JSON

Partha Das Chowdhury

University of BRISTOL

The Basics

- Sending data to be displayed on a webpage?
- Many programming languages support it.
- Supports numbers, booleans, strings, null, arrays (ordered sequences of values), and objects
- Does not support functions, regular expressions etc.
- JSON cannot be called or constructed.

The Basics

Strings in JSON.   →   { "name":"Joe" } (Strings in double quotes)

Numbers in JSON   →   { "age":20 }

JSON can be objects.   →   { "student":{ "name":"Joe", "age":30} }

Use the (.) notation to access a value.

JSON cannot contain functions JavaScript objects can contain functions

```javascript
const data = {
    "name": "CyberJoe",
    "hobby": "hacking",
    "language" : ["JavaScript", "HTML", "CSS"]
}
// accessing JSON object
console.log(data.name);
console.log(data.hobby);
console.log(data.language[0]);
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

ja21121@C02DWCVPML7H Teaching 2023 % node json.js
Debugger attached.
CyberJoe
hacking
JavaScript

```javascript
1    const obj = {name: "John", age: 30, city: "New York"};
2    const myJSON = JSON.stringify(obj);
3    console.log(myJSON);
```

JavaScript object to JSON object

PROBLEMS   OUTPUT   DEBUG CONSOLE   **TERMINAL**

```
ja21121@C02DWCVPML7H Teaching 2023 % node index.js
Debugger attached.
{"name":"John","age":30,"city":"New York"}
Waiting for the debugger to disconnect...
ja21121@C02DWCVPML7H Teaching 2023 % 
```

```javascript
1    const string = '{"name":"John","age":30,"city":"New York"}';
2    const myJSON = JSON.parse(string);
3    console.log(myJSON);
```

JSON object to JavaScript Object

PROBLEMS   OUTPUT   DEBUG CONSOLE   **TERMINAL**

```
ja21121@C02DWCVPML7H Teaching 2023 % node index.js
Debugger attached.
{ name: 'John', age: 30, city: 'New York' }
Waiting for the debugger to disconnect...
ja21121@C02DWCVPML7H Teaching 2023 %
```

JSON vs XML

- Easy to parse
- Example from www.w3schools.com
- JSON can use arrays

JSON Example

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

XML Example

```xml
<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>
```

Thank You