

Java para Internet

Java Servlets

Luis Fernando Llana Díaz
llana@sip.ucm.es

Departamento de Sistemas Informáticos y Programación
Universidad Complutense de Madrid

Escuela Complutense 2005

Introducción

Servidor de aplicaciones

Mis primeros servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros servlets

Juntando todo

Paquetes, Clases e interfaces

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

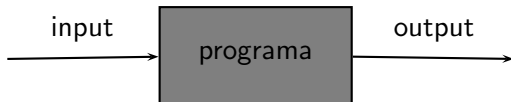
Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

¿Qué es un programa?



En pascal:

```
program holaMundo (input,output);  
var  
    nombre : String;  
begin  
    readLn(nombre);  
    writeLn('Hola ',nombre);  
end.
```

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

Un servlet

- ▶ input → `HttpServletRequest`
- ▶ output → `HttpServletResponse`

```
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
public class HolaMundo extends HttpServlet {
    public final void doGet(final HttpServletRequest request,
                           final HttpServletResponse response)
        throws ServletException, IOException {
        ServletOutputStream out = response.getOutputStream();
        String nombre = request.getParameter("nombre");
        response.setContentType("text/html");
        if (nombre==null) {
            out.println("<p>Hola Mundo</p>");
        } else {
            out.println("<p>Hola "+nombre+" </p>");
        }
    }
}
```

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

Servidor de aplicaciones: Tomcat

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

Los severlet son controlados por servidor de aplicaciones.

Tomcat: <http://jakarta.apache.org/tomcat/index.html>

Cuando una petición llega

1. Si el objeto de la clase no ha sido creado

- 1.1 Carga la clase

- 1.2 Crea el objeto

- 1.3 Invoca el método `init`.

2. Llama al método `service`.

Si debe eliminar al servlet llama al método `destroy`.

Interfaz `javax.servlet.Servlet`

- ▶ `void destroy()`
- ▶ `void init(ServletConfig config)`
- ▶ `void service(ServletRequest req, ServletResponse res)`

Clase `javax.servlet.http.HttpServlet`

- ▶ `void doGet(HttpServletRequest req, HttpServletResponse resp)`
- ▶ `void doPost(HttpServletRequest req, HttpServletResponse resp)`

Servlet contador I

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

```
public class Contador extends HttpServlet {
    private int contador;

    /*
     * Metodos para guardar el contador antes de
     * finalizar y leerlo al empezar.....
     */

    private synchronized int incrementa() {
        contador++;
        return contador;
    }

    public final void doGet(final HttpServletRequest request,
                           final HttpServletResponse response)
        throws ServletException, IOException {
        int n=incrementa();
        ServletOutputStream out = response.getOutputStream();
        String nombre = request.getParameter("nombre");
        response.setContentType("text/html");
        out.println(" <p>Numero de accesos:"+n+" </p>");
    }
}
```

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

Servlet contador II

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

```
private ServletConfig config;
private final static String strFichero="fichero";

public final void init(final ServletConfig c)
    throws ServletException {
    config = c;
    ServletContext ctx = config.getServletContext();
    String fichero = config.getInitParameter(strFichero);
    try {
        BufferedReader f
            = new BufferedReader(
                new FileReader(ctx.getRealPath(fichero)));
        contador = Integer.parseInt(f.readLine());
        f.close();

    } catch (FileNotFoundException e) {
        contador = 0;
    } catch (IOException e) {
        contador = 0;
    }
}
```


Servlet contador III

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

```
public final void destroy() {  
    ServletContext ctx = config.getServletContext();  
    String fichero = config.getInitParameter(strFichero);  
    try {  
        FileWriter f = new FileWriter(ctx.getRealPath(fichero));  
        f.write(contador+"\n");  
        f.close();  
    } catch (IOException e) {  
        throw new RuntimeException(e.getLocalizedMessage());  
    }  
}
```

Aplicación Hola I

```
TOMCAT_HOME/webapps/hola
|-- build.xml
|-- contador.txt
|-- html
|   |-- cabecera.html
|   |-- pie.html
|-- prj.el
|-- src
|   |-- Contador.java
|   |-- Debug.java
|   |-- EnvoltorioHTML.java
|   |-- Fecha.java
|   |-- HolaMundo.java
|   |-- IncluyeFichero.java
|   |-- Login.java
|   |-- Sesion.java
'-- WEB-INF
    |-- classes
    |   |-- Contador.class
    |   |-- Debug.class
    |   |-- EnvoltorioHTML.class
    |   |-- Fecha.class
    |   |-- HolaMundo.class
    |   |-- IncluyeFichero.class
    |   |-- Login.class
    |   |-- Sesion.class
    |-- lib
    '-- web.xml
```

