



Array

Um array é um conjunto ordenado de valores. Cada valor é chamado de elemento e cada elemento tem uma posição numérica no array, conhecida como índice. Os arrays em JavaScript são não tipados: um elemento do array pode ser de qualquer tipo e diferentes elementos do mesmo array podem ser de tipos diferentes

Como criar array?

A declaração direta ou representação literal é aquela em que o array é representado pelos símbolos de colchetes, que podem ou não conter elementos. Veja a sintaxe:

```
8 // Criando array com colchetes
9 var carros = ["Gol", "Palio", "Uno"];
```

Outra sintaxe para a declaração de um array é por meio da utilização do construtor new. Construtor É uma função que cria um objeto.

```
// Criando array com construtor
// Aqui esta importando o construtor Array
var carros = new Array("Gol", "Palio", "Uno");
```

Mas qual é a diferença do Array com o Vetor ?

- Array podem ser listas simples e sequenciais, e também, coleções compostas, isso significa que um array pode ter uma ou várias dimensões.

Ex:

```
var carros = new Array("Gol", "Palio", "Uno");
console.log(typeof carros);
```

- Já nos vetores, cada elemento é caracterizado por um único índice que aponta sua posição na lista. Enquanto nas matrizes, os elementos são reconhecidos por meio de dois ou mais índices que, em conjunto, apontam a posição do item na coleção.

```
var carros = ["Gol", "Palio", "Uno"];
console.log(typeof carros);
```

(Observação: As resposta serão identicas pois em javascript praticamente tudo é objeto, menos tipos primitivos. Então sua operação são parecidas)



Entregar os Exemplos e as tentativas que estão na cor vermelha

“Os valores dos exemplos não precisam ser necessariamente iguais ao exemplo”.

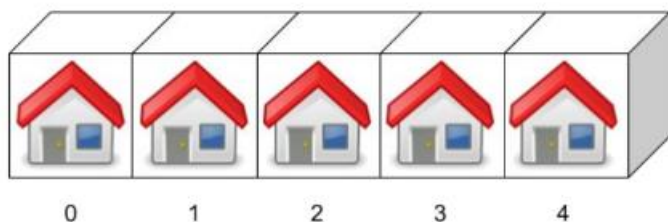
Agora vamos manipular o array:

Criando array:

```
var frutas = new Array("Maçã", "Pêra", "Laranja", "Uva", "Manga");  
console.log(frutas);
```

Agora tente acessar vamos acessar a posição 2:

(Para ajudar sempre imagine posições como blocos)



```
// Acessando um elemento  
console.log(frutas[2]);
```

*Tente acessar a posição 4

Alterando o elemento 2 do array:

*Tente alterar o elemento 6 execute e tente comentar no seu código e ao professor o que está acontecendo.

```
frutas[2] = "Banana";  
console.log(frutas);
```

Adicionando

```
frutas.push("Maracujá");  
console.log(frutas);
```



**Tente colocar no início do array*

Excluindo:

```
frutas.pop();  
console.log(frutas);
```

**Tente remover no início do Array.*

Verificando o tamanho do array:

```
console.log(frutas.length);
```

Invertendo a ordem do array:

```
console.log(frutas.reverse());
```

Transformando array em string:

```
console.log(frutas.toString());
```

Transformando array em string e separando por um caracter:

```
console.log(frutas.join(" - "));
```

Agora vamos manipular o Vetor:

Criando:

```
let vetor = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100];  
console.log(vetor);
```

Como já dito o funcionamento do array é bem parecido com do array. Porém como o conceito do vetor é ser mais rígidos. Então algumas diferenças como:

Adicionando:



```
vetor[vetor.length] = 200;  
console.log(vetor);
```

Removendo:

```
frutas[frutas.length - 1] = undefined;
```

*Tente fazer as demais manipulações do array no vetor