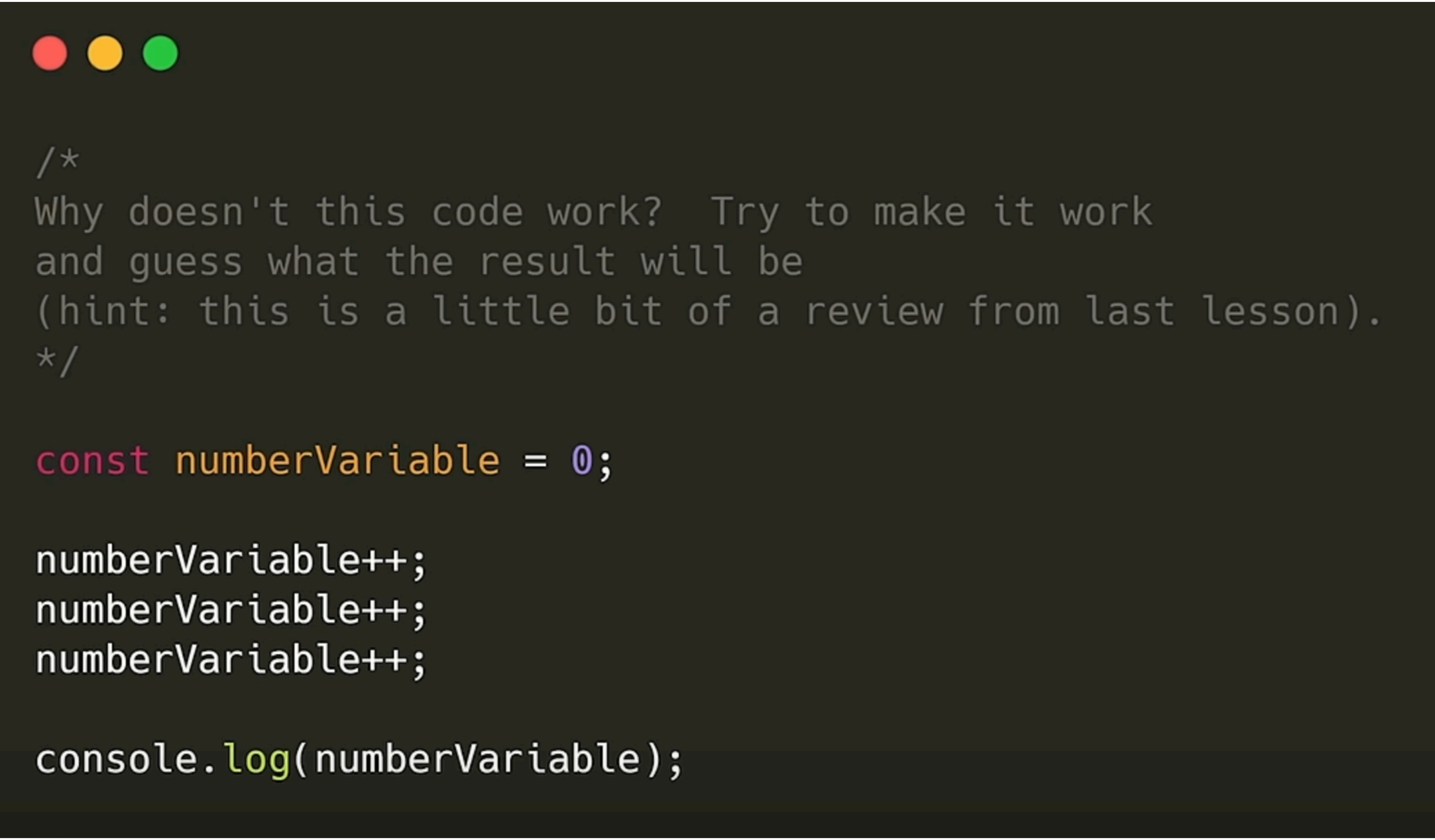
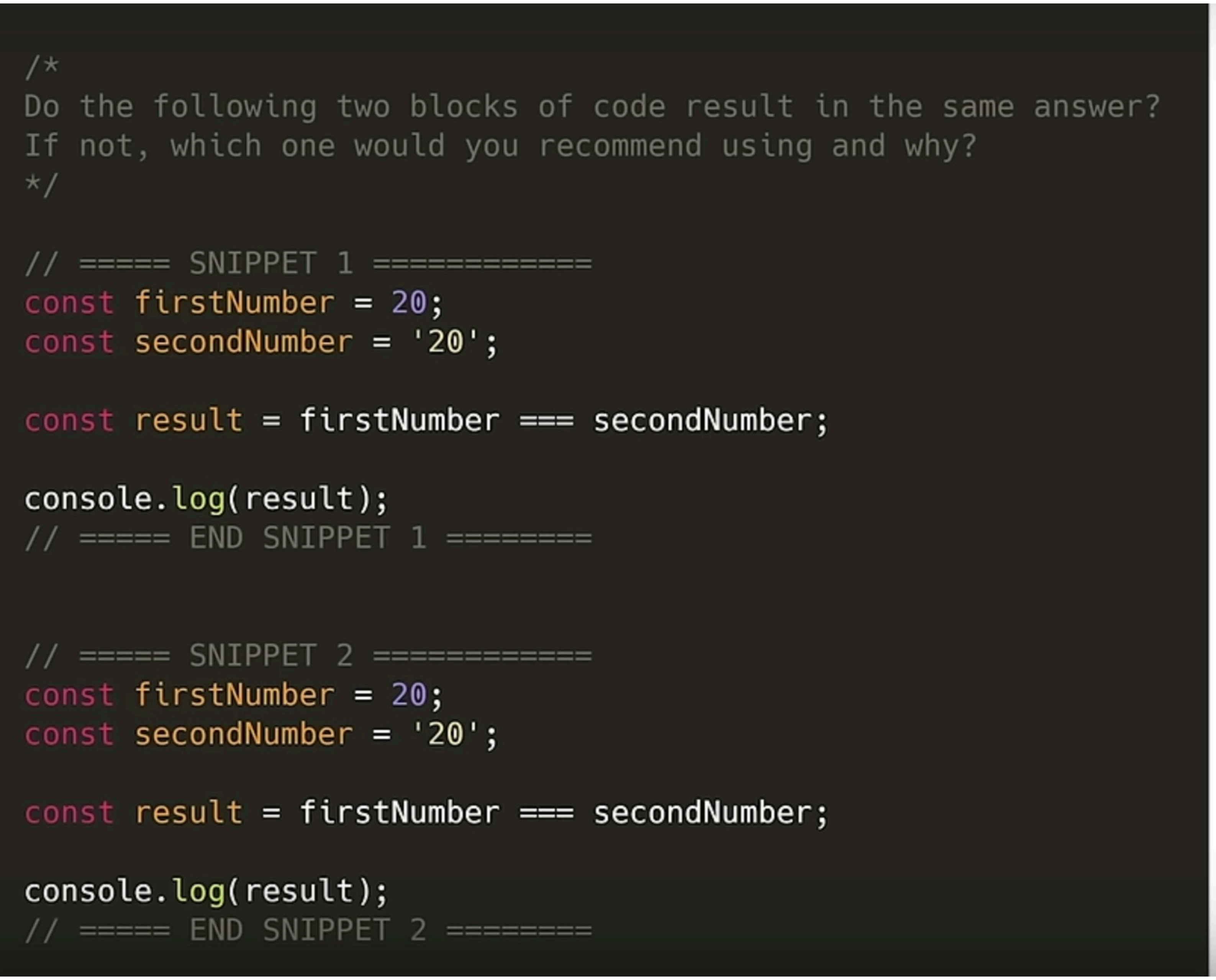
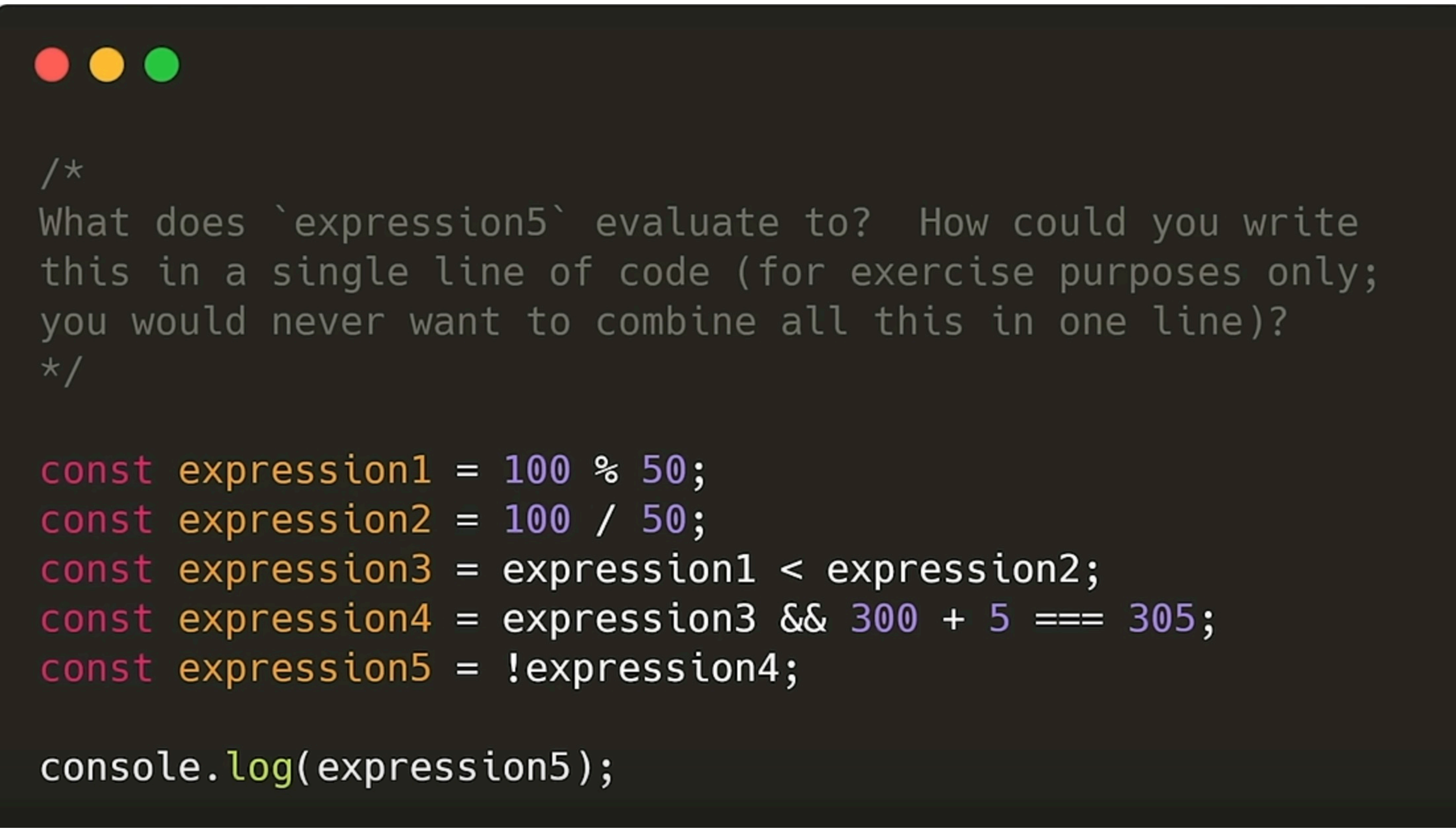


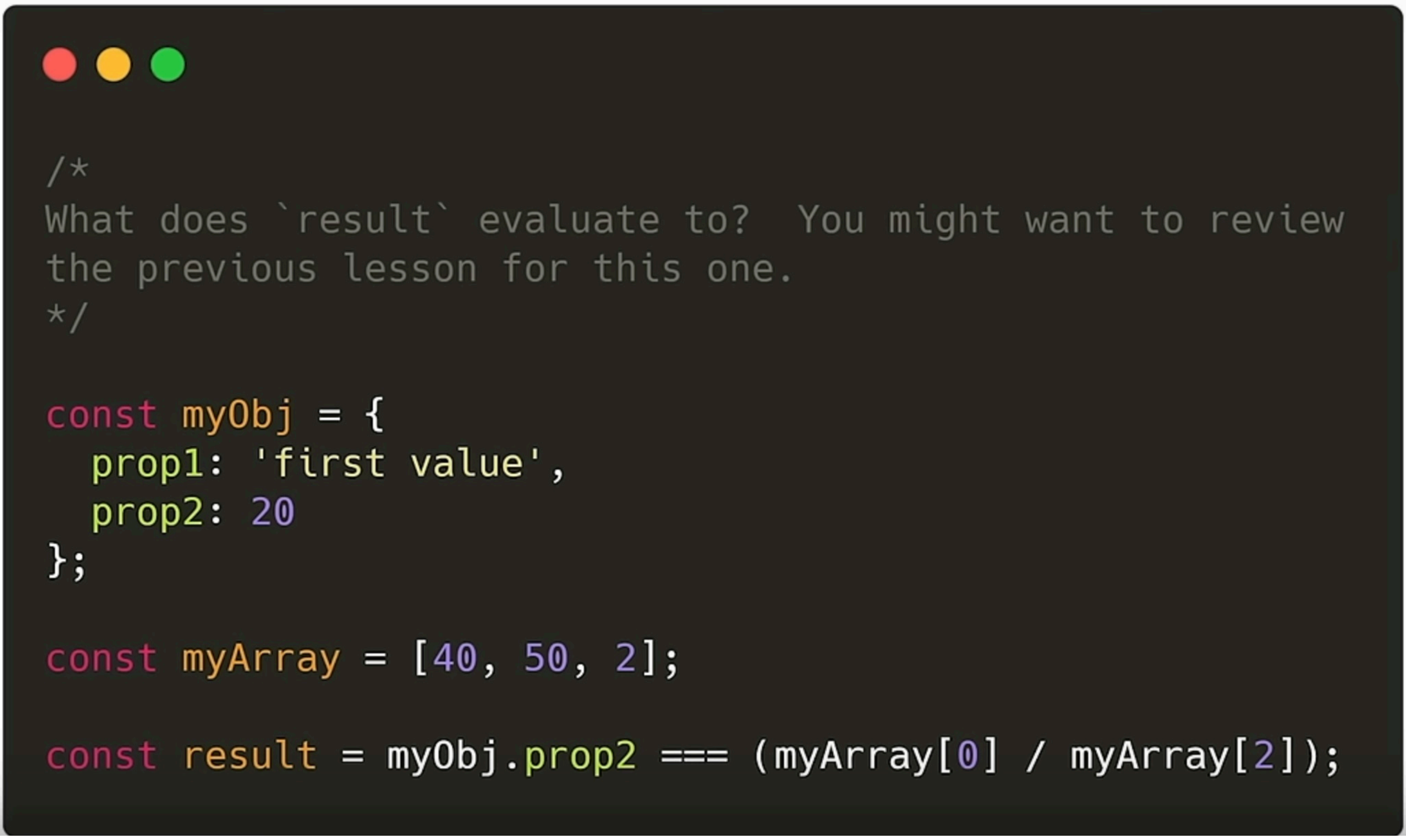
```
/*
Why doesn't this code work?  Try to make it work
and guess what the result will be
(hint: this is a little bit of a review from last lesson).
*/

const numberVariable = 0;

numberVariable++;
numberVariable++;
numberVariable++;

