# Blog sobre Java EE

Estás aquí: Inicio/Java EE/REST/JSON-P y Java Enterprise Edition 8

# JSON-P y Java Enterprise Edition 8

15 mayo, 2018 por Cecilio Álvarez Caules — Deja un comentario

# Anúnciate con Google - Con Google AdWo

Empieza logy y consigue 756 paraetu primera campaña al invertir tuso primera cúrtectura mis libros mis cursos adwords.google.com

JSON-P (JSON Processing) es una de las novedades de Java Enterprise Edition 8 .¿Para que sirve JSON-P? . Esta nueva especificación sirve para procesar estructuras JSON de una forma sencilla y acceder a los datos de forma muy directa. La especificación se encarga de definir un puntero a la estructura de datos que necesitamos dentro de un documento JSON. Vamos a ver una pequeña introducción. En primer lugar vamos a definir las dependencias de Maven que necesitamos.

```
cproject xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3
 1
 2
        <modelVersion>4.0.0</modelVersion>
 3
        <groupId>com.arquitecturajava
 4
        <artifactId>JSONP</artifactId>
 5
        <version>1</version>
 6
        <dependencies>
 7
     <dependency>
8
        <groupId>org.glassfish
9
        <artifactId>javax.json</artifactId>
10
        <version>1.1.2
     </dependency>
11
12
        </dependencies>
13
     </project>
```

Es momento de construir nuestro fichero JSON que va a tener unos datos muy sencillos.

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información.

-- 1 )

En este caso estamos ante unos datos de una persona y los cursos que ha realizado. Eclipse nos puede mostrar una estructura a detalle del arbol:

```
Task List

▼{}

▼{}

▼{}

Persona
Semail: "pepe@pepe.com"
Sombre: "pepe"

▼[] cursos

▼{}

Stitulo: "java"

▼{}

Stitulo: ".net"

▼{}

Stitulo: "javascript"
```

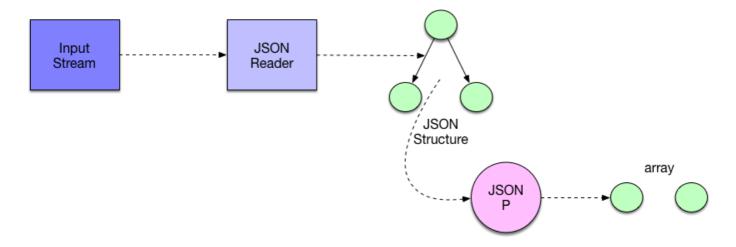
Podemos usar JSON-P (JSON Processing) para acceder a la información de los cursos de una forma muy directa.

```
1
     package com.arquitecturajava;
 2
 3
     import java.io.IOException;
 4
     import java.io.InputStream;
 5
 6
     import javax.json.Json;
 7
     import javax.json.JsonArray;
 8
     import javax.json.JsonPointer;
 9
     import javax.json.JsonReader;
10
     import javax.json.JsonStructure;
     import javax.json.JsonValue;
11
12
13
     public class Principal {
14
15
         public static void main(String[] args) {
16
17
             try (InputStream is = JsonPointer.class.getClassLoader().getResourc
18
                      JsonReader jsonreader = Json.createReader(is)) {
19
                  JsonStructure jsonEstructura = jsonreader.read();
20
21
22
23
                  JsonPointer jsonPointer = Json.createPointer("/persona/cursos")
24
                  JsonValue jv = jsonPointer.getValue(jsonEstructura);
25
                  JsonArray lista=jv.asJsonArray();
26
                  <u>lista.forFach(Svstem.out::nrintln):</u>
```

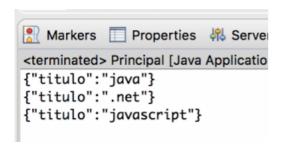
Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información.

# **Utilizando JSON-P**

En este caso el código es muy sencillo de entender. Partimos de un inputStream que se encarga de leer nuestro fichero JSON. Una vez tenemos el fichero JSON utilizamos un JSON Reader para leer el fichero. Realizado este paso leemos el fichero construimos una estructura en memoria a esa estructura la aplicamos un puntero (JSON pointer). Este puntero indica que vamos a seleccionar unicamente los cursos de esta persona.



Realizado este paso convertimos el resultado del puntero y lo imprimimos en la consola como un sencillo array.



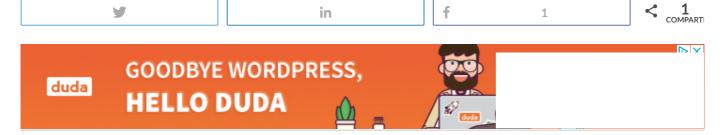
Acabamos de filtrar los datos de una estructura JSON utilizando JSON-P (JSON processing) una de las JSRs de Java EE 8 . El manejo de datos JSON cada día es más habitual y esta JSR nos puede aportar una solución rápida y elegante.

#### Otros artículos relacionados

- 1. JAX-RS Client y JSON
- 2 REST ISON v Iava

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información.





Archivada en: REST

## Leave a Reply

Be the First to Comment!



Start the discussion

☑ Subscribe ▼

**BUSCAR** 

Buscar en este sitio ...

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información.

# **PACKJAVACORE**

### Mis Cursos de Java Gratuitos

#### Java Herencia



Java JDBC



#### **Servlets**

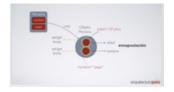


#### Introduccion JPA

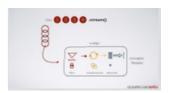


### Mis Cursos de Java

### Programación Orientada a Objeto en Java



#### **Java APIS Core**



Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información.



#### Arquitectura Java Solida con Spring



#### **POPULAR**

Arquitecturas REST y sus niveles

Angular 5 Hello World y su funcionamiento

Nuevo Curso:Arquitectura Java Sólida con Spring 4.3 y Anotaciones

Java 9 Modules y el concepto de modularidad

Java 8 Lambda Syntax ,simplificando nuestro código

El concepto de Java Annotations y su funcionamiento

Spring REST Test utilizando Rest Assured

Spring @Qualifier utilizando @Autowired

Java 9 Collections y sus novedades

Java Composite Pattern y recursividad

#### CONTACTO

contacto@arquitecturajava.com

#### LO MAS LEIDO

JSON-P y Java Enterprise Edition 8

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información.

¿Cuales son las certificaciones Java?

Java Override y encapsulación

Ejemplo de Java Singleton (Patrones y ClassLoaders)

REST HTTP return codes y sus curiosidades

¿Qué es Gradle?

Usando el patron factory

REST JSON y Java

Ejemplo de JPA , Introducción (I)

Uso de Java Generics (I)

Comparando java == vs equals

¿Qué es un Microservicio?

Mis Libros

Spring MVC Configuración (I)

Introducción a Servicios REST

Copyright © 2018 · eleven40 Pro Theme en Genesis Framework · WordPress · Acceder

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información.