

Einrichtung der Eclipse EE und des Tomcat-Servers

Patrick Weber

Januar 2024

Inhaltsverzeichnis

1	Installation von Eclipse EE	1
2	Installation des Tomcat-Servers in Eclipse	2
3	Installation prüfen.....	7

Installation von Eclipse EE

Zunächst muss „Eclipse EE“ (Enterprise Edition) installiert werden. Hierfür kann der Installer genutzt werden, der unter folgendem Link heruntergeladen werden kann: <https://www.eclipse.org/downloads/>

Nach dem Start des Eclipse-Installers wählen wir die Version **Eclipse IDE for Enterprise Java and Web Developers** (siehe Abbildung 1.1) und folgen den Anweisungen des Installationsdialogs.

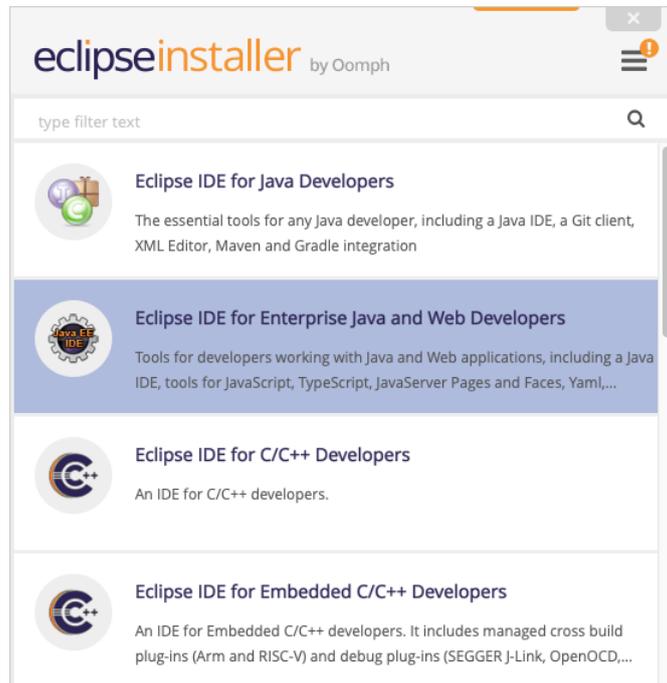


Abbildung 1.1: Der Eclipse-Installer

Installation des Tomcat-Servers in Eclipse

Nachdem Eclipse EE installiert ist, muss der Tomcat-Server heruntergeladen und konfiguriert werden. Hierfür öffnen wir die **Preferences...** in Eclipse. Diese finden wir unter :

- **Window** → **Preferences...** (unter Windows oder Linux)
- **Eclipse** → **Settings...** (unter MacOS)

Dort navigieren wir zu **Server** → **Runtime Environments** und fügen über den Button **Add...** eine neue Server-Laufzeitumgebung hinzu (siehe Abbildung 2.1).

