killer application* of the *Web* at CERN:
  - was the laboratory's telephone directory
  - A very simple and mundane beginning for one of the technological wonders of the computer age.

# W3C and the WHATWG

- *Internet* technologies and protocols are managed by the World Wide Web Consortium (W3C) (URL: <https://www.w3.org/>)
- From 2004 the *HTML standard* has been maintained by WHATWG :
  - Web Hypertext Application Technology Working Group (<https://whatwg.org/>)
- The following slides show the WHATWG web site where you can see:
  - The welcome (or home) page
  - Access to the HTML standards
  - Access to an HTML validation checker
  - HTML “living standard” web page with multiple links
  - The GitHub page (a research sharing page)
  - Other links useful in communication and collaboration





# Welcome to the WHATWG community

Maintaining and evolving HTML since 2004

→ [Read the HTML Living Standard](#)

→ [See the other Living Standards developed at the WHATWG](#)

[Blog](#)

[FAQ](#)

[GitHub](#)

[Policies](#)



**Participate**

Get started with contributing to the WHATWG



**IRC**

Chat with other members of the WHATWG community



**Twitter**

Keep track of spec changes and other announcements



**HTML checker**

Validate your HTML documents

Queries can be directed either as a [new meta issue](#) or as an email to the [Steering Group](#).

Copyright © 2018 WHATWG (Apple, Google, Mozilla, Microsoft). This work is licensed under a [Creative Commons Attribution 4.0 International License](#).



## WHATWG – Standards

**Standards**

**FAQ**

**Policies**

**Participate**

The WHATWG works on a number of technologies that are fundamental parts of the web platform. They are organised somewhat arbitrarily based on the preferences of those editing the standard for those technologies.

### [Compatibility](#)

The Compatibility Standard describes a collection of non-standard (and often vendor-prefixed) CSS properties and DOM APIs that web browsers need to support for compatibility with the de facto web.

### [Console](#)

The Console Standard defines APIs for console debugging facilities.

### [DOM](#)

The DOM Standard defines the core infrastructure used to define the web.

### [Encoding](#)

The Encoding Standard defines how character encodings work on the web.

### [Fetch](#)

The Fetch Standard defines the networking model for resource retrieval on the web.

### [Fullscreen API](#)

The Fullscreen API Standard defines how web pages can take over a user's entire screen (at the user's request), e.g., for gaming or to watch a video.



# WHATWG – HTML Conformance Checkers

Standards

FAQ

Policies

Participate

Here are the currently known online conformance checkers that track the [HTML Standard](#):

- [Nu Html Checker](#)

Address of page to check:

Check

Please [let us know](#) if you are working on a conformance checker.

Copyright © 2018 WHATWG (Apple, Google, Mozilla, Microsoft). This work is licensed under a [Creative Commons Attribution 4.0 International License](#).

# HTML

Living Standard — Last Updated 9 May 2020



<a href="#">One-Page Version</a> html.spec.whatwg.org	<a href="#">Multipage Version</a> /multipage	<a href="#">Developer Version</a> /dev	<a href="#">PDF Version</a> /print.pdf	<a href="#">Translations</a> 日本語・简体中文	<a href="#">FAQ</a> on GitHub	<a href="#">Join us on IRC</a> #whatwg on Freenode	
<a href="#">Contribute on GitHub</a> whatwg/html repository	<a href="#">Commits</a> on GitHub	<a href="#">Snapshot</a> as of this commit	<a href="#">Twitter Updates</a> @htmlstandard	<a href="#">Open Issues</a> filed on GitHub	<a href="#">Open an Issue</a> whatwg.org/newbug	<a href="#">Tests</a> web-platform-tests html/	<a href="#">Issues for Tests</a> ongoing work

## Table of contents

- 1 Introduction
- 2 Common infrastructure
- 3 Semantics, structure, and APIs of HTML documents
- 4 The elements of HTML
- 5 Microdata
- 6 User interaction
- 7 Loading Web pages
- 8 Web application APIs
- 9 Communication
- 10 Web workers
- 11 Web storage
- 12 The HTML syntax
- 13 The XML syntax
- 14 Rendering
- 15 Obsolete features
- 16 IANA considerations
- Index
- References
- A Acknowledgments



Why GitHub?

Team

Enterprise

Explore

Marketplace

Pricing

Search

Sign in

Sign up



WHATWG

Please leave your sense of logic at the door. All are welcome to participate.

#whatwg, Freenode

https://whatwg.org/ Verified

Repositories 36

Packages

People 69

**Grow your team on GitHub**

GitHub is home to over 50 million developers working together. Join them to grow your own development teams, manage permissions, and collaborate on projects.

[Sign up](#)

Dismiss

Pinned repositories

- html**

HTML Standard

HTML ★ 3.5k 🍴 1.2k
- streams**

Streams Standard

HTML ★ 993 🍴 107
- fetch**

Fetch Standard

HTML ★ 1.5k 🍴 191
- dom**

DOM Standard

HTML ★ 951 🍴 186
- url**

URL Standard

HTML ★ 299 🍴 108
- encoding**

Encoding Standard

HTML ★ 152 🍴 59

Find a repository...

