

Além desta propriedade de comprimento, o JavaScript fornece uma API rica para Trabalhando com strings:

Seja s = "Olá, mundo"; // Comece com algum texto.

// obtendo partes de uma corda

s.substring (1,4) // => "ell": o 2º, 3º e 4º caracteres.

s.slice (1,4) // => "ell": a mesma coisa

s.slice (-3) // => "rld": últimos 3 caracteres

S.Split (",") // => ["Hello", "World"]: dividido em String delimiter

// pesquisando uma string

S.IndexOf ("L") // => 2: Posição da primeira letra L

S.IndexOf ("L", 3) // => 3: posição de primeiro "L" em ou depois de 3

S.IndexOf ("zz") // => -1: s não inclui o

Substring "ZZ"

S.LastIndexOf ("L") // => 10: Posição da última letra L

// funções de pesquisa booleana no ES6 e mais tarde

s.startswith ("inferno") // => true: a string começa com esses

s.endswith ("!") // => false: s não termina com isso

s.includes ("ou") // => true: s inclui substring "ou"

// Criando versões modificadas de uma string

S.Replace ("LLO", "YA") // => "Heya, mundo"

s.toLowerCase () // => "Olá, mundo"

S.ToupPercase () // => "Olá, mundo"

s.Normalize () // Unicode NFC Normalização: ES6

S.Normalize ("NFD") // NFD Normalização. Também "nfkc", "Nfkd"

// Inspeccionando caracteres individuais (16 bits) de uma string

S.Charat (0) // => "H": o primeiro caractere

S.Charat (S.Length-1) // => "D": o último caractere

S.Charcodet (0) // => 72: número de 16 bits no posição especificada

S.CodePointat (0) // => 72: ES6, trabalha para pontos CodePoints>