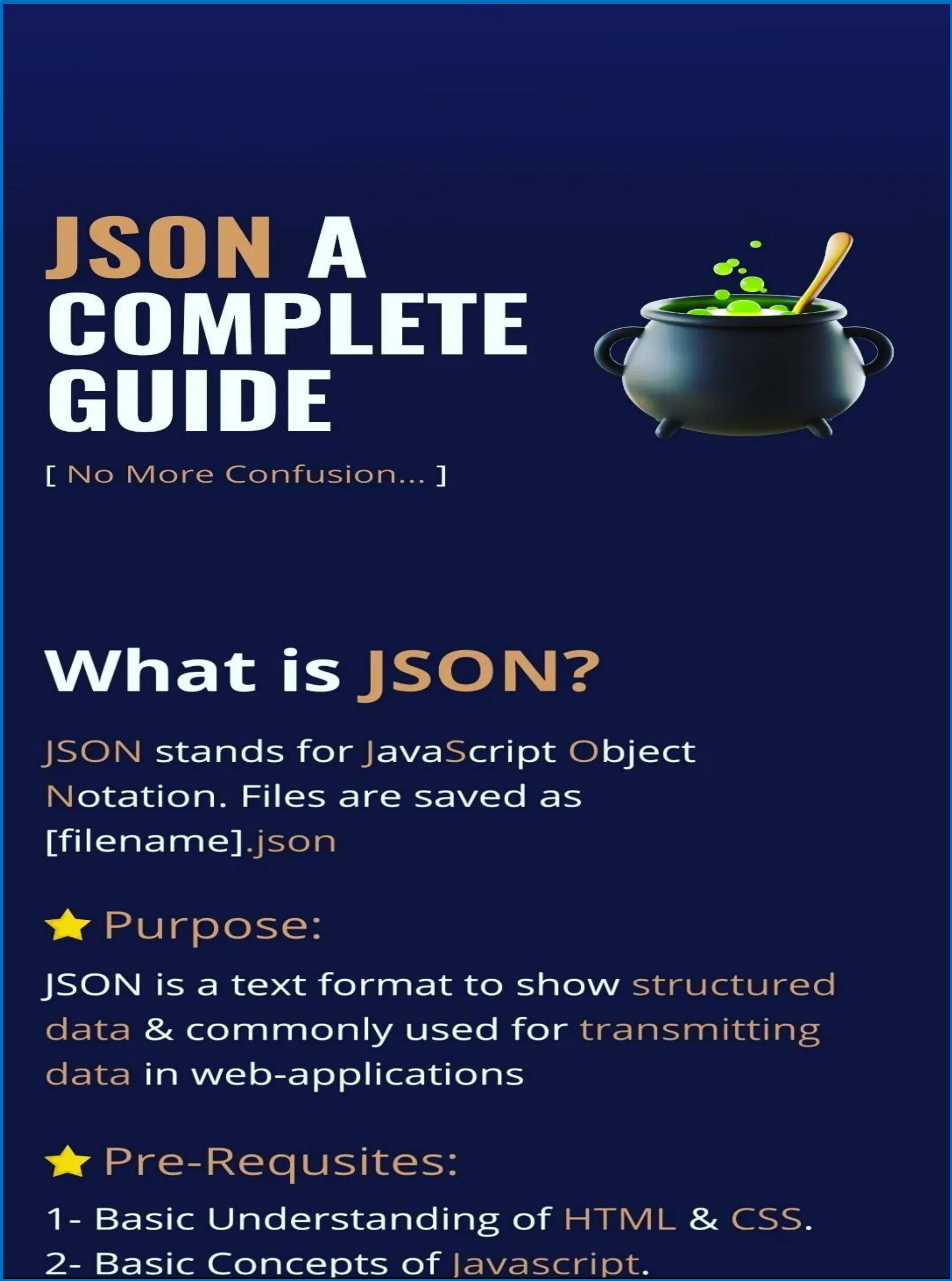


JSON A COMPLETE GUIDE

[No More Confusion...]

What is JSON?

