

1. Subscript numbering of arrays in JavaScript always begins with what number? *0*
2. How can you determine the length of an array? *Use the array.length property*
3. Does JavaScript initialize array elements? If so, to what value? *NO*
4. Does JavaScript perform bounds checking? What is bounds checking?

No, JavaScript does not check to make sure an array element exists before referencing it.

5. How do you copy one array to another?

You must use a loop to copy each individual element of the first array into the second array

Given these declarations:

```
var i = 3;  
var arr = {10,15,7,20,11,2};  
var x = -1;  
var y = 100;
```

Show the values in x and y after executing each of the following:

6. `x = arr[3];`

x = 20

7. `x = 2 * arr[0];`

x = 20

8. `x = arr[1] + arr[2];`

x = 22

9. `x = arr[i] + i;`
`y = x + arr[5];`

x = 23 y = 25

10. `x = arr[i] + 1;`
`y = arr[i+1];`

x = 21 y = 11

Given the following array:

```
var a = {10,15,7,20,11,2,99,42,84,-3};
```

Write a loop to do each of the following:

11. Add 1 to every element of a.

```
function addOne(a) {  
  for (var i = 0; i < a.length; i++) {  
    a[i]++;  
  }  
}
```

12. Print the number of negative numbers in a.

```
function numNegs(a) {  
  var ctr = 0;  
  for (var i = 0; i < a.length; i++) {  
    if (a[i]<0) {  
      ctr++;  
    }  
  }  
  return ctr;  
}
```

13. Create a new array b which is the same size as a, and copy all elements from a into b.

```
function copy(a) {  
  var b = [];  
  for (var i = 0; i < a.length; i++) {  
    b.push(a[i]);  
  }  
  return b;  
}
```

14. Print the elements of a in reverse order.

```
function reverse(a) {  
  var b = [];  
  for (var i = 0; i < a.length; i++) {  
    b.push(a[a.length-1-i]);  
  }  
  return b;  
}
```

15. Count the number of elements in a that have values between 10 and 20 inclusive.

```
function numBetween(a,min,max) {  
  var ctr = 0;  
  for (var i = 0; i < a.length; i++) {  
    if (a[i]<=max && a[i] >= min) {  
      ctr++;  
    }  
  }  
  return ctr;  
}
```