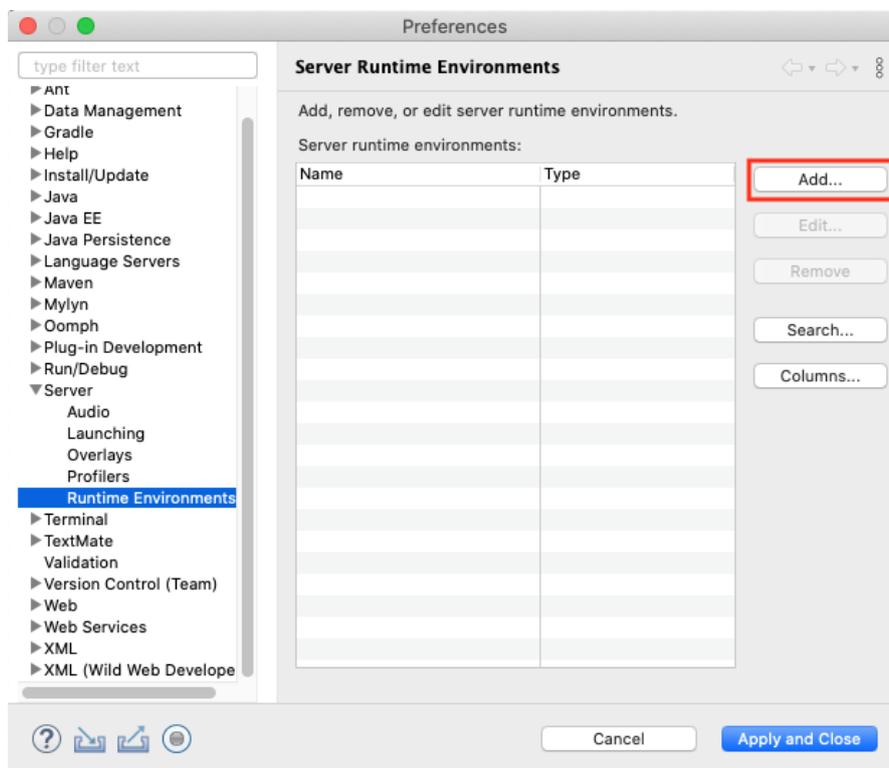


Abbildung 2.1: Eine neue Server-Laufzeitumgebung hinzufügen

In der Kategorie **Apache** wählen wir nun einen Tomcat-Server. Unterstützt werden Versionen der Servlet API **Jakarta Servlet ≥ 5.0** .

Achtung: An dieser Stelle wählen wir deshalb unbedingt eine Version ≥ 10.0 und die Checkbox **Create a new local server** aus (siehe Abbildung 2.2)