Type: All

Language: All



# The Development of HTML

- HTML 2.0
  - November 24 1995
- HTML 3.2
  - January 14 1997
- HTML 4.0
  - December 18 1997
- HTML 4.01
  - December 24 1999
- HTML 5
  - October 28 2014
- HTML 5.1
  - November 1, 2016
- **HTML 5.2**
  - December 14, 2017



# Overview

# Web Systems Programming

# Writing and Running Computer Code

- Computer programming can use a word processor (such as Windows Notepad – **NOT Microsoft Word**)
- All ‘real-world’ practical programming uses an IDE and programming language framework
  - An IDE is NetBeans
  - A language framework is
    - A set of libraries and templates (re-usable code modules)
    - A set of *application programming interfaces* (termed an API)
- In this course you will be using the *NetBeans IDE* with the related *language framework*



# Program Language Framework

- In this course you will be using the NetBeans IDE
- NetBeans has language frameworks for programming languages
- The following slide shows:
  - The NetBeans IDI interface for JavaScript
  - Shown is an example of the framework available APIs where the available APIs shown with the details of the API
  - The APIs are shown in the dropdown list accessed using the *dot* ( . ) operator
  - Selecting an API plus the dot operator will show other related APIs

Projects Services Files

JS\_Event\_onclick

- Site Root
- index.html
- Unit Tests

Navigator

JavaScript

- displayDate(): undefined
- document

CSS

HTML

index.html

Source

History

```
1 <!DOCTYPE html>
2 <!--
3 A JavaScript file impl
4 -->
5 <html>
6 <head>
7 <title>JavaScript
8 <meta charset=
9 <meta name="vi
10 </head>
11 <body>
12 <p>Click the b
13 <button onclick
14 <p id="demo"><
15 <script>
16 function d
17 document
18 document.g
19 }
20 </script>
21 </body>
22 </html>
23
24
25
```

document.getElementById(elementId)

Returns the Element that has an ID attribute with the given value. If no such element exists, this returns null. If more than one element has an ID attribute with that value, what is returned is undefined. The DOM implementation is expected to use the attribute Attr.isId to determine if an attribute is of type ID. Note: Attributes with the name "ID" or "id" are not of type ID unless so defined.

#### Parameters:

String elementId The unique id value for an element.

#### Returns:

- getElementById(elementId: String): Eleme... JS Platform
- getElementsByClassName(names: String): N... JS Platform
- getElementsByName(elementName: String): ... JS Platform
- getElementsByTagName(tagname: String): N... JS Platform
- getElementsByTagNameNS(namespaceURI: Str... JS Platform
- getFeature(feature: String, version: Str... JS Platform
- getOwnPropertyDescriptor(obj: Object, pr... JS Platform
- getOwnPropertyDescriptors(obj: Object): ... JS Platform
- getOwnPropertyNames(obj: Object): Array JS Platform
- getOwnPropertySymbols(obj: Object): Arra... JS Platform
- getPrototypeOf(obj: Object): Object JS Platform
- getSelection(): Selection JS Platform
- getUserData(key: String): DOMUserData JS Platform

# Programming Templates

PHP\_Random\_Number - Apache NetBeans IDE 11.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Projects Services Files

PHP\_Random\_Number

- Source Files
  - index.php
- Include Path
  - C:\xampp

Navigator

- html
  - head
    - meta
    - title
  - body

Filters: [Icons]

index.php

Source History

```
1 <!DOCTYPE html>
2 <!--
3 To change this license header, choose License Headers in Project Properties.
4 To change this template file, choose Tools | Templates
5 and open the template in the editor.
6 -->
7 <html>
8   <head>
9     <meta charset="UTF-8">
10    <title></title>
11  </head>
12  <body>
13    <?php
14      // put your code here
15    ?>
16  </body>
17 </html>
18
```

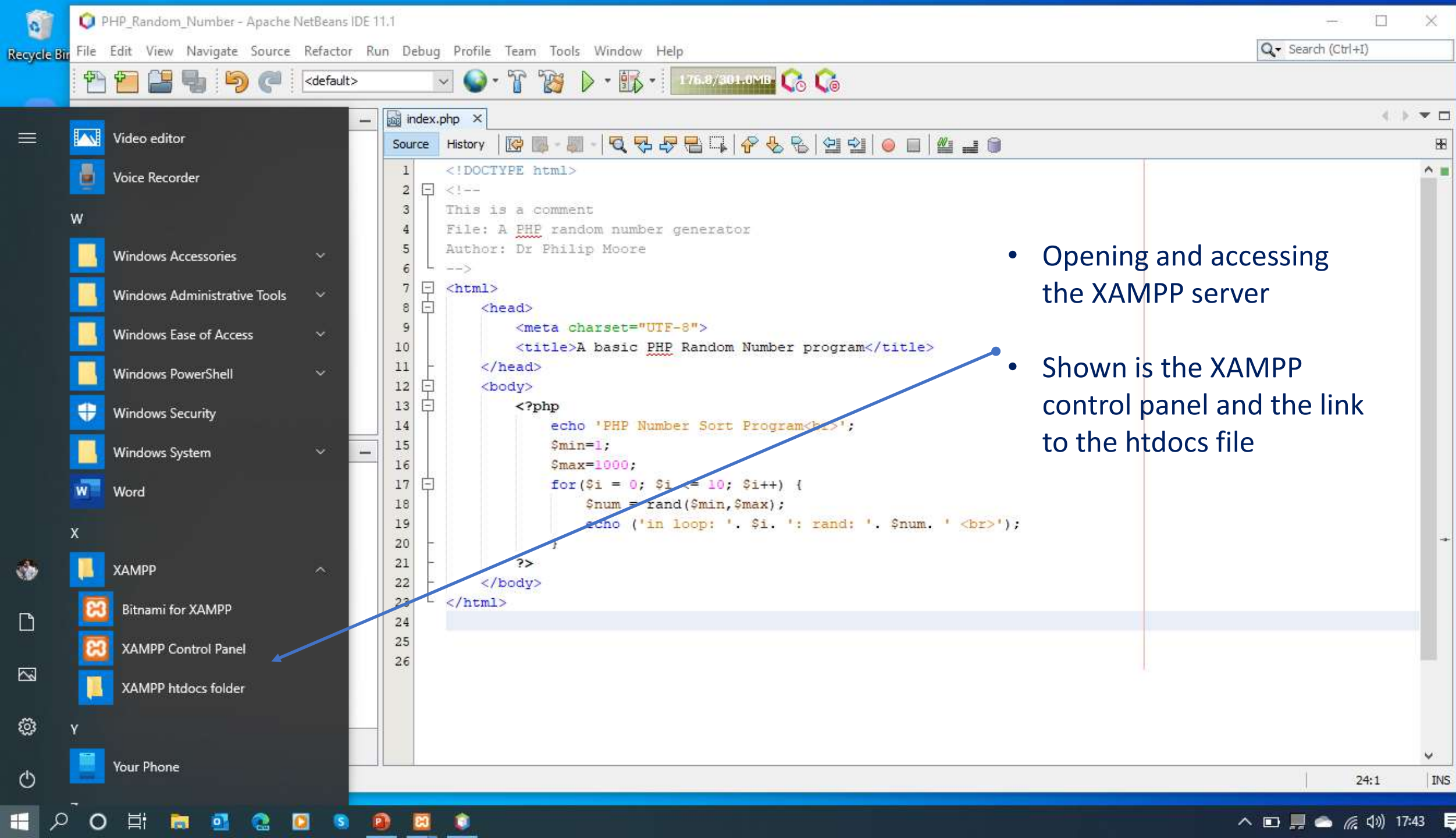
- This slide shows the basic NetBeans PHP template which you can use and change with your PHP program code
- There are similar templates for the other programming languages

