

```
1 console.clear();
2
3 /*
4
5 =====
6
7         Strings Interview Questions
8
9 =====
10
11 */
12
13 /*
14
15 1. Write a function to count the number of vowels in a string.
16
17 */
18
19 function countVowels(str) {
20
21     let count = 0;
22
23     for (let char of str) {
24
25         if ("aeiouAEIOU".includes(char)) {
26
27             count++;
28         }
29     }
30
31     return count;
32 }
33
34 console.log("The number of vowels in the string is: " + countVowels("Hello World"));
35
36 /*
37
```

```
38 2. Write a function to reverse a string.
39
40 */
41
42 function reverseString(str) {
43
44     return str.split('').reverse().join('');
45 }
46
47 console.log("Reversed string: " + reverseString("Hello"));
48
49 /*
50
51 3. Write a function to check if a string is a palindrome.
52
53 */
54
55 function isPalindrome(str) {
56
57     let reversed = str.split('').reverse().join('');
58
59     return str === reversed;
60 }
61
62 console.log("Is 'racecar' a palindrome? " + isPalindrome("racecar"));
63
64 /*
65
66 4. Write a function to count occurrences of a character in a string.
67
68 */
69
70 function countCharacter(str, char) {
71
72     return str.split(char).length - 1;
73 }
74
75 console.log("Occurrences of 'l' in 'hello': " + countCharacter("hello", "l"));
```

```
76
77  /*
78
79  5. Write a function to remove all spaces from a string.
80
81  */
82
83  function removeSpaces(str) {
84
85      return str.replace(/\s+/g, '');
86  }
87
88  console.log("String without spaces: " + removeSpaces("Hello World"));
89
90  /*
91
92  6. Write a function to capitalize the first letter of each word in a string.
93
94  */
95
96  function capitalizeWords(str) {
97
98      return str.split(' ').map(word => word.charAt(0).toUpperCase() + word.slice(1)).join(' ');
99  }
100
101  console.log("Capitalized words: " + capitalizeWords("hello world"));
102
103  /*
104
105  7. Write a function to find the first non-repeating character in a string.
106
107  */
108
109  function firstNonRepeatingChar(str) {
110
111      for (let char of str) {
112
113          if (str.indexOf(char) === str.lastIndexOf(char)) {
```

○

```
114
115         return char;
116     }
117 }
118
119     return null;
120 }
121
122 console.log("First non-repeating character: " + firstNonRepeatingChar("swiss"));
123
124 /*
125
126 8. Write a function to check if two strings are anagrams.
127
128 */
129
130 function areAnagrams(str1, str2) {
131
132     return str1.split('').sort().join('') === str2.split('').sort().join('');
133 }
134
135 console.log("Are 'listen' and 'silent' anagrams? " + areAnagrams("listen", "silent"));
136
137 /*
138
139 9. Write a function to check if a string is a valid palindrome ignoring spaces and punctuation.
140
141 */
142
143
144 function isValidPalindrome(str) {
145
146     let cleaned = str.toLowerCase().replace(/[^a-z0-9]/g, '');
147
148     return cleaned === cleaned.split('').reverse().join('');
149 }
150
151 console.log("Is 'A man, a plan, a canal: Panama' a valid palindrome? " + isValidPalindrome("A man, a plan, a canal: Panama"));
```

```
152
153 /*
154
155 10. Write a function to replace all occurrences of a substring in a string.
156
157 */
158
159 function replaceSubstring(str, oldSub, newSub) {
160
161     return str.split(oldSub).join(newSub);
162 }
163
164 console.log("Replacing 'world' with 'everyone': " + replaceSubstring("Hello world", "world", "everyone"));
165
166 /*
167
168 11. Write a function to find the longest word in a string.
169
170 */
171
172 function longestWord(str) {
173
174     return str.split(' ').reduce((longest, word) => word.length > longest.length ? word : longest, "");
175 }
176
177 console.log("Longest word: " + longestWord("The quick brown fox jumps over the lazy dog"));
178
179 /*
180
181 12. Write a function to find the most frequent character in a string.
182
183 */
184
185 function mostFrequentChar(str) {
186
187     let freq = {};
188
189     let maxChar = ''
```

O

```
190
191     let maxCount = 0;
192
193     for (let char of str) {
194
195         freq[char] = (freq[char] || 0) + 1;
196
197         if (freq[char] > maxCount) {
198
199             maxCount = freq[char];
200
201             maxChar = char;
202         }
203     }
204
205     return maxChar;
206 }
207
208 console.log("Most frequent character: " + mostFrequentChar("banana"));
209
210 /*
211
212 13. Write a function to find all permutations of a given string.
213
214 */
215
216 function stringPermutations(str) {
217
218     if (str.length <= 1) return [str];
219
220     let perms = [];
221
222     for (let i = 0; i < str.length; i++) {
223
224         let rest = stringPermutations(str.slice(0, i) + str.slice(i + 1));
225
226         for (let perm of rest) {
227
```

```
228         perms.push(str[i] + perm);
229     }
230 }
231
232 return perms;
233 }
234
235 console.log("Permutations of 'abc': ", stringPermutations("abc"));
236
237 /*
238
239 14. Write a function to compress a string using the counts of repeated characters (e.g., "aaabbc" -> "a3b2c1").
240
241 */
242
243 function compressString(str) {
244
245     let compressed = "";
246
247     let count = 1;
248
249     for (let i = 0; i < str.length; i++) {
250
251         if (str[i] === str[i + 1]) {
252
253             count++;
254         }
255
256         else {
257
258             compressed += str[i] + count;
259
260             count = 1;
261         }
262     }
263
264     return compressed.length < str.length ? compressed : str;
265 }
```

O

```
266
267 console.log("Compressed string: " + compressString("aaabbc"));
268
269 /*
270
271 15. Write a function to generate all valid substrings of a given string.
272
273 */
274
275 function substrings(str) {
276
277     let result = [];
278
279     for (let i = 0; i < str.length; i++) {
280
281         for (let j = i + 1; j <= str.length; j++) {
282
283             result.push(str.slice(i, j));
284         }
285     }
286
287     return result;
288 }
289
290 console.log("Substrings of 'abc': ", substrings("abc"));
291
292 /*
293
294 16. Write a function to check if a string contains balanced parentheses.
295
296 */
297
298 function isBalanced(str) {
299
300     let stack = [];
301
302     let pairs = { ')': '(', '}': '{', ']': '[' };
303
```



```
304     for (let char of str) {
305
306         if (Object.values(pairs).includes(char)) {
307
308             stack.push(char);
309         }
310
311         else if (pairs[char]) {
312
313             if (stack.pop() !== pairs[char]) return false;
314         }
315     }
316
317     return stack.length === 0;
318 }
319
320 console.log("Is '(a + b) * (c - d)' balanced? " + isBalanced("(a + b) * (c - d)"));
321
322 /*
323
324 17. Write a function to find the longest common prefix among an array of strings.
325
326 */
327
328 function longestCommonPrefix(arr) {
329
330     if (!arr.length) return "";
331
332     let prefix = arr[0];
333
334     for (let str of arr) {
335
336         while (str.indexOf(prefix) !== 0) {
337
338             prefix = prefix.slice(0, -1);
339
340             if (!prefix) return "";
341         }
342     }
```

O

```
342     }
343
344     return prefix;
345 }
346
347 console.log("Longest common prefix: " + longestCommonPrefix(["flower", "flow", "flight"]));
348
349 /*
350
351 18. Write a function to determine if one string is a rotation of another.
352
353 */
354
355 function isRotation(str1, str2) {
356
357     return str1.length === str2.length && (str1 + str1).includes(str2);
358 }
359
360 console.log("Is 'erbottlewat' a rotation of 'waterbottle'? " + isRotation("waterbottle", "erbottlewat"));
361
362 /*
363
364 19. Write a function to remove duplicate characters from a string.
365
366 */
367
368 function removeDuplicates(str) {
369
370     return [...new Set(str)].join('');
371 }
372
373 console.log("String after removing duplicates: " + removeDuplicates("banana"));
```

O