

```
1 // Harflar bilan
2 console.log(String.fromCharCode(72, 105)); // "Hi"
3 console.log(String.fromCharCode(72, 105)); // "Hi"
4
5 // Emoji bilan
6 console.log(String.fromCharCode(128512)); // noto'g'ri natija: "???"
7 console.log(String.fromCodePoint(128512)); // to'g'ri natija: "😄"
8
9 // -----
10
11 ✓ for (const match of "abc123xyz456".matchAll(/\d+/g)) {
12     | console.log(match[0]); // 123, keyin 456
13 }
14
15 ✓ // -----
16
17 // 1. at(index)
18 // Berilgan indeksdagi belgini qaytaradi.
19
20 console.log("Hello".at(1)); // "e"
21 ✓ // -----
22 // 2. charAt(index)
23 // Indeksdagi belgini qaytaradi(an'anaviy usul).
24
25 console.log("Hello".charAt(1)); // "e"
26 ✓ // -----
27 // 3. charCodeAt(index)
28 // Indeksdagi belgining Unicode raqamini qaytaradi.
29
30 console.log("A".charCodeAt(0)); // 65
31 ✓ // -----
32 // 4. codePointAt(index)
33 // Unicode kod nuqtasini qaytaradi(ko'proq emoji uchun ishlatiladi).
34
35 console.log("😄".codePointAt(0)); // 128512
36 ✓ // -----
37 // 5. concat(str)
38 // Ikki stringni birlashtiradi.
39
40 console.log("Hello".concat(" World")); // "Hello World"
41 ✓ // -----
```

```
41 // -----
42 // 6. endsWith(substr)
43 // String ma'lum bir substring bilan tugashini tekshiradi.
44
45 console.log("Hello".endsWith("lo")); // true
46 // -----
47 // 7. includes(substr)
48 // String ichida substring bor - yo'qligini tekshiradi.
49
50 console.log("Hello".includes("ell")); // true
51 // -----
52 // 8. indexOf(substr)
53 // Substring birinchi paydo bo'lish indeksini qaytaradi.
54
55 console.log("Hello".indexOf("l")); // 2
56 // -----
57 // 9. lastIndexOf(substr)
58 // Substring oxirgi paydo bo'lish indeksini qaytaradi.
59
60 console.log("Hello".lastIndexOf("l")); // 3
61 // -----
62 // 10. localeCompare(str)
63 // Ikki stringni mahalliy til qoidalariga ko'ra solishtiradi.
64
65 console.log("a".localeCompare("b")); // -1 (a oldinda)
66 // -----
67 // 11. match(regex)
68 // Regexga mos keluvchi birinchi qismni topadi.
69
70 console.log("abc123".match(/\d+/)); // ["123"]
71 // -----
72 // 12. matchAll(regex)
73 // Regexga mos keluvchi barcha qismlarni iterator shaklida qaytaradi.
74
75 for (const match of "abc123xyz456".matchAll(/\d+/g)) {
76     console.log(match[0]); // 123, keyin 456
77 }
78 // -----
79
80 // 13. normalize()
81 // Stringni unicode normalizatsiyasiga keltiradi.
82
83 console.log("e\u0301".normalize()); // "é"
84 // -----
```

```
84  ✓ // -----
85    // 14. padEnd(targetLength, padString)
86    // String oxiriga to'ldiruvchi belgi qo'shadi.
87
88    console.log("Hi".padEnd(5, ".")); // "Hi..."
89  ✓ // -----
90    // 15. padStart(targetLength, padString)
91    // String boshiga to'ldiruvchi belgi qo'shadi.
92
93    console.log("5".padStart(3, "0")); // "005"
94  ✓ // -----
95    // 16. raw()
96    // Template literalning xom(raw) ko'rinishini qaytaradi.
97
98    console.log(String.raw`Hello\nWorld`); // Hello\nWorld
99  ✓ // -----
100   // 17. repeat(count)
101   // Stringni count marta takrorlaydi.
