

SetCharacterEncodingFilter

#FORMAT Java

```
/*
    Copyright 2004 The Apache Software Foundation

    Licensed under the Apache License, Version 2.0 (the "License");
    you may not use this file except in compliance with the License.
    You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

    Unless required by applicable law or agreed to in writing, software
    distributed under the License is distributed on an "AS IS" BASIS,
    WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
    See the License for the specific language governing permissions and
    limitations under the License.
*/

package org.apache.servlet;

import java.io.IOException;

import javax.servlet.Filter;
import javax.servlet.FilterChain;
import javax.servlet.FilterConfig;
import javax.servlet.ServletException;
import javax.servlet.ServletRequest;
import javax.servlet.ServletResponse;

/**
    <p>Example filter that sets the character encoding to be used in parsing the
    incoming request, either unconditionally or only if the client did not
    specify a character encoding. Configuration of this filter is based on
    the following initialization parameters:</p>
    <ul>
    <li><strong>encoding</strong> - The character encoding to be configured
    for this request, either conditionally or unconditionally based on
    the <code>ignore</code> initialization parameter. This parameter
    is required, so there is no default.</li>
    <li><strong>ignore</strong> - If set to "true", any character encoding
    specified by the client is ignored, and the value returned by the
    <code>selectEncoding()</code> method is set. If set to "false",
    <code>selectEncoding()</code> is called <strong>only</strong> if the
    client has not already specified an encoding. By default, this
    parameter is set to "true".</li>
    </ul>
    <p>Although this filter can be used unchanged, it is also easy to
    subclass it and make the <code>selectEncoding()</code> method more
    intelligent about what encoding to choose, based on characteristics of
    the incoming request (such as the values of the <code>Accept-Language</code>
    and <code>User-Agent</code> headers, or a value stashed in the current
    user's session.</p>
    @author Craig McClanahan
    @version $Revision: 267129 $ $Date: 2004-03-18 11:40:35 -0500 (Thu, 18 Mar 2004) $
*/

public class SetCharacterEncodingFilter implements Filter {

    // ---- Instance Variables

    /**
    The default character encoding to set for requests that pass through
    this filter.
    */
    protected String encoding = null;

    /**
    The filter configuration object we are associated with. If this value
    is null, this filter instance is not currently configured.
    */
    protected FilterConfig filterConfig = null;

    /**
    Should a character encoding specified by the client be ignored?
    */
    protected boolean ignore = true;

    // ---- Public Methods
}
```

- Take this filter out of service. */

```

public void destroy() {
    this.encoding = null;
    this.filterConfig = null;
}

```
- Select and set (if specified) the character encoding to be used to interpret request parameters for this request. *
- @param request The servlet request we are processing
- @param result The servlet response we are creating
- @param chain The filter chain we are processing *
- @exception IOException if an input/output error occurs
- @exception [ServletException](#) if a servlet error occurs */

```

public void doFilter(ServletRequest request, ServletResponse response,
FilterChain chain)
    throws IOException, ServletException {
    // Conditionally select and set the character encoding to be used
    if (ignore | (request.getCharacterEncoding() == null)) {
        String encoding = selectEncoding(request);
        if (encoding != null)
            request.setCharacterEncoding(encoding);
    }
    // Pass control on to the next filter
    chain.doFilter(request, response);
}

```
- Place this filter into service. *
- @param filterConfig The filter configuration object */

```

public void init(FilterConfig filterConfig) throws ServletException {
    this.filterConfig = filterConfig;
    this.encoding = filterConfig.getInitParameter("encoding");
    String value = filterConfig.getInitParameter("ignore");
    if (value == null)
        this.ignore = true;
    else if (value.equalsIgnoreCase("true"))
        this.ignore = true;
    else if (value.equalsIgnoreCase("yes"))
        this.ignore = true;
    else
        this.ignore = false;
}

```
- Select an appropriate character encoding to be used, based on the characteristics of the current request and/or filter initialization parameters. If no character encoding should be set, return `null`.
-
- The default implementation unconditionally returns the value configured by the `encoding` initialization parameter for this filter. *
- @param request The servlet request we are processing */

```

protected String selectEncoding(ServletRequest request) {
    return (this.encoding);
}

```

}