JSON stands for JavaScript Object Notation. Files are saved as [filename].json

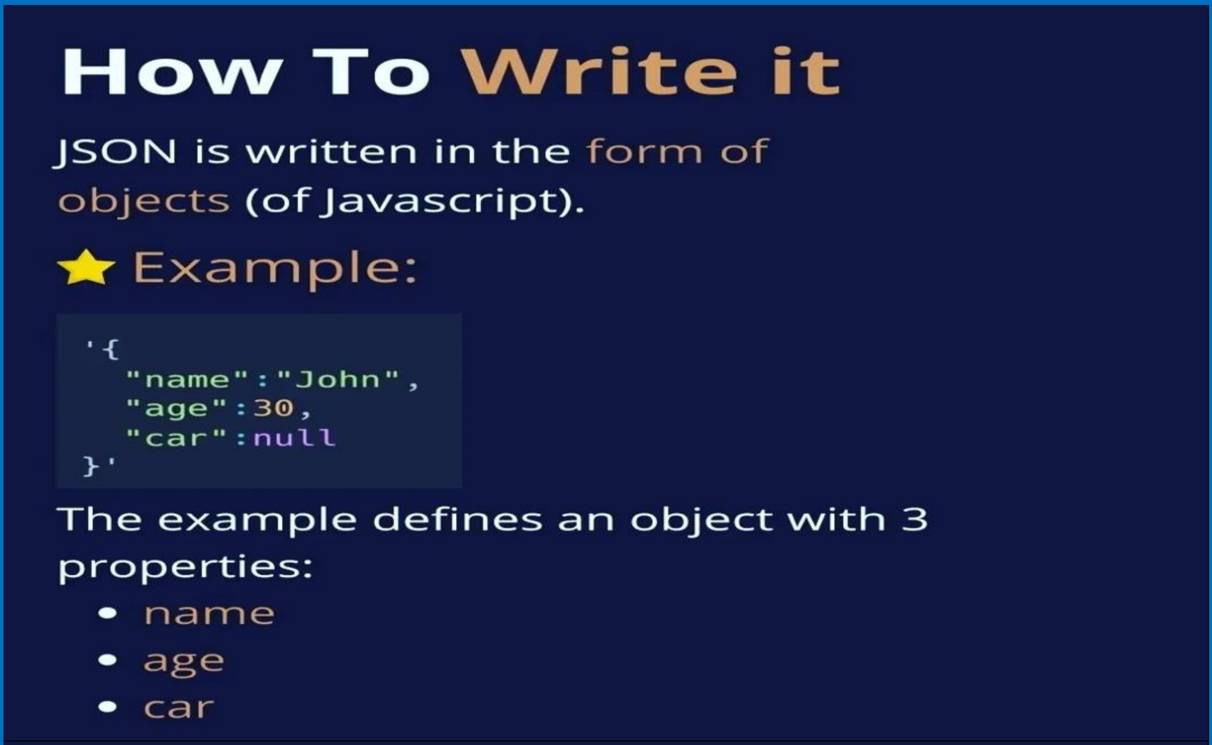
⭐ Purpose:

JSON is a text format to show structured data & commonly used for transmitting data in web-applications

⭐ Pre-Requsites:

1- Basic Understanding of HTML & CSS.
2- Basic Concepts of Javascript.

