

METODY OBIEKTU STRING

✖Zmiana stylu:

+big() –powiększona czcionka

+small() –pomniejszona czcionka

+bold() –wytluszczona czcionka

+italics() –kursywa

+blink() –tekst migający (nie działa w IE)

+fixed() –czcionka o stałej szerokości

+strike() –przekreślona czcionka

+fontcolor(*kolor*) –zmiana koloru czcionki (*kolor*= nazwa koloru | rgb(n,n,n) | #xxxxxx)

+fontsize(*rozmiar_czcionki*) –zmiana rozmiaru czcionki

+toLowerCase() –małe litery

+toUpperCase() –wielkie litery

+sub() –indeks dolny

+sup() –indeks górny

+link(*url*) –tekst będzie wyświetlony jako hiperłącze do dokumentu o adresie *url*

✖Wyszukiwanie:

- +charAt(*pozycja*) –zwraca znak występujący na określonej pozycji (pierwsza pozycja to 0)
- +charCodeAt(*pozycja*) –zwraca kod Unicode znaku występującego na określonej pozycji
- +indexOf(*tekst*, *pozycja_początkowa*) –zwraca pozycję, od której zaczyna się *tekst*, poszukiwanie rozpoczyna się od *pozycji_początkowej*(wielkość liter jest istotna, gdy *tekstnie* występuje zwracana jest wartość -1, drugi argument opcjonalny)
- +lastIndexOf(*tekst*, *pozycja_końcowa*) –zwraca pozycję, od której zaczyna się *tekst*, tym razem poszukiwanie wstecz od *pozycji_końcowej*(wielkość liter jest istotna, gdy *tekstnie* występuje w danym łańcuchu zwracana jest wartość -1, drugi argument opcjonalny)
- +match(*tekst*) –podobne do indexOf(), ale zamiast pozycji zwraca *tekst*(null jeśli *tekst* nie występuje w ciągu, wielkość liter ma znaczenie)
- +search(*tekst*) –wyszukuje w ciągu wzorca *tekst* i podobnie jak indexOf zwraca pozycję pierwszego wystąpienia *tekst* (wartość "-1" jeśli *tekstnie* występuje w ciągu, wielkość liter istotna, ale można ją zignorować ustawiając flagę i, wówczas argument *tekst* umieszcza się w parze znaków / a nie " i a po nich flagę i

✖Zamiana tekstu:

- +fromCharCode(*liczba_1*, *liczba_2*, ... , *liczba_n*) –zwraca łańcuch znaków powstały z zamiany ciągu wartości Unicode na odpowiadające im znaki (metoda jest statyczna, co oznacza, że jedynym obiektem, dla którego może być wywołana, jest sam obiekt String)
- +replace(*stary_tekst*, *nowy_tekst*) –zamiana w danym ciągu podciągu *stary_tekst* na *nowy_tekst*(wielkość liter jest istotna, ustawienie flagi izmienia sposób interpretacji wielkości liter, flaga g włącza zastępowanie wszystkich wystąpień podciągu *stary_tekst* w łańcuchu)

✖Wyodrębnianie podciągu znaków:

+slice(*pozycja_początkowa*,*pozycja_końcowa*) –wycinanie fragmentu tekstu od *pozycja_początkowa* do *pozycja_końcowa* (obydwa argumenty numeryczne, wartość ujemna oznacza pozycję od końca, musi być spełniony warunek *pozycja_początkowa* < *pozycja_końcowa*, drugi argument opcjonalny –brak oznacza, że *pozycja_końcowa*= koniec łańcucha)

+substr(*pozycja_początkowa*, *długość*)-wycina z łańcucha liczbę znaków *długość* począwszy od *pozycja_początkowa* (ujemna wartość pierwszego argumentu oznacza pozycję od końca, drugi argument opcjonalny –brak= koniec łańcucha)

+substrting(*pozycja_początkowa*,*pozycja_końcowa*) –podobnie jak slice

+split(*separator*, *liczba*)–podzielenie łańcucha znaków na tablicę ciągów, pierwszy argument jest znakiem, który rozgranicza ciągi w łańcuchu znaków (np. "." przy podziale na zdania, " " przy podziale na wyrazy, "" przy podziale na litery, itd.) , drugi opcjonalny argument służy do określenia liczby ciągów

✕ Łączenie tekstów:

+ concat(*tekst1*, *tekst2*, ...) – dodaje do obiektu łańcuchy znakowe będące argumentami metody (musi wystąpić co najmniej jeden argument)

✕ Inne metody:

+ anchor(*nazwa_kotwicy*) – ustawienie kotwicy o nazwie *nazwa_kotwicy* w pliku HTML

✕ Przykład:

★ skrypt:

```
<script type="text/javascript">  
var tekst="Jakiś tekst."  
document.write(tekst.anchor("kotwica1"))  
</script>
```

★ jest równoważny zapisowi HTML:

` Jakiś tekst. `

+ `toSource()` – reprezentuje kod źródłowy obiektu (nie działa w IE)

+ `valueOf()` – zwraca prymitywną wartość obiektu `String`, wartość dziedziczona przez wszystkie obiekty potomne obiektu `String`, metoda zazwyczaj wywoływana automatycznie przez JavaScript