console.log(numberVariable);
```



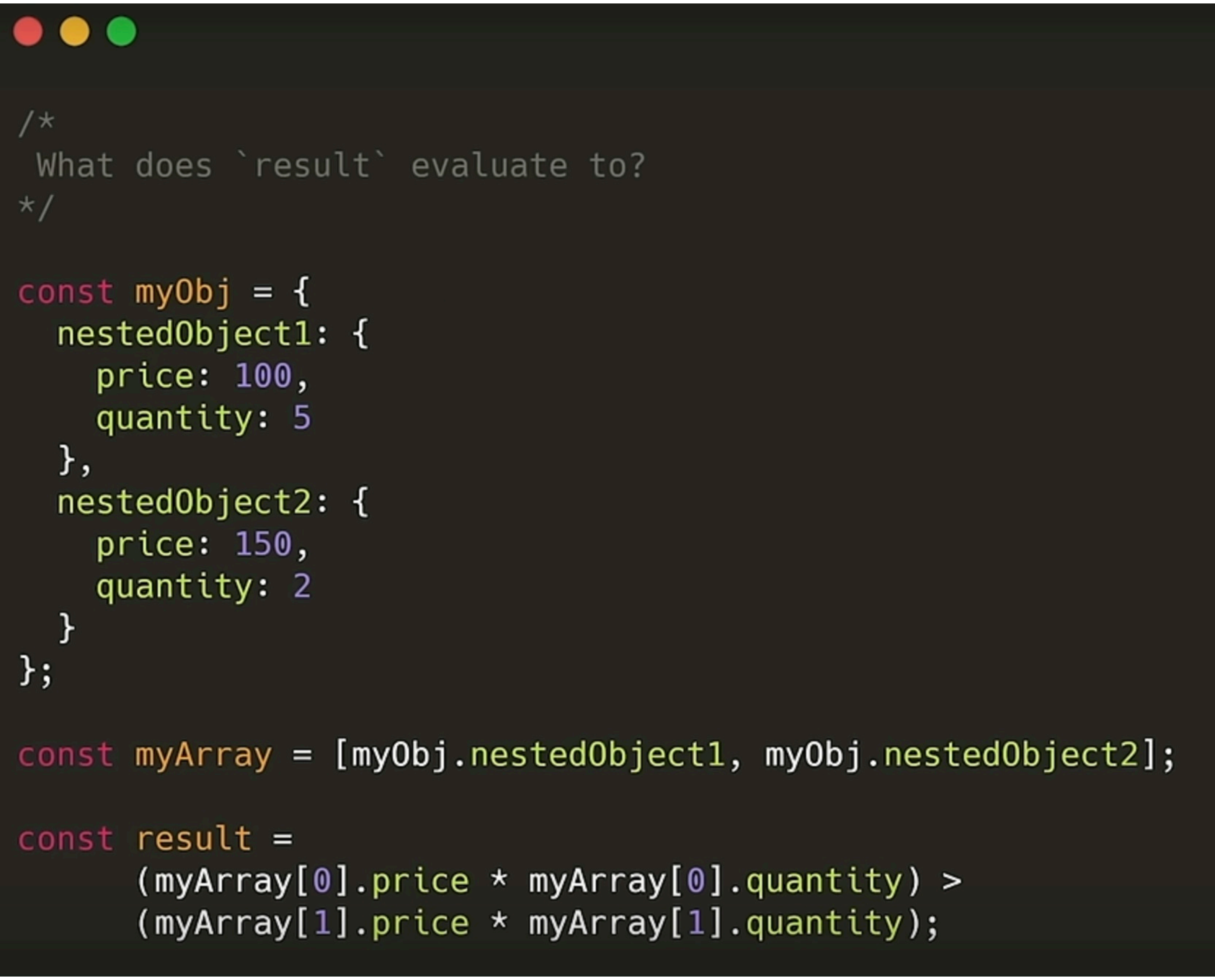
```
/*
Do the following two blocks of code result in the same answer?
If not, which one would you recommend using and why?
*/

// ===== SNIPPET 1 ============
const firstNumber = 20;
const secondNumber = '20';

const result = firstNumber === secondNumber;

console.log(result);
// ===== END SNIPPET 1 ========


// ===== SNIPPET 2 ============
const firstNumber = 20;
const secondNumber = '20';

const result = firstNumber === secondNumber;

console.log(result);
// ===== END SNIPPET 2 ========
```



```
/*
What does `expression5` evaluate to?  How could you write
this in a single line of code (for exercise purposes only;
you would never want to combine all this in one line)?
*/

const expression1 = 100 % 50;
const expression2 = 100 / 50;
const expression3 = expression1 < expression2;
const expression4 = expression3 && 300 + 5 === 305;
const expression5 = !expression4;

console.log(expression5);
```



```
/*
What does `result` evaluate to?  You might want to review
the previous lesson for this one.
*/

const myObj = {
  prop1: 'first value',
  prop2: 20
};

const myArray = [40, 50, 2];

const result = myObj.prop2 === (myArray[0] / myArray[2]);
```



```
/*
 What does `result` evaluate to?
*/

const myObj = {
  nestedObject1: {
    price: 100,
    quantity: 5
  },
  nestedObject2: {
    price: 150,
    quantity: 2
  }
};

const myArray = [myObj.nestedObject1, myObj.nestedObject2];

const result =
      (myArray[0].price * myArray[0].quantity) >
      (myArray[1].price * myArray[1].quantity);
```