

Quiz 07: JS Forms

Due Oct 27, 2023 at 2:59am**Points** 6**Questions** 6**Time Limit** None

Attempt History

	Attempt	Time	Score
LATEST	<u>Attempt 1</u>	45 minutes	6 out of 6

Score for this quiz: **6** out of 6

Submitted Oct 26, 2023 at 6:53pm

This attempt took 45 minutes.

Question 1

1 / 1 pts

Which fat arrow function is the closest equivalent to this function:

```
const catify = function(cat) {
```

```
  return `${cat}, the Cat`;
```

```
}
```

☐ `const catify = () => { return `${cat}, the Cat` };`

☐ `const catify = (cat) => { `${cat}, the Cat`; }`

☐ `function catify(cat) { return "${cat}, the Cat"; }`

Correct!

☒ `const catify = cat => `${cat}, the Cat`;`

Question 2

1 / 1 pts

Which statement is the most correct way to get the event object?

☐ The event object is automagically available inside the event callback

☐ The event object is a global object called `event`

Correct!

☒ The event object is passed to the event handler callback

☐ The event object is a global object called `e`

Question 3

1 / 1 pts

Which statement is NOT true about event propagation?

Correct!

- ☐ Event Propagation is sometimes called "Event Bubbling"
- ☒ Event propagation only happens when you turn the feature on
- ☐ Event Propagation happens AFTER the event handlers on the element itself are called
- ☐ Event Propagation means the event fires on the ancestors of an element in turn (for most events, and unless stopped)

Question 4

1 / 1 pts

A Web dev writes a page with a menu that will automatically change the page contents when links in the navigation are clicked, without actually having the browser navigate to a new page.

However, it is not working as intended and the browser is navigating to the link hrefs when the links are clicked.

Which statement is the most FALSE?

Correct!



They need to use ids on their elements instead of classes, since classes use "." in their selectors and ids use "#"



They probably didn't include a `event.preventDefault();` call in their event handler



Semantically, they probably shouldn't be using links for these control actions



The dev should have a single handler listening on a common ancestor of all the link elements rather than a listener on each of the different link elements

Question 5

1 / 1 pts

Which statement is MOST TRUE about Destructuring?



Destructuring destroys an object, separating it into different variables, 1 for each property

Correct!

Destructuring is declaring and assigning variables based on some properties or elements of objects/arrays



Destructuring creates efficient memory patterns for encrypted web traffic



Destructuring is only used for faking named function parameters

Question 6

1 / 1 pts

What statement is the MOST TRUE reason you would use `event.target` instead of `querySelector` to get the element node you intend to read/modify?



`event.target` is the only way to get the `.value` property for form fields



`event.target` bypasses the DOM and lets you interact directly with the browser's rendering intelligence system



`querySelector()` returns a list of nodes, so `event.target` during an event handler callback allows you to select the exact element the listener is attached to

Correct!

You can have an event listener on an ancestor element and have a single handler callback respond to the event, using `event.target` to know which element had the event happen to it

Quiz Score: 6 out of 6