
Protokoll zur SEW-Übung „TempConverter“

**SEW
4AHITM 2015/16**

Eren Sefer

Note:

Betreuer: Michael Borko

Dominik Dolezal

Version 1.0

Begonnen am 06. Juni 2016

Beendet am 06. Juni 2016

INHALTSVERZEICHNIS

1	Aufgabenstellung	3
2	Aufwandschätzung	3
3	Link zum Git-Hub Repo	3
4	Durchführung der Aufgabenstellung	4
4.1	Schritt 1	4
4.2	Schritt 2	5
4.3	Schritt 3	5
4.4	Schritt 4	6
4.5	Schritt 6	7
5	Zeitaufzeichnung	7
6	Literaturverzeichnis	7

1 AUFGABENSTELLUNG

Folge der Anleitung [1] und erstelle einen Temperatur-Konverter und erweitere das Projekt, sodass auch Grad Kelvin unterstützt werden!

Abgabe: Protokoll (inkl. Kopf- und Fußzeile, Screenshots, Ergebnis, ...) + Code auf GitHub pushen

2 AUFWANDSCHÄTZUNG

Arbeitspaket	Geschätzter Aufwand
Anleitung befolgen und Temperatur-Konverter zum Laufen bringen	15 min
Aufgabe erweitern	30 min
Protokoll	30 min
Summe	1 h 15 min

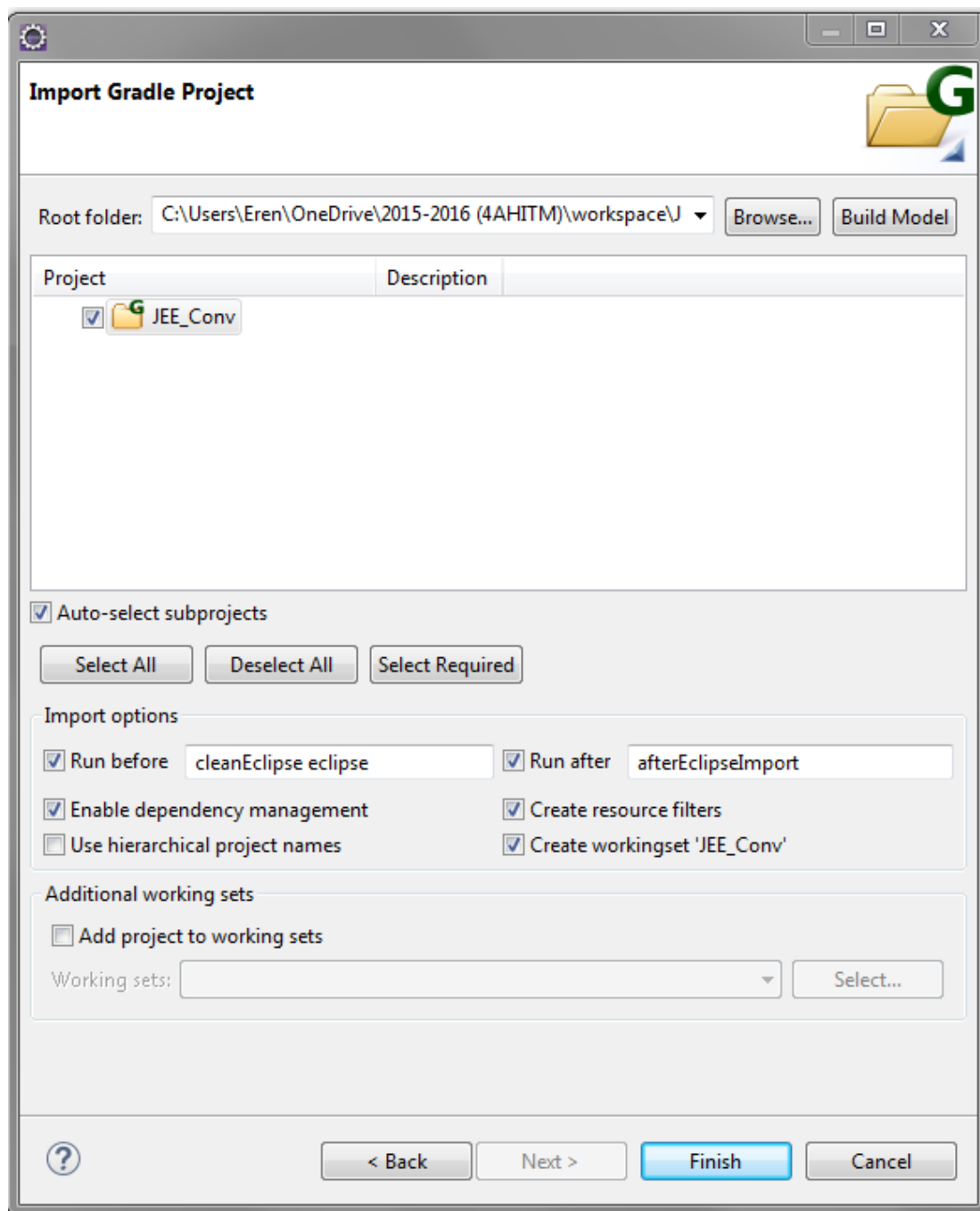
3 LINK ZUM GIT-HUB REPO

Link: https://github.com/esefer-tgm/JEE_Conv

4 DURCHFÜHRUNG DER AUFGABENSTELLUNG

4.1 SCHRITT 1

Als Basis wurde das Beispiel das „HelloJSF“ kopiert und entsprechend der Anleitung umgeändert, sodass es in Eclipse importiert werden kann. Nachdem notwendige Änderungen durchgeführt wurden, kann man das Projekt unter „File - Import – Gradle (STS) – Gradle (STS) Project“ das Projekt importieren. Nachdem über „Browse...“ der Pfad zum erstellten Projekt gezeigt wurde, kann über das „Build Model“ notwendige Abhängigkeiten über Eclipse erstellt werden und somit ist das Importieren des Projektes abgeschlossen.



4.2 SCHRITT 2

Nachdem das Projekt erfolgreich hinzugefügt wurde konnten nun die Erweiterungen hinzugefügt werden.

Diese Methoden wurden zusätzlich zu der `TemperatureConvertor.java` Datei hinzugefügt:

```
public void celsiusToKelvin() {
    