

Array em JavaScript - métodos

O método ou recurso **push** permite adicionar um elemento ao array, 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.push("Sistemas");

  document.getElementById("ver2").innerHTML = dsw;
</script>

</body>
</html>
```

WEB,Internet,QI,TI

WEB,Internet,QI,TI,Sistemas

Quando o método ou recurso **push** é utilizado e exibido, um elemento é adicionado ao array e o comprimento ou quantidade de itens do array será retornado, 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;

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

  document.getElementById("ver3").innerHTML = dsw;
</script>

</body>
</html>
```

WEB,Internet,QI,TI

5

WEB,Internet,QI,TI,Sistemas

O método ou recurso **shift** permite remover o primeiro elemento do array, deslocando os demais para índices inferiores, 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.shift();
document.getElementById("ver2").innerHTML = dsw;
</script>

</body>
</html>
```

WEB,Internet,QI,TI

Internet,QI,TI

Observa-se na imagem acima que o elemento com conteúdo “WEB” foi removido, deslocando os demais elementos para índices inferiores.

Ainda sobre o método **shift** é interessante entender que há como retornar o conteúdo do elemento removido, utilizando o recurso de atribuição, conforme exemplo:

```
<!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;

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

document.getElementById("ver3").innerHTML = dsw;
</script>

</body>
</html>
```

WEB,Internet,QI,TI

WEB

Internet,QI,TI

O método **unshift** adiciona um elemento no início do array e desloca os elementos existentes para índices superiores, 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.unshift("NOVO");
document.getElementById("ver2").innerHTML = dsw;
</script>
</body>
</html>
```

WEB,Internet,QI,TI

NOVO,WEB,Internet,QI,TI

O método **unshift** permite retornar o comprimento do array após a inserção do novo elemento, sendo que para tanto é necessário utilizar o recurso de atribuição, conforme exemplo:

```
<!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;

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

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

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

WEB,Internet,QI,TI
5
NOVO,WEB,Internet,QI,TI

O recurso **delete** permite a exclusão do conteúdo do array, mas não remove o índice, conforme exemplo:

```
<!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;

  delete dsw[2];

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

  document.getElementById("ver3").innerHTML = dsw[2];

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

WEB,Internet,QI,TI
WEB,Internet,,TI
undefined

O recurso ou método **concat** permite juntar um array a outro array, criando um terceiro array detentor dos conteúdos concatenados, conforme exemplo:

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

  <p id="ver1"></p>
  <p id="ver2"></p>
  <p id="ver3"></p>

<script>
  const dsw1 = ["WEB", "Internet", "QI", "TI"];

  const dsw2 = ["DEV", "FRONT-END"];

  document.getElementById("ver1").innerHTML = dsw1;

  document.getElementById("ver2").innerHTML = dsw2;

  dsw3 = dsw1.concat(dsw2);

  document.getElementById("ver3").innerHTML = dsw3;

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

WEB,Internet,QI, TI

DEV,FRONT-END

WEB,Internet,QI, TI,DEV,FRONT-END

O recurso ou método **concat** permite ainda juntar mais de dois arrays, criando um array detentor dos conteúdos concatenados, conforme exemplo:

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

  <p id="ver1"></p>
  <p id="ver2"></p>
  <p id="ver3"></p>
  <p id="ver4"></p>

<script>
  const dsw1 = ["WEB", "Internet", "QI", "TI"];

  const dsw2 = ["DEV", "FRONT-END"];

  const dsw3 = ["BACK-END", "TÉCNICO"];

  document.getElementById("ver1").innerHTML = dsw1;

  document.getElementById("ver2").innerHTML = dsw2;

  document.getElementById("ver3").innerHTML = dsw3;

  dsw4 = dsw1.concat(dsw2,dsw3);

  document.getElementById("ver4").innerHTML = dsw4;

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

WEB,Internet,QI, TI

DEV,FRONT-END

BACK-END,TÉCNICO

WEB,Internet,QI, TI,DEV,FRONT-END,BACK-END,TÉCNICO