JSON (JavaScript Object Notation)

"Code with Passion!"



Topics

- What is and Why JSON?
- JSON Object
- JavaScript Object and JSON text conversion
- JSON in Java

What is & Why JSON?

What is JSON?

- JSON is a text format representation of JavaScript object
 - More precisely, it is a text format representation of a JavaScript object literal with some restrictions (see the next slide for these restrictions)
 - Readily usable in JavaScript code you can easily convert JSONbased text format representation to JavaScript object
- Lighter text-based data-interchange format
 - Compared to XML
- Simple format
 - Easy for humans to read and write
 - > Easy for machines to parse and generate

JSON Restrictions

- A key must be double quoted "" string in JSON
 - Single quoted string is not allowed
- The values can be one of the following it cannot be a function for example
 - > a string
 - > a number
 - > another (JSON-based) JavaScript object
 - > an array
 - > true
 - > false
 - > null

Example

- The following is a valid JavaScript object literal but it is not a valid JSON format because
 - > The keys are not in double-quoted strings
 - > The value cannot be a function

```
{
    property1 : true,
    showMessage : function (msg) { alert(msg) }
}
```

Why Use JSON over XML?

- Lighter and faster than XML as on-the-wire text-based data format
- JSON data is typed while XML data is typeless
 - > JSON types: string, number, array, boolean, etc
 - > XML data are all string
- Native data form for JavaScript code
 - JSON data is readily converted a JavaScript object in your JavaScript code vs. XML data needed to be parsed and assigned to variables through tedious DOM APIs
 - Retrieving values is as easy as reading from an object property in your JavaScript code

Where is JSON Used?

- Data format in communication protocols
- Data format
 - NoSQL document databases such as CouchBase, MongoDB
- Configuration information

JSON Notation

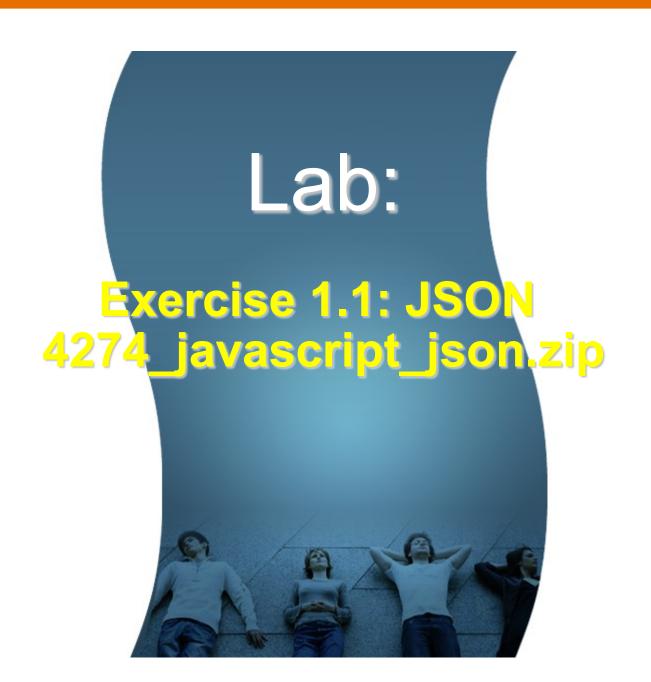
JSON Notation

- A collection of name/value pairs with some restrictions
 - Aligned with JavaScript object format remember JavaScript object is also a collection of name/value pairs
 - JSON notation can be used in the JavaScript language with no extra work
- A JSON notation begins with { (left brace) and ends with } (right brace)
- Each name is followed by: (colon) and the name/value pairs are separated by, (comma)

```
{"name": "sangshin", "hobby": "golf"}
```

Example: JSON-format JavaScript Object

- In this example, a JSON-format JavaScript object is created containing a single member "bindings", which contains an array containing three objects, each containing "ircEvent", "method", and "regex" members
- Members can be retrieved using dot or subscript operators myJSONObject.bindings[0].method // "newURI"



JavaScript Object to/from JSON Text Conversion

JSON Text to JavaScript Object Conversion in JavaScript code

- To convert a JSON text into an JavaScript object (some people call it JSON object), use the eval() function
 - > eval('(' + <JSON-text> + ')') invokes the JavaScript compiler
 - Since <JSON-text> is a valid representation of JavaScript syntax, the compiler will correctly parse the text and produce a JavaScript object

```
var myJSONtext = "{'my_text': 'Life is good!',
    'my_number':35}";
var myObject = eval('(' + myJSONtext + ')');
```

Security & JSON Parser

- eval() can compile and execute any JavaScript program, so there can be security issues (i.e., cross-site scripting)
- Use of eval() is strongly discouraged
- Use eval() only when the source can be trusted
- When security is a concern the source cannot be trusted -, it is better to use a JSON parser
- A JSON parser will only recognize JSON text and so is much safer
 in other words, it will not execute malicious JavaScript code

var myObject = JSON.parse(myJSONtext);

Object to Text Conversion

var myJSONText = JSON.stringify(myJSONObject2);

You can convert JavaScript object into JSON text



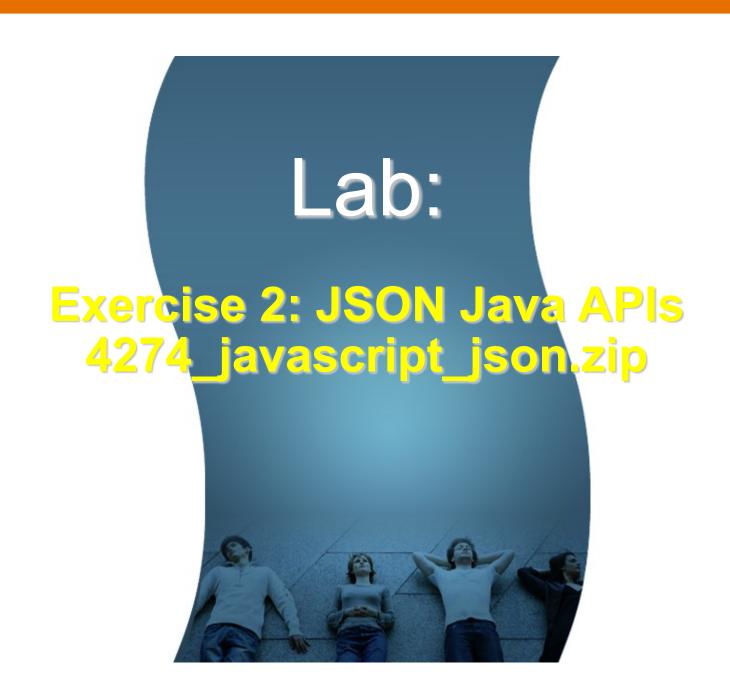
JSON in Java

JSON Tools for Java Developer

- Parser
 - Parse JSON text files and convert these to a Java model
- Renderer
 - > Render a Java representation into text
- Serializer
 - Serialize plain POJO clusters to a JSON representation
- Validator
 - Validate the contents of a JSON file using a JSON schema

JSON Java Packages

- Jackson
 - http://wiki.fasterxml.com/JacksonHome
 - Used internally by Jersey
- Douglas Crockford's implementation
 - https://github.com/douglascrockford/JSON-java
- Json-lib
 - http://json-lib.sourceforge.net/
 - > Based on Douglas Crockford's implementation
- Google GSON
 - http://code.google.com/p/google-gson/



Code with Passion!

