

Array em JavaScript - métodos

O método ou recurso **splice** permite adicionar um elemento ao array a partir de uma posição, viabilizando ainda a substituição dos elementos subsequentes, como demonstra-se:

```
<!DOCTYPE html>
<html>
<body>

  <p id="ver1"></p>
  <p id="ver2"></p>

<script>
  const dsw = ["WEB", "Internet", "QI", "TI"];

  document.getElementById("ver1").innerHTML = dsw;

  dsw.splice(1,0,"NOVO ITEM");

  document.getElementById("ver2").innerHTML = dsw;

</script>
</body>
</html>
```

WEB,Internet,QI,TI

WEB,NOVO ITEM,Internet,QI,TI

Ao analisar o método ou recurso **splice** aplicado na imagem abaixo notamos que será adicionado o elemento “NOVO ITEM” na posição 1 e nenhum elemento do array dsw será substituído:

Posição do "NOVO ITEM"

```
dsw.splice(1,0,"NOVO ITEM");
```

Quantos itens serão substituídos

O método ou recurso **splice** permite adicionar mais de um elemento ao array a

partir de uma posição, como demonstra-se:

```
<!DOCTYPE html>
<html>
<body>

  <p id="ver1"></p>
  <p id="ver2"></p>

<script>
  const dsw = ["WEB", "Internet", "QI", "TI"];

  document.getElementById("ver1").innerHTML = dsw;

  dsw.splice(1,0,"NOVO ITEM",5465462, "3º ITEM NOVO");

  document.getElementById("ver2").innerHTML = dsw;

</script>
</body>
</html>
```

WEB,Internet,QI,TI

WEB,NOVO ITEM,5465462,3º ITEM NOVO,Internet,QI,TI

O método ou recurso **splice** permite ainda substituir elemento(s) do array a partir de uma posição, como demonstra-se:

```
<!DOCTYPE html>
<html>
<body>

  <p id="ver1"></p>
  <p id="ver2"></p>

<script>
  const dsw = ["WEB", "Internet", "QI", "TI"];

  document.getElementById("ver1").innerHTML = dsw;

  dsw.splice(1,2,"NOVO ITEM",5465462, "3º ITEM NOVO");

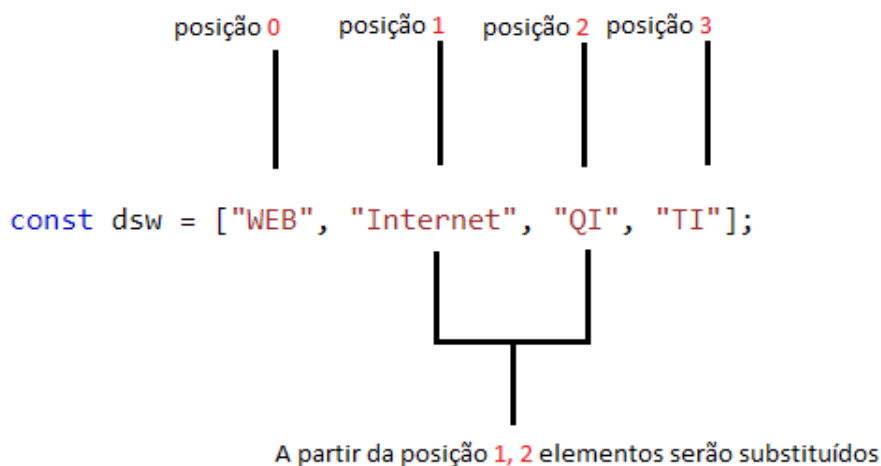
  document.getElementById("ver2").innerHTML = dsw;

</script>
</body>
</html>
```

WEB,Internet,QI,TI

WEB,NOVO ITEM,5465462,3º ITEM NOVO,TI

Observa-se no exemplo a acima que método **splice** irá substituir os elementos “Internet” e “QI”, pois foi determinado que a partir da posição 1, 2 elementos seriam substituídos, senão vejamos:



O método **splice** pode ser utilizado para remover elementos do array, como observa-se:

```
<!DOCTYPE html>
<html>
<body>

  <p id="ver1"></p>
  <p id="ver2"></p>

<script>

  const dsw = ["WEB", "Internet", "QI", "TI"];

  |
  document.getElementById("ver1").innerHTML = dsw;

  dsw.splice(1,2);

  document.getElementById("ver2").innerHTML = dsw;

</script>
</body>
</html>
```

WEB,Internet,QI,TI

WEB,TI

Verifica-se que a partir da posição 1 do array dsw 2 elementos foram removidos, transformando o array dsw num array com comprimento ou length igual a 2, isto é, com os elementos “WEB” e “TI”.