Background scanning of projects... 60%

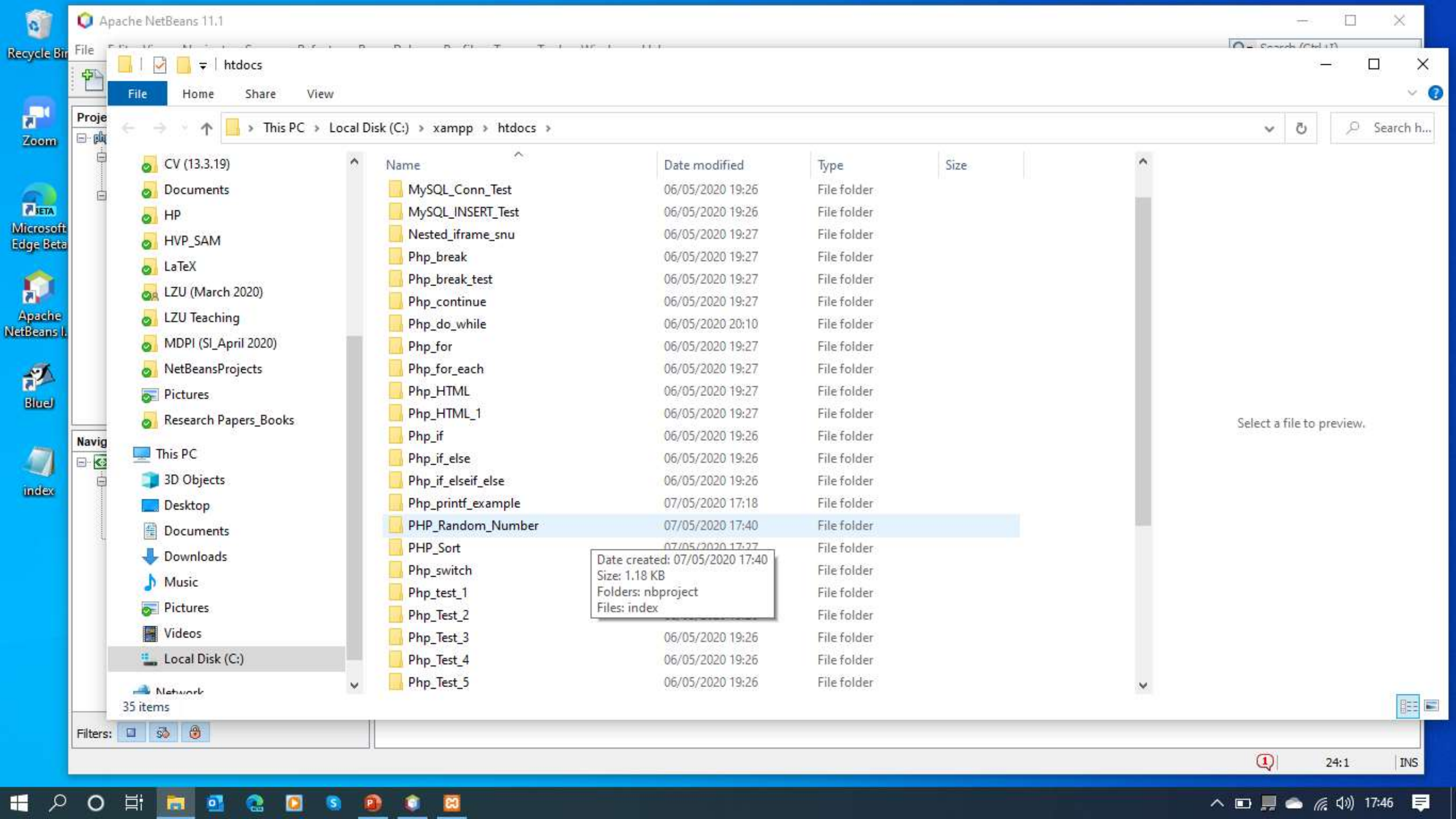
1:1 INS

# NetBeans IDE and XAMPP server





- Opening and accessing the XAMPP server
- Shown is the XAMPP control panel and the link to the htdocs file



