Test Challenges

REHAN MEMON

Name Swap Challenge

Reverse the order of two provided strings.

Examples/ "Abraham Lincoln" - "Lincoln Abraham“

"Hank Aaron" - "Aaron Hank“

Approach 1 –Using Built in Functions// Tools

String-prototype split()

http://bit.ly/string-split

Array. prototype.reverse()

http://bit.ly/array-reverse-method

Array. prototype. Join

http://bit.ly/array-join

Name Swap Challenge

```
function nameSwap (str) {
// Write Code Block here to solve the challenge
}
console. Log (nameSwap ('Abraham Lincoln'));
console. log (nameSwap ('Hank Aaron'));
```

Name Swap Challenge

// Reverse the order of two provided strings.

// Examples

// "Abraham Lincoln" - "Lincoln Abraham“

// "Hank Aaron" - "Aaron Hank“

/// Approach 1 - Add array indices/indexes//

/// String-prototype.split()

// http://bit.ly/string-split

// Accessing array elements

/ / http://bit.ly/arrays-bracket-notation

Name Swap Challenge

```
function nameSwap (str) {


}


console. log (nameSwap ('Abraham Lincoln')) ;
console. log (nameSwap ('Hark Aaron')) ;
```

Remove Odd Numbers From an Array

// Challenge
// Create a function that takes an array of
numbers and returns only the even values.
// The function should return an array containing
only even numbers

Remove Odd Numbers From an Array

```
// Examples
// [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] -> [2, 4, 6, 8, 10]
// [21, 26, 28, 29] -> [26, 28]
// Approach 1 - for loop
 // Tools
// for statement
// http://bit.ly/for-loop-javascript
// remainder/modulus operator
 // http://bit.ly/remainder-operator
// Array.prototype.push()
// http://bit.ly/array-push
```

```
//Function evenOnly(arr){


//  }


//Console.log(evenOnly([1, 2, 3, 4, 5, 6, 7, 8, 9, 10]));
//Console.log(evenOnly([21, 26, 28, 29]));
```

// Remove Odd Numbers From an Array

// Remove Odd Numbers From an Array

// Challenge

// Create a function that takes an array of numbers and returns only the even values.

// The function should return an array containing only even numbers// Examples

// [1, 2, 3, 4, 5, 6, 7, 8, 9, 101 →> [2, 3, 4, 6, 8, 10]

11 [21, 26, 28, 29] -> [26, 28]

Approach 1- 1/ filter method

// Tools

/ filter method1

/ http://bit.ly/array-filter-method

// remainder/modulus operator

/ http://bit.ly/reminder-operator

// Repeat a String

// Challenge

// write a function that takes two arguments, a string and a number.

// The function should return a single string that contains the original string,

// repeated the number of times specified by the second argument,

// or an empty string if the number is negative.

// Examples

// 'car', 4 -> 'carcarcarcar'

// 'bar', 3 -> 'barbarbar'

```
// Approach 1 - for loop

// Tools
  // for loop
    // http://bit.ly/for-loop-javascript
  // arithmetic operators
    // http://bit.ly/arithmetic-operators-js
```

Repeat a String

```
function repeatString(str, num) {
 // Step 1 -> block of code


}


 // tests
  console.log(repeatString('car', 4));
  console.log(repeatString('bar', 3));
```

// Repeat a String // Approach 1 - while loop

```
// Challenge
  // write a function that takes two arguments, a string and a number.
  // The function should return a single string that contains the original string, repeated the number of times specified by the second argument, or an empty string if the number is negative.

// Examples
  // 'car', 4 -> 'carcarcarcar'
  // 'bar', 3 -> 'barbarbar'
```

// Repeat a String
// Approach 1 - while loop

- // Tools
  - // while loop
    - // http://bit.ly/while-loop
  - // arithmetic operators
    - // http://bit.ly/arithmetic-operators-js

// Find the longest String

```
// Challenge
  // Given a string of words return the length of the longest word(s)

// Examples
  // 'Hi, where is the airport?' -> 7
  // 'Thanks for stopping by!' -> 8
```