@AAKASH KANOJIYA



How To Write it

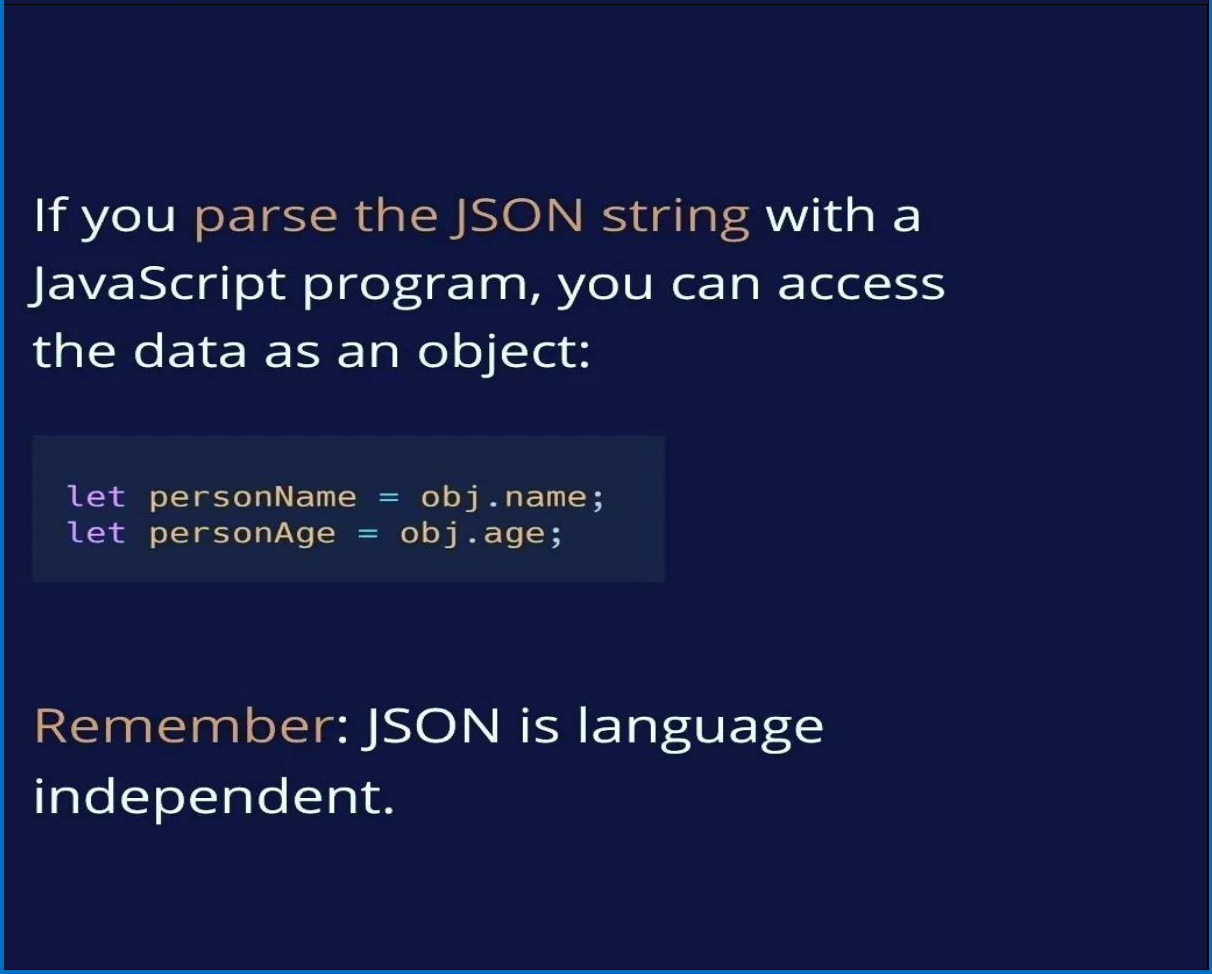
JSON is written in the form of objects (of Javascript).

⭐ Example:

```
'{
   "name":"John",
   "age":30,
   "car":null
}'
```

The example defines an object with 3 properties:

