

Web Service Project Sample Template

Table of contents

- 1 Introduction.....2
- 2 Using the Web Service Template.....2
 - 2.1 Copy and Rename the Template Folder..... 2
 - 2.2 Edit the build.properties File.....2
 - 2.3 Build and Deploy the Template Web App..... 3
 - 2.4 To Run the Template Web Service..... 3

1. Introduction

Use the "blank" web service sample (located at `<BeehiveRoot>/samples/wsm-blank`) as a template for building your own web service applications.

Note:

`<BeehiveRoot>` refers to the top-level directory of your Beehive installation. A typical value for `<BeehiveRoot>` would be `C:/apache/apache-beehive-1.0`.

The template contains everything needed in a basic web service application, except for the runtime JAR files.

2. Using the Web Service Template

The following instructions assume that you have completed all of required and optional steps in the Beehive set up procedure at [Installation and Setup](#) (`../setup.html`) .

To use the template, follow these steps:

2.1. Copy and Rename the Template Folder

Copy the folder `<BeehiveRoot>/samples/wsm-blank` to your development directory.

Rename `wsm-blank` to something more appropriate to your application. Below we assume that the path to the renamed `wsm-blank` is **`<Project-Folder>`**.

2.2. Edit the `build.properties` File

In this section you will edit the `build.properties` file--the file that sets the build-related properties for your web service.

Open the file `<Project-Folder>/WEB-INF/src/build.properties` in a text editor.

Edit the `beehive.home` property so it points to the top-level folder of your beehive installation.

Edit the `service.name` property to something appropriate for your application.

For example, if your beehive installation resides at

Web Service Project Sample Template

, then your `build.properties` file would appear as follows.

```
beehive.home=C:/apache/apache-beehive-1.0  
service.name=<SomeContext>
```

Note:

Windows users must use forwardslashes (/) not backslashes (\) in the `build.properties` file.

Note:

The value of `<SomeContext>` will determine (1) the name of the WAR file produced when the web service is compiled and the (2) part of the URL used by Tomcat to access the web service.
The WAR file will be named `<SomeContext>WS.war`
The URL will be `http://localhost:8080/<SomeContext>WS/`

2.3. Build and Deploy the Template Web App

To build the template web service, run the following Ant command. (The result of the build will be a WAR file: `<SomeContext>WS.war`. The WAR file will be saved to the parent directory of `<Project-Folder>`. Assume that this parent directory is named `C:/webDevelopmentDirectory`.)

```
ant  
-f <Project-Folder>WEB-INF\src\build.xml  
clean  
build  
war
```

Copy and Paste version:

```
ant -f <Project-Folder>\WEB-INF\src\build.xml clean build war
```

To deploy the web application copy the application's WAR file to Tomcat's `webapps` dir.

```
copy C:\webDevelopmentDirectory\<SomeContext>WS.war  
%CATALINA_HOME%\webapps
```

2.4. To Run the Template Web Service

You can now try out the web service by pointing your browser at the following link.

- <http://localhost:8080/<SomeContext>WS/>

Java, J2EE, and JCP are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

© 2004, Apache Software Foundation