

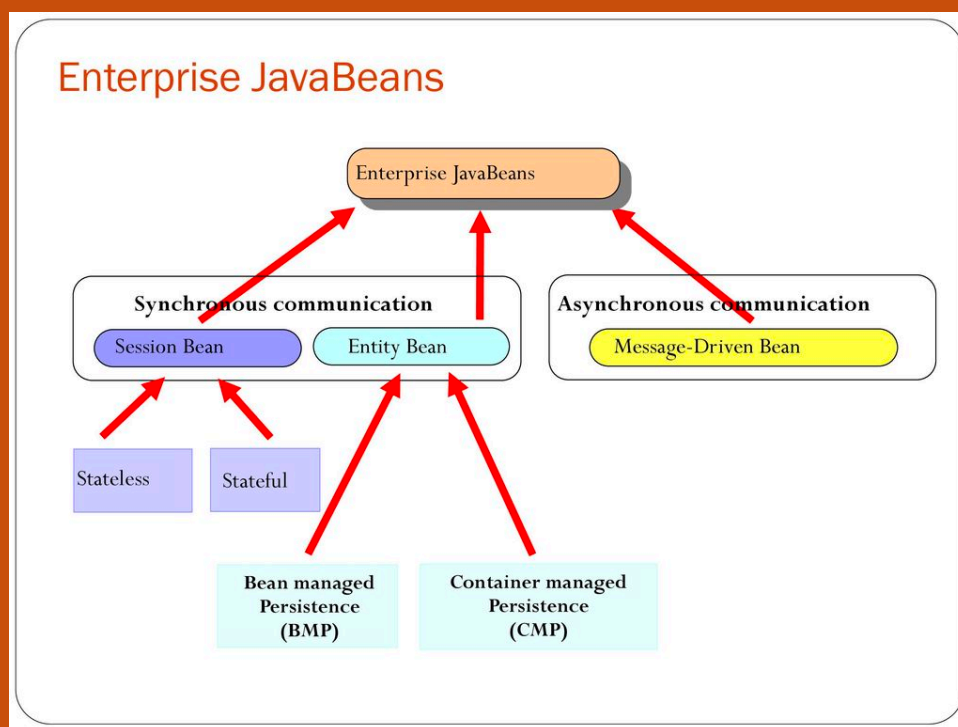


---

# DISTRIBUTED APPLICATIONS

*Wintersemester 2022/2023*

---



---

## PROGRAMMING

*Exercise 9 - BMI Session Bean*

---

## GITHUB-REPOSITORY

Please see the full Code in my Github-Repo: [https://github.com/DziadekM/distributed-applications/tree/main/Exercise 9/ejbstateless](https://github.com/DziadekM/distributed-applications/tree/main/Exercise%209/ejbstateless)

## SETUP

- Server: Wildfly
- IDE: VS-Code
- Tools: Apache Maven

## IMPORTANT FILES

### BMIServlet.java

```
import java.io.*;
import java.lang.Math;

import jakarta.ejb.EJB;
import jakarta.servlet.*;
import jakarta.servlet.http.*;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;

public class BMIServlet extends HttpServlet {

    @EJB
    // private BMISessionBean bmiCalculator;
    private BMICalculatorLocal calc = null;

    protected void doPost(HttpServletRequest req, HttpServletResponse res)
        throws ServletException, IOException {

        // Stream
        PrintWriter out = res.getWriter();

        // Parameter vom Formular holen für die BMI-Berechnung
        String firstname = req.getParameter("firstname");
        String height = req.getParameter("height");
        String weight = req.getParameter("weight");

        try {
            double bmi = calc.calculateBMI(
                Double.parseDouble(weight),
                Double.parseDouble(height));
            double result = (int) Math.round(bmi);

            // bind Attribute to the RequestObject
            req.setAttribute("firstname", firstname);
            req.setAttribute("bmi", bmi);
            res.setHeader("bmi", String.valueOf(result));

            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Ihr BMI Ergebnis</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Ihr BMI Ergebnis</h1>");
            out.println("<p>Hallo " + firstname + " ihr BMI ist " + result + "</p>");
            out.println("</body>");
            out.println("</html>");

            /*
             * Ein RequestDispatcher-Objekt kann verwendet werden, um eine Anfrage an die
             * Ressource weiterzuleiten oder um die Ressource in eine Antwort
             * einzuschließen. Die Ressource kann dynamisch oder statisch sein.
             */

        } catch (Exception e) {
            req.getRequestDispatcher("BMIErrrorPage.html").forward(req, res);
        }
    }
}
```

## BMISessionBean.java

```
import jakarta.ejb.Stateless;

@Stateless
public class BMISessionBean implements BMICalculatorLocal {

    @Override
    public Double calculateBMI(Double weight, Double height) {
        System.out.println("Hello, here is the BMISessionBean!");

        return weight / (height * height);
    }
}
```

