JavaScript Chapter 2 Quiz - DOM

Total points 57/71

Read each question carefully before answering!

The respondent's email (**nong.phloeut@student.passerellesnumeriques.org**) was recorded on submission of this form.

✓ What does DOM mean in JavaScript? 5/5

○ Documents Or Mobiles

○ DavaScript OavaScript MavaScript

● Document Object Model ✓

○ Degree Of Merger

○ Degree Of Monotonicity

○ Don't Open Me

○ It doesn't stand for anything, it was the name of Albert Einstein's dog

○ "DOM!" is the sound Chrome makes when it throws an error

✓ Which of these statements is **false**? 5/5

○ Elements can have any number of child elements.

○ The function getElementById(<ID>) gets the element that matches the id

○ style.display can be changed to show or hide an element

● A child element must have exactly one other child element of its own ✓

○ The 'document' element is the root, or top, of the DOM tree

✗ Which **two** of these statements are **false**? (Remember: select only the false statements, not the true ones!) 0/5

☐ The DOM is an array of properties.

☐ All elements are nodes in the DOM tree.

☐ Two elements at the same level of the hierarchy are known as 'siblings'.

☐ All elements (except 'document') are either a child element of 'document', or the child element of a child element.

☐ "document" is a parent node

☐ A parent node may have multiple child nodes.

☐ Child nodes may also be parent nodes.

☑ Child nodes may have zero, one, or multiple siblings. ✗

☑ Nodes with children cannot have properties. ✓

☐ An HTML Collection is an array-like collection (list) of HTML elements.

Correct answer

☑ The DOM is an array of properties.

☑ Nodes with children cannot have properties.

✓ Which of these will get the text of the first child element of the <select> element, with id "mySelect"? 4/4

● let text = document.getElementById("mySelect").firstElementChild.textContent; ✓

○ let text = documentation.getElementById("mySelect").text.<select>.select;

○ let text<select> = documents.getElementsByClass("mySelect").element-child.textContents().first;

○ let text(<select>) = website.document.get-element-by-id("mySelect").firstElement(<select>).text();

○ string text = documents("mySelect").firstElementChild.select().text;

○ text text = document.firstElementChild("mySelect").textContents().toString();

✓ What does this code do to a page containing a <div id="wrapper">? 5/5

```
let newElement = document.createElement("h2")
newElement.textContent = "hello";
document.getElementById("wrapper").appendChild(newElement);
```

○ Changes <div> to <h2>

○ Creates an <h2> element, containing the text "hello", at the bottom of the page

○ Creates an <h2> element inside <div>, containing the string "newElement"

● Creates an <h2> element, containing the text "hello", inside the bottom of the div. ✓

○ Replaces the current contents of the <div> with <h2>hello</h2>

○ Replaces the current contents of the <div> with <h2></h2>

○ Replaces the current contents of the <div> with "wrapper"

○ Replaces the current contents of the <div> with "appendChild"

○ Replaces the current contents of the <div> with "newElement"

○ Throws an error

✓ What are 2 differences between getElementById() and getElementsByClassName()? Select **TWO** answers. 5/5

☑ getElementById() returns one element, getElementsByClassName() returns an HTMLCollection of multiple (or zero) elements ✓

☐ getElementById() returns multiple elements, getElementsByClassName() returns one element.

☑ getElementById() uses the 'id' property of an element, getElementsByClassName() uses the 'class' property ✓

☐ getElementById() should be used for Chrome, getElementsByClassName() is for the Edge Browser

☐ getElementById() is not part of standard JavaScript, the user must create their own getElementById() function using getElementsByClassName()

☐ Trick question, there is no such function as getElementById()

☐ Error

✓ Which 5 of these are **events**? 4/4

☑ Mouse click ✓

☑ Keyboard press ✓

□ 1 is less than 3

☑ Form submitted ✓

☑ Focus on form field ✓

□ Upgrade to Windows 11

☑ Mouseover ✓

□ Faulty RAM

□ for() loop

✓ What are **two** ways of telling JavaScript how to react to events? 5/5

□ Define a dictionary of events at the top of the page, e.g. events = [Clicks:myButton, Mouseover:anotherButton]

□ Create a new ".events" file containing the list of events, separated by commas

☑ Set an Event Handler property in an element, e.g. <button id="myButton" onclick="displayDate()"> ✓

□ Add the events by dragging them into the Console in Inspection Mode

☑ Add an EventListener to the DOM, using e.g. myButton.addEventListener("click", displayDate) ✓

□ Reinstall Windows

✗ (Read this question carefully)
The following function is called by an event handler: 0/5

```
function myFunction(event) {
  let et = event.target;
  let targetType = et.tagName;
}
```

What does **et** contain?