- name
- age
- car

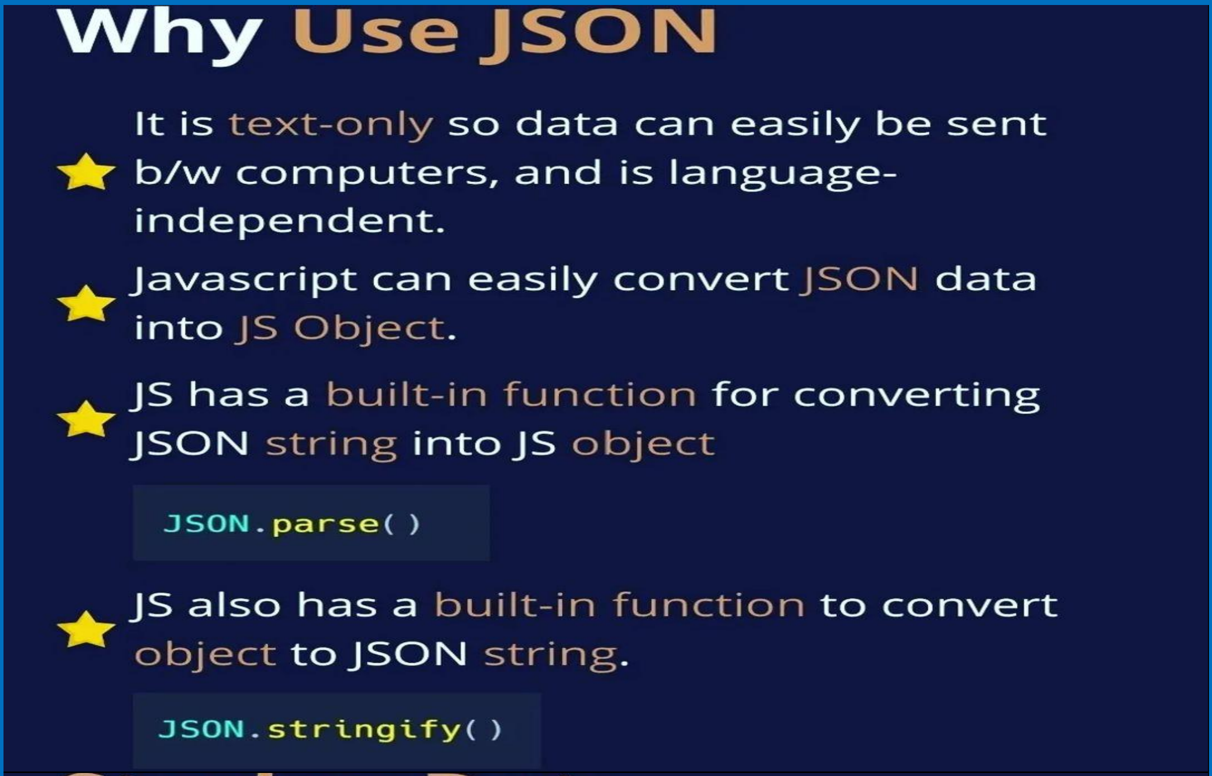


If you parse the JSON string with a JavaScript program, you can access the data as an object:

```
let personName = obj.name;
let personAge = obj.age;
```

Remember: JSON is language independent.