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones
Filtros
Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

Fichero web.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app>
  <display-name>Servlets de prueba</display-name>
  <servlet>
    <servlet-name>hola</servlet-name>
    <servlet-class>HolaMundo</servlet-class>
  </servlet>
  <servlet>
    <servlet-name>contador</servlet-name>
    <servlet-class>Contador</servlet-class>
    <init-param>
      <param-name>fichero</param-name>
      <param-value>contador.txt</param-value>
    </init-param>
  </servlet>
  <servlet-mapping>
    <servlet-name>invoker</servlet-name>
    <url-pattern>/http/*</url-pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>invoker</servlet-name>
    <url-pattern>/servlet/*</url-pattern>
  </servlet-mapping>
</web-app>
```

Índice

Introducción

Servidor de
aplicaciones

Mis primeros servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

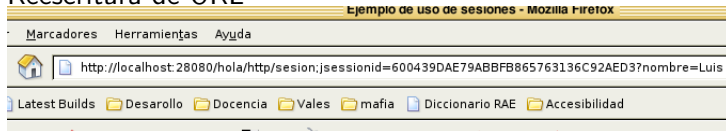
Juntando todo

Paquetes, Clases e
interfaces

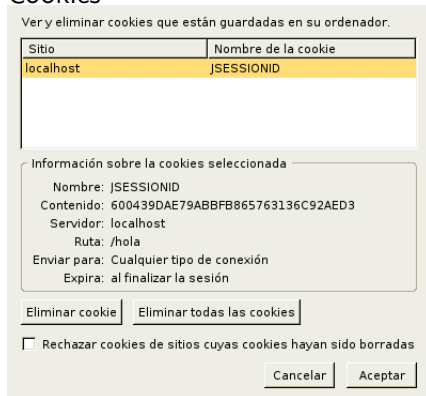
Memoria entre diferentes invocaciones a servlets.

```
HttpSession sesion = request.getSession();
String nombre =(String)sesion.getAttribute(strNombre);
Integer numAccesos = (Integer)sesion.getAttribute(strNumAccesos);
if (numAccesos==null) {
    numAccesos = new Integer(0);
}
sesion.setAttribute(strNumAccesos , new Integer(numAccesos.intValue()+1));
.....
.....
if (borroSesion!=null) {
    contenido.append(sesionBorrada(response));
    sesion.removeAttribute(strNumAccesos);
    sesion.removeAttribute(strNombre);
}
```

► Reescritura de URL



► Cookies

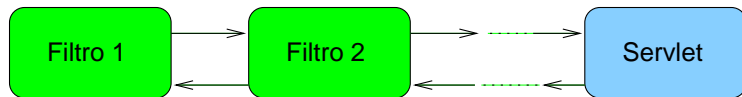
[Índice](#)[Introducción](#)[Servidor de aplicaciones](#)[Mis primeros servlets](#)[Aplicación Hola](#)[Avanzando](#)[Sesiones](#)[Filtros](#)[Invocando otros servlets](#)[Juntando todo](#)[Paquetes, Clases e interfaces](#)

Toda URL debe ser *codificada*. Funciona tanto si tiene como si no tiene las *cookies* habilitadas.

```
private StringBuffer  
    formularioContinua(HttpServletResponse response) {  
    StringBuffer sb = new StringBuffer();  
    sb.append("<form type=\"get\" action=\"\")  
    /* Se codifica la URL, para que sea  
       independiente de las cookies */  
    sb.append(response.encodeURL(config.getServletName()));  
    sb.append("\"><br>");  
    sb.append("    <p><button name=\"continuar\" type=\"submit\">");  
    sb.append("Continuar</button></p>\n");  
    sb.append("    <p><button name=\"");  
    sb.append(strBorraSesion);  
    sb.append("\n\" type=\"submit\">Borrar sesion</button></p>\n");  
    sb.append("</form>\n");  
    return sb;  
}
```

[Índice](#)[Introducción](#)[Servidor de
aplicaciones](#)[Mis primeros
servlets](#)[Aplicación Hola](#)[Avanzando](#)[Sesiones](#)[Filtros
Invocando otros
servlets](#)[Juntando todo](#)[Paquetes, Clases e
interfaces](#)

Transformar la salida del servlet



Aplicaciones:

- ▶ autenticación,
- ▶ cifrado,
- ▶ transformaciones XML,
- ▶ lo que se quiera....

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

