1. What will be the output of the following JavaScript?

1 / 1 point

```
const a = true;
if(!a) {
    console.log("Green");
} else {
    console.log("Blue");
}
```

- ○ Green
- ◉ Blue
- ○ Nothing

Correct

That's correct! The NOT operator results in the condition being false. Therefore, the code inside the else statement will execute and Blue will be output.

2. What will be the output of the following JavaScript?

1 / 1 point

```
var x = 2;
x += 5;
console.log(x);
```

- 2
- 3
- 5
- 7

Correct

That's correct! The x variable is initially assigned the value 2. Then 5 is added to the variable and the result is stored in the variable. Therefore, 7 is output by the code.

3. What is the data type of the *x* variable in the following code?

1 / 1 point

```
1
2   var x = "Hello";
3
```

- Number
- BigInt
- String
- Boolean

Correct

That's correct! Text wrapped in double quotes represents a string data type.

4. What will the following JavaScript code output? **1 / 1 point**

```
1
2   var x = 10;
3
4   if(x > 10) {
5       console.log("Apple");
6   } else if(x > 5) {
7       console.log("Pear");
8   } else {
9       console.log("Orange");
10  }
11
```

- ○ Apple
- ◉ Pear
- ○ Orange

Correct

That's correct. The *x* variable is equal to 10 so the first condition fails but the second condition succeeds. Therefore, the code inside the else if statement executes and Pear is output.

5. What will the following JavaScript code output?

1 / 1 point

```
var result = 0;

for(var i = 0; i < 5; i++) {
    result += 2;
}

console.log(result);
```

- ○ 0
- ○ 2
- ○ 5
- ◉ 10

Correct

That's correct! The loop will run 5 times and each time add 2 to the result variable. Therefore, 10 will be output.

6. What will the following JavaScript code output?

1 / 1 point

```
console.log(result);
var result = 7;
```

- null
- undefined
- 7

Correct
That's correct. Since result is not yet defined when it is output, the value *undefined* is output.

7. What's missing from this JavaScript function declaration? **1 / 1 point**

```
1
2   function(a,b) {
3       return a + b
4   }
5
```

- The function name.
- The assignment operator.
- The dot notation.

Correct
Well done. When coding function declarations, you need to give them a name.

8. What is the output of the code below?

1 / 1 point

```
1
2   var car = { mileage: 200 }
3   var carMileage = 100
4   console.log(car.mileage)
5
```

- (●) 200
- ( ) 100
- ( ) 300

Correct
Well done. You can access the mileage property on the car object using the dot notation.

9. True or False. You use the pop method on an array to remove the last item from it.

1 / 1 point

- (●) True
- ( ) False

Correct
Well done. The pop method removes the last item from an array.

10. True or False. The second argument passed to the addEventListener function is the actual function that will handle the event, when it gets triggered.

1 / 1 point

- (●) True
- ( ) False

Correct
Well done. The second argument passed to the addEventListener handles the event.

11. How can you add an HTML attribute to an HTML element using JavaScript?

1 / 1 point

- (●) By invoking the `setAttribute` method on a given element.
- ( ) By invoking the `getAttribute` method on a given element.
- ( ) By invoking the `createAttribute` method on a given element.

Correct
Well done. For example, to add an id attribute to an element, you can run `setAttribute('id', 'sub-heading')`

12. What does this code do?

1 / 1 point

```
1
2   function addFive(val) {
3     return val + 5;
4   };
5   module.exports = addFive;
6
```

- (●) It defines the addFive function and exports it as a Node module so that it can be used in other files.
- ( ) This syntax is invalid.
- ( ) It allows you to invoke the addFive function without the parentheses.

Correct

Well done. It's a way to export the addFive function as a module that can be used elsewhere.