





JS - Objects



VS





Arrays VS Objects

Archivi vs Citofoni

Un **Array**:

è una struttura di dati che può contenere più valori ai quali si può accedere per mezzo di un **indice numerico**.



Un **Object** *oggetto letterale*:

è una struttura di dati che può contenere più valori ai quali si può accedere per mezzo di un **indice di nomi**.





Objects

La sintassi

Come si scrive un oggetto letterale?

1. Creiamo una variabile
2. Apriamo le parentesi graffe
3. Inseriamo una serie di coppie
nome: valore

```
1 const palla = {  
2  
3   "colore": ['red', 'blue'],  
4   tipo: 'pallina da golf',  
5   'chiave': <tipoDiDato>  
6  
7 };  
8
```



Una coppia (detta anche *chiave/valore*) viene detta **proprietà**.
Attenzione alle virgolette!



Objects

La sintassi

Ci ricorda qualcosa?

```
1 {  
2  
3   chiave : valore,  
4  
5   chiave : valore,  
6  
7   chiave : valore  
8  
9 }
```





Objects

La sintassi

Come si recupera un valore da un object?

dot notation

parentesi quadre come per gli array

```
1 const palla = {  
2   "colore": ['red', 'blue'],  
3   'tipo': 'pallina da golf'  
4 };  
5  
6  
7 console.log(palla.tipo); // 'pallina da golf'  
8  
9  
10  
11 console.log(palla['tipo']); // 'pallina da golf'  
12
```



Objects

La sintassi

Come si aggiunge un valore ad un object?

Dot notation

Se la proprietà esiste già si cambia il suo valore.
Se non esiste sarà aggiunta.

```
1 palla.peso = 46;
```

Quadre

Se la proprietà esiste già si cambia il suo valore.
Se non esiste sarà aggiunta.

```
1 palla['peso'] = 46;
```



Nelle quadre passiamo una stringa che corrisponde alla chiave della proprietà.



Javascript - JS

Objects

La sintassi



**Quando usiamo la dot notation
e quando le parentesi quadre?**



Objects

La sintassi



E se vogliamo stampare tutte le proprietà di un oggetto senza doverne conoscere il numero e il nome?



Objects - for..in

Un ciclo per gli oggetti

- keyword:
for
- condizione:
Andiamo avanti finché esistono delle chiavi da analizzare all'interno dell'oggetto
 - **let key** indica la chiave sulla quale operiamo
 - keyword **in** - stiamo ciclando all'interno di
 - **nome dell'oggetto**

```
1 const palla = {
2   "prezzo": 12,
3   "tipo": "pallina da golf"
4 };
5
6
7 for (let key in palla) {
8   console.log(palla[key]);
9 }
10
11
12 // Risultato atteso:
13 // 12
14 // pallina da golf
15 // non per forza in questo ordine
```



Le quadre qui sono necessarie per poter usare la variabile key.



Strutture dati

Time to Recap



Variabili con tipi di dato semplice:

servono per immagazzinare un singolo valore, che può essere modificato nel tempo.



Array:

servono per salvare una lista ordinata di dati. L'indice numerico viene utilizzato per accedere al dato e può servire per identificare l'ordine della lista.



Oggetti:

servono per salvare, nello stesso "contenitore", più informazioni riguardanti una unica entità. La chiave della proprietà viene usata per accedere al valore corrispondente.

E combinazioni di questi tipi:

- Array di oggetti
- Oggetti che contengono array tra le proprietà
- ...



Develop a Developer Mind

Come affrontare la risoluzione di un nuovo problema

1. In quale struttura dati salverò i miei dati?

- a. Me ne serve più di una?
- b. Quale è la scelta migliore in termini di complessità?

2. Ci sono dei cicli?

- a. La scelta dei cicli è coerente con le strutture dati che ho scelto?
- b. Potrebbe esserci una via più semplice se cambiassi struttura dati?

3. Casi Limite

- a. L'algoritmo che ho immaginato funziona nei casi limite?
- b. Potrebbe esserci situazioni in cui non dà il risultato desiderato?



Array Bidimensionali

Marty! Non stai pensando quadrimensionalmente!

Abbiamo visto cos'è un array:

una struttura dati che può contenere una "serie" ordinata di elementi.

E se l'elemento contenuto fosse a sua volta un array?

```
1 const iscritti =  
2  
3 [  
4  
5 "Luca", // iscritti[0] = Luca  
6 "Marco" // iscritti[1] = Marco  
7  
8 ];  
9
```

```
1 const classi =  
2  
3 [  
4  
5 ['Mattia', 'Giacomo'],  
6 ['Martina', 'Giorgia']  
7  
8 ];
```

Se **iscritti[0]** era il primo elemento dell'array, cioè **"luca"**

classi[0] sarà l'array **["Mattia", "Giacomo"]**

classi[0][0] sarà **"Mattia"**, **classi[1][0]** sarà?





Array Bidimensionali

Marty! Non stai pensando quadrimensionalmente!

E se un array contiene degli objects?

```
1 const classi =  
2  
3 [  
4  
5   {  
6     'nome' : 'Classe 1',  
7     'numero_studenti' : 10  
8   },  
9  
10  {  
11    'nome' : 'Classe 2',  
12    'numero_studenti' : 15  
13  }  
14  
15 ];  
16  
17
```





LIVE CODING



ESERCITAZIONE