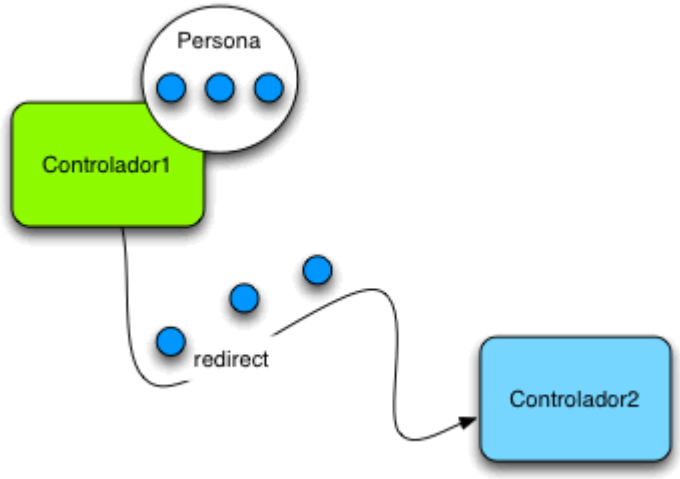
Acabamos de ver como usar Java equals y hashcode

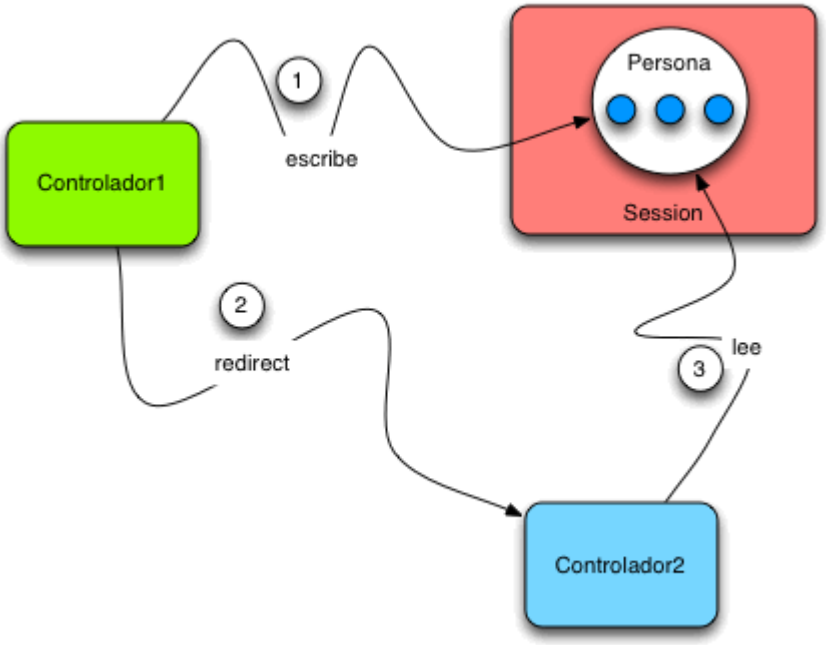
Spring MVC Flash Attributes es una de las características menos entendidas a nivel de manejo de estado dentro de Spring MVC. ¿ Para qué sirven exactamente los Spring MVC Flash Attributes? . Normalmente cuando navegamos entre Controladores de Spring framework usamos operaciones como “redirect:/controlador2” desde por ejemplo el controlador1. Esto obliga a que toda la información que queramos enviar desde el primer controlador al segundo pase a través de la url como parámetro de la petición.



