**Course Outline**

In this course you will learn three key website programming and design languages: HTML, CSS and JavaScript.

You will create a web page using basic elements to control layout and style. Additionally, your web page will support interactivity.

At the end of the course, you will be able to:

1. define the purpose of HTML, CSS and JavaScript

2. make a simple web page using HTML

3. use CSS to control text styles and layout

4. use CSS libraries such as bootstrap to create responsive layouts

5. use JavaScript variables and functions

6. manipulate web page content using JavaScript

7. respond to user input using JavaScript

In this course, you will complete:

2 assignments writing HTML, CSS and JavaScript, each taking ~1 hour to complete

2 quizzes, each taking ~20 minutes to complete

multiple practice quizzes, each taking ~5 minutes to complete

### ****Grading and assessment information****

After each lesson there are practice quizzes which will test your knowledge from the video lecture you have just watched. Your attempts at these quizzes will not count towards your final grade for the course and will not affect your ability to achieve the course certificate.

**Certificate**

In order to achieve the course certificate you must complete all module summary quizzes and both peer assessments.

All modules have a graded quiz which each count towards 10% of your overall grade for the course.

There is a peer assessment at the end of modules 1.2 and 1.4 which each count towards 30% of your overall grade for the course.

The pass mark for this course is 65%.

# URLs for this lecture

### ****URLs for this lecture****

Here are the URLs Kate will be referring to in her lecture on setting up a dev environment:

1. At 1.57 : <http://www.w3.org/History/19921103-hypertext/hypertext/WWW/TheProject.html>
2. At 2.39: **Plain text editors**:

**TextWrangler** (OSX) <http://www.barebones.com/products/textwrangler/>

**Notepad++** (PC) <https://notepad-plus-plus.org/>

**Sublime Text** (OSX, Windows, Linux) <http://www.sublimetext.com/>

**Atom** (OSX, Windows, Linux) <https://atom.io>

# URL for this lecture

### ****URL for this lecture****

Here is the URL to the world's first web page that Matthew will be looking at in the following lecture:

<http://info.cern.ch/hypertext/WWW/TheProject.html>

Welcome to the second module of 'Responsive website basics'.

In this section of the course we will have a look at linking external CSS files to your HTML documents, controlling fonts with CSS and using CSS to customise hyperlink formatting and to control text layout. We will also install the bootstrap library and implement a responsive grid layout- Enjoy!

**Learning Objectives**

* Use external CSS files
* Modify fonts using CSS
* Describe what responsive layout is
* Use the bootstrap library to make a responsive grid

# URL for this lecture

### ****URL for this lecture****

Download Bootstrap: <http://getbootstrap.com/>

When prompted to in the following lecture, copy the bootstrap.css file from the CSS folder into your project directory.

In this module we will write simple JavaScript programs and learn how to write programs that can respond to user input such as clicking on HTML elements. We will also take a look at JavaScript functions and use jQuery to manipulate web pages. Finally, you will learn how to write your own javaScript functions including anonymous functions.

 Write programs that can respond to user input

 Use Javascript functions

 Use jquery to manipulate web pages

 Write new Javascript functions including anonymous functions

# JavaScript Resources

### JavaScript Resources

JavaScript is a complex language that would be impossible to teach entirely in a course such as this, but there are many additional resources that you can use to learn more.

This online book is very good:

<http://eloquentjavascript.net>

The w3schools have good tutorials:

<http://www.w3schools.com/js/>

as does Mozilla:

<https://developer.mozilla.org/en-US/Learn/JavaScript>

There are many, many more resources out there and a web search should quickly throw up answers to most questions

<head>

</head>

<body>

    <!-- The onclick attribute is the code

             that happens when the element is clicked.

             The value of the attribute is some

             javascript code.

             In this case it creates an alert

             pop up dialog

    -->

    <h1 id="title" onclick="alert('hello');" >

            Hello

    </h1>

</body>

<head>

</head>

<body>

    <!--

        In this case the onclick functions

        writes something to the console.

        The console is an object so we use the

        dot (.) notation to call a function

        on it

    -->

    <h1 id="title"

        onclick="console.log('hello');" >

            Hello

    </h1>

</body>

# jQuery

### jQuery

In this course we will be using the library jQuery to manipulate HTML DOM. While it is possible to do this in plain JavaScript, jQuery makes it much easier.

The main jQuery site and download is here:

<https://jquery.com>

jQuery has very extensive documentation, which you can find here:

<http://api.jquery.com>

Having said that, the API documentation isn't the best place to start learning. The w3schools have a number of tutorials:

<http://www.w3schools.com/jquery/>

as well as the jQuery site itself:

<https://learn.jquery.com>

<head>

   <!-- to use jQuery we need to import it like this -->

   <script src=" http://code.jquery.com/jquery-1.11.3.min.js"></script>

</head>

<body>

    <!--

        in this example use use jQuery to

            change the content itself.

            The $ is shorthand for the jQuery function

         We are passing in a CSS selector which

         gets this element by its id.

         The html function sets the html content

         of an element

    -->

    <h1 id="title" onclick="$('#title').html('Goodbye');">

        Hello

    </h1>

</body>

<head>

       <script src = "http://code.jquery.com/jquery-1.11.3.min.js"></script>

</head>

<body>

    <h1 id="title" onclick="sayHello()"> Hello </h1>

</body>

<!--

    the script tag is where you can put

    more complex scripts

-->

<script type="text/javascript">

    function sayHello() {

        alert('Hello');

    };

</script>

<head>

<script src="http://code.jquery.com/jquery-1.11.3.min.js"></script>

</head>

<body>

    <h1 id="title" onclick="sayGoodbye();"> Hello </h1>

</body>

<script type="text/javascript">

    function sayGoodbye(){

        console.log("goodbye");

        $("#title").html("Goodbye");

        $("#title").click(function () {

            $("#title").html("Hello");

            $("#title").off("click");

        });

    };

</script>

<!DOCTYPE html>

<head>

<script src="http://code.jquery.com/jquery-1.11.3.min.js">

</script>

</head>

<body>

<!-- this element will display the variable. It is empty as it will be filled from javascript. -->

<h1 id="number" onclick="count();">

</h1>

<script type="text/javascript">

// this is a variable, we start it at 0

var counter = 0;

// display the variable in the

// element "number"

$("#number").text(counter);

// when this function is called it runs

// everything inside the curly brackets

function count() {

// firstly one is added to the variable

// ‘counter'

counter = counter + 1;

// set the h1 element (with id “number”)

// with the value of ‘counter'

$("#number").text(counter);

}

</script>

</body>

</html>