Übung Java Reflection URLClassLoader

Ausgangslage

Mit dieser Übung soll eine Klasse, definiert über eine URL (Uniform Resource Locator) geladen und instanziiert werden.Hierzu kann die Java Klasse java.net.URLClassLoader verwendet werden. Die Klasse wird hierzu über die Methode loadClass(...) geladen, wobei diese über den Klassennamen referenziert wird.URLClassLoader erlaubt das Laden von Klassen via URL Directory (z.B. http://localhost/transfer/) oder JAR-Datei (z.B. http://localhost/transfer/my.jar). Es sind mehrere URL Pfade (Array) definierbar.Für unsere Übung wollen wir eine Klasse in ein Web Server Verzeichnis kopieren und alsdann via URLClassLoader laden.

Web Server

Zuerst benötigen wir einen lokal ausgeführten Web Server oder den schreibenden Zugang zu einem Web Server im Internet. Der Simtech Web Server programmiert in Go (Golang) bietet uns die ideale Möglichkeit und ist auf Apple, Unix und Windows verfügbar:Version Windows 64 Bit webserver.windows.x64.exeVersion Windows 32 Bit webserver.windows.x86.exeVersion Linux 64 Bit webserver.linux.x64Version Linux 32 Bit webserver.linux.x86Version Darwin 64 Bit webserver.darwin.x64Version Darwin 32 Bit webserver.darwin.x86Laden Sie die richtige Version auf Ihren PC und starten Sie das Programm mit der Hilfefunktion wie folgt:Starten Sie den Server im aktuellen Verzeichnis z.B. wie folgt:Der lokale Web Server sollte nun über die URL http://localhost:80 erreichbar sein, er meldet "404 page not found".Definieren Sie nun eine index.html Datei z.B. gemäss dem folgenden Script und speichern Sie solche im Web Server Verzeichnis ab:<!DOCTYPE html>
<html>
<head>
 <title>Web Server Root</title>
</head>
<body>
 Web Server is up and running
</body>
</html>Der lokale Web Server meldet nun "Web Server is up and running".

Das JAR (Java Archive)

Für diese Übung benötigen wie eine Java Klasse exportiert in einem JAR File (JavaArchive). Wir verwenden hierzu die folgende Java Klasse in einem eigenen Eclipse Projekt:package ch.std.reflection;
public class HelloClass {
 private String name;
 public HelloClass() {
 this("James");
 }
 public HelloClass(String name) {
 this.name = name;
 }
 @Override
 public String toString() {
 return "Hello " + this.name;
 }
}
Via Eclipse Export Funktion erstellen wir das Java Archive JAR mit der Klasse ch.std.reflection.HelloClassEin Beispiel als Alternative finden Sie hier hello.jar

Vorgehen

Lösen Sie bitte die Aufgabe wie folgt:Installieren Sie den lokalen Web Server gemäss der obigen Beschreibung.Installieren Sie die Datei hello.jar in das Root Verzeichnis des lokalen Web ServersStarten Sie den lokalen Web Server und verifizieren Sie dass die hello.jar Datei z.B via Browser URL http://localhost/hello.jar geladen werden kann.Erzeugen Sie die Datei URLClassLoaderTest.java mit einem geeigneten Editor.Mit der Methode main(...) soll der 1. Command Line Parameter die URL (z.B: http://localhost/hello.jar) definieren, die dem absoluten Suchpfad entspricht. Der 2. Command Line Parameter definiert den Namen der Klasse (z.B. ch.std.reflection.HelloClass).Instanziieren Sie die Klasse java.net.URLClassLoader und laden Sie die gesuchte Klasse.Bilden Sie eine Instanz der geladenen Klasse und geben Sie deren Inhalt über die Methode toString() aus.Testen Sie das Programm und verifizieren Sie das Resultat.Das Programm sollte den folgenden Output ausgeben:

Lösung

Eine mögliche Lösung finden Sie hier

Kontakt

Simtech AG
Finkenweg 23
3110 Münsingen
Schweiz

Impressum

Das Copyright für sämtliche Inhalte dieser Website liegt bei Simtech AG, Schweiz.
Beachten Sie auch unsere Hinweise zum Urheberrecht, Datenschutz und Haftungsauschluss.
Jeder Hinweis auf Fehler nehmen wir gerne entgegen.

Copyright

2024 Simtech AG, All rights reserved, Powered by stack.ch written in Golang by Daniel Schmutz

https://www.simtech-ag.ch/online-ausbildung-jpf2-kurs-java-advanced-übungen-reflection