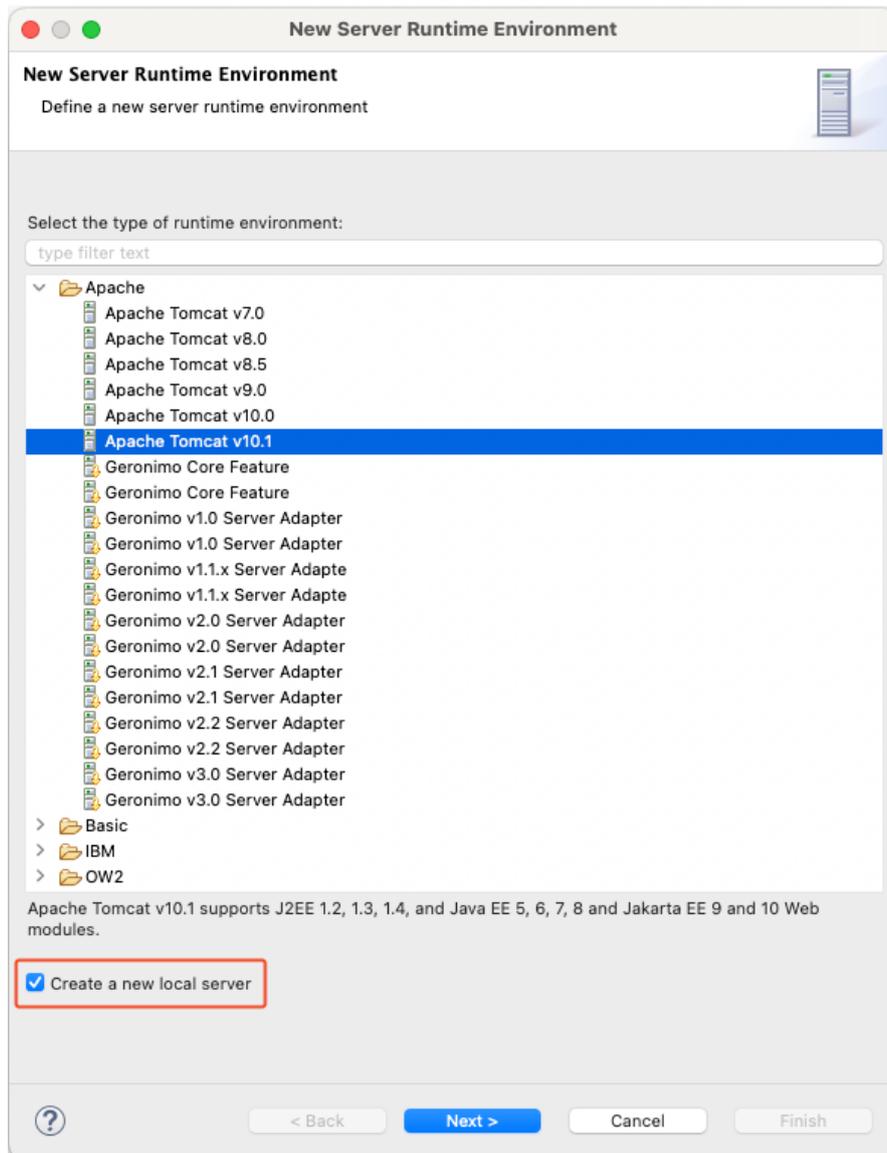


Abbildung 2.2: Apache Tomcat auswählen

Über den Button **Download and Install...** können wir den Tomcat-Server über Eclipse herunterladen (siehe Abbildung 2.3).

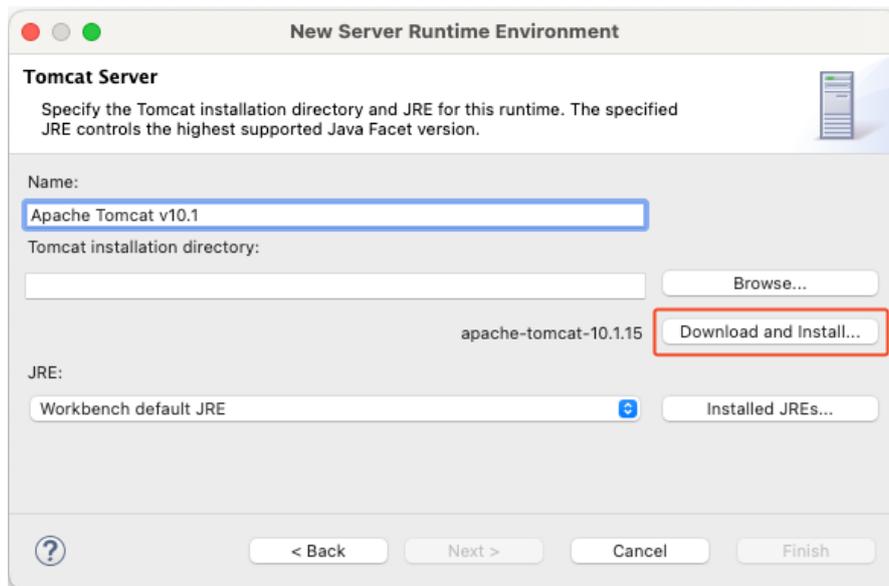


Abbildung 2.3: Apache Tomcat herunterladen

Wir akzeptieren das Lizenzabkommen und bestätigen über **Finish** (siehe Abbildung 2.4).