@AAKASH KANOJIYA



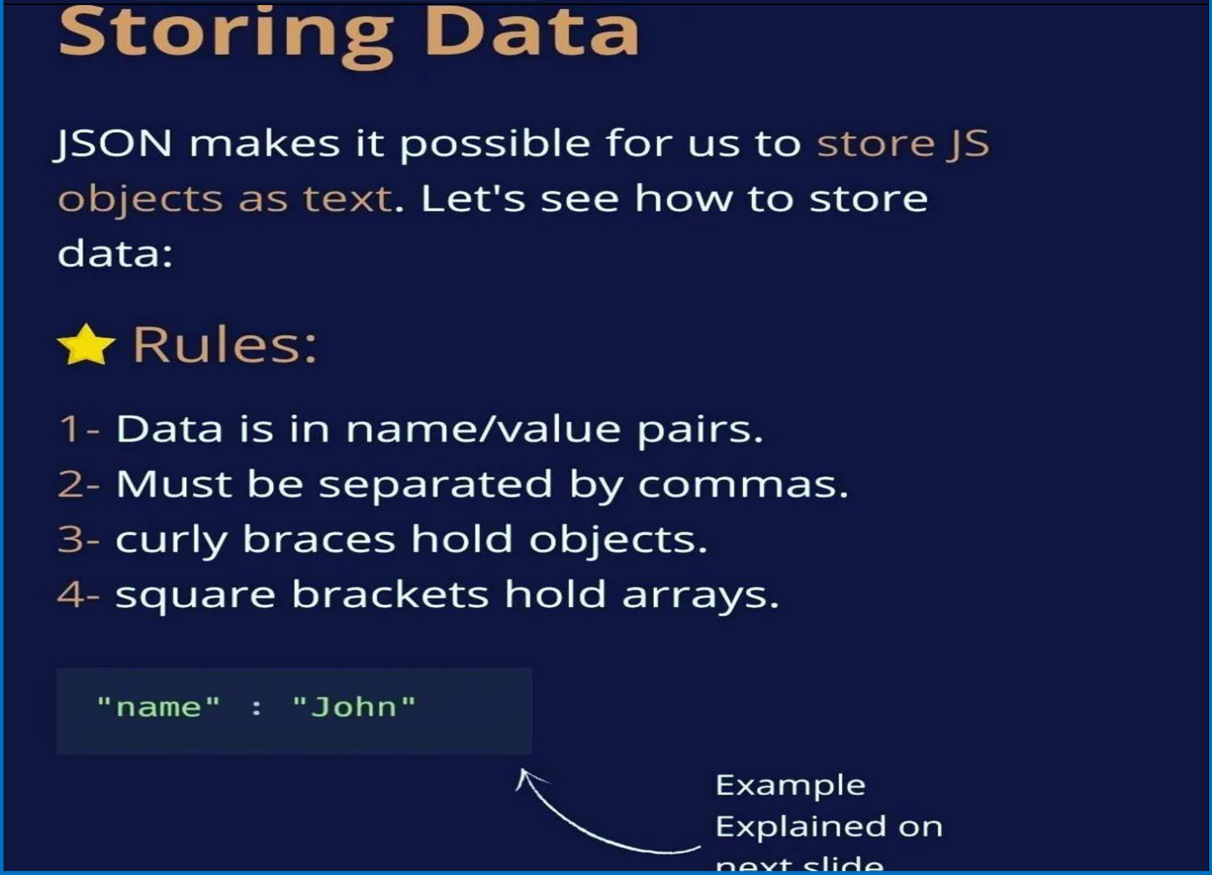
Why Use JSON

- It is text-only so data can easily be sent b/w computers, and is language-independent.
- Javascript can easily convert JSON data into JS Object.
- JS has a built-in function for converting JSON string into JS object

```
JSON.parse()
```

- JS also has a built-in function to convert object to JSON string.

```
JSON.stringify()
```



Storing Data

JSON makes it possible for us to store JS objects as text. Let's see how to store data:

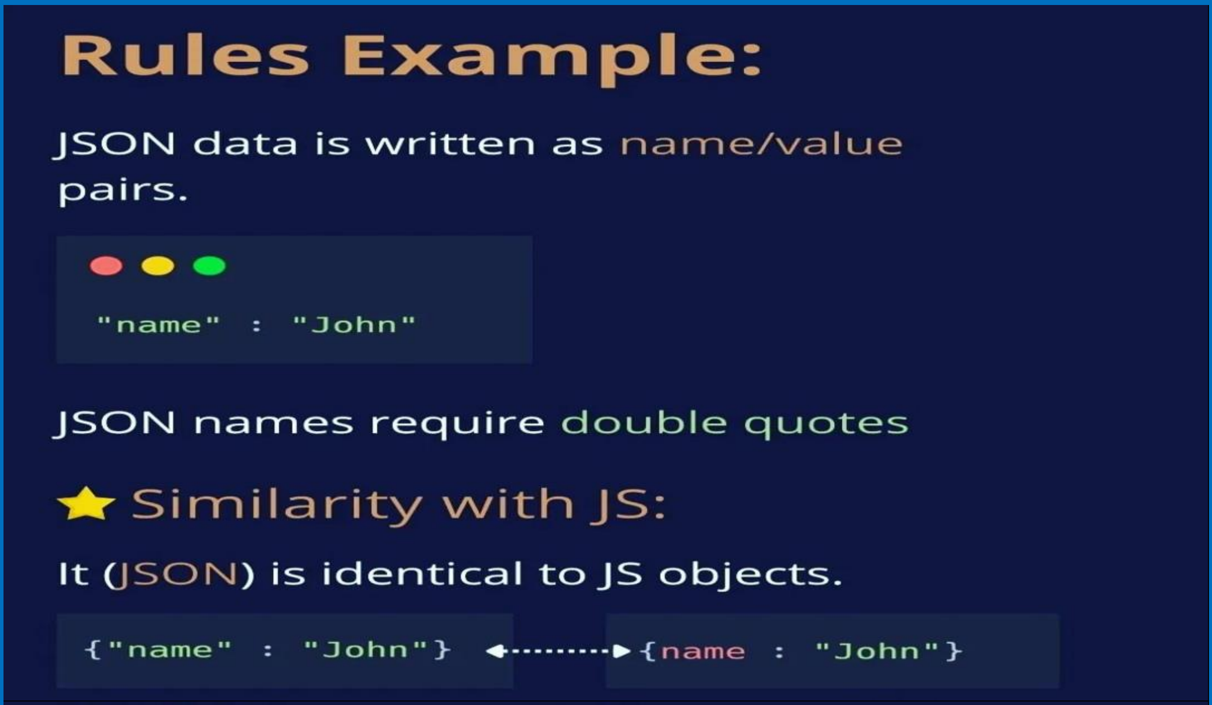
★ Rules:

1- Data is in name/value pairs.
2- Must be separated by commas.
3- curly braces hold objects.
4- square brackets hold arrays.

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

Example
Explained on
next slide

@AAKASH KANOJIYA



Rules Example:

JSON data is written as name/value pairs.