Date created: 07/05/2020 17:40  
Size: 1.18 KB  
Folders: nbproject  
Files: index

PHP\_Random\_Number - Apache NetBeans IDE 11.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

194.0/301.0MB

Projects Services Files

PHP\_Random\_Number

- Source Files
  - index.php
- Include Path
  - C:\xampp

Navigator

- html
  - head
    - meta
    - title
  - body

Filters: [ ] [ ] [ ]

index.php

```
1 <!DOCTYPE html>
2 <!--
3 This is a comment
4 File: A PHP random number
5 Author: Dr Philip Moore
6 -->
7 <html>
8 <head>
9 <meta charset="UTF-8">
10 <title>A basic PHP
11 </head>
12 <body>
13 <?php
14 echo 'PHP Num
15 $min=1;
16 $max=1000;
17 for($i = 0; $i < $max; $i++)
18     $num = rand($min, $max);
19     echo ($i
20 }
21 ?>
22 </body>
23 </html>
24
25
26
```

XAMPP Control Panel v3.2.4 [ Compiled: Jun 5th 2019 ]

XAMPP Control Panel v3.2.4

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	13292 4692	80, 443	Stop Admin Config Logs
<input type="checkbox"/>	MySQL	6652	3306	Stop Admin Config Logs
<input type="checkbox"/>	FileZilla			Start Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat			Start Admin Config Logs

17:43:53 [main] All prerequisites found  
17:43:53 [main] Initializing Modules  
17:43:53 [main] Starting Check-Timer  
17:43:53 [main] Control Panel Ready  
17:44:05 [Apache] Attempting to start Apache app...  
17:44:06 [Apache] Status change detected: running  
17:44:06 [mysql] Attempting to start MySQL app...  
17:44:07 [mysql] Status change detected: running



PHP\_Random\_Number - Apache NetBeans IDE 11.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Projects Services Files

PHP\_Random\_Number

- Source Files
  - index.php
- Include Path
  - C:\xampp

Navigator

- html
  - head
    - meta
    - title
  - body

Filters: [ ] [ ] [ ]

index.php

Source History

```
1 <!DOCTYPE html>
2 <!--
3 This is a comment
4 File: A PHP random number generator
5 Author: Dr Philip Moore
6 -->
7 <html>
8 <head>
9 <meta charset="UTF-8">
10 <title>A basic PHP Random Number program</title>
11 </head>
12 <body>
13 <?php
14 echo 'PHP Number Sort Program<br>';
15 $min=1;
16 $max=1000;
17 for($i = 0; $i <= 10; $i++) {
18     $num = rand($min,$max);
19     echo ('in loop: '. $i. ': rand: '. $num. ' <br>');
20 }
21 ?>
22 </body>
23 </html>
24
25
26
```

- A simple PHP program
- Note: the `<?php ... ?>` tags
- Creating and running this PHP program will be shown in this week's laboratory tutorial

PHP\_Random\_Number - Apache NetBeans IDE 11.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

222.5/301.0MB

Search (Ctrl+I)

Projects Services Files

PHP\_Random\_Number

- Source Files
  - index.php
- Include Path
  - C:\xampp

Navigator

- html
  - head
    - meta
    - title
  - body

Filters: [Icons]

Browser

With NetBeans Connector

- Embedded WebKit Browser
- Chrome
- IDE's default browser
- Chrome
- Internet Explorer
- Microsoft Edge

index.php

Source

```
1 <
2 <
3 TR
4 F
5 A
6
7 <
8
9
10
11
12
13
14 per Sort Program<br>';
15
16
17 for($i = 0; $i <= 10; $i++) {
18     $num = rand($min,$max);
19     echo ('in loop: '. $i. ': rand: '. $num. ' <br>');
20 }
21 ?>
22 </body>
23 </html>
24
25
26
```

generator

-8">

Random Number program</title>

per Sort Program<br>';

for(\$i = 0; \$i <= 10; \$i++) {

\$num = rand(\$min,\$max);

echo ('in loop: '. \$i. ': rand: '. \$num. ' <br>');

}

?>

</body>

</html>

24:1 INS

PHP\_Random\_Number - Apache NetBeans IDE 11.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window

Projects Services Files

PHP\_Random\_Number

- Source Files
  - index.php
- Include Path
  - C:\xampp

Navigator

- html
  - head
    - meta
    - title
  - body

Filters: [ ] [ ] [ ]

A basic PHP Random Number Generator

localhost/PHP\_Random\_Number/index.php

### PHP Number Sort Program

```
in loop: 0: rand: 462
in loop: 1: rand: 127
in loop: 2: rand: 277
in loop: 3: rand: 362
in loop: 4: rand: 360
in loop: 5: rand: 67
in loop: 6: rand: 826
in loop: 7: rand: 559
in loop: 8: rand: 895
in loop: 9: rand: 315
in loop: 10: rand: 554
```

```
1 <!DOCTYPE html>
2 <!--
3 This is a comment
4 File: A PHP random number generator
5 Author: Dr Philip Moore
6 -->
7 <html>
8 <head>
9 <meta charset="UTF-8">
10 <title>A basic PHP Random Number Generator</title>
11 </head>
12 <body>
13 <?php
14 echo 'PHP Random Number Generator';
15 $min=1;
16 $max=1000;
17 for($i = 0; $i < 10; $i++)
18 {
19     $num = rand($min,$max);
20     echo ('in loop: '. $i. ': rand: '. $num. ' <br>');
21 }
22 ?>
23 </body>
24 </html>
```



PHP\_Random\_Number - Apache NetBeans IDE 11.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

