## Congratulations! You passed!

**Grade received** 100% **To pass** 80% or higher

Go to next item

**1.** What will print out when the following code runs? **1 / 1 point**

```
const meal = ["soup", "steak", "ice cream"]
let [starter] = meal;
console.log(starter);
```

- (•) soup
- ( ) ice cream
- ( ) steak

**Correct**
That's correct! The first item in the `meal` array is de-structured to the `starter` variable. Therefore, the code will print out `soup`.

**2.** The for-of loop works for Object data types. **1 / 1 point**

- ( ) true
- (•) false

**Correct**
That's correct! The for-of loop must use an iterable such as arrays.

**3.** What will print out when the following code runs? **1 / 1 point**

```
let food = "Chocolate";
console.log(`My favourite food is ${food}`);
```

- (•) My favourite food is Chocolate
- ( ) My favourite food is ${food}

**Correct**
That's correct! The template literal will replace `${food}` with `Chocolate`.

**4.** What values will be stored in the `set` collection after the following code runs? **1 / 1 point**

```
let set = new Set();
set.add(1);
set.add(2);
set.add(3);
set.add(2);
set.add(1);
```

- ( ) 1, 2, 3, 2, 1
- (•) 1, 2, 3

**Correct**
That's correct! A Set only stores unique items.

**5.** What value will be printed out when the following code runs?

1 / 1 point

```
let obj = {
    key: 1,
    value: 4
};

let output = { ...obj };
output.value -= obj.key;

console.log(output.value);
```

- 1
- 2
- 3 (selected)
- 4

**Correct**

That's correct! The spread operator ... will copy the properties from obj . Therefore, when output is created, it's value property will be equal to 4. Then 1 is substracted from the value property and result is stored back in the value property. Therefore, 3 will be printed out.

**6.** What value will be printed out when the following code runs?

1 / 1 point

```
function count(...basket) {
    console.log(basket.length)
}

count(10, 9, 8, 7, 6);
```

- 10, 9, 8, 7, 6
- 1, 2, 3, 4, 5
- 6
- 5 (selected)

**Correct**

That's correct! The rest operator ... allows a function to accept an indefinite amount of parameters. The length property of the basket variable will return 5 because there were 5 parameters passed to the method call. Therefore, 5 will be printed out.