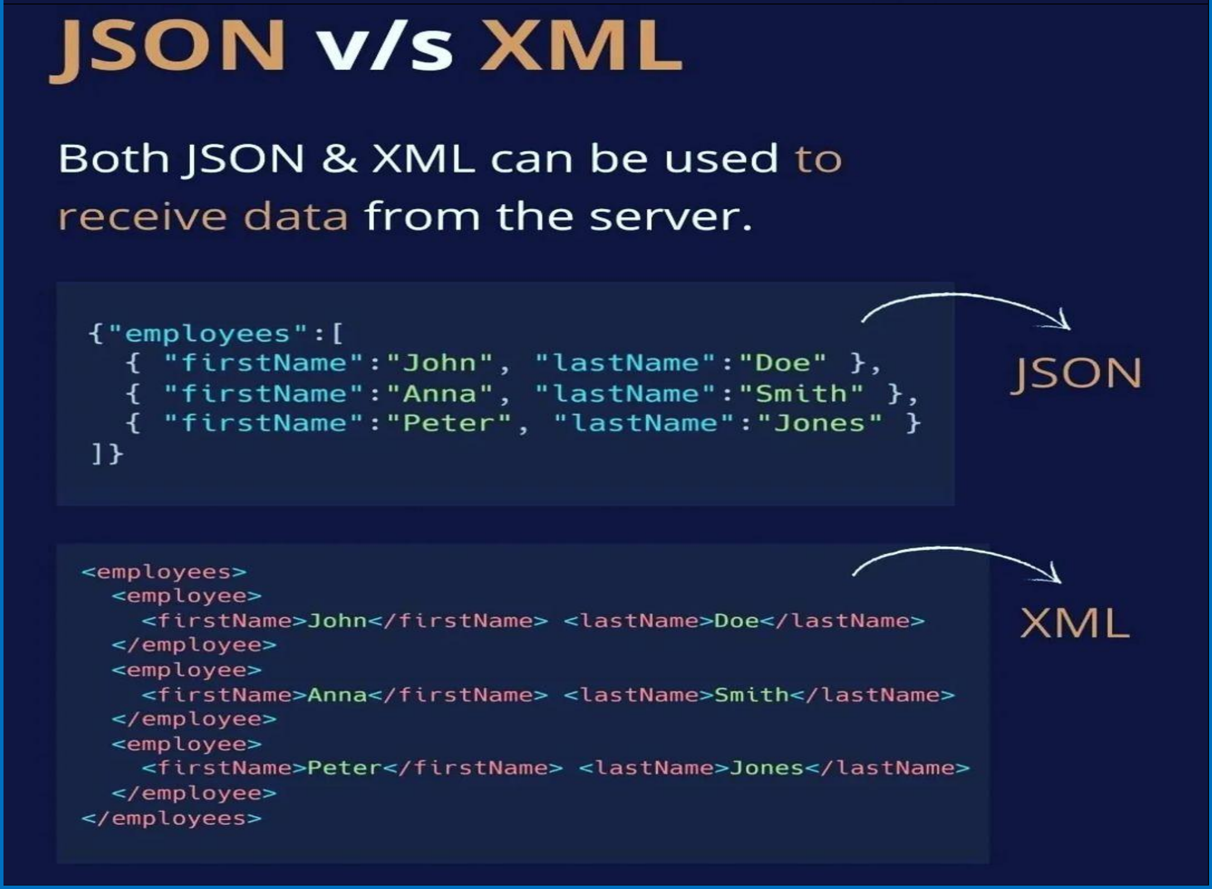
```
"name" : "John"
```

JSON names require double quotes

⭐ Similarity with JS:

It (JSON) is identical to JS objects.

```
{"name" : "John"} ↔ {name : "John"}
```



