

JSON

JSON (JavaScript Object Notation) è un semplice formato per lo scambio di dati. Per le persone è facile da leggere e scrivere, mentre per le macchine risulta facile da generare e analizzarne la sintassi. Si basa su un sottoinsieme del Linguaggio di Programmazione JavaScript, Standard ECMA-262 Terza Edizione - Dicembre 1999.

JSON è un formato di testo completamente indipendente dal linguaggio di programmazione, ma utilizza convenzioni conosciute dai programmatori di linguaggi della famiglia del C, come C, C++, C#, Java, JavaScript, Perl, Python, e molti altri. Questa caratteristica fa di JSON un linguaggio ideale per lo scambio di dati.

<u>value</u>	<u>object</u>	<u>string</u>	<u>number</u>
object	{ }	" "	int
array	{ members }	" chars "	int frac
string		chars	int exp
number	members	char	int frac exp
true	pair	char chars	int
false	pair , members	char	digit
null	pair	any-Unicode-character- except-"-or-\-or- control-character	digit1-9 digits
	string : value	"	- digit
		\	- digit1-9 digits
		\	frac
		\	. digits
	<u>array</u>	\b	exp
	[]	\f	e digits
	[elements]	\n	digits
	elements	\r	digit
	value	\t	digit digits
	value , elements	\u four-hex-digits	e
			e, e+, e-
			E, E+, E-

Oggetti JS e JSON

- Collezione chiave/valore
- Lista ordinata di valori
- { "nome" : "Mauro" }
- In js non sono obbligatorie gli apici sulle chiavi
- In js gli oggetti possono contenere funzioni

JS PARSING JSON

- eval();
- JSON.parse();
- JSON.stringify();//il contrario di parse
- caniuse.com/json

JSON vs XML

- Più leggero
- Più compatto
- Pesa meno KB
- Indipendente dal linguaggio

JSON è basato su due strutture:

- Un insieme di coppie nome/valore. In diversi linguaggi, questo è realizzato come un oggetto, un record, uno struct, un dizionario, una tabella hash, un elenco di chiavi o un array associativo.
- Un elenco ordinato di valori. Nella maggior parte dei linguaggi questo si realizza con un array, un vettore, un elenco o una sequenza.

Queste sono strutture di dati universali. Virtualmente tutti i linguaggi di programmazione moderni li supportano in entrambe le forme. E' sensato che un formato di dati che è interscambiabile con linguaggi di programmazione debba essere basato su queste strutture.

In JSON, assumono queste forme:

- Un **oggetto** è una serie non ordinata di nomi/valori. Un oggetto inizia con {parentesi graffa sinistra e finisce con }parentesi graffa destra. Ogni nome è seguito da :due punti e la coppia di nome/valore sono separata da ,virgola.
- Un **array** è una raccolta ordinata di valori. Un array comincia con [parentesi quadra sinistra e finisce con]parentesi quadra destra. I valori sono separati da ,virgola.
- Un **valore** può essere una stringa tra virgolette, o un numero, o vero true o falso false o nullo null, o un oggetto o un array. Queste strutture possono essere annidate.
- Una **stringa** è una raccolta di zero o più caratteri Unicode, tra virgolette; per le sequenze di escape utilizza la barra rovesciata. Un singolo carattere è rappresentato come una stringa di caratteri di lunghezza uno. Una stringa è molto simile ad una stringa C o Java.
- Un **numero** è molto simile ad un numero C o Java, a parte il fatto che i formati ottali e esadecimali non sono utilizzati.
- I caratteri di **spaziatura** possono essere inseriti in mezzo a qualsiasi coppia di token.

A parte alcuni dettagli di codifica, questo descrive totalmente il linguaggio.