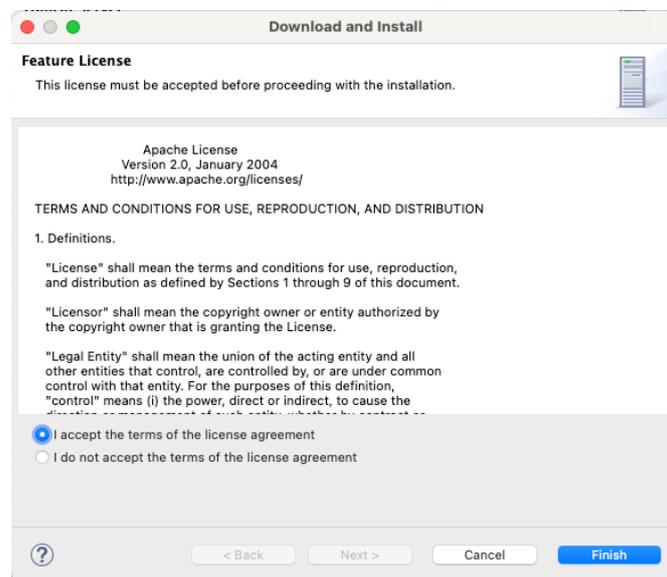


Abbildung 2.4: Lizenzabkommen akzeptieren

Im Anschluss öffnet sich ein Dialog, in dem wir ein Installationsverzeichnis für den Tomcat-Server auswählen (zum Beispiel: `~/tomcat`).

Achtung: Zunächst wird eine Fehlermeldung angezeigt (siehe Abbildung 2.5).

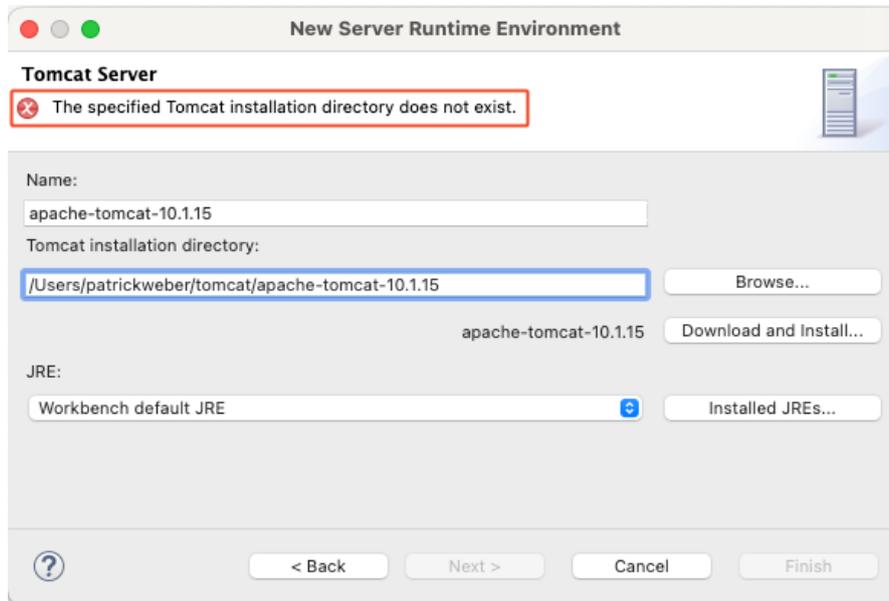


Abbildung 2.5: Fehlermeldung während der Tomcat im Hintergrund heruntergeladen wird

Der Download erfolgt jedoch im Hintergrund, wobei der Fortschritt in der Statusleiste (unten rechts) angezeigt wird (siehe Abbildung 2.6)



Abbildung 2.6: Fortschrittsanzeige des Downloads in der Statusleiste

Sobald dieser abgeschlossen wurde, verschwindet die Fehlermeldung und der Dialog kann über **Finish** beendet werden.

Danach wird der installierte Tomcat-Server angezeigt. Diesen Dialog bitte über **Apply and Close** schließen (siehe Abbildung 2.7).

3

Installation prüfen

Um die Installation zu prüfen, kann ein Beispiel-Servlet deployed werden. Hierzu legen wir zunächst ein neues **Dynamic Web Project** mit dem Namen **FOPT** über **File** → **New** → **Dynamic Web Project** an (siehe Abbildung 3.1).