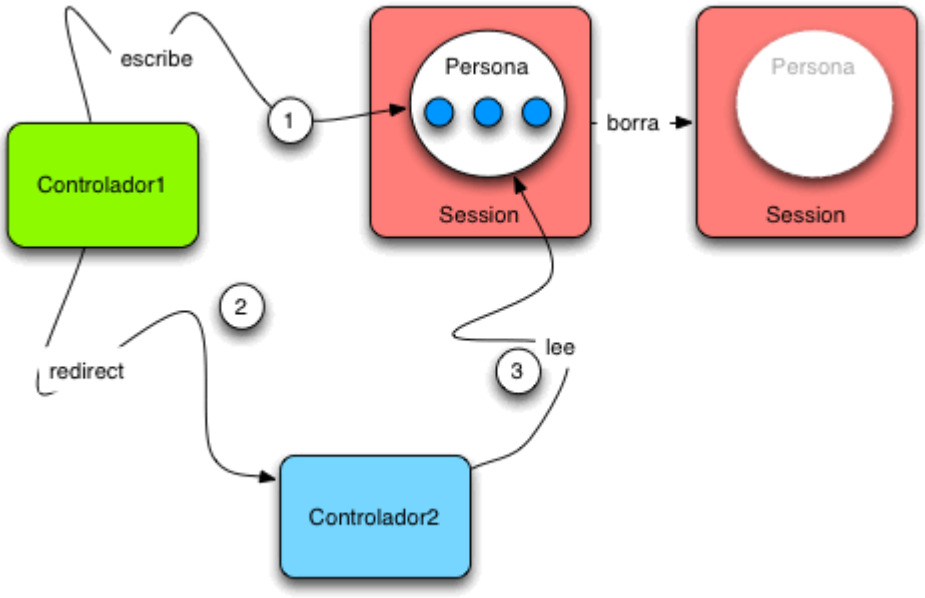
Esto tiene varios problemas , en primer lugar que la información enviada no puede ser muy compleja, en segundo lugar que hay un límite claro de tamaño de la url y por último que la información es muy visible. Spring MVC Flash Attributes aporta un enfoque distinto a la hora de manejar este tipo de información .

¿Qué son los Spring MVC Flash Attributes?

Se trata de una solución de Spring MVC bastante elegante a la hora de compartir información . Habitualmente cuando queremos almacenar información durante varias peticiones utilizamos objetos de Session. Estos objetos son muy versátiles pero lamentablemente penalizan ya que se almacenan en memoria durante bastante tiempo, 30 minutos por defecto en Tomcat.



Los atributos flash funcionan apoyándose en las Sessiones pero solo almacenan los datos durante el instante que dura la redirección.



Vamos a ver un ejemplo en código;

```
package com.arquitecturajava;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.servlet.mvc.support.RedirectAttributes;

@Controller
public class ControladorA {

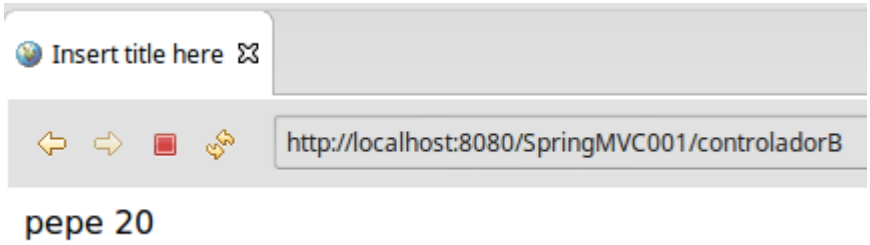
 @RequestMapping("/controladorA")
 public String metodoA(RedirectAttributes atributos) {

 Persona p= new Persona();
 p.setNombre("pepe");
 p.setEdad(20);
 atributos.addFlashAttribute("persona",p);

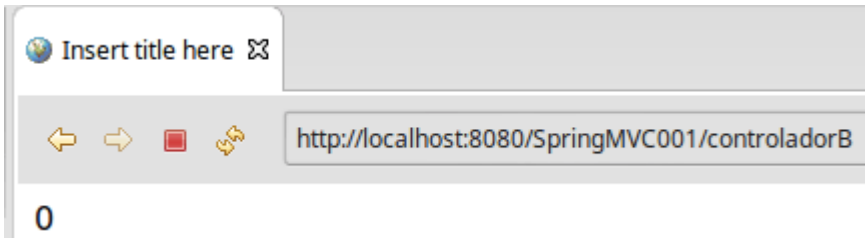
 return "redirect:/controladorB";

 }
}
```

Disponemos de un controladorA que añade como objeto a la session una Persona. Si invocamos la aplicación con http://localhost/SpringMVC001/controlarA el resultado que nos mostrará el navegador será:



Spring MVC nos ha redirigido al controladorB y nos ha pasado los atributos utilizando un objeto Flash y apoyandose en la sesión. Podemos comprobar que este era un objeto flash simplemente recargando la página ya que el navegador cargará directamente el controladorB y como los datos han sido borrados de la sesión ya no dispondremos de la Persona. Es así de sencillo.



Tengamos en cuenta este tipo de atributos nos serán útiles en muy diversas situaciones ya que no mantendrán datos en la sesión de forma absurda.

Otros artículos relacionados :

Introducción a Spring MVC

EJB vs Spring

Spring Framework y aislamiento