



# DESARROLLO DE SOFTWARE I

**Eric Gustavo Coronel Castillo**

[gcoronelc.blogspot.com](http://gcoronelc.blogspot.com)

[ecoronel@uch.edu.pe](mailto:ecoronel@uch.edu.pe)



## TEMA: Sesiones



En esta sesión aprenderás a manejar datos de sesión.





## Parámetro HttpSession

Puedes definir un parámetro de tipo HttpSession para que accedas de manera transparente al objeto de sesión y puedas guardar los datos relevantes de la sesión del usuario, como por ejemplo su identificación.

```
@RequestMapping(value = "procesar.htm", method = RequestMethod.POST)
public String procesar(HttpSession session){

    return "vistaDestino";
}
```



## Anotación @SessionAttributes

Esta anotación sirve para acceder o establecer las variables de sesión. Las variables se declaran en la clase, y se accede a ellas a través del modelo.

```
@Controller
@SessionAttributes("usuario")
public class ProcesoController {

    @RequestMapping(value="/procesar.htm")
    public void procesar(Model model) {
        model.addAttribute("usuario", "Gustavo Coronel");
        . . .
    }
}
```



## Anotación @SessionAttributes

También puedes definir varios parámetros de sesión con @SessionAttributes.

```
@Controller
@SessionAttributes({"usuario","catalogo"})
public class ProcesoController {

    @RequestMapping(value="/procesar.htm")
    public void procesar(Model model) {
        model.addAttribute("usuario", "Gustavo Coronel");
        . . .
    }

    . . .
    . . .
}
```



## Parámetro SessionStatus

Si lo que necesitas es eliminar los datos de la sesión, debes crear un parámetro de tipo **SessionStatus** en el controlador y ejecutar el método **setComplete()**.

```
@Controller
@SessionAttributes({"usuario","catalogo"})
public class ProcesoController {

    @RequestMapping(value="/salir.htm")
    public String salir(SessionStatus status) {
        status.setComplete();
        return "index"
    }

    : : :
    : : :
}
```