WEB TECHNOLOGIES

XML vs. JSON

Prof. Pavan A C

Department of Computer Science and Engineering

Acknowledgement:

Teaching Assistants(Harini B and Chandana MS)

XML vs. JSON

XML

- XML stands for “extensible markup language”.
- XML was designed in 1996 and officially became a W3C standard in 1998.
- It was created to better represent data formats with a hierarchical structure.
- XML
  - Structures, stores, and transfers information and describes what the data is.
- HTML
  - Display data and focus on how the data looks.

XML Vs JSON HTML5

XML PROS

- This data format fully supports hierarchical data structures and is very appropriate when receiving complex data as a response.
- It is also very human readable. Most browsers have built in XML readers that allow you to inspect XML files.
- Since XML was the first standard hierarchical data format, most APIs have built in functionality to automatically convert XML data streams into native data structures like objects.

XML Vs JSON HTML5

XML CONS

- This data format is about three times as large as CSV. This is because each data element has an associated open and close parameter tag.

XML vs. JSON

JSON

- JSON stands for (Javascript Object Notation).
- It was invented in 2001 and became popularized by Yahoo and Google in 2005 and 2006.
- It was created as an alternative to XML. Like XML, however, it represents hierarchical data with the use of commas, curly braces and brackets.

XML Vs JSON HTML5

JSON PROS

- This data format supports hierarchical data while being smaller in size than XML.
- As its name implies, it was also created to more easily **parse data into native Javascript** objects, making it very useful for web applications.
- JSON is the best of both worlds with respect to CSV and XML. It's simple and compact like CSV, but supports hierarchical data like XML. Unlike XML, JSON formats are only about twice as large as CSV formats.

XML Vs JSON HTML5

JSON CONS

- This data format has a little bit less support than XML.
- Since JSON is relatively newer than XML, fewer APIs exist to automatically convert JSON to native data structures.
- However, this is rapidly changing because newer APIs and plugins are supporting both XML and JSON.

XML vs. JSON

Format

XML

```
<empinfo>
   <employees>
      <employee>
         <name>James Kirk</name>
         <age>40></age>
      </employee>
      <employee>
         <name>Jean-Luc Picard</name>
         <age>45</age>
      </employee>
      <employee>
         <name>Wesley Crusher</name>
         <age>27</age>
      </employee>
   </employees>
</empinfo>
```

JSON

```
{  "empinfo" :
      {
            "employees" :  [
            {
                  "name" : "James Kirk",
                  "age" : 40,
            },
            {
                  "name" : "Jean-Luc Picard",
                  "age" : 45,
            },
            {
                  "name" : "Wesley Crusher",
                  "age" : 27,
            }
                              ]
      }
}
```

XML vs. JSON

Differences

JSON	XML
Based on JavaScript language	Derived from SGML
Way of representing objects	Markup language to represent data items
No support for namespaces	Supports namespaces
Support arrays	Doesn’t support arrays
Files are very easy to read as compared to XML	Documents are comparatively difficult to read and interpret
Doesn’t use end tag	Has start and end tags
Less secure	More secured than JSON
Doesn’t supports comments	Supports comments
Supports only UTF-8 encoding	Supports various encoding

XML Vs JSON HTML5

JSON

JSON String

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

XML Vs JSON HTML5

JSON

- JavaScript has a built in function for converting JSON strings into JavaScript objects:

  JSON.parse()

- JavaScript also has a built in function for converting an object into a JSON string:

  JSON.stringify()

- You can receive pure text from a server and use it as a JavaScript object.
- You can send a JavaScript object to a server in pure text format.
- You can work with data as JavaScript objects, with no complicated parsing and translations.

XML Vs JSON HTML5

XML Parser

A text string is defined:

```
text = "<bookstore><book>" +
"<title>Everyday Italian</title>" +
"<author>Giada De Laurentiis</author>" +
"<year>2005</year>" +
"</book></bookstore>";
```

XML Vs JSON HTML5

XML Parser

Parses a text string into an XML DOM object, and extracts the info from it with JavaScript:

An XML DOM parser is created:

```
parser = new DOMParser();
```

The parser creates a new XML DOM object using the text string:

```
xmlDoc = parser.parseFromString(text,"text/xml");
```

XML DOM object to XML string:

XMLSerializer interface provides serializeToString() method to contruct an XML string

```
var oSerializer = new XMLSerializer();
    var sXML = oSerializer.serializeToString(xmlDoc);
    console.log(sXML);
```

XML Vs JSON HTML5

MCQ

1. Which of the following is the correct JavaScript function to convert a JSON string into a JavaScript object?

A) JSON.stringify()
B) parseJSON()
C) JSON.parse()
D) JSON.toObject()

2. In the following code, what will be the output in the console?

javascript

```
let person = { "name": "John", "age": 30 };

console.log(JSON.stringify(person));
```

A) [object Object]
B) {"name":"John","age":30}
C) undefined
D) name=John,age=30

XML Vs JSON HTML5
MCQ

3. Which statement about XML is true?

A) XML does not support comments
B) Every XML element must have both a start and end tag
C) XML supports arrays natively
D) XML cannot be read by browsers

4. Using JavaScript and the DOMParser, which line successfully parses an XML string str into an XML DOM object?

A) let obj = JSON.parse(str, 'application/xml');
B) let obj = new XMLSerializer().parseFromString(str, 'text/xml');
C) let obj = (new DOMParser()).parseFromString(str, 'text/xml');
D) let obj = parseDOM(str, 'text/xml');

XML Vs JSON HTML5
MCQ Answers

- Answer 1: C) JSON.parse()
- Answer 2: B) {"name":"John","age":30}
- Answer 3: B) Every XML element must have both a start and end tag
- Answe 4: C) let obj = (new DOMParser()).parseFromString(str, 'text/xml');

XML Vs JSON HTML5

JSON

When storing data, the data has to be a certain format, and regardless of where you choose to store it, text is always one of the legal formats.

JSON makes it possible to store JavaScript objects as text.

A common use of JSON is to exchange data to/from a web server.

When receiving data from a web server, the data is always a string.

Parse the data with JSON.parse(), and the data becomes a JavaScript object.

use the JavaScript function JSON.parse() to convert text into a JavaScript object:

Convert a JavaScript object into a string with `JSON.stringify()`

THANK YOU

Prof. Pavan A C

Department of Computer Science and Engineering