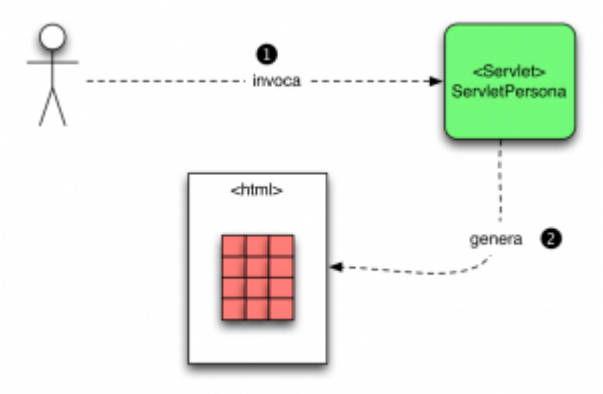
Hoy en dia nos toca trabajar más y más con el formato JSON en el mundo Java. Hay situaciones en las que nuestro servidor de aplicaciones tiene soporte (JEE6) y otras muchas en las que no y nos tenemos que apoyar en frameworks de terceros (Apache CXF). Por ultimo hay situaciones en las que simplemente tenemos que cubrir de forma rápida el problema y seguir adelante .Es en estas situaciones en las que la libreria de Google (Google JSON) es muy práctica y nos permite generar estructuras JSON de forma sencilla .Vamos a verlo a continuación utilizando un sencillo Servlet. En un primer lugar nuestro Servlet imprimirá por pantalla en formato HTML una lista de objetos dentro de una tabla.



Vamos a mostrar el código fuente de la clase Persona y el del Servlet que imprime una lista de personas :

Persona.java

```
package com.arquitecturajava.negocio;

public class Persona {

public Persona(String nombre, int edad) {
 super();
 this.nombre = nombre;
```

```
 this.edad = edad;
 }
 private String nombre;
 private int edad;
 public String getNombre() {
 return nombre;
 }
 public void setNombre(String nombre) {
 this.nombre = nombre;
 }
 public int getEdad() {
 return edad;
 }
 public void setEdad(int edad) {
 this.edad = edad;
 }
}
```

ServletDatos.java

```
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import com.arquitecturajava.negocio.Persona;

public class ServletDatos extends HttpServlet {
 private static final long serialVersionUID = 1L;

public ServletDatos() {
```

```
 super();
 // TODO Auto-generated constructor stub
 }

protected void doGet(HttpServletRequest request,
 HttpServletResponse response) throws ServletException, IOException {

List<Persona> lista = new ArrayList<Persona>();
 lista.add(new Persona("pedro", 20));
 lista.add(new Persona("ana", 30));

PrintWriter pw = response.getWriter();
 pw.println("<html>");
 pw.println("<body>");

for (Persona p : lista) {

pw.println("<p>" + p.getNombre());
 pw.println(p.getEdad() + "</p>");

}
 pw.println("</body>");
 pw.println("</html>");

}
}
```

El servlet nos mostrará la siguiente información.

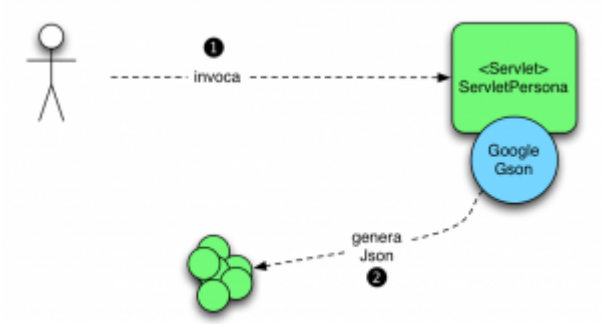


Objetos y JSON (GSON)

Realizada esta operación vamos a modificar el Servlet para utilizar la librería de Google (GSON) y que la información que se genere sea en formato JSON.La librería se encuentra disponible en la siguiente URL.

https://code.google.com/p/google-gson/

Instalada la librería modificaremos el Servlet para que genere los datos en formato JSON.



Vamos a ver el código:

```
package com.arquitecturajava.servlets;

import java.io.IOException;
import java.io.PrintWriter;
```

```
import java.util.ArrayList;
import java.util.List;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import com.arquitecturajava.negocio.Persona;
import com.google.gson.Gson;

public class ServletDatosJSON extends HttpServlet {
 private static final long serialVersionUID = 1L;

 public ServletDatosJSON() {
 super();
 // TODO Auto-generated constructor stub
 }
 protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

 List<Persona> lista = new ArrayList<Persona>();
 lista.add(new Persona("pedro", 20));
 lista.add(new Persona("ana", 30));
 Gson gson= new Gson();
 PrintWriter pw=response.getWriter();
 pw.println( gson.toJson(lista));
 pw.close();

 }
```

}

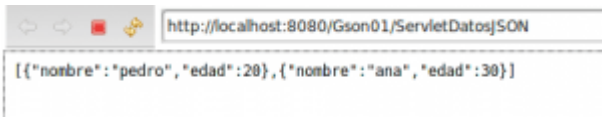
Como podemos ver la libreria es muy sencilla de utilizar . En primer lugar nos creamos un objeto Gson.

Gson gson= new Gson();

Para mas adelante pasarle la lista de personas y que la convierta

pw.println(gson.toJson(lista));

Realizados estos cambios el Servlet devuelve lo siguiente :



Como podemos ver utilizar la librería es muy sencillo