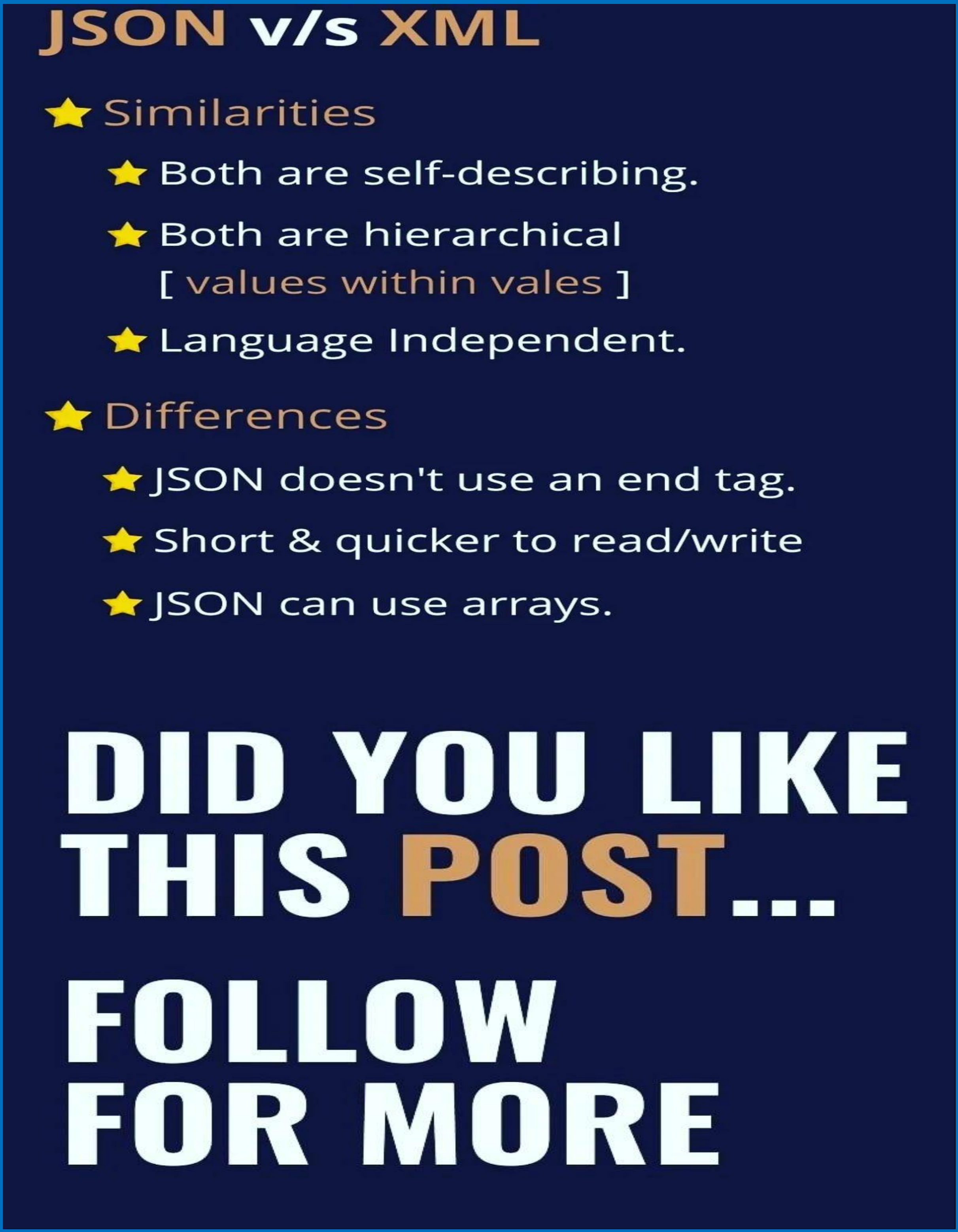
JSON v/s XML

Both JSON & XML can be used to receive data from the server.

```
{"employees":[
  { "firstName":"John", "lastName":"Doe" },
  { "firstName":"Anna", "lastName":"Smith" },
  { "firstName":"Peter", "lastName":"Jones" }
]}
```

```
<employees>
  <employee>
    <firstName>John</firstName> <lastName>Doe</lastName>
  </employee>
  <employee>
    <firstName>Anna</firstName> <lastName>Smith</lastName>
  </employee>
  <employee>
    <firstName>Peter</firstName> <lastName>Jones</lastName>
  </employee>
</employees>
```

@AAKASH KANOJIYA



JSON v/s XML

★ Similarities

- ★ Both are self-describing.
- ★ Both are hierarchical
  [values within vales]
- ★ Language Independent.

★ Differences

- ★ JSON doesn't use an end tag.
- ★ Short & quicker to read/write
- ★ JSON can use arrays.

DID YOU LIKE THIS POST...

FOLLOW FOR MORE

@AAKASH KANOJIYA