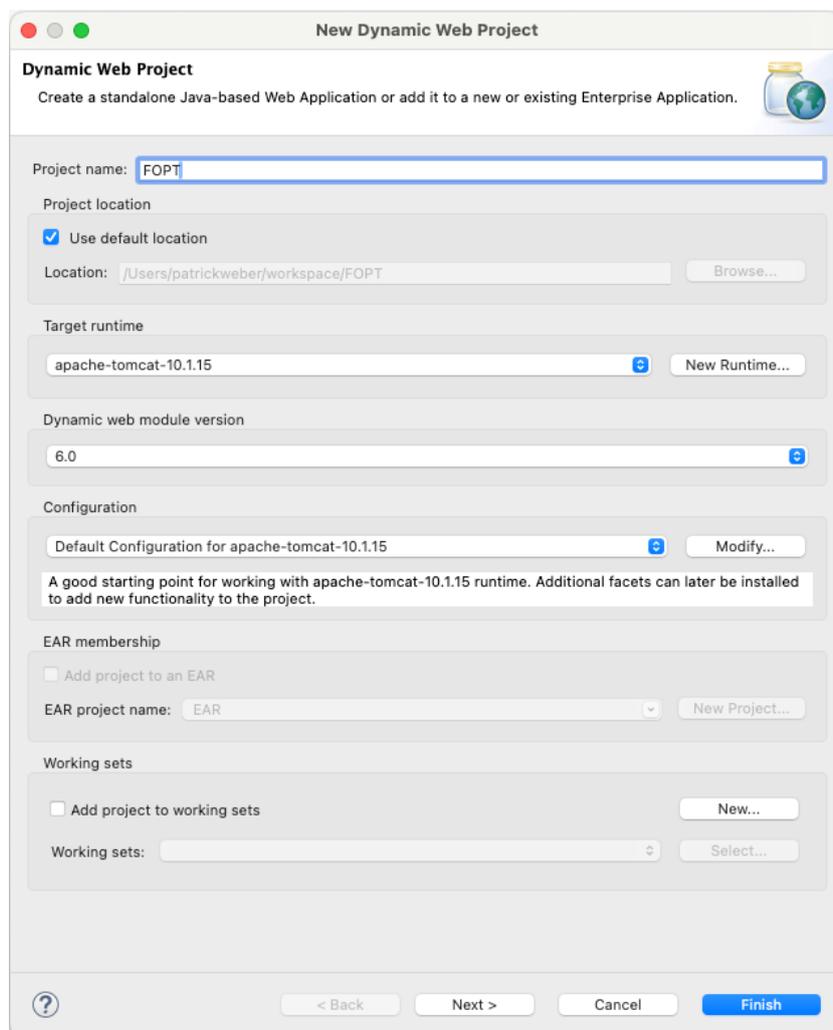


Abbildung 3.1: Neues **Dynamic Web Project** anlegen

Danach das Package `servlets.hello` erzeugen und die Klasse `HelloServlet` erstellen. Diese könnte zum Beispiel wie in Abbildung 3.2 aussehen.

```
package servlets.hello;

import java.io.IOException;
import java.io.PrintWriter;

import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;

@SuppressWarnings("serial")
@WebServlet("/HalloWelt")
public class HelloServlet extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws
        IOException {
        resp.setContentType("text/html");
        PrintWriter out = resp.getWriter();
        out.println("<html>");
        out.println("<head>");
        out.println("<title> Hallo Welt</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1> Hallo Welt</h1>");
        out.println("Herzlich willkommen!");
        out.println("</body></html>");
    }
}
```

Abbildung 3.2: Die Klasse `HelloServlet`

Um das Servlet zu deployen, klicken wir mit der rechten Maustaste auf `HelloServlet.java`. Dann `Run As` → `Run on Server` (siehe Abbildung 3.3).

