

Section 9 – Quiz

Question 1:

Are events only applicable for web browsers and JavaScript?

-

Yes.

-

No.

Answer: B

Question 2:

Which of the below statements is true?

-

All events share the same set of properties and methods.

-

There are many different types of events, including form events, mouse events, touch events, keyboard events and so on ... there are tons!

-

All events inherit from the Array object.

Answer: B

Question 3:

There are many different types of events in the web browser.

But, where do all of these events come from? In other words, who gives us access to all of these events?

-

JavaScript gives us access to all of these events.

-

The backend server (like Node.js) gives us access to these events.

-

The browser gives us access to many different types of events.

Answer: C

Question 4:

What is the Event interface?

-

It forms the basis of every event that is fired in the DOM. It provides properties and meaningful values for all event types.

Although each event has its own specific type of interface (e.g. the 'click' event inherits from the MouseEvent interface), all interfaces are actually related to one another and form a hierarchy. The Event interface is the root or top of the hierarchy.

-

It is a function that is executed when the browser loads, allowing JavaScript to access events and define actions that are triggered when the event fires.

-

Although the the Event interface only applies to some events (such as the click event), it is very important because it contains useful properties and methods.

Answer: A

Question 5:

What is the MouseEvent interface?

-

It is a useful function that is executed whenever the user presses a key on his/her keyboard.

-

The `MouseEvent` interface inherits the properties and methods of the `Event` interface, and contains useful properties and methods like `button`, `clientX`, `clientY` and `shiftKey`.

Many user actions, such as the click event, mousemove event, mouseup and mousedown events, inherit from the MouseEvent interface.

Answer: B