## web.xml

```
<!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>BMI - EJB</display-name>
  <welcome-file-list>
    <welcome-file>BMI.html</welcome-file>
  </welcome-file-list>

  <servlet>
    <servlet-name>BMIServlet</servlet-name>
    <servlet-class>BMIServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>BMIServlet</servlet-name>
    <url-pattern>/calculateServlet</url-pattern>
  </servlet-mapping>
</web-app>
```

## BMI.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>BMI-Rechner (Exercise 8)</title>
    <link rel="stylesheet" href="style.css">
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
      crossorigin="anonymous">
  </head>
  <body>
    <div class="form-body">
      <div class="row">
        <div class="form-holder">
          <div class="form-content">
            <div class="form-items">
              <h3>Calculate your BMI</h3>
              <p>Fill in the data below.</p>
              <form name="bmiForm" action="calculateServlet"
                method="POST">

                <div class="col-md-12">
                  <label>Your Firstname</label>
                  <input class="form-control" type="text"
                    name="firstname" placeholder="Firstname"
                    required>
                </div>
                <div class="col-md-12">
                  <label>Your Height (m)</label>
                  <input class="form-control"
                    type="number"
                    name="height"
                    placeholder="1.60" step="0.01" min="1"
                    max="2.8"
                    required>
                </div>
                <div class="col-md-12">
                  <label>Your Weight (kg)</label>
                  <input class="form-control"
                    type="number"
                    name="weight" placeholder="65.2"
                    step="0.01" min="1" max="300"
                    required>
                </div>
                <div class="form-button mt-4 spacingbutton">
                  <button id="submit" type="submit"
                    class="btn
                    btn-primary">Submit</button>
                  <button id="reset" type="reset"
                    class="btn
                    btn-primary">Reset</button>
                </div>
              </form>
            </div>
          </div>
        </div>
      </div>
    </div>
  </body>
</html>
```

## pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.localejb</groupId>
  <artifactId>localejb</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>war</packaging>

  <name>localejb Maven Webapp</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

  <dependencies>

    <dependency>
      <groupId>org.wildfly</groupId>
      <artifactId>wildfly-webservices-jakarta</artifactId>
      <version>27.0.0.Beta1</version>
      <type>pom</type>
    </dependency>

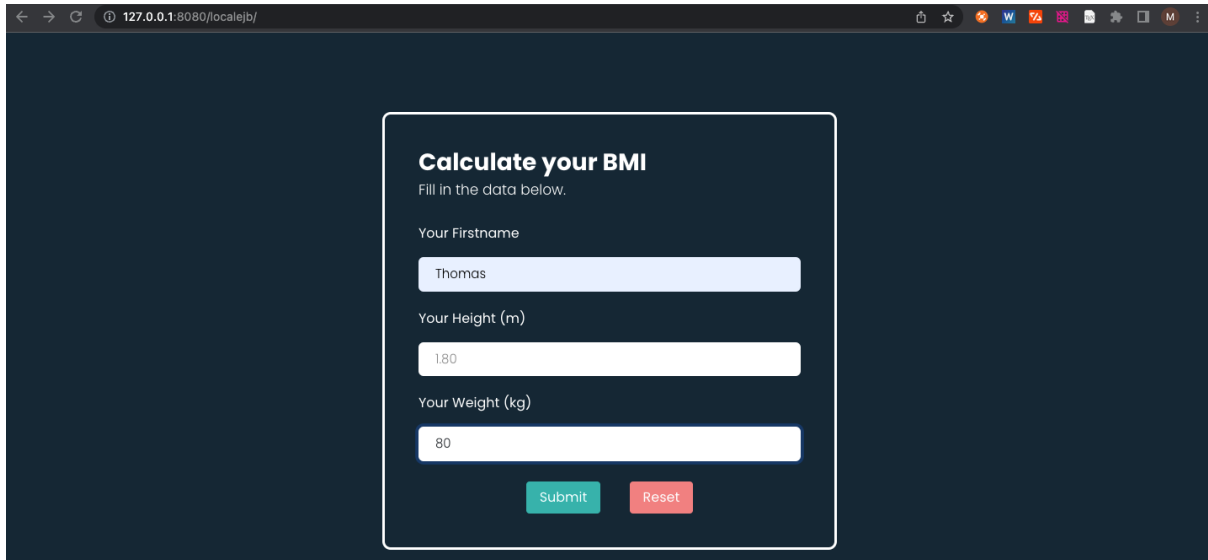
    <dependency>
      <groupId>jakarta.servlet</groupId>
      <artifactId>jakarta.servlet-api</artifactId>
      <version>6.0.0</version>
    </dependency>

    <dependency>
      <groupId>jakarta.ejb</groupId>
      <artifactId>jakarta.ejb-api</artifactId>
      <version>4.0.1</version>
    </dependency>

    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

  <build>
    <finalName>localejb</finalName>
    <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->
      <plugins>
        <plugin>
          <artifactId>maven-clean-plugin</artifactId>
          <version>3.1.0</version>
        </plugin>
        <!-- see http://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin_bindings_for_war_packaging -->
        <plugin>
          <artifactId>maven-resources-plugin</artifactId>
          <version>3.0.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-compiler-plugin</artifactId>
          <version>3.8.0</version>
        </plugin>
        <plugin>
          <artifactId>maven-surefire-plugin</artifactId>
          <version>2.22.1</version>
        </plugin>
        <plugin>
          <artifactId>maven-war-plugin</artifactId>
          <version>3.2.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-install-plugin</artifactId>
          <version>2.5.2</version>
        </plugin>
        <plugin>
          <artifactId>maven-deploy-plugin</artifactId>
          <version>2.8.2</version>
        </plugin>
      </plugins>
    </pluginManagement>
  </build>
</project>
```