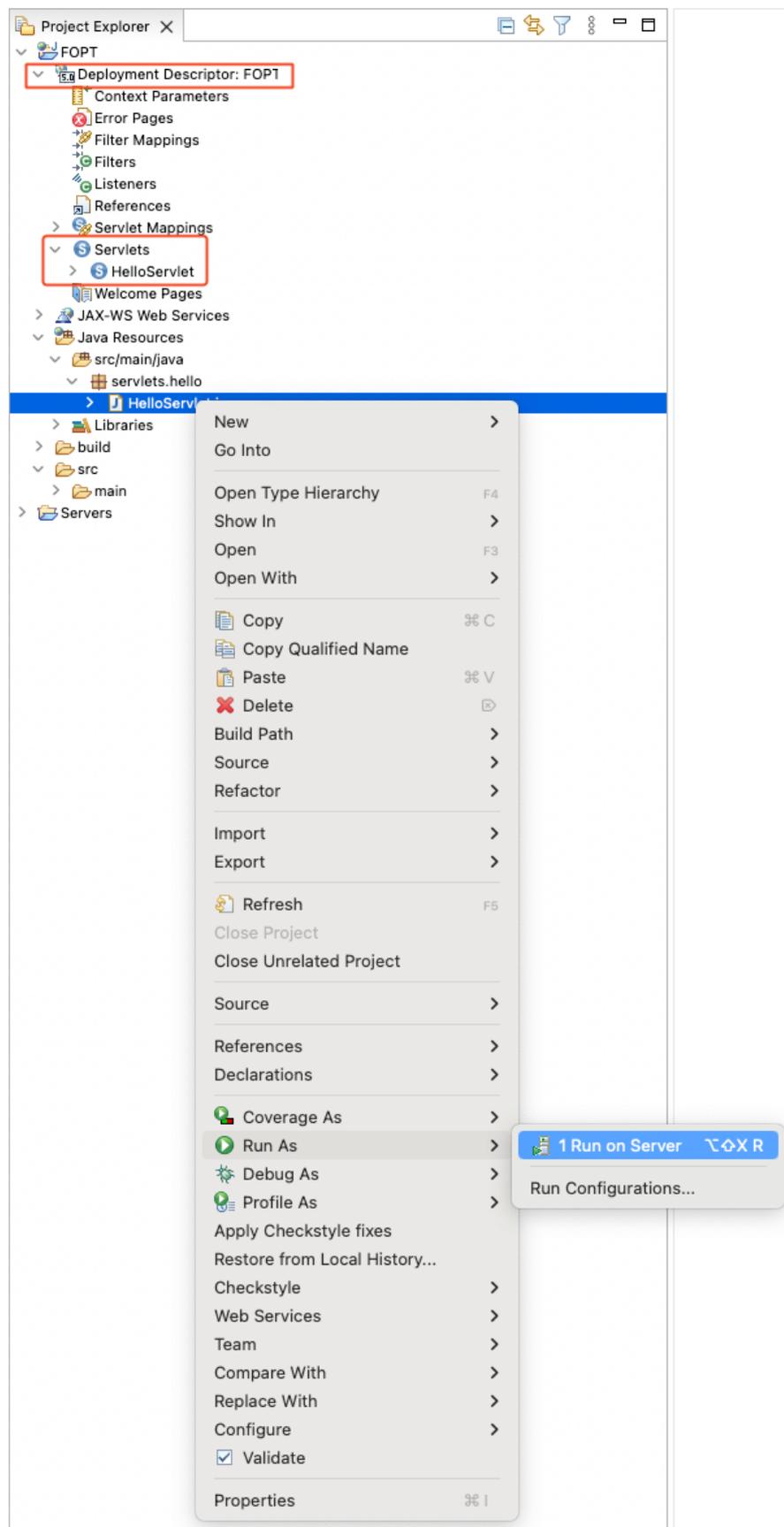


Abbildung 3.3: Servlet auf dem Tomcat deployen

Alternativ kann das Servlet auch unter **Deployment Descriptor: FOPT** → **Servlets** → **HelloServlet** gefunden werden. Sollte die hier gezeigte Ansicht nicht dargestellt werden, lässt sich diese unter **Window** → **Perspective** → **Open Perspective** → **Other...** → **Java EE** einstellen.