173.0/344.0MB

Projects Services Files

PHP\_Random\_Number

- Source Files
  - index.php
- Include Path
  - C:\xampp

Navigator

- html
  - head
    - meta
    - title
  - body

Filters: [ ] [ ] [ ]

Browser

With NetBeans Connector

- Embedded WebKit Browser
- Chrome
- IDE's default browser
- Chrome
- Internet Explorer
- Microsoft Edge

index.php

Source

```
1 <?php
2 <
3 TR
4 F
5 A
6
7 <?
8
9
10
11
12
13
14
15
16
17 for($i = 0; $i <= 10; $i++) {
18     $num = rand($min,$max);
19     echo ('in loop: '. $i. ' : rand: '. $num. ' <br>');
20 }
21 ?>
22 </body>
23 </html>
24
25
26
```

generator

-8">

Random Number program</title>

per Sort Program<br>';

for(\$i = 0; \$i <= 10; \$i++) {

\$num = rand(\$min,\$max);

echo ('in loop: '. \$i. ' : rand: '. \$num. ' <br>');

}

?>

</body>

</html>

24:1 INS

Apache NetBeans 11.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Projects Services Files

PHP\_Random\_Number

- Source Files
  - index.php
- Include Path
- C:\xampp

index.php

Source History

```
1 <!DOCTYPE html>
2 <!--
3 This is a comment
4 File: A PHP random number generator
5 Author: Dr Philip Moore
6 -->
7 <html>
8 <head>
9 <meta charset="UTF-8">
10 <title>A basic PHP Random Number program</title>
11 </head>
```

A basic PHP Random Number... X

http://localhost/PHP\_Random\_Number/index.php

100%

PHP Number Sort Program

in loop: 0: rand: 89  
in loop: 1: rand: 678  
in loop: 2: rand: 125  
in loop: 3: rand: 790  
in loop: 4: rand: 536  
in loop: 5: rand: 52  
in loop: 6: rand: 220  
in loop: 7: rand: 346  
in loop: 8: rand: 646  
in loop: 9: rand: 202  
in loop: 10: rand: 191

Filters: [ ] [ ] [ ]

# Computer Programming

# Computer Programming

- There are only three operations in any computer program
- ***Sequential*** operations:
  - Run one-after another without repetition
- ***Selection*** operations (conditional actions):
  - Select a course of action from several available alternatives
- ***Iteration*** operations (repeating actions):
  - Repeat until a condition is satisfied or termination criteria is reached
- A computer program mixes the three processes to achieve a desired result
- In this course we will learn how to use the three operations in JavaScript and PHP

# Web Systems Programming (1)

- There are many programming languages used in web programming including
  - *JavaScript / PHP / Python / Ruby / Java / Scala (a fork of Java) / C / C++ / Perl ...*
  - We will introduce HTML 5 (this is not a programming language)
- In this course we only consider
  - *JavaScript and PHP with MySQL*
- The other programming languages are beyond the scope of this course



# Web Systems Programming (2)

- When we consider web systems programming we will consider two approaches
  - **Client-side** programming
    - The program code is located and run within a web browser
    - The results are displayed by the web browser
    - A web-browser is termed a thin client
  - **Server-side** programming
    - The program code is located and run in a web server
    - The results are returned and displayed by the web browser
- *Client-side* programming uses HTML and JavaScript
- *Server-side* programming uses HTML and PHP
- HTML, JavaScript, and PHP may be used in a single program

# Web Systems Programming (3)

- *JavaScript* is a *client-side* programming language
- *PHP* is a *server-side* programming language
- JavaScript can not connect directly to MySQL server
  - There are exceptions which require '*bridging*' such as
    - Node.js (or)
    - RESTful application programming interface (API)
- To connect to the database server (and MySQL)
  - We must use PHP
- In this course we will connect to MySQL server using PHP

# Program Code Documentation

# Documenting Program Code

- In preparing an HTML file with JavaScript, PHP, and MySQL
  - It is important that the program code is documented
- By documentation we refer to *inserting comments* in the code
- Proper documentation is important because:
  - Comments provide information to understand the purpose and reason for the program code
  - Understanding is important for multiple programmers
  - Understanding is also important to remind the programmer of program logic and purpose
- Proper documentation is very important in the software life cycle
  - To enable maintenance and updating of software

# Example HTML Web Page File (template)

- Sample HTML File
- The basic HTML shows:
  - HTML <tags>
  - How to insert comments
  - Indented program code
- Note:
  - The syntax is specific to HTML
  - Other programming languages use a different syntax

```
<!-- this is a comment -->
```

