Hyperiondev

TASK

Introduction to jQuery

Visit our website

Hyperiondev Copyright © 2018 Hyperion Development. All rights reserved.

Introduction

Welcome to The First jQuery Task!

In this task, you will learn to use jQuery. You will learn what jQuery is, what it is used for, how it is used and the benefits of using it.

Get in touch

Connect for support

Remember that with our courses, you're not alone! You can contact your mentor to get support on any aspect of your course.

The best way to get help is to login to **www.hyperiondev.com/portal** to start a chat with your mentor. You can also schedule a call or get support via email.

Your mentor is happy to offer you support that is tailored to your individual career or education needs. Do not hesitate to ask a question or for additional support!

Copyright © 2018 Hyperion Development. All rights reserved.

LEARNING ABOUT JQUERY

jQuery is simply a *JavaScript library* which enhances the functionality of JavaScript in a huge way due to its unique simplicity. A JavaScript library is a collection of pre-written JavaScript code that you can reuse to speed up your development. Other benefits of using JQuery:

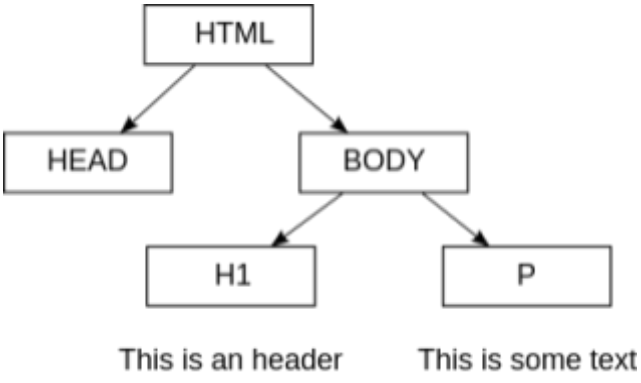
- Simplifies DOM manipulation (don't worry, DOM will be discussed shortly!)
- Simplifies event handling and AJAX calls (you will learn about AJAX later).
- Makes your sites more compatible with various browsers.

jQuery is widely used by many big industry leaders such as Google, Microsoft, IBM, and even Netflix.

THE DOCUMENT OBJECT MODEL (DOM)

The Document Object Model (DOM) is a cross-platform and language-independent application programming interface that treats an HTML, XHTML or XML document as a *tree structure* wherein each node is an object representing a part of the document. The objects can be manipulated programmatically and any visible changes occurring, as a result, may then be reflected in the display of the document.[1]

You will learn what a tree is in greater detail later. For now, consider the image below. In this image, an HTML document is shown as a tree structure. Parent nodes have child nodes, e.g. the "BODY" node is the parent of the "H1" node and the "P" node. "H1" and "P" are sibling nodes.



[1] "Document Object Model - Wikipedia, the free encyclopedia." 2011. 4 Jul. 2016 <https://en.wikipedia.org/wiki/Document_Object_Model>

Copyright © 2018 Hyperion Development. All rights reserved.

LINKING JQUERY TO YOUR HTML

To be able to use jQuery, you can either download the library or simply use Google's hosted version thanks to their own CDN (content delivery network). All that is required is for you to include the following lines in the HTML file of your site:

```
<head>
   <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
</head>
```

One big advantage of using the hosted jQuery from Google is that many users already have downloaded jQuery from Google or Microsoft when visiting another site. As a result, it will be loaded from a cache when they visit your site, which leads to faster loading time. Also, most CDN's will make sure that once a user requests a file from it, it will be served from the server closest to them, which also leads to faster loading time.

JQUERY SYNTAX

To help you better understand the syntax of jQuery, let's consider the following line of code:

```
$(document).ready(something);
```

By breaking down each part of this line, you'll understand just how jQuery works:

- $() - This tells us that something in jQuery is about to happen (think of this as the key to using jQuery).
- (document) - This tells us that some jQuery operation is about to happen on the HTML document itself.
- .ready() - This is a basic function in jQuery (think along the lines of the onload() method from JavaScript) which will do something when the document has loaded.
- ready(something) - This lets us know that some action will be carried out once the document has completely loaded.

The basic jQuery command can be generalised into the format below:

Copyright © 2018 Hyperion Development. All rights reserved.

```
$(selector).action();
```

In this task, you'll learn more about the key features of jQuery:

- Functions
- Selectors
- Events and Effects

FUNCTIONS IN JQUERY

By now, you've most likely become accustomed to using functions. Everything in jQuery happens thanks to functions, thus it's pivotal to understand them as the rest becomes much easier to grasp thereafter.

The most important thing to remember is that your functions should only operate once a document has completely loaded. This makes sense because, as you can imagine, you can't manipulate objects that don't exist! Thus the standard jQuery definition can be seen below:

```
$(document).ready(function(){
  //jQuery events and effects go here
});
```

Here jQuery ensures the document is completely loaded (is ready) before any events take place within the function. A simpler way to do this is:

```
$(function(){
  //jQuery events and effects go here
);
```

This code does the exact same thing as the first code example; it is simply a more efficient way of accomplishing the same task. The choice of function declaration is up to you.

Copyright © 2018 Hyperion Development. All rights reserved.

