What is The JSON(JavaScript Object Notation)?

JavaScript Object Notation(JSON) is a lightweight text-based open standard designed for human-readable data interchange. It is derived from the JavaScript programming language for representing simple data structures and associative arrays, called objects. And JSON is language-independent, with parsers available for virtually every programming language. Uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python,php
The JSON format is often used for serializing and transmitting structured data over a network connection. When third party data interchane(REST Services) then JSON may used there LIKE SHOP .It is primarily used to transmit data between a server and web application, serving as an alternative to XML.

Who is the Father or creater of JSON ?

Douglas Crockford called as the Father of JSON

---

what is the file extension of JSON

The JSON filename extension is .json.

Explain Json with php

Json is too much easy with php There is no installation needed to use these functions; they are part of the PHP core. nothing more need to know just only use { ,[and create json format string and use three php function json_encode() to get JSON representation of a value, json_decode() for Decodes a JSON string, ¦json_last_error() to get the last error occurred in process.
write your desire string in below format and use php funtions :

```
$string='{
"firstName": "Rohit",
"lastName": "Singh",
"age": 26,
"address": {
"streetAddress": "Mira Road Thane ",
"city": "Mumbai",
"state": "maharshtra",
"postalCode": "401107"
},
"phoneNumber": [
{ "type": "home", "number": "022 333-1234" },
{ "type": "fax", "number": "022 444-4567" }
]
}';
$decodeString = json_decode($string);
echo 'First Name – '.$decode->{"firstName"};
echo 'Last Name – '.$decode->{"lastName"};
echo 'Address – '.$decode->{"address"}->{"streetAddress"};
```

Out put : Print below

First Name – Rohit

Last Name – Singh

Address – Mira Road Thane

Why Use JSON over XML

Lighter and faster than XML as on-the-wire data format

JSON objects are typed while XML data is typeless

JSON types: string, number, array, boolean,

XML data are all string

Native data form for JavaScript code

Data is readily accessible as JSON objects in your JavaScript code whereas 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

---

Explain JSON Structures

A collection of name/value pairs

In various languages, this is realized as an object, record, struct, dictionary, hash table, keyed list, or associative array

An ordered list of values

In most languages, this is realized as an array, vector, list, or sequence

These are universal data structures supported

A JSON object is an unordered set of name/value pairs

A JSON object begins with { (left brace) and ends with } (right brace)

Each name is followed by : (colon) and the name/value pairs are separated by , (comma)

Compare JSON with JavaScript

JSON is a subset of the object literal notation of JavaScript

JSON can be used in the JavaScript language with no muss or fuss

Example: JSON Object

var myJSONObject = {“bindings”: [{"ircEvent": "PRIVMSG", "method": "newURI", "regex": "^http://.*"},

{"ircEvent": "PRIVMSG", "method": "deleteURI", "regex": "^delete.*"},

{"ircEvent": "PRIVMSG", "method": "randomURI", "regex": "^random.*"}

]

};

In this example, a JSON 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”

Text to Object Conversion in

JavaScript code

var myObject = eval(‘(‘ + myJSONtext + ‘)’);

To convert a JSON text into an JSON object, use

the eval() function > eval() invokes the JavaScript compiler

Since JSON is a proper subset of JavaScript, the compiler will

correctly parse the text and produce an object structure

---

what is the Security in JSON Parser

Security in JSON Parser is explained by below examples

// Include http://www.json.org/json.js

var myObject = myJSONtext.parseJSON();

eval() can compile and execute any JavaScript program, so there can be security issues (cross-site scripting)

Use eval() 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

Object to Text Conversion

var myJSONText = myObject.toJSONString();

You can convert JSON object into JSON text

JSON does not support cyclic data structure

Do not give cyclical structures to the JSON stringifier

Do you know JSON Tools for Java Developer

Some of JSON tool for java developer is

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

JSONObject Java Class

A JSONObject is an unordered collection of name/value pairs

The put methods adds a name/value pair to an object

The texts produced by the toString methods strictly conform to the JSON syntax rules

myString = new JSONObject().put(“JSON”, “Hello, World!”).toString();

// myString is {“JSON”: “Hello, World”}

How to Generate or Send JSON Data at the Server Side

Create JSONObject Java object

Add name and value pairs using put method

Convert it to String type using toString method and send it to the client with content-type as “text/xml” or “text/plain”

myString = new JSONObject().put(“JSON”, “Hello, World!”).toString();

// myString is {“JSON”: “Hello, World”}

How to Receive JSON Data at the Client Side

JSON data is received as a string

Calling eval() will generate JSON object in JavaScript code

var JSONdata = eval(req.responseText);

Once you have JSON object, you can use . notation to access its properties

var name = JSONdata.name;

var address = JSONdata.addresses[3];

var streetname = JSONdata.addresses[3].street;

How to Generate/Send JSON Data at the Client Side

Create JSON JavaScript object

Use “POST” HTTP method in the open method of the XMLHttpRequest object

Pass JSON JavaScript object in the send method of XMLHttpRequest object

var carAsJSON = JSON.stringify(car);

var url = “JSONExample?timeStamp=” + new Date().getTime(); createXMLHttpRequest();

xmlHttp.open(“POST”, url, true);

xmlHttp.onreadystatechange = handleStateChange;

xmlHttp.setRequestHeader(“Content-Type”, “application/x-www-form-urlencoded”);

xmlHttp.send(carAsJSON);

How to Receive JSON Data at the Server Side

Read the JSON data as a String type

Create JSONObject Java object from the string String json = readJSONStringFromRequestBody(request);

//Use the JSON-Java binding library to create a JSON object in Java JSONObject jsonObject = null;

try {

jsonObject = new JSONObject(json);

}

catch(ParseException pe) {

}

What is JSON-RPC? What is JSON-RPC-Java?

JSON-RPC is a simple remote procedure call protocol similar to XML-RPC although it uses the lightweight JSON format instead of XML

JSON-RPC-Java is a Java implementation of the JSON-RPC protocol

---

Why JSON-RPC-Java?

It allows you to transparently call server-side Java code from JavaScript with an included lightweight JSON-RPC JavaScript client
It is designed to run in a Servlet container such as Tomcat and can be used with J2EE Application servers to allow calling of plain Java or EJB methods from within a JavaScript DHTML web application

---

Features of JSON-RPC-Java

Dynamically call server-side Java methods from JavaScript DHTML web applications. No Page reloading.
Asynchronous communications.
Transparently maps Java objects to JavaScript objects.
Lightweight protocol similar to XML-RPC although much faster.
Leverages J2EE security model with session specific exporting of objects.
Supports Internet Explorer, Mozilla, Firefox, Safari, Opera and Konqueror

---