Section 3 Quiz

Polyfills

We've built our very own Polyfill. But as I always say, lets not get lost in the all the detail. What I mean by this is that we sometimes need to take a step back and recap what we've learnt.

A brilliant way to recap is to test yourself.

Good luck.

# Question 1:

What is a polyfill?

1. A “Polyfill” is a way for us to optimize our CSS and HTML code by 'minifying' it so that it can be rendered by the browser faster.
2. A “Polyfill” is a forEach() method that allows us to loop through an array.
3. A “Polyfill” is an hand-made version of a functionality that is not yet implemented in all browsers.

## Answer:

C - Well done. As an interesting side note, before the real Fetch API was released to the public, developers eager to try it installed a Fetch polyfill which enabled the feature. This was done before the “real” Fetch was implemented by browser’s vendors.

# Question 2:

Is Polyfilling limited to only JavaScript?

1. No.
2. Yes.

## Answer:

A - Although Javascript is one of the main technologies that can be used in Polyfilling, it is not limited to just JavaScript. Remember, a Polyfill is using any technology to “fill in the gaps” that older browsers have in order to add support for some functionality. As an interesting side note, the term “poly” in polyfilling is means “many” and the word "many" implies that many different technologies can be used, not just JavaScript.

# Question 3:

Is a Polyfill part of the HTML5 Standard?

1. Yes.
2. No.

## Answer:

B - Remember, a Polyfill is an informal term that’s used to describe some custom specific code that helps you get some missing functionality. And, a Polyfill does not even have to be used just for HTML5 – it could just as well apply to some CSS code to add support for CSS3.

# Question 4:

Are Polyfills only used for older browsers?

1. Yes.
2. No.

## Answer:

B - This is a bit of a tricky question. The practice of Polyfilling is not limited to just adding support for older browsers. It also applies to newer browsers. This is because even newer browsers may not implement all of the features in the HTML5 Standard. What if you are writing an HTML5 application that uses some cool feature that is not present in the latest version of a popular browser (like Firefox or Chrome)? You gusssed it, we must then add some Polyfill code to take care of that.

# Question 5:

1. What are the 2 steps taken when polyfilling?
2. The first step is detecting which features are present in a given browser. If that feature does not exist, then the second step is to "patch” in support with helper scripts (like Javascript) for any missing features in that browser.
3. The first step is to import the jQuery library so you have access to the Polyfill function. The second step is writing your custom code.
4. The first step is to open your console and access the polyfill method. The second step is to implement that polyfill method by passing in the required arguments.

Answer: A