```

public class EnvoltorioHTML implements Filter {
    private static final String cabecera="html/cabecera.html";
    private static final String pie="html/pie.html";
    private static final String strTitulo="titulo";
    private FilterConfig conf;
    public void init(FilterConfig filterConfig)
        throws ServletException {
        conf = filterConfig;
    }
    public void destroy() {
        conf = null;
    }
    public void doFilter(ServletRequest request,
                        ServletResponse response,
                        FilterChain filterChain)
        throws IOException, ServletException {
        String fichero =
            conf.getServletContext().getRealPath(cabecera);
        String cabecera = IncluyeFichero.incluye(fichero);
        String titulo = conf.getInitParameter(strTitulo);
        ServletOutputStream out = response.getOutputStream();
        cabecera = cabecera.replaceAll(" <title>",
                                       " <title>"+titulo);

        out.println(cabecera);
        filterChain.doFilter(request,response);
        fichero = conf.getServletContext().getRealPath(pie);
        out.println(IncluyeFichero.incluye(fichero));
    }
}

```


html/cabecera.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-15">
    <title></title>
  </head>
  <body>
```

html/pie.html

```
<hr>
<address>
  <a href="mailto:luis@ramonvazquez.net">
    Luis Fernando Llana Díaz
  </a>
</address>
</body>
</html>
```

Índice

Introducción

Servidor de aplicaciones 15

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

```

..... parametros globales de la aplicacion
<filter>
  <filter-name>envoltorioHola</filter-name>
  <filter-class>EnvoltorioHTML</filter-class>
  <init-param>
    <param-name>titulo</param-name>
    <param-value>Hola Mundo</param-value>
  </init-param>
</filter>
<filter>
  <filter-name>envoltorioContador</filter-name>
  <filter-class>EnvoltorioHTML</filter-class>
  <init-param>
    <param-name>titulo</param-name>
    <param-value>Ejemplo de contador</param-value>
  </init-param>
</filter>

<filter-mapping>
  <filter-name>envoltorioHola</filter-name>
  <url-pattern>/http/hola</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>envoltorioContador</filter-name>
  <url-pattern>/http/contador</url-pattern>
</filter-mapping>
definicion de servlets .....

```

```

public class Login implements Filter {
    .....
    public final void doFilter(final ServletRequest request,
                              final ServletResponse response,
                              final FilterChain filterChain)
        throws IOException, ServletException {
        HttpSession session =
            ((HttpServletRequest)request).getSession();
        String nombre=(String)session.getAttribute(Sesion.strNombre);
        if (nombre==null) {
            nombre = request.getParameter(strNombre);
            if (nombre==null) {
                pideNombre(((HttpServletRequest)request,
                           (HttpServletResponse)response);
            } else {
                session.setAttribute(Sesion.strNombre,nombre);
                filterChain.doFilter(request, response);
            }
        } else {
            filterChain.doFilter(request, response);
        }
    }
}

```

[Índice](#)[Introducción](#)[Servidor de
aplicaciones](#)[Mis primeros
servlets](#)[Aplicación Hola](#)[Avanzando](#)[Sesiones](#)[Filtros](#)[Invocando otros
servlets](#)[Juntando todo](#)[Paquetes, Clases e
interfaces](#)

```

public class Login implements Filter {
    private static final String strServlet="servlet";
    private static final String strNombre="nombre";
    private FilterConfig config;
    public void init(FilterConfig filterConfig)
        throws ServletException {
        config = filterConfig;
    }
    public void destroy() { }
    private void pideNombre(final HttpServletRequest request,
        final HttpServletResponse response)
        throws IOException, ServletException {
        ServletOutputStream out = response.getOutputStream();
        String servlet = config.getInitParameter(strServlet);
        response.setContentType("text/html");
        out.println("<form type=\"get\" action=\"\"+
            response.encodeURL(servlet)+\"\">\n");
        out.println("    <p>\n");
        out.println("        <label for=\""+strNombre+"\">Nombre:</label>");
        out.println("        <input id=\""+strNombre+"\" name=\""+strNombre+"\" type=\"text\"");
        out.println("        <button type=\"submit\">Login</button>\n");
        out.println("    </p>\n");
        out.println("</form>");
    }
    public final void doFilter(.....)
}