É possível criar um novo array com elementos removidos através do método splice, como demonstra-se:

```
<!DOCTYPE html>
<html>
<body>|
  <p id="ver1"></p>
  <p id="ver2"></p>
  <p id="ver3"></p>
<script>
  const dsw = ["WEB", "Internet", "QI", "TI"];

  document.getElementById("ver1").innerHTML = dsw;

  novoArray = dsw.splice(1,2);

  document.getElementById("ver2").innerHTML = dsw;

  document.getElementById("ver3").innerHTML = novoArray;

</script>
</body>
</html>
```

WEB,Internet,QI,TI

WEB,TI

Internet,QI

O método slice permite criar um novo array a partir de um existente, mas sem remover itens do array existente, como exemplifica-se:

```
<!DOCTYPE html>
<html>
<body>
  <p id="ver1"></p>
  <p id="ver2"></p>
  <p id="ver3"></p>
<script>
  const dsw = ["WEB", "Internet", "QI", "TI"];

  document.getElementById("ver1").innerHTML = dsw;

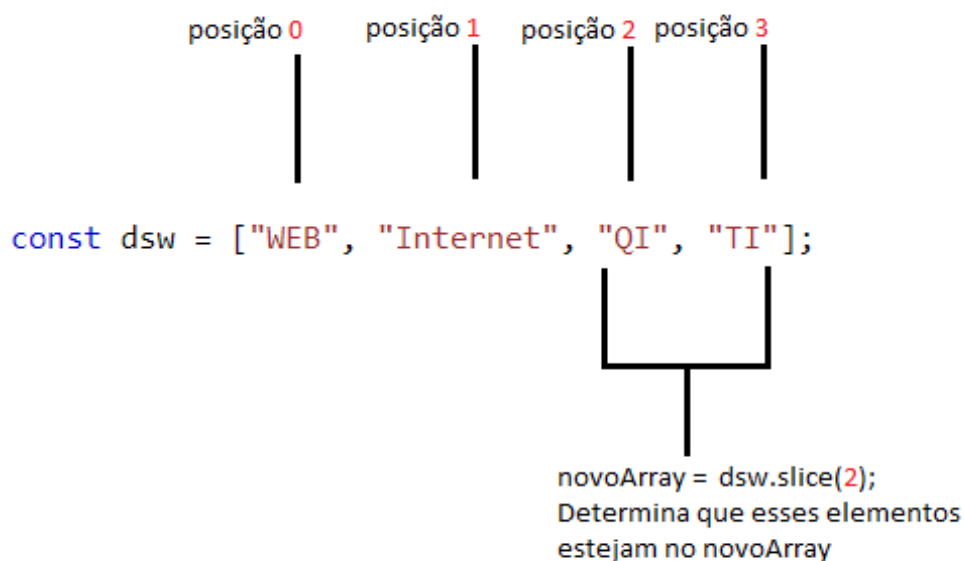
  novoArray = dsw.slice(2);
  document.getElementById("ver3").innerHTML = novoArray;

</script>
</body>
</html>
```

WEB,Internet,QI,TI

QI,TI

Com o uso do método slice no array dsw, verifica-se no exemplo acima que da posição 2 até o final do array dsw todos os elementos farão parte do novoArray, senão vejamos:



Há como utilizar mais um argumento como segundo parâmetro do método slice no array dsw, determinando a posição inicial e o limite de obtenção de elementos de um array para a construção de um novo, como demonstra-se:

```
<!DOCTYPE html>
<html>
<body>
  <p id="ver1"></p>
  <p id="ver2"></p>
  <p id="ver3"></p>
<script>
  const dsw = ["WEB", "Internet", "QI", "TI"];

  novoArray = dsw.slice(0,2);

  document.getElementById("ver3").innerHTML = novoArray;

</script>
</body>
</html>
```

WEB,Internet

Salienta-se que o segundo argumento determina o limite, ou seja, o elemento da posição correspondente ao limite não é incluído no novoArray, que no exemplo acima é o elemento “QI”.