

JSON: STEP-BY-STEP



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

Objects are
enclosed in **curly
braces**

 john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

Information is
stored in
"key": "value"
pairs

json john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

Information is
stored in
"key": "value"
pairs

keys are of
datatype **string**



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

values **must be**
one of the following
data types:

- string
- number
- boolean
- null
- array
- object



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

values **must be**
one of the following
data types:

- string
- number
- boolean
- null
- array
- object

json john.json

```
{  
  "name": "John",  
  "age": 30,  
  "employment": "employed",  
  "hasChildren": true,  
  "parents": [{"name": "John", "age": 60}],  
  "pet": {"name": "Brutus", "species": "dog", "age": 7}  
}
```



strings are any kind of characters enclosed in “ “

- “word”
- “This is also a string.”
- “7 bananas”

values **must be**

non
array
object



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

values **must be**
one of the following
data types:

string
number
boolean
null
array
object

json john.json

```
{  
  "name": "John",  
  "age": 42,  
  "employed": true,  
  "hasChildren": false,  
  "parent": "John",  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```



numbers can be:

- integers (e.g. 42)
- floats (e.g. 0.0005)

values **must be**

boolean
null
array
object



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

values **must be**
one of the following
data types:

- string
- number
- boolean**
- null
- array
- object

json john.json

```
{  
  "name": "John",  
  "age": 30,  
  "employed": true,  
  "hasChildren": false,  
  "parents": [  
    {  
      "name": "Brutus",  
      "species": "dog",  
      "age": 7  
    }  
  ]  
}
```



a *boolean* has one of two possible values

- true / false
- 1 / 0

values **must be**

boolean
null
array
object



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

values **must be**
one of the following
data types:

string
number
boolean
null
array
object

json john.json

```
{  
  "name": "John",  
  "age": 30,  
  "employment": "Software Engineer",  
  "hasChildren": true,  
  "parents": [{"name": "John", "age": 60}],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```



null can only have the value NULL.
The variable of data type *null* has **no value assigned** to it.

values **must be**

boolean
null
array
object



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

values **must be**
one of the following
data types:

string
number
boolean
null
array
object

json john.json

```
{  
  "name": "John",  
  "age": 30,  
  "employment": "Software Engineer",  
  "hasChildren": true,  
  "parents": [{"name": "John", "age": 60}, {"name": "Mary", "age": 55}],  
  "pet": {"species": "dog", "age": 7},  
}
```



An *array* is a collection of elements. Can be understood as a list.

- ["Annika", "Silke"]
- [1,2,3]
- ["some string", 0.5, true]

values **must be**

array
object

json john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

values **must be**
one of the following
data types:

string
number
boolean
null
array
object

json john.json

```
{
```

```
  "name": "Bill",
```

```
  "age": 36,
```

```
  "employment": "Postdoc",
```

```
  "hasChildren": false,
```

```
  "parent": "John",
```

```
  "pet": "cat",
```

```
  "name": "John",
```

```
  "species": "human",
```

```
  "age": 7,
```

```
}
```

```
}
```



An *object* contains key/value pairs, separated by commas and is enclosed by { }

```
{  
  "name": "Bill",  
  "jobTitle": "Postdoc",  
  "city": "New York",  
  "age": 36  
}
```

values **must be**

object



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```

Data is
**separated by
commas**



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```



john.json

```
{  
  "name": "John",  
  "age": 27,  
  "employed": true,  
  "hasCar": null,  
  "parents": ["Anna", "Michael"],  
  "pet": {  
    "name": "Brutus",  
    "species": "dog",  
    "age": 7  
  }  
}
```



john.xml

```
<name>John</name>  
<age>27</age>  
<employed>TRUE</employed>  
<hasCar>NULL</hasCar>  
<parents>Anna</parents>  
<parents>Michael</parents>  
<pet>  
  <name>Brutus</name>  
  <species>dog</species>  
  <age>7</age>  
</pet>
```

XML

```
<example>
  <superhero>Wonder Woman</superhero>
  <publisher>DC Comics</publisher>
  <identities>
    <identity>Princess Diana</identity>
    <identity>Diana Prince</identity>
  </identities>
  <pet>
    <name>Jumpa</name>
    <species>kangaroo</species>
  </pet>
</example>
```

JSON

```
{
  "superhero": "Wonder Woman",
  "publisher": "DC Comics",
  "identities": [
    "Princess Diana",
    "Diana Prince"
  ],
  "pet": {
    "name": "Jumpa",
    "species": "kangaroo"
  }
}
```

YAML

```
---
superhero: Wonder Woman
publisher: DC Comics
identities:
  - Princess Diana
  - Diana Prince
pet:
  name: Jumpa
  species: kangaroo
```

if you are interested in YAML, also see <https://yaml.org/>

QUESTIONS?