# **[Java构造和解析Json数据的两种方法详解一](http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/3096001.html)**

      在[www.json.org](http://www.json.org/" \t "http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/_blank)上公布了很多JAVA下的json构造和解析工具，其中org.json和json-lib比较简单，两者使用上差不多但还是有些区别。下面首先介绍用json-lib构造和解析Json数据的方法示例。

**用org.son构造和解析Json数据的方法详解请参见我下一篇博文：**[Java构造和解析Json数据的两种方法详解二](http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/24/3096437.html" \t "http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/_blank)

**一、介绍**

      JSON-lib包是一个beans,collections,maps,java arrays 和XML和JSON互相转换的包，主要就是用来解析Json数据，在其官网[http://www.json.org/上](http://www.json.org/" \t "http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/_blank)有详细讲解，有兴趣的可以去研究。

**二、下载jar依赖包：可以去**[这里](http://download.csdn.net/detail/lanxuezaipiao/5440331" \t "http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/_blank)**下载**

**三、基本方法介绍**

1. List集合转换成json方法

List list = new ArrayList();list.add( "first" );list.add( "second" );JSONArray jsonArray2 = JSONArray.fromObject( list );

2. Map集合转换成json方法

[IMG_257](http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/javascript:void(0);)

Map map = new HashMap();map.put("name", "json");map.put("bool", Boolean.TRUE);map.put("int", new Integer(1));map.put("arr", new String[] { "a", "b" });map.put("func", "function(i){ return this.arr[i]; }");JSONObject json = JSONObject.fromObject(map);

[IMG_258](http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/javascript:void(0);)

3. Bean转换成json代码

JSONObject jsonObject = JSONObject.fromObject(new JsonBean());

4. 数组转换成json代码

boolean[] boolArray = new boolean[] { true, false, true };JSONArray jsonArray1 = JSONArray.fromObject(boolArray);

5. 一般数据转换成json代码

JSONArray jsonArray3 = JSONArray.fromObject("['json','is','easy']" );

6. beans转换成json代码

[IMG_259](http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/javascript:void(0);)

List list = new ArrayList();JsonBean2 jb1 = new JsonBean2();jb1.setCol(1);jb1.setRow(1);jb1.setValue("xx");JsonBean2 jb2 = new JsonBean2();jb2.setCol(2);jb2.setRow(2);jb2.setValue("");list.add(jb1);list.add(jb2);JSONArray ja = JSONArray.fromObject(list);

[IMG_260](http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/javascript:void(0);)

**四、演示示例**

**这里以基本的几个常用方法进行测试**

[IMG_261](http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/javascript:void(0);)

package com.json;import java.util.ArrayList;import java.util.HashMap;import java.util.List;import java.util.Map;import net.sf.json.JSONArray;import net.sf.json.JSONObject;/\*\* \* 使用json-lib构造和解析Json数据 \* \* @author Alexia \* @date 2013/5/23 \* \*/public class JsonTest { /\*\* \* 构造Json数据 \* \* @return \*/ public static String BuildJson() { // JSON格式数据解析对象 JSONObject jo = new JSONObject(); // 下面构造两个map、一个list和一个Employee对象 Map<String, String> map1 = new HashMap<String, String>(); map1.put("name", "Alexia"); map1.put("sex", "female"); map1.put("age", "23"); Map<String, String> map2 = new HashMap<String, String>(); map2.put("name", "Edward"); map2.put("sex", "male"); map2.put("age", "24"); List<Map> list = new ArrayList<Map>(); list.add(map1); list.add(map2); Employee employee = new Employee(); employee.setName("wjl"); employee.setSex("female"); employee.setAge(24); // 将Map转换为JSONArray数据 JSONArray ja1 = JSONArray.fromObject(map1); // 将List转换为JSONArray数据 JSONArray ja2 = JSONArray.fromObject(list); // 将Bean转换为JSONArray数据 JSONArray ja3 = JSONArray.fromObject(employee); System.out.println("JSONArray对象数据格式："); System.out.println(ja1.toString()); System.out.println(ja2.toString()); System.out.println(ja3.toString()); // 构造Json数据，包括一个map和一个Employee对象 jo.put("map", ja1); jo.put("employee", ja2); System.out.println("\n最终构造的JSON数据格式："); System.out.println(jo.toString()); return jo.toString(); } /\*\* \* 解析Json数据 \* \* @param jsonString Json数据字符串 \*/ public static void ParseJson(String jsonString) { // 以employee为例解析，map类似 JSONObject jb = JSONObject.fromObject(jsonString); JSONArray ja = jb.getJSONArray("employee"); List<Employee> empList = new ArrayList<Employee>(); // 循环添加Employee对象（可能有多个） for (int i = 0; i < ja.size(); i++) { Employee employee = new Employee(); employee.setName(ja.getJSONObject(i).getString("name")); employee.setSex(ja.getJSONObject(i).getString("sex")); employee.setAge(ja.getJSONObject(i).getInt("age")); empList.add(employee); } System.out.println("\n将Json数据转换为Employee对象："); for (int i = 0; i < empList.size(); i++) { Employee emp = empList.get(i); System.out.println("name: " + emp.getName() + " sex: " + emp.getSex() + " age: " + emp.getAge()); } } /\*\* \* @param args \*/ public static void main(String[] args) { // TODO Auto-generated method stub ParseJson(BuildJson()); }}

[IMG_262](http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/javascript:void(0);)

运行结果如下

**五、与org.json比较**

      json-lib和org.json的使用几乎是相同的，我总结出的区别有两点：

      1. org.json比json-lib要轻量得多，前者没有依赖任何其他jar包，而后者要依赖ezmorph和commons的lang、logging、beanutils、collections等组件

      2. json-lib在构造bean和解析bean时比org.json要方便的多，json-lib可直接与bean互相转换，而org.json不能直接与bean相互转换而需要map作为中转，若将bean转为json数据，首先需要先将bean转换为map再将map转为json，比较麻烦。

      总之，还是那句话—适合自己的才是最好的，大家要按需选取使用哪种方法进行解析。最后给大家介绍两款解析Json数据的工具：一是在线工具JSON [Edit（](http://braincast.nl/samples/jsoneditor/" \t "http://www.cnblogs.com/lanxuezaipiao/archive/2013/05/23/_blank)http://braincast.nl/samples/jsoneditor/）；另一个是Eclipse的插件JSON Tree Analyzer，都很好用，推荐给大家使用！