Arrays Problem Set

printReverse()

Write a function *printReverse()* that takes an array as an argument and prints out the elements in the array in reverse order(don't actually reverse the array itself)

```
printReverse([1,2,3,4]);
//4
//3
//2
//1

printReverse(["a","b","c"]);
//"c"
//"b"
//"a"
```

isUniform()

write a function *isUniform()* which takes an array as an argument and returns true if all elements in the array are identical

```
isUniform([1,1,1,1]);          //true
isUniform([2,1,1,1]);          //false
isUniform(["a", "b", "p"]);    //false
isUniform(["b", "b", "b"]);    //true
```

sumArray()

Write a function *sumArray()* that accepts an array of numbers and returns the sum of all numbers in the array

```
sumArray([1,2,3]);          //6
sumArray([10,3,10,4]);      //27
sumArray([-5,100]);         //95
```

max()

Write a function max*()* that accepts an array of numbers and returns the maximum number in the array

```
max([1,2,3]);         //3
max([10,3,10,4]);     //10
max([-5,100]);        //100
```

contains()

Write a function *contains()* which takes 2 arguments: an array, and an element to search for. Return true is the element exists in the provided array

```
//Don't use the built-in Array.indexOf()

contains([10,15,20], 15);             //true
contains(["hello", "bye"], "bye");    //true
contains([10,15,20], 11);             //false
```

reverse()

Write a function *reverse()* that takes a single array as an argument and returns a reversed copy of the array

This is similar to the earlier *printReverse()* except that this function should return an array rather than just print the elements

```
//Don't use the built-in Array.reverse()

reverse([1,2,3]);              //[3,2,1]
reverse(["a", "b", "c"]);      //["c", "b", "a"]
```