102
103   console.log("ha".repeat(3)); // "hahaha"
104  ✓ // -----
105   // 18. replace(searchValue, newValue)
106   // Birinchi mos kelgan qismni almashtiradi.
107
108   console.log("foo bar".replace("bar", "baz")); // "foo baz"
109  ✓ // -----
110   // 19. replaceAll(searchValue, newValue)
111   // Barcha mos kelgan qismlarni almashtiradi.
112
113   console.log("foo bar bar".replaceAll("bar", "baz")); // "foo baz baz"
114  ✓ // -----
115   // 20. search(regex)
116   // Regex bo'yicha qidiruv qiladi va indeksni qaytaradi.
117
118   console.log("abc123".search(/\d/)); // 3
119  ✓ // -----
120   // 21. slice(start, end)
121   // Stringdan qism ajratib oladi.
122
123   console.log("Hello".slice(1, 4)); // "ello"
124  ✓ // -----
```

```
124 // -----
125 // 22. split(separator)
126 // Stringni bo'laklarga ajratib, arrayga aylantiradi.
127
128 console.log("a,b,c".split(",")); // ["a", "b", "c"]
129 // -----
130 // 23. substr(start, length)
131 // Boshlang'ich indeksdan boshlab, berilgan uzunlikdagi qismini oladi.
132
133 console.log("Hello".substr(1, 3)); // "ell"
134 // -----
135
136 // 24. substring(start, end)
137 // slice ga o'xshaydi, faqat start va end indeksleri o'zgartirilsa, tartibni tuzatadi.
138
139 console.log("Hello".substring(1, 4)); // "ell"
140 // -----
141 // 25. toLocaleLowerCase()
142 // Mahalliy til qoidalariga mos holda kichik harflarga o'zgartiradi.
143
144 console.log("Istanbul".toLocaleLowerCase("tr")); // "istanbul"
145 // -----
146 // 26. toLocaleUpperCase()
147 // Mahalliy til qoidalariga mos holda katta harflarga o'zgartiradi.
148
149 console.log("istanbul".toLocaleUpperCase("tr")); // "İSTANBUL"
150 // -----
151 // 27. toLowerCase()
152 // Kichik harflarga o'zgartiradi.
153
154 console.log("HELLO".toLowerCase()); // "hello"
155 // -----
156 // 28. toString()
157 // String qiymatini qaytaradi(odatda avtomatik chaqiriladi).
158
159 console.log(("Hello").toString()); // "Hello"
160 // -----
161 // 29. toUpperCase()
162 // Katta harflarga o'zgartiradi.
163
164 console.log("hello".toUpperCase()); // "HELLO"
165 // -----
```

```
155 // -----
156 // 28. toString()
157 // String qiymatini qaytaradi(odatda avtomatik chaqiriladi).
158
159 console.log(("Hello").toString()); // "Hello"
160 // -----
161 // 29. toUpperCase()
162 // Katta harflarga o'zgartiradi.
163
164 console.log("hello".toUpperCase()); // "HELLO"
165 // -----
166 // 30. toWellFormed()
167 // Yaxshi shakldagi unicode stringini qaytaradi.
168
169 console.log("≡".toWellFormed()); // "≡" (unicode to'g'ri formatda)
170 // -----
171 // 31. trim()
172 // Boshlanish va oxiridagi bo'sh joylarni olib tashlaydi.
173
174 console.log("  hello  ".trim()); // "hello"
175 // -----
176 // 32. trimEnd()
177 // Oxiridagi bo'sh joylarni olib tashlaydi.
178
179 console.log("  hello  ".trimEnd()); // "  hello"
180 // -----
181 // 33. trimStart()
182 // Boshlanishdagi bo'sh joylarni olib tashlaydi.
183
184 console.log("  hello  ".trimStart()); // "hello  "
185 // -----
186 // 34. valueOf()
187 // Primitiv string qiymatini qaytaradi.
188
189 console.log(("Hello").valueOf()); // "Hello"
```