SELECTORS IN JQUERY

jQuery selectors are one of the most important parts of the jQuery library. They offer you the ability to select and manipulate a range of HTML elements. They are essentially used to find HTML elements based on their name, id, classes, types, attributes, values of attributes and much more, allowing you to have full control over how your web page functions. It's based on the existing CSS Selectors, with the addition of some further customised selectors. All selectors in jQuery start with the dollar sign and parentheses: $(). Let's take a look at some of the most common selectors:

- **Element Selectors:** The element selector selects elements based on the element type. You could select all elements on an HTML page, for example, using the following code:

```
$("div")
```

- **ID Selectors:** The ID selector selects elements based on the element id attribute which was assigned to the element in HTML. For example, you could select a specific element ID (in this case "demoID"), using the following code:

```
$("#demoID")
```

- **Class Selectors:** The class selector selects elements based on the element which class attribute which it has been assigned to. For example, you could select a specific element class (in this case "demoClass"), using the following code:

```
$(".demoClass")
```

- **Other selectors:** The above selectors are the three most commonly used selectors in jQuery, although many variations are created with a combination of these. It's definitely worth playing around with each of these selectors to become familiar with them - this is what leads to creative web pages! Further examples (which include combinational selectors) can be seen below:

Copyright © 2018 Hyperion Development. All rights reserved.

Selector	Description
$(“*”)	Selects all HTML elements
$(this)	Selects current HTML element
$(“div.last”)	Selects element with class=”last”
$(“button”)	Selects button element
$(“[href]”)	Selects all link-based elements

For a full list of interactive selectors, visit this interactive page here.

SEPARATE SCRIPTING FILES

As previously discussed in the JavaScript segment, it’s generally recommended to use a separate script file rather than having the script within your HTML file as this allows for code reusability as well as more efficient code layout. To create a scripting file for your HTML as well as incorporating the jQuery library, simply use the following code:

```
<head>
   <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"> </script>
   <script src="script.js"></script>
</head>
```

A note from our coding mentor
Sarah

Note: There are some good YouTube videos for those learning jQuery here. For more links to good video resources see this Hyperiondev Blog post.

Copyright © 2018 Hyperion Development. All rights reserved.

EVENTS IN JQUERY

Events are represented by some action, and jQuery generates a response at the exact moment that the action occurs. Events can range from a mouse click to a specific keyboard button being pressed by the user. This allows for many customisable animations to be created.

Examples of common events used in jQuery can be found below:

Mouse Events	Keyboard Events	Form Events
Click	Key press	Submit
Double click	Key down	Change
Mouse enter	Key up	Blur

EFFECTS IN JQUERY

Effects can be better described as actions which do something to HTML elements on your web page. They're really straightforward if you're able to grasp all the previous sections, so go back and revise if you feel the need to!

Examples of common effects used in jQuery can be found below:

Effect	Description
Hide	Hides the selected element
Show	Reveals the selected element
Fade	Fades the element in or out
Slide	Slides the element
Animate	Animates some property of the element
Stop	Stops any effect of an element

Though there aren't too many effects, each feature offers countless customisable options to create a unique effect each time!

Copyright © 2018 Hyperion Development. All rights reserved.

EVENT SYNTAX

Effect methods are implemented in the following order:

- Select object where event will occur
- Specify type of event
- Create function to carry out effect
- Specify and customise desired effect

Code to represent this process is demonstrated below:

```
$("p").click(function(){
   $("this").hide('slow');
});
```

In the code above, a function is created whereby, when a <p> element is clicked, it will slowly fade away (take note of the argument).

EFFECT CHAINING

jQuery allows for you to chain actions all into a single command - the possibilities are endless! (Though the commands should make sense when being executed in that order). Below is an example of a chained effect:

```
$("button").css("color", "blue").slideDown(500).slideUp(500);
```

Here we have created a command that will select a button, change its colour, slide it down the page, and then back up (think of an accordion - there's even an accordion function!)

Copyright © 2018 Hyperion Development. All rights reserved.

Compulsory Task

Follow these steps:

- Create a basic HTML file with a few basic elements.
- Import the jQuery library into your HTML document.
- Create a jQuery command that will create an alert once the page has loaded.
- Create a jQuery command that will change the background once the page has loaded.
- Create a jQuery function which changes only a single paragraph's styling
- Create a jQuery function which fades out any object that is clicked on. (You'll need to do a bit of research for this one.)
- Create a jQuery function which creates a drop-down menu with the use of an accordion animation style when each section is hovered over.
- Create a jQuery function which contains a chained effect to slide all the elements of your page around repeatedly whilst changing the background.
- Create a jQuery function which fades a picture in and out over a period of 3 seconds each when the respective buttons are clicked.
- Create a jQuery function which can stop the animation of the above effect whilst in progress.

Once you have completed the task in line with the instructions above, click the button below to request your mentor to review your work and provide feedback. If you have any questions while attempting the task, leave your mentor a note on the comments.txt file in your Student Dropbox folder.

Copyright © 2018 Hyperion Development. All rights reserved.

Completed the task(s)?

Ask your mentor review your work!

Review work

Things to look out for:

1. Make sure that you have installed and setup all programs correctly. Follow the instructions in the FAQs document that accompanies this task to configure Sublime Text for debugging and executing JavaScript.

2. If you are not using Windows, please ask your mentor for alternative instructions if needed.

Rate us

Share your thoughts

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.

Copyright © 2018 Hyperion Development. All rights reserved.