

Tapestry5Utf8Encoding

How to use UTF-8 encoding in T5

T5.0.5 now supports UTF-8 encoding by default and doesn't need the `decoratePageResponseRenderer()` below. The `Utf8Filter` changes are still needed to decode form submissions.

Follow the instructions below to use "foreign" characters, i.e. ones outside ISO-8859-1.

In your [AppModule.java](#) file, add the following methods

```
public RequestFilter buildUtf8Filter(
    @InjectService("RequestGlobals") final RequestGlobals requestGlobals)
{
    return new RequestFilter()
    {
        public boolean service(Request request, Response response, RequestHandler handler)
            throws IOException
        {
            requestGlobals.getHttpServletRequest().setCharacterEncoding("UTF-8");
            return handler.service(request, response);
        }
    };
}

// this method is not needed in Tapestry 5.0.5 or later
public static PageResponseRenderer decoratePageResponseRenderer(
    @InjectService("PageMarkupRenderer") final PageMarkupRenderer markupRenderer,
    @InjectService("MarkupWriterFactory") final MarkupWriterFactory markupWriterFactory,
    final Object delegate)
{
    return new PageResponseRenderer()
    {
        public void renderPageResponse(Page page, Response response) throws IOException
        {
            MarkupWriter writer = markupWriterFactory.newMarkupWriter();
            markupRenderer.renderPageMarkup(page, writer);
            PrintWriter pw = response.getPrintWriter("text/html; charset=UTF-8");
            writer.toMarkup(pw);
            pw.flush();
        }
    };
}
```

Now update your `contributeRequestHandler()` method

(Note that in earlier versions this was called `contributeRequestFilters`.)

```
public void contributeRequestHandler(OrderedConfiguration<RequestFilter> configuration,
    @InjectService("TimingFilter") final RequestFilter timingFilter,
    @InjectService("Utf8Filter") final RequestFilter utf8Filter)
{
    configuration.add("Utf8Filter", utf8Filter); // handle UTF-8
    configuration.add("Timing", timingFilter);
}
```

Utf8Filter with tapestry-upload

If you use file upload form with fields, in order to decode the field values properly `Utf8Filter` needs to implement `HttpServletRequestFilter` instead of `RequestFilter` and add the filter before `MultipartFilter`.

```
public HttpServletRequestFilter buildUtf8Filter()
{
    return new HttpServletRequestFilter()
    {
        public boolean service(HttpServletRequest request, HttpServletResponse response,
            HttpServletRequestHandler handler) throws IOException
        {
            request.setCharacterEncoding("UTF-8");
            return handler.service(request, response);
        }
    };
}

public void contributeHttpServletRequestHandler(
    OrderedConfiguration<HttpServletRequestFilter> configuration,
    @InjectService("Utf8Filter")
    HttpServletRequestFilter utf8Filter)
{
    configuration.add("Utf8Filter", utf8Filter, "before:MultipartFilter");
}
```