Hier nun den zuvor erstellen Tomcat-Server auswählen und über den Button **Finish** bestätigen (siehe Abbildung 3.4).

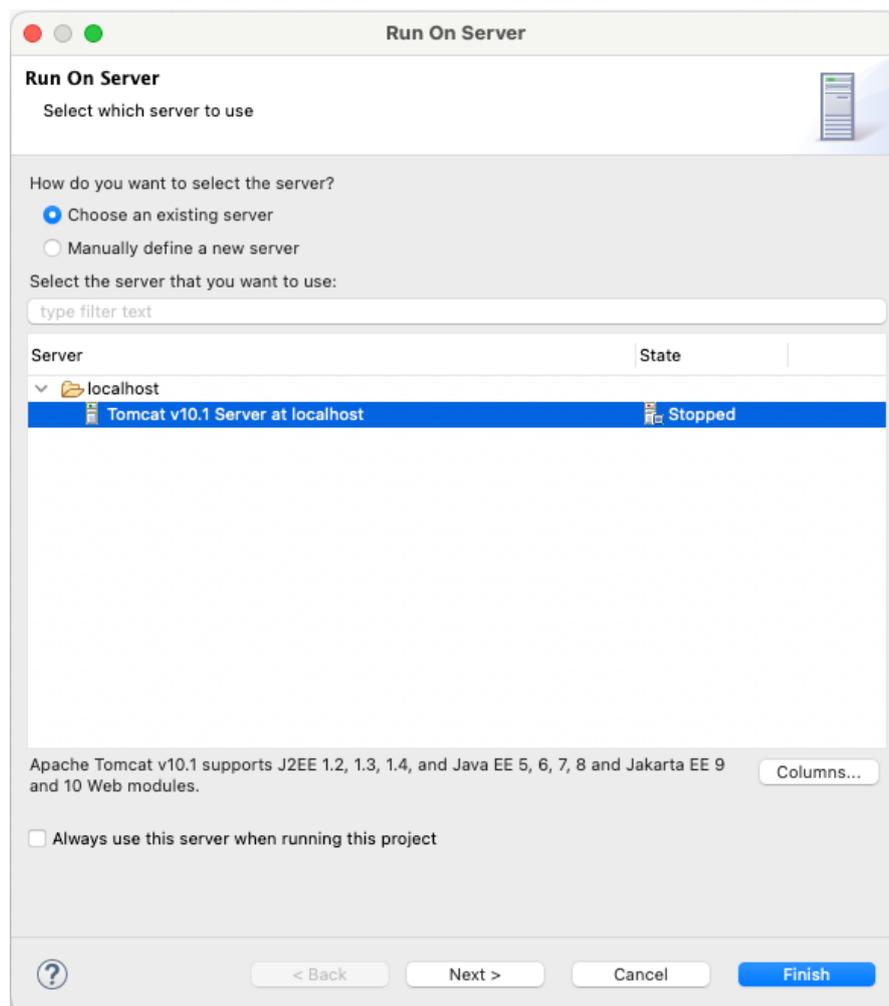


Abbildung 3.4: Servlet auf dem Tomcat deployen

Das hat zur Folge, dass der Tomcat-Server gestartet und das Servlet deployed wird. Nach einem kurzen Augenblick wird der Inhalt des Servlets in Eclipse angezeigt (siehe Abbildung 3.5).

