Hyperion Development

Task: JavaScript III

www.hyperiondev.com

Introduction

Welcome to The Third JavaScript Task!

In this task you will learn how to apply JavaScript to your html. The task also introduces types of scripting implementation used in JavaScript. You will also learn to write scripts that respond to HTML events.

You don't have to take our courses alone! This course has been designed to be taken with an online mentor that marks your submitted code by hand and supports you to achieve your career goals on a daily basis.

To access this mentor support simply navigate to www.hyperiondev.com/support.

Incorporating JavaScript within HTML

Up until now, you've learnt the very basics of JavaScript as a language, but you haven't yet used it on a web page. Before you can learn more JavaScript, it's important to showcase how it's applied to web pages as a scripting language.

Here is a single JavaScript statement, which creates a pop-up dialog saying "Hello World!":

```
alert("Hello World!");
```

For the browser to execute the statement, it must be placed inside a `<script>` element. This element describes which section of the HTML code contains executable code, and will be described in further detail later.

```
<script type="text/javascript">
  alert("Hello World!");
</script>
```

The `<script>` element should then be nested inside the `<head>` element of an HTML document. Assuming the page is viewed in a browser that has JavaScript enabled, the browser will execute the statement as the page is loading.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>First Script</title>
    <script type="text/javascript">
      alert("Hello World!");
    </script>
  </head>
  <body>
    <p>Page content here</p>
  </body>
</html>
```

The above program can be used as a starting point for any new programs that you develop in the future! Copy and paste the code above into Notepad and save the files as test.html. Make sure you use the extension html. Open the html file to see what this code does.

Note: It's pivotal that you grasp what has just been explained! If not, do some reasearch and some playing around with the code to better understand it. If you are still unsure, contact your mentor.

The Script Element

All JavaScript, when placed in an HTML document, needs to be within a script element. A script element is used to link to an external JavaScript file, or to contain inline scripting. Inline scripts are used when you have small pieces of scripts within the HTML file. A script element to link to an external JavaScript file looks like:

```
<script type="text/javascript" src="script.js"></script>
```

A script element that contains inline JavaScript looks like:

```
<script type="text/javascript">
     Some scripting code here
</script>
```

Inline scripting has the advantage that both your HTML and your JavaScript are in one file, which is convenient for quick development and testing. Having your JavaScript in a separate file is recommended for JavaScript functions though. Functions, which will be discussed soon, that can potentially be used in more than one page. It is advisable to separate content (html) from behaviour (JavaScript). We'll discuss why later but to get into the habit of using external javaScript files, we'll focus on using a separate scripting files whenever possible.

To do this, you'll need to create a script.js file within the folder of the html file that will be calling it. To call the script file use the following code:

```
<script type="text/javascript" src="script.js"></script>
```

JavaScript Events

An event is basically an action that occurs that your program responds to. You have encountered HTML events. HTML events are things that the browser or a user does such as when a web page has completely loaded, an input field was changed, or a button was

clicked. JavaScript is now able to change how those events occur, and actually modify the HTML event actions.

The most common events you'll deal with when getting started are:

Event	Description
onchange	Some HTML element has been modified
onclick	An HTML element has been clicked on
onmouseover	An HTML element was hovered over
onmouseout	Mouse cursor moves off HTML element
onkeydown	A keyboard button is pressed
onload	The HTML page has finished loading

JavaScript contains many HTML methods which it uses to modify the HTML element. A method is a unit of code that contains all the coding instructions needed to accomplish a specific task. An example is the getElementById() method. Here is a basic example of how this method can be used:

```
document.getElementById("greet").innerHTML = "Hello JavaScript";
```

Here the method looks within the document for an HTML element with the id "greet" and changes the content of that element to now be "Hello JavaScript".

Let's now look at using the method on an HTML event - the button:

```
<button onclick="getElementById("time").innerHTML=Date()">What's the
time?</button>
```

Here we have a button which will be displayed. When clicked, it will display the current date and time (thanks to the built in Date() method), which will then be displayed within the HTML element with the id "time".

JavaScript can essentially change any visual property of a webpage. This means that it can also change the CSS of a webpage:

```
<button onclick="document.getElementById(“text”).style.fontSize='40px'">Click
Here To Change The Size!</button>
```

This creates a button which can be clicked to change the font size of some element with an id of “text”. By now you can probably see that it’s possible to edit any kind of styling with this method.

There are many online references with example on more Javascript Event examples. It’s advisable to do some further research as it’s impossible to cover all of them within a task - you’ll also require this do some of the task components.

Compulsory Task

Follow these steps:

- Create a basic HTML file with a few headings, paragraphs, forms, and images. Ensure that each item has an id assigned to it.
- Create buttons to do the following:
  - Change the size of a heading
  - Change the font style of a paragraph
  - Highlight a paragraph
  - Hide an Image
  - Alternate between two images
- Change an image when the mouse moves onto it, and then back to the original image when the mouse moves off of it.
- Create a script which creates an alert on the page when a user right clicks on a specific image
- Create an alert on the page when a user enters data into a form.
- Create an alert to let the user know when the page has loaded
- Highlight all paragraphs when the mouse moves over them (then revert back once it's moved past)

SHARE YOUR THOUGHTS WITH US

Hyperion strives to provide internationally-excellent course content that helps you achieve your learning outcomes. Think the content of this task, or this course as a whole, can be improved or think we've done a good job? Click here to share your thoughts anonymously.