



Coding Bootcamp

JavaScript: Drive in

Array iteration Methods

I metodi di iterazione degli Array

Gli Array in JavaScript hanno dei **metodi built-in** che è possibile utilizzare per **iterare** su ogni elemento della lista.

Questi metodi sono delle **Higher Order Functions**, ovvero delle funzioni che accettano come argomento una funzione e che invocano la funzione fornita.

Ognuno di questi metodi, per ogni elemento presente nell'array, esegue la **callback** passata come argomento e, a seconda del metodo utilizzato, **restituisce un valore** specifico.

forEach()

Il metodo built-in `forEach()` esegue la **callback** fornita come argomento una volta per ogni elemento presente nell'Array. Ecco la sintassi:

```
[1,5,12,24].forEach((element, index, array) => console.log(element, index, array)); // undefined
```

ARGOMENTI

- Il primo parametro che il metodo `forEach()` fornisce alla **callback** è l'**elemento** dell'array su cui si sta iterando.
- Il secondo è l'**indice** dell'elemento stesso all'interno dell'array.
- Il terzo è l'**intero array**.

RETURN

`forEach()` restituisce come valore `undefined`.

map()

Il metodo built-in `map()` esegue la **callback** fornita come argomento una volta per ogni elemento presente nell'Array. A differenza di `forEach()` restituisce un array contenente i valori di **return** della callback:

```
const newArray = [1,5,12].map((element, index, array) => element + 5); // [6,10,17]
```

ARGOMENTI

- Il primo parametro che il metodo `map()` fornisce alla **callback** è l'**elemento** dell'array su cui si sta iterando.
- Il secondo è l'**indice** dell'elemento stesso all'interno dell'array.
- Il terzo è l'**intero array**.

RETURN

`map()` restituisce come valore un array composto da tutti i valori di **return** di ogni invocazione della callback.

filter()

Il metodo built-in `filter()` restituisce un nuovo array contenente solo gli elementi dell'array originale per i quali la condizione booleana restituita dal `return` della callback sia vera:

```
const newArray = [1,5,12].filter((element, index, array) => element < 10); // [1,5]
```

ARGOMENTI

- Il primo parametro che il metodo `filter()` fornisce alla **callback** è l'**elemento** dell'array su cui si sta iterando.
- Il secondo è l'**indice** dell'elemento stesso all'interno dell'array.
- Il terzo è l'**intero array**.