○ the element that originally triggered the event.

● the name (as a string) of the element that triggered the event ✗

○ the id (as a string) of the element that originally triggered the event

○ the id (as a string) of the parent element currently triggering the event

○ the name of the function called by the element, e.g. myFunction

Correct answer

● the element that originally triggered the event.

✗ An event is triggered on a button. How are the event handlers called by default? 0/4

● Only the button's event handler is called. ✗

○ First the event handler of the button is called, then its parent's, then its parent's parent, and so forth unless stopped using stopPropagation().

○ The 'document' object's handler is called first, then the child's, all the way down to the button's.

○ The event handlers for the button and its parents are called at the same time.

○ The event handlers for the button and its parents are called in random order.

Correct answer

● First the event handler of the button is called, then its parent's, then its parent's parent, and so forth unless stopped using stopPropagation().

✓ Read this code carefully. It should change the text in <h2> to 'Clicked!' when the user clicks on it, but it has some missing pieces. What are they? 5/5

```
<body>
  <h2 MISSING_A>Click on this text!</h2>

  MISSING_B
  function changeText(id) {
    id.textContent = "Clicked!";
  }
  </script>
</body>
```

● MISSING_A: onclick="changeText(this)" MISSING_B: <script> ✓

○ MISSING_A: on-click="changeText()" MISSING_B: <javascript>

○ MISSING_A: click="changeText(this.value)" MISSING_B: <html>

○ MISSING_A: onclick=changeText(this) MISSING_B: {}

○ MISSING_A: onclick="changeText(Clicked!, id)" MISSING_B: <Script>

✓ Which 4 of these are **<input>** field **types** in a form?
In other words, what are valid things to use as **<input type=???>** 3/3

□ class

☑ text ✓

☑ checkbox ✓

☑ password ✓

□ lunch

☑ radio ✓

□ television

□ video

□ python

✓ How might we get the contents of the myTextField field of a form named myForm? 4/4

● let contents = document.myForm.myTextfield.value; ✓

○ let contents = formProcessor.get(myForm).get(myTextField);

○ var contents = MYTEXTFIELD;

○ string contents = docs.getForm(myForm).value;

○ let contents = myTextField.myForm.text;

○ CTRL-F to Find, 'Contents' in search box

✓ How might we get the contents of the myTextField box by id, if we don't have a form? 4/4

○ let contents = html.get('myTextField');

○ let contents = javascript.getText('myTextField');

● let contents = document.getElementById('myTextField').value; ✓

○ let contents = getContentsOfInputBox('myTextField');

○ let contents = myTextField;

○ myTextField++;

✓ Read this code fragment carefully. It should display the value of the currently selected myChoice radio button.
But, it has a line missing in the middle, indicated by ??? marks. What should the missing line be?

4/4

```
let result = "";
let myRadios = document.getElementsByName("myChoice");
for (radio of myRadios) {
  if (radio.checked) {
        ???
  }
}
document.getElementById("output").textContent = result;
```

○ set result = radio;

○ for (i in radio) { result = i.value };

● result = radio.value; ✓

○ let result = document.getElementsByName(radio.checked).value;

○ result = myRadios.radio.checked;

○ result = isChecked(myRadios);

○ myRadios++;

✓ Which code will tell JavaScript to call myFunction() whenever myRadio is changed? 4/4

● myRadio.addEventListener("change", myFunction); ✓

○ myRadio.set.onChange(myFunction);

○ document.EventListener(myRadio, myFunction, change);

○ doc.myRadio.event.change(element);

○ MYRADIO.LISTENER

○ int radio(void);