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>First Web Page</title>
```

```
  </head>
```

```
  <body>
```

```
    <p>
```

```
      <!-- Insert text here -->
```

```
      My first web page!
```

```
    </p>
```

```
  </body>
```

```
</html>
```

# Web Systems Programming and HTML

# Web Systems Programming and HTML

- **HTML** is not a programming language – it is a markup language
- In this course we will consider HTML
  - HTML is the structure within which JavaScript and PHP is embedded
- The current version is HTML 5
  - HTML 5 builds on HTML 4 and HTML 5 replaces some <tags>
  - The <tags> for HTML 4 remain in HTML 5
  - There are <tags> in HTML 4 which are **deprecated** in HTML 5
  - In current web-browsers the implementation of HTML 5 is variable

# Hypertext and HTML

- The term *Hypertext*
  - Originally referred to text stored in electronic form with cross-referenced links between pages.
  - It has developed into a broader term that refers to: objects (text, images, files, etc) that can be linked to other objects.
  - HTTP is a language for describing how text, graphics, and files containing other information are organized and linked.
- The *Hypertext Markup Language* (HTML) is a standard '*markup language*' for the creation of web pages and web-systems applications.
- When used with *Cascading Style Sheets* (CSS) and scripting languages it forms the main technologies of the *Internet*



# Web-Systems Markup Languages

# Web-Systems Markup Languages

- There are 3 web-systems *markup-languages* the:
  - *HyperText Markup Language* (HTML)
  - *eXtensible Markup Language* (XML)
  - *eXtensible Hypertext Markup Language* (XHTML)

# HTML - XML

- HTML
  - Is a legacy technology of the World Wide Web Consortium (W3C)
- XML
  - Is a W3C standard for text document markup
  - NOT a language itself
  - A set of rules for creating other markup languages
  - Specific type definitions can be created using **Document Type Definition** (DTD)
- The design goals for *XML* are:
  - Simplicity, generality, and usability across the Internet
  - XML focuses on documents
  - XML is widely used to represent data structures as used in web-systems and services

# Document - Type Definition (DTD)

- The W3C Recommendation:
  - Use a Document Type Definition to identify the type of markup language used in a web page
- XHTML 1.0 Transitional
  - This is the least strict specification for XHTML 1.0.
  - It allows the use of both Cascading Style Sheets and traditional formatting instructions such as fonts
- XHTML 1.0 Strict
  - Requires exclusive use of Cascading Style Sheets
- XHTML 1.0 Frameset
  - Required for pages using XHTML frames (deprecated in HTML 5)

# XHTML 1.0 Transitional DTD

- The XHTML version and type is declared in the Document Type Definition (DTD) called the !DOCTYPE
  - The !DOCTYPE Identifies the document type to browsers and XHTML code validators
  - The second line of your XHTML page:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!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">  an opening tag
```

```
.... page info goes here
```

```
</html> a closing tag
```

# XHTML

- In *eXtensible Hypertext Markup Language (XHTML)* the W3C
  - Adds the structure and *extensibility* of the *Extensible Markup Language (XML)* to *HTML*
- *XHTML*
  - Uses the elements and attributes of HTML (with)
  - the syntax of XML
  - *XHTML* is compatible with *XML*
- *XHTML* is a '*markup language*' that
  - defines a set of rules for encoding documents in a format in both human-readable and machine-readable formalism

# XHTML

- XHTML 1.0
  - It is the latest version of HTML
  - It is a W3C specification (plus related specifications) are free open standards that define a common XML standard
  - Allows only a document's content and structure to appear in a valid XHTML document, and not its formatting
  - Formatting is specified with Cascading Style Sheets (CSS)
- XHTML uses:
  - The **elements** and attributes of HTML
  - With the syntax of the *eXtensible Markup Language* (XML)

# XML Syntax

- An XML document must be *well-formed* and must:
  - Use lowercase for <tags>
  - Use opening and closing <tags>
    - `<body> ... </body>`
  - A closed stand-alone <tag> with special syntax
    - `<hr />`
- XML documents begin with an XML directive.
  - The basic form of this directive is:
    - `<?xml version="1.0" encoding="UTF-8"?>`
- This is the first line of your XHTML file



# XML Syntax Structure

- XML has the following structure:
  - `<element1  
  attribute1="value">content</element1>`
  - `<element1 attribute1="value"></element1>`
  - `<element1 attribute1="value" />`

# XML Syntax

- In XML there are *elements* in a document
- The following example demonstrates elements and markup <tags>:
  - <element1 attribute1="value">content</element1>
  - **<food type="fruit" color="red">A Red Apple</food>**
- Examples of markup (Tag), Content, and Attribute can be found at [w3schools.com](http://w3schools.com)

# HTML 5

- HTML 5 is currently
  - The latest HTML standard (2018) is HTML 5.2
  - Is generally supported by the major web-browsers (we will consider this later)
  - Provides improved integration of multimedia
  - HTML 5 is easy to work with
  - No XML directive and lengthy DOCTYPE declaration is required

# HTML 5

- HTML 5 requires:
  - No XML directive
  - No lengthy DOCTYPE declaration
- The syntax is similar to (nearly identical) XHTML

```
<!DOCTYPE html>
```

```
<html>
```

```
...
```

```
</html>
```

# Document Object Model (DOM)

# Document Object Model (DOM)

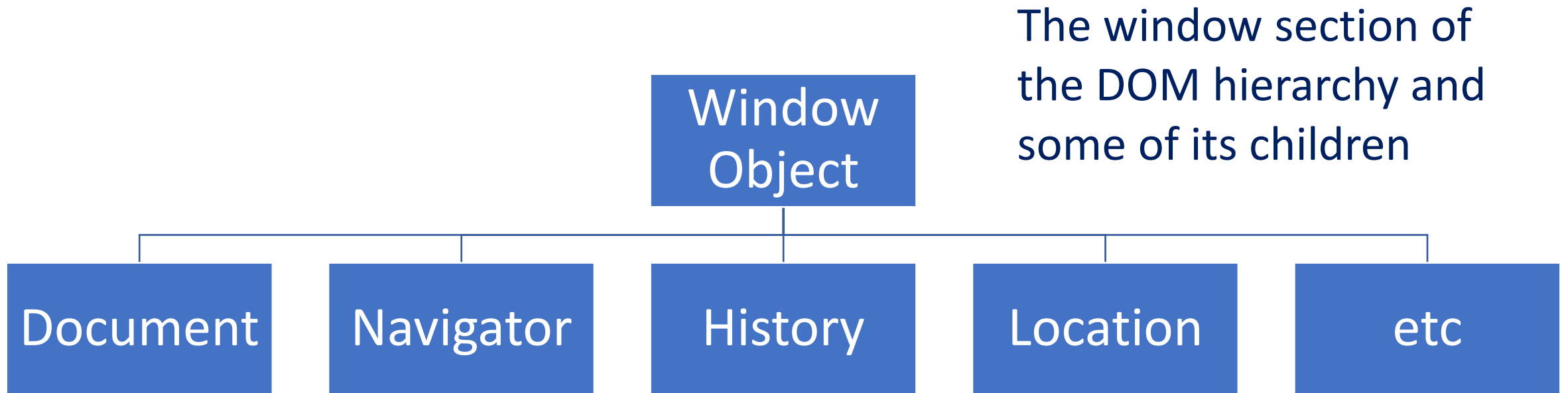
- The DOM is a World Wide Web Consortium (W3C) standard for accessing documents
- The W3C Document Object Model (DOM)
  - A platform and *language-neutral* (e.g., Chinese) interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document
- The W3C DOM standard is separated into 3 different parts:
  - Core DOM - standard model for all document types
  - XML DOM - standard model for XML documents
  - HTML DOM - standard model for HTML documents