// Approach 1 - Using Built in Functions & Regular Expressions & For Loop

```
// Tools
  // split() method - splits a string into an array
    // http://bit.ly/string-split
  // String.prototype.replace()
    // http://bit.ly/string-replace-method
  // Online Regex Tool
    // http://bit.ly/regular-expressions-javascript
    // https://regex101.com/
    // https://regex101.com/r/c19vza/1
```

```
function longestWordLength(str) {


}

    console.log(longestWordLength
('Hi, where is the airport?'));
    console.log(longestWordLength
('Thanks for stopping by!'));
```

// Alphabetical Order

```
// Challenge
  // Create a function that takes a string and returns a string with its letters
  // in alphabetical order.

// Examples
  // "hello"  "olleh"
  // "goobye"  "eyboog"
  // "now"  "won"
  // "javascript"  "aacijprstv"
```

// Approach 1 – // if statement / for loop

- // Tools
  - // String.prototype.split()
    - // http://bit.ly/string-split
  - // Array.prototype.sort()
    - // http://bit.ly/array-sort
  - // Array.prototype.join()
    - // http://bit.ly/array-join

```
function alphaOrder(str) {


}


console.log(alphaOrder("hello"));
console.log(alphaOrder("goodbye"));
```

// Alphabetical Order

// Challenge

// Create a function that takes a string and returns a string with its letters

// in alphabetical order.

// Examples

// "hello" "ehllo"

// "goodbye" "bdegooy"

// Approach 1 - // if statement / for loop

```
// Tools
  // Spread Operator
    // https://codeburst.io/javascript-the-spread-operator-a867a71668ca
  // Array.prototype.sort()
    // http://bit.ly/array-sort
  // Array.prototype.join()
    // http://bit.ly/array-join
```

// Reverse a String

```
// Challenge

// Reverse the provided string.

// You may need to turn the string into an array before you can reverse it.

// Your result must be a string.


// Examples

// 'car' -> 'rac'

// 'bar' -> 'rab'
```

// Reverse a String
// Approach 1 - // Using Built in Functions

// Tools

// String.prototype.split()

// http://bit.ly/string-split

// Array.prototype.reverse()

// http://bit.ly/array-reverse-method

// Array.prototype.join()

// http://bit.ly/array-join

// Reverse a String

```
// Challenge
  // Reverse the provided string.
  // You may need to turn the string into an array before you can reverse it.
  // Your result must be a string.

// Examples
  // 'car' -> 'rac'
  // 'bar' -> 'rab’
// Tools
  // spread operator
    // http://bit.ly/spread-operator
  function revString(str) {

}

 // tests
 console.log(revString('car'));
 console.log(revString('bar'));
```

// Reverse a String

```
// Challenge
  // Reverse the provided string.
  // You may need to turn the string into an array before you can reverse it.
  // Your result must be a string.

// Examples
  // 'car' -> 'rac'
  // 'bar' -> 'rab'

// Tools
  // for loops
    // http://bit.ly/for-loop-javascript
```

// Reverse a String

```
// Challenge
  // Reverse the provided string.
  // You may need to turn the string into an array before you can reverse it.
  // Your result must be a string.

// Examples
  // 'car' -> 'rac'
  // 'bar' -> 'rab'

// Tools
  // for of
    // https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...of
```

// Determine if a string is a Palindrome

```
// Challenge
  // A palindrome is a text that is spelled the same front to back.

// Examples
  // 'Eva, Can I Stab Bats In A Cave?' -> true
  // 'Was It A Rat I Saw?' -> true
  // 'A nut for a jar of tuna?' -> true
// Note
  // Ignore capitalization and spaces or any special characters.
```

```
// Tools
  // String.prototype.replace()
    // http://bit.ly/string-replace-method
  // Online Regex Tool
    // http://bit.ly/regular-expressions-javascript
    // https://regex101.com/
  // String.prototype.toLowerCase()
    // http://bit.ly/lowercase-string
```