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

```
1
2      const meal = ["soup", "steak", "ice cream"]
3      let [starter] = meal;
4      console.log(starter);
5
```

- ● soup
- ○ ice cream
- ○ steak

✓ **Correcto**
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.

- ○ true
- ● false

✓ **Correcto**
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
2      let food = "Chocolate";
3      console.log(`My favourite food is ${food}`);
4
```

- ● My favourite food is Chocolate
- ○ My favourite food is ${food}

✓ **Correcto**
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
2          let set = new Set();
3          set.add(1);
4          set.add(2);
5          set.add(3);
6          set.add(2);
7          set.add(1);
8
```

○ 1, 2, 3, 2, 1

◉ 1, 2, 3

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

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

```
1
2          let obj = {
3              key: 1,
4              value: 4
5          };
6
7          let output = { ...obj };
8          output.value -= obj.key;
9
10         console.log(output.value);
11
```

○ 1

○ 2

◉ 3

○ 4

✓ **Correcto**
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
2          function count(...basket) {
3              console.log(basket.length)
4          }
5
6          count(10, 9, 8, 7, 6);
7
```

○ 10, 9, 8, 7, 6

○ 1, 2, 3, 4, 5

○ 6

◉ 5

✓ **Correcto**
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.