## OUTPUT

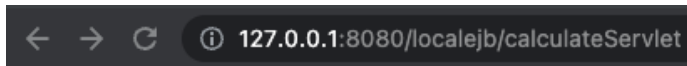


**Calculate your BMI**  
Fill in the data below.

Your Firstname

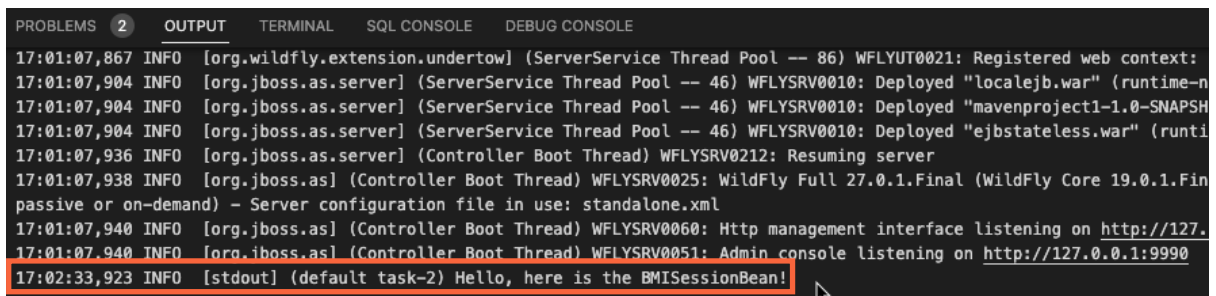
Your Height (m)

Your Weight (kg)



## Ihr BMI Ergebnis

Hallo Thomas ihr BMI ist 25.0



```
PROBLEMS 2 OUTPUT TERMINAL SQL CONSOLE DEBUG CONSOLE
17:01:07,867 INFO [org.wildfly.extension.undertow] (ServerService Thread Pool -- 86) WFLYUT0021: Registered web context:
17:01:07,904 INFO [org.jboss.as.server] (ServerService Thread Pool -- 46) WFLYSRV0010: Deployed "localejb.war" (runtime-n
17:01:07,904 INFO [org.jboss.as.server] (ServerService Thread Pool -- 46) WFLYSRV0010: Deployed "mavenproject1-1.0-SNAPSH
17:01:07,904 INFO [org.jboss.as.server] (ServerService Thread Pool -- 46) WFLYSRV0010: Deployed "ejbstateless.war" (runti
17:01:07,936 INFO [org.jboss.as.server] (Controller Boot Thread) WFLYSRV0212: Resuming server
17:01:07,938 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: WildFly Full 27.0.1.Final (WildFly Core 19.0.1.Fin
passive or on-demand) - Server configuration file in use: standalone.xml
17:01:07,940 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0060: Http management interface listening on http://127.
17:01:07,940 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0051: Admin console listening on http://127.0.0.1:9990
17:02:33,923 INFO [stdout] (default task-2) Hello, here is the BMISessionBean!
```

## TASK 2

1. **How do Stateless and Stateful Session Beans differ? Please give two examples for the use of Stateless and Stateful Session Beans**

→ **Stateless Beans (zustandslos)**: hier müssen bei jedem Aufruf alle relevanten Daten übertragen werden, damit der Service ohne gespeicherte Daten überhaupt funktionieren kann z.B. einfache Suchanfrage auf Webseiten

→ **Stateful Beans (Zustand)**: Session Beans mit einem Zustand müssen gewisse Informationen über einen Zeitraum speichern.

2. **What is the division of task between EJB programmer and EJB container? What does the container do for you?**

Die Aufgabe des Programmierers ist es EJBs zu definieren, damit sie als geeignete Middleware fungieren können. Der EJB Container fängt die Kommunikation von Clients mit dem Server ab und verarbeitet sie mit entsprechenden Java Beans.

## TASK 1

Enterprise JavaBeans sind standardisierte Komponenten innerhalb eines mit Java-EE erstellten Servers. Sie können dem Nutzer bei der Entwicklung helfen, indem sie die Umsetzung gewisser Business-logic innerhalb der Software vereinfachen.

EJB sind Komponenten, da sie oft die Zustände gewisser Aktionen speichern können müssen. Bei Session Beans besteht die Möglichkeit, diese zustandslos und somit als Service zu verwenden