```
console.clear();

/*

=====================================================================

                Strings Interview Questions

=====================================================================

*/

/*

1. Write a function to count the number of vowels in a string.

*/

function countVowels(str) {

    let count = 0;

    for (let char of str) {

        if ("aeiouAEIOU".includes(char)) {

            count++;
        }
    }

    return count;
}

console.log("The number of vowels in the string is: " + countVowels("Hello World"));

/*

```

```
2. Write a function to reverse a string.

*/

function reverseString(str) {

    return str.split('').reverse().join('');
}

console.log("Reversed string: " + reverseString("Hello"));

/*

3. Write a function to check if a string is a palindrome.

*/

function isPalindrome(str) {

    let reversed = str.split('').reverse().join('');

    return str === reversed;
}

console.log("Is 'racecar' a palindrome? " + isPalindrome("racecar"));

/*

4. Write a function to count occurrences of a character in a string.

*/

function countCharacter(str, char) {

    return str.split(char).length - 1;
}

console.log("Occurrences of 'l' in 'hello': " + countCharacter("hello", "l"));
```

```

/*

5. Write a function to remove all spaces from a string.

*/

function removeSpaces(str) {

    return str.replace(/\s+/g, '');
}

console.log("String without spaces: " + removeSpaces("Hello World"));

/*

6. Write a function to capitalize the first letter of each word in a string.

*/

function capitalizeWords(str) {

    return str.split(' ').map(word => word.charAt(0).toUpperCase() + word.slice(1)).join(' ');
}

console.log("Capitalized words: " + capitalizeWords("hello world"));

/*

7. Write a function to find the first non-repeating character in a string.

*/

function firstNonRepeatingChar(str) {

    for (let char of str) {

        if (str.indexOf(char) === str.lastIndexOf(char)) {

            return char;
        }
    }

    return null;
}

console.log("First non-repeating character: " + firstNonRepeatingChar("swiss"));

/*

8. Write a function to check if two strings are anagrams.

*/

function areAnagrams(str1, str2) {

    return str1.split('').sort().join('') === str2.split('').sort().join('');
}

console.log("Are 'listen' and 'silent' anagrams? " + areAnagrams("listen", "silent"));

/*

9. Write a function to check if a string is a valid palindrome ignoring spaces and punctuation.

*/


function isValidPalindrome(str) {

    let cleaned = str.toLowerCase().replace(/[^a-z0-9]/g, '');

    return cleaned === cleaned.split('').reverse().join('');
}

console.log("Is 'A man, a plan, a canal: Panama' a valid palindrome? " + isValidPalindrome("A man, a plan, a canal: Panama"));
```

```javascript

/*

10. Write a function to replace all occurrences of a substring in a string.

*/

function replaceSubstring(str, oldSub, newSub) {

    return str.split(oldSub).join(newSub);
}

console.log("Replacing 'world' with 'everyone': " + replaceSubstring("Hello world", "world", "everyone"));

/*

11. Write a function to find the longest word in a string.

*/

function longestWord(str) {

    return str.split(' ').reduce((longest, word) => word.length > longest.length ? word : longest, "");
}

console.log("Longest word: " + longestWord("The quick brown fox jumps over the lazy dog"));

/*

12. Write a function to find the most frequent character in a string.

*/

function mostFrequentChar(str) {

    let freq = {};

    let maxChar = ''

    let maxCount = 0;

    for (let char of str) {

        freq[char] = (freq[char] || 0) + 1;

        if (freq[char] > maxCount) {

            maxCount = freq[char];

            maxChar = char;
        }
    }

    return maxChar;
}

console.log("Most frequent character: " + mostFrequentChar("banana"));

/*

13. Write a function to find all permutations of a given string.

*/

function stringPermutations(str) {

    if (str.length <= 1) return [str];

    let perms = [];

    for (let i = 0; i < str.length; i++) {

        let rest = stringPermutations(str.slice(0, i) + str.slice(i + 1));

        for (let perm of rest) {

            perms.push(str[i] + perm);
        }
    }

    return perms;
}

console.log("Permutations of 'abc': ", stringPermutations("abc"));

/*

14. Write a function to compress a string using the counts of repeated characters (e.g., "aaabbc" -> "a3b2c1").

*/

function compressString(str) {

    let compressed = "";

    let count = 1;

    for (let i = 0; i < str.length; i++) {

        if (str[i] === str[i + 1]) {

            count++;
        }

        else {

            compressed += str[i] + count;

            count = 1;
        }
    }

    return compressed.length < str.length ? compressed : str;
}
```

```javascript

console.log("Compressed string: " + compressString("aaabbc"));

/*

15. Write a function to generate all valid substrings of a given string.

*/

function substrings(str) {

    let result = [];

    for (let i = 0; i < str.length; i++) {

        for (let j = i + 1; j <= str.length; j++) {

            result.push(str.slice(i, j));
        }
    }

    return result;
}

console.log("Substrings of 'abc': ", substrings("abc"));

/*

16. Write a function to check if a string contains balanced parentheses.

*/

function isBalanced(str) {

    let stack = [];

    let pairs = { ')': '(', '}': '{', ']': '[' };

    for (let char of str) {

        if (Object.values(pairs).includes(char)) {

            stack.push(char);
        }

        else if (pairs[char]) {

            if (stack.pop() !== pairs[char]) return false;
        }
    }

    return stack.length === 0;
}

console.log("Is '(a + b) * (c - d)' balanced? " + isBalanced("(a + b) * (c - d)"));

/*

17. Write a function to find the longest common prefix among an array of strings.

*/

function longestCommonPrefix(arr) {

    if (!arr.length) return "";

    let prefix = arr[0];

    for (let str of arr) {

        while (str.indexOf(prefix) !== 0) {

            prefix = prefix.slice(0, -1);

            if (!prefix) return "";
        }
    }

    return prefix;
}

console.log("Longest common prefix: " + longestCommonPrefix(["flower", "flow", "flight"]));

/*

18. Write a function to determine if one string is a rotation of another.

*/

function isRotation(str1, str2) {

    return str1.length === str2.length && (str1 + str1).includes(str2);
}

console.log("Is 'erbottlewat' a rotation of 'waterbottle'? " + isRotation("waterbottle", "erbottlewat"));

/*

19. Write a function to remove duplicate characters from a string.

*/

function removeDuplicates(str) {

    return [...new Set(str)].join('');
}

console.log("String after removing duplicates: " + removeDuplicates("banana"));
```