# HTML DOM

- The HTML DOM is a standard object model and programming interface for HTML which defines:
  - The HTML elements as objects
  - The properties of all HTML elements
  - The methods to access all HTML elements
  - The events for all HTML elements
- In simple terms: the HTML DOM is a standard for how to *get*, *change*, *add*, or *delete* HTML elements
- The DOM and JavaScript will be introduced in later weeks

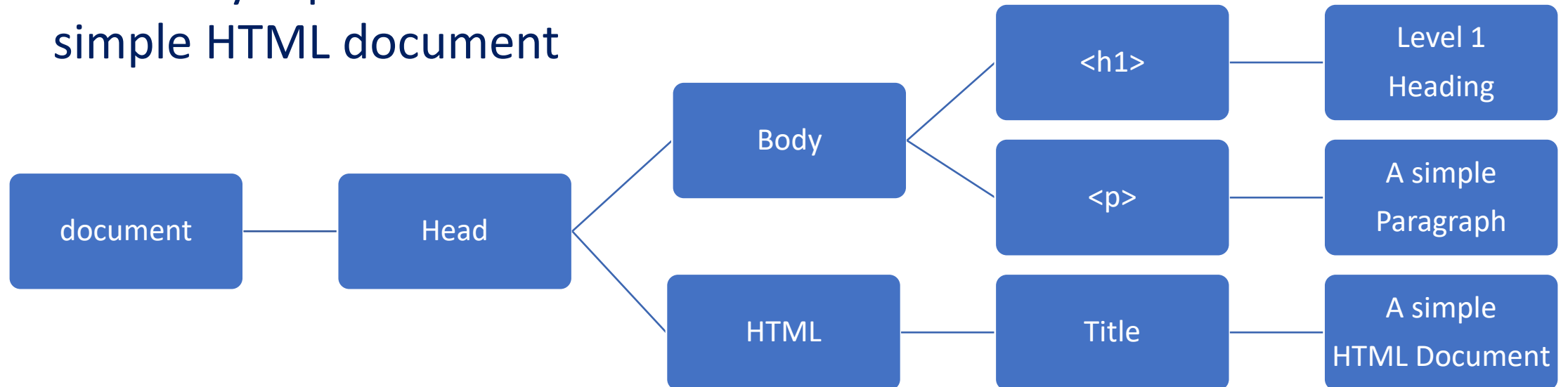


# HTML DOM Hierarchy



# HTML DOM Example

How the HTML DOM hierarchy represents a simple HTML document



# First HTML Web Page

- Creating a Sample HTML File
- This simple HTML code prints
  - “My first web page!”
- You will learn about HTML
- You will learn how to add style to web-pages using
  - Cascading Style Sheets (CSS)
- The following slide shows the output in a web browser

```
<!-- this is a comment -->
<!DOCTYPE html>
<html>
<head>
<title>First Web Page</title>
</head>
<body>
<p>
My first web page!
</p>
</body>
</html>
```

My first web page!

# Web Systems Browser Support

# Web Systems and Web-Browser Support

- Web-based systems must accommodate
  - A range of browsers
  - Implemented in a range of devices
  - A range of screen resolutions
- The support for HTML 4 is achieved in all major browsers
- The support for HTML 5 is not universal
  - Web-pages and web-sites must
    - Incorporate HTML that enables the web-pages web-sites to be viewed in a range of current and older browsers and devices

# HTML5 TEST how well does your browser support html5?

your browser other browsers compare

news device lab about the test

desktop browsers tablets mobiles other

latest search

## OVERVIEW

	Chrome	Opera	Firefox	Edge	Safari
Upcoming	68 528		60 497	18 496	11.2 477
Current	66 528	45 518	59 491	17 492	11.1 471
Older	65 528	37 489	58 486	16 476	11 452
	64 528	30 479	57 486	15 473	10.1 406
	63 528	12.10 309	56 478	14 460	10.0 383
	62 528		55 478	13 433	9.1 370
	61 526		54 474	12 377	9.0 360
	60 523		53 474	Internet Explorer	8.0 354
				11 312	



HTML5 TEST how well does your browser support html5?

your browser other browsers compare

news | device lab | about the test

desktop browsers tablets mobiles other latest search

## OVERVIEW

	Chrome	Opera Mobile	Edge	iOS	Firefox Mobile
Upcoming			18 496		
Current		37 481	17 492	10.3 415	
Older	52 486	12.10 289	16 476	10.2 391	48 466
	51 488		15 473	10.0 389	47 459
			14 460	9.3 378	
			13 433	9.0 368	
			12 377	8.0 360	
			Internet Explorer	7.0 327	
			11 312		

