Section 5 - Quiz: Traversing the DOM

Top of Form

Question 1:

The DOM can be represented as a tree of nodes, and often we want to walk around these branches.

Which directions can we traverse the DOM?

* 

**1. Upwards**

**2. Sideways**

**3. Downwards**

* 

**1. Upwards**

**2. Downwards**

* 

**1. Upwards**

**2. Sideways**

Bottom of Form

Answer: A

Top of Form

Question 2:

What are nodes often compared to?

* 

**The nodes in the DOM are referred to as branches, and there is only one type of branch, being an element node.**

* 

**The nodes in the DOM are referred to as *parents*, *children*, and *siblings*, depending on their relation to other nodes.**

* 

**Nodes are referred to as HTML elements, because they represent each and every HTML element in the DOM tree.**

Bottom of Form

Answer: B

Top of Form

Question 3:

What is **true**about this block of code?

1. <body>
2. <div>
3. <h1>Keep going</h1>
4. <h2>Stay motivated</h2>
5. </div>
6. <p> Slowly slowly catch a monkey </p>
7. </body>

* 

**body is the parent of <div>, <h1> and <h2> elements**

* 

**The <p> element is the child of the <div> element**

* 

**<h1> and <h2> are siblings.**

Bottom of Form

Answer: C

Top of Form

Question 4:

What is a similarity between *sibling*methods and *children*methods?

* 

**Both siblings and children have a set of methods that allow us to find an element with either the ID of that element, or its class name.**

* 

**Both siblings and children have a set of methods that allow us to traverse all nodes, and a set of methods that allow us to traverse only element nodes.**

* 

**They both allow us to traverse UP the DOM tree, but not down.**

Bottom of Form

Answer: B

Top of Form

Question 5:

Which of the below methods is **NOT**a valid method to find a child of a node?

* 

**lastChild**

* 

**firstElementChild**

* 

**childrenNodes**

* 

**childNodes**

Bottom of Form

Answer: C

Top of Form

Question 6:

When you access the ***children*** property on a parent, you get returned a collection of the child elements the parent has.

For example, let's take this code:

1. <body>
2. <ul>
3. <li>Item 1</li>
4. <li>Item 2</li>
5. </ul>
6. </body>

If we use the children property on the <ul> tag, we will get returned to us:

****

Question: is this collection the same as a JavaScript Array?

* 

**Yes.**

* 

**No.**

Bottom of Form

Answer: B