```

Invocando otros servlets

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

```
public class Fecha extends HttpServlet {
    public static final String strServlet="/http/fecha";
    public static final String strID="id";
    public final void doGet(final HttpServletRequest request,
                            final HttpServletResponse response)
        throws ServletException, IOException {
        ServletOutputStream out = response.getOutputStream();
        response.setContentType("text/html");
        Date hoy = new Date();
        DateFormat df = DateFormat.getDateInstance(DateFormat.FULL);

        out.println(" <div id=\""+request.getAttribute(strID)+"\">");
        out.println(df.format(hoy));
        out.println(" </div>");
    }
}
```

Índice

Introducción

Servidor de
aplicacionesMis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

Incluir el resultado del servlet `/http/fecha` en cualquier parte

```
.....  
RequestDispatcher rd =  
    request.getRequestDispatcher(Fecha.strServlet);  
request.setAttribute(Fecha.strID, "fecha");  
rd.include(request, response);  
.....
```

El servlet /http/sesion I

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

```
..... otros filtros
<filter>
  <filter-name>envoltorioSesion</filter-name>
  <filter-class>EnvoltorioHTML</filter-class>
  <init-param>
    <param-name>titulo</param-name>
    <param-value>Ejemplo de uso de sesiones</param-value>
  </init-param>
</filter>
<filter>
  <filter-name>loginSesion</filter-name>
  <filter-class>Login</filter-class>
  <init-param>
    <param-name>servlet</param-name>
    <param-value>sesion</param-value>
  </init-param>
</filter>
... otros filtros
```

El servlet /http/sesion II

```
..... otras disposiciones de filtros
<!-- el orden es importante -->
<filter-mapping>
  <filter-name>envoltorioSesion</filter-name>
  <url-pattern>/http/sesion</url-pattern>
</filter-mapping>

<filter-mapping>
  <filter-name>loginSesion</filter-name>
  <url-pattern>/http/sesion</url-pattern>
</filter-mapping>
... otras disposiciones de filtros
```

```
..... otros servlets
<servlet>
  <servlet-name>sesion</servlet-name>
  <servlet-class>Sesion</servlet-class>
</servlet>
<servlet>
  <servlet-name>fecha</servlet-name>
  <servlet-class>Fecha</servlet-class>
</servlet>
... otros servlets
```

Java para Internet

Luis Fernando
Llana Díaz
llana@sip.ucm.es

Índice

Introducción

Servidor de
aplicaciones

Mis primeros
servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e
interfaces

Paquetes I

► `javax.servlet`:

Interfaces: `Filter`, `FilterChain`, `FilterConfig`,
`RequestDispatcher`, `Servlet`,
`ServletConfig`, `ServletContext`,
`ServletContextAttributeListener`,
`ServletContextListener`,
`ServletRequest`, `ServletResponse`,
`SingleThreadModel`

Clases: `GenericServlet`,
`ServletContextAttributeEvent`,
`ServletContextEvent`,
`ServletInputStream`,
`ServletOutputStream`,
`ServletRequestWrapper`,
`ServletResponseWrapper`

► `javax.servlet.http`

Interfaces: `HttpServletRequest`,
`HttpServletResponse`, `HttpSession`,
`HttpSessionActivationListener`,
`HttpSessionAttributeListener`,
`HttpSessionBindingListener`,
`HttpSessionContext`,
`HttpSessionListener`

Clases: `Cookie`, `HttpServletRequest`,
`HttpServletRequestWrapper`,
`HttpServletResponseWrapper`,
`HttpSessionBindingEvent`,
`HttpSessionEvent`, `HttpUtils`

Índice

Introducción

Servidor de
aplicaciones

Mis primeros servlets

Aplicación Hola

Avanzando

Sesiones

Filtros

Invocando otros
servlets

Juntando todo

Paquetes, Clases e interfaces