File

Paste

Clipboard

39

40

41

42

43

44

# HTML5 TEST how well does your browser support html5?

your browser other browsers compare

news device lab about the test

desktop browsers tablets mobiles other

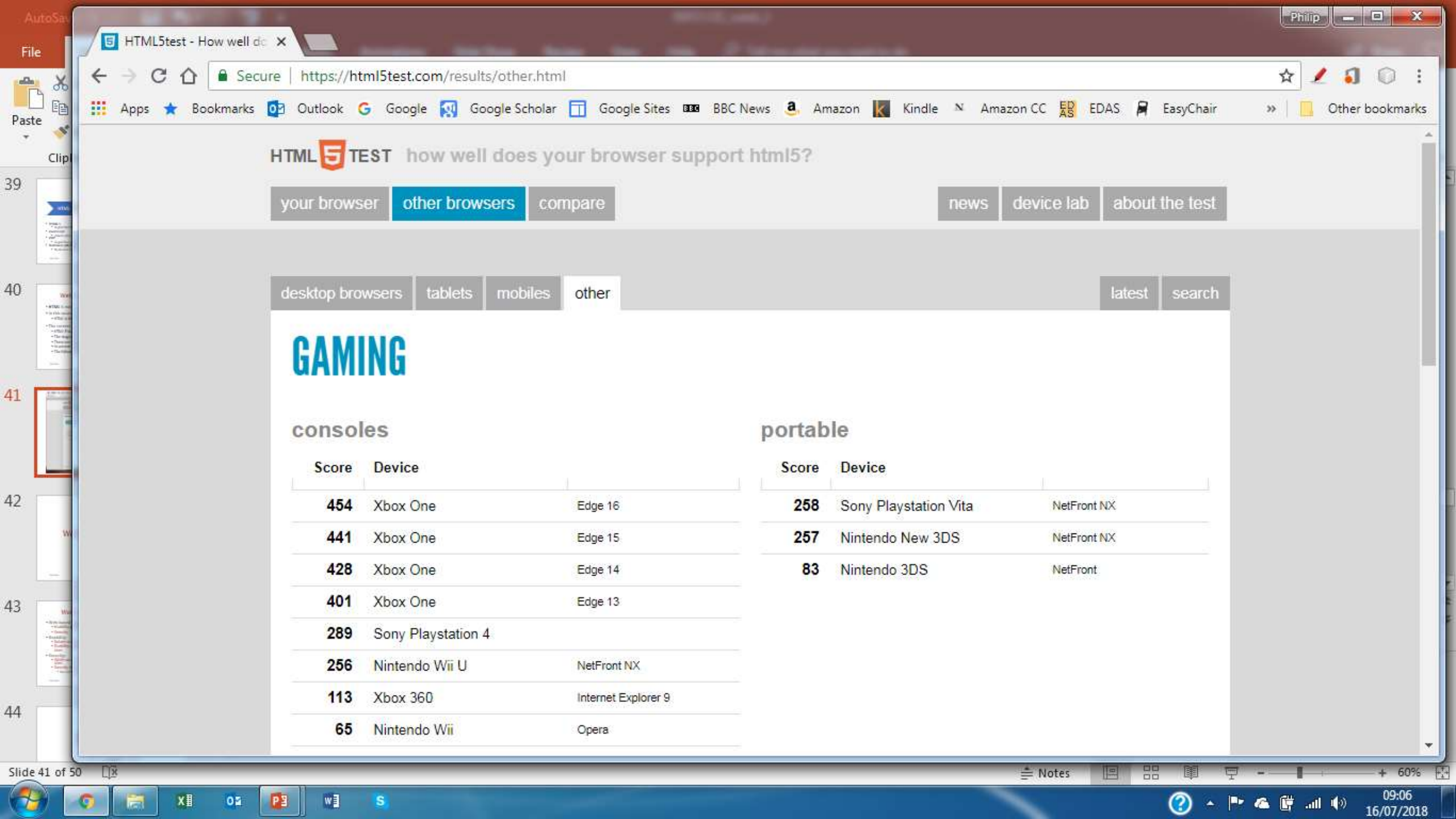
latest search

## OVERVIEW

	Chrome	Samsung Internet	Opera Mobile	Edge	iOS	Firefox Mobile
Upcoming						
Current		6.2 517	37 481	14 444	10.3 415	
Older	52 486	2.0 418	12.10 289	13 417	10.2 391	48 466
	51 488	3.0 444		Internet Explorer	10.0 389	47 459
		5.4 490		11 310	9.3 378	
		5.0 484		10 263	9.0 368	
		4.0 469		9 113	8.0 360	
					7.0 327	

Screenshot saved

Notes



## HTML5 TEST how well does your browser support html5?

your browser other browsers compare

news device lab about the test

desktop browsers tablets mobiles other

latest search

## GAMING

## consoles

Score	Device	
454	Xbox One	Edge 16
441	Xbox One	Edge 15
428	Xbox One	Edge 14
401	Xbox One	Edge 13
289	Sony Playstation 4	
256	Nintendo Wii U	NetFront NX
113	Xbox 360	Internet Explorer 9
65	Nintendo Wii	Opera

## portable

Score	Device	
258	Sony Playstation Vita	NetFront NX
257	Nintendo New 3DS	NetFront NX
83	Nintendo 3DS	NetFront

# Conclusion

- We have provided:
  - An introduction to web systems programming and the core technologies
  - An overview of the history of the *Internet* and *hypertext*
  - An introduction the HTML and related technologies
- In this week's practical laboratory session we will:
  - Introduce the course software
  - Introduce the XAMPP integrated server environment
  - Create and run a simple '*stateless*' html web page