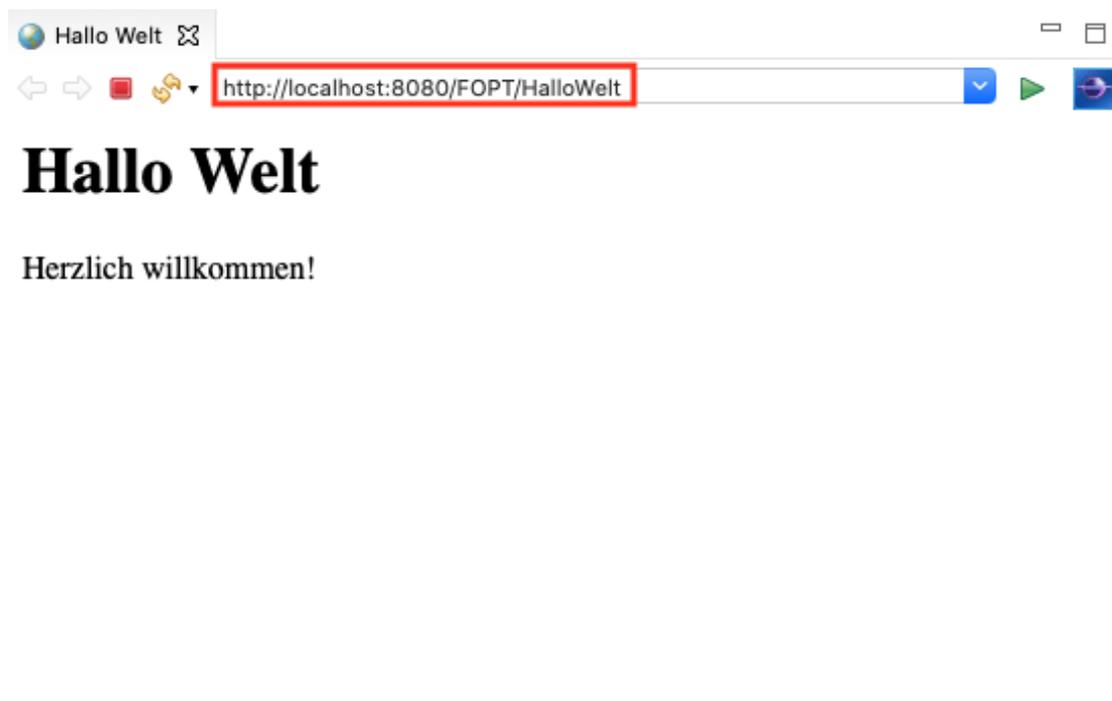


Abbildung 3.5: Inhalt des HelloWorld-Servlets

Das Servlet kann auch über den Browser aufgerufen werden. Die URL setzt sich wie folgt zusammen: **http://localhost:8080/<Name des Projekts><Wert der @WebServlet-Annotation>**

Die Log-Ausgaben des Tomcat-Servers werden unter **Console** angezeigt (siehe Abbildung 3.6).

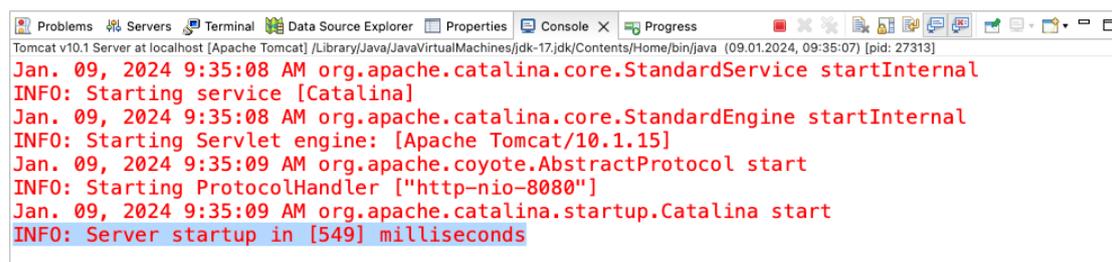


Abbildung 3.6: Log-Ausgaben des Tomcat