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

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

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

```
7/29/25, 11:21 AM
                                                                     string-interview-questions is
  114
  115
                   return char;
  116
           }
  117
  118
           return null;
  119
  120
  121
       console.log("First non-repeating character: " + firstNonRepeatingChar("swiss"));
  122
  123
  124
       /*
  125
       8. Write a function to check if two strings are anagrams.
  126
  127
  128
       */
  129
       function areAnagrams(str1, str2) {
  130
  131
           return str1.split('').sort().join('') === str2.split('').sort().join('');
  132
  133
  134
       console.log("Are 'listen' and 'silent' anagrams? " + areAnagrams("listen", "silent"));
  135
  136
 137
       /*
  138
       9. Write a function to check if a string is a valid palindrome ignoring spaces and punctuation.
  139
  140
       */
  141
  142
  143
  144
       function isValidPalindrome(str) {
  145
           let cleaned = str.toLowerCase().replace(/[^a-z0-9]/g, '');
  146
  147
  148
           return cleaned === cleaned.split('').reverse().join('');
  149
  150
      console.log("Is 'A man, a plan, a canal: Panama' a valid palindrome? " + isValidPalindrome("A man, a plan, a canal: Panama"));
```

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

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

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

```
266
     console.log("Compressed string: " + compressString("aaabbc"));
267
268
269
    /*
270
    15. Write a function to generate all valid substrings of a given string.
271
272
    */
273
274
    function substrings(str) {
275
276
277
         let result = [];
278
         for (let i = 0; i < str.length; i++) {</pre>
279
280
             for (let j = i + 1; j <= str.length; j++) {</pre>
281
282
                 result.push(str.slice(i, j));
283
284
285
         }
286
         return result;
287
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
    function isBalanced(str) {
298
299
         let stack = [];
300
301
         let pairs = { ')': '(', '}': '{', ']': '[' };
302
303
```

```
304
         for (let char of str) {
305
306
             if (Object.values(pairs).includes(char)) {
307
308
                 stack.push(char);
309
             }
310
             else if (pairs[char]) {
311
312
                 if (stack.pop() !== pairs[char]) return false;
313
314
             }
315
         }
316
317
         return stack.length === 0;
318
319
     console.log("Is '(a + b) * (c - d)' balanced?" + isBalanced("(a + b) * (c - d)"));
320
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
         if (!arr.length) return "";
330
331
         let prefix = arr[0];
332
333
         for (let str of arr) {
334
335
             while (str.indexOf(prefix) !== 0) {
336
337
338
                 prefix = prefix.slice(0, -1);
339
                 if (!prefix) return "";
340
341
```

```
7/29/25, 11:21 AM
  342
  343
           return prefix;
  344
  345
  346
  347
       console.log("Longest common prefix: " + longestCommonPrefix(["flower", "flow", "flight"]));
  348
       /*
  349
  350
      18. Write a function to determine if one string is a rotation of another.
  351
  352
      */
  353
  354
       function isRotation(str1, str2) {
  355
  356
  357
           return str1.length === str2.length && (str1 + str1).includes(str2);
  358
  359
       console.log("Is 'erbottlewat' a rotation of 'waterbottle'? " + isRotation("waterbottle", "erbottlewat"));
  360
  361
      /*
  362
  363
      19. Write a function to remove duplicate characters from a string.
  364
  365
      */
  366
  367
      function removeDuplicates(str) {
  368
  369
           return [...new Set(str)].join('');
  370
  371
  372
  373 console.log("String after removing duplicates: " + removeDuplicates("banana"));
```