FREE eBook

LEARNING wicket

Free unaffiliated eBook created from
Stack Overflow contributors.

#wicket

Table of Contents

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: wicket

It is an unofficial and free wicket ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official wicket.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with wicket

Remarks

Wicket is a component-based framework, which puts it in stark contrast to some of the earlier solutions to the sometimes monotonous task of web programming. Like other frameworks, Wicket builds on top of Sun's servlet API.

However, unlike frameworks like Struts or Spring MVC, the developer using Wicket is mostly removed from the request/response nature that is inherent with the web and Servlets. Instead of building controllers that must service many users and threads simultaneously, taking in requests, returning responses, and never storing any state, the Wicket developer thinks in terms of stateful components. Instead of creating a controller or action class, he or she creates a page, places components on it, and defines how each component reacts to user input.

Wicket uses plain XHTML for templating. Each component is bound to a named element in the XHTML and becomes responsible for rendering that element in the final output. The page is simply the top-level containing component and is paired with exactly one XHTML template.

Each component is backed by its own model, which represents the state of the component. The framework does not have knowledge of how components interact with their models, which are treated as opaque objects automatically serialized and persisted between requests. More complex models, however, may be made detachable and provide hooks to arrange their own storage and restoration at the beginning and end of each request cycle.

In Wicket, all server side state is automatically managed. You should never directly use an HttpSession object or similar wrapper to store state. Instead, state is associated with components. Each server-side page component holds a nested hierarchy of stateful components, where each component’s model is, in the end, a POJO (Plain Old Java Object).

Versions

Version	Release Date
8.0.0-M4	2017-02-07
7.6.0	2016-12-29
6.26.0	2016-12-29

Examples

Installation or Setup

My first Wicket setup, app showing Hello World on home screen:

```
import org.apache.wicket.protocol.http.WebApplication;
public class HelloWorldApplication extends WebApplication {

    public HelloWorldApplication() {
    }

    @Override
    public Class getHomePage() {
        return HelloWorld.class;
    }
}
```

HelloWorld.java

```
import org.apache.wicket.markup.html.WebPage;
import org.apache.wicket.markup.html.basic.Label;
public class HelloWorld extends WebPage {
    public HelloWorld() {
        add(new Label("message", "Hello World!"));
    }
}
```

HelloWorld.html

```
<html>
   <body>
       <span wicket:id="message">Message goes here</span>
    </body>
</html>
```

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app
  PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
  "http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>
    <display-name>My first Wicket App</display-name>
    <filter>
        <filter-name>HelloWorldApplication</filter-name>
        <filter-class>org.apache.wicket.protocol.http.WicketFilter</filter-class>
        <init-param>
          <param-name>applicationClassName</param-name>
          <param-value>org.apache.wicket.examples.helloworld.HelloWorldApplication</param-
value>
        </init-param>
    </filter>
    <filter-mapping>
        <filter-name>HelloWorldApplication</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>
</web-app>
```

pom.xml

```
    <dependency>
        <groupId>org.apache.wicket</groupId>
        <artifactId>wicket-core</artifactId>
        <version>${wicket.version}</version>
    </dependency>
```

Read Getting started with wicket online: https://riptutorial.com/wicket/topic/3834/getting-started-with-wicket

Credits

S. No	Chapters	Contributors
1	Getting started with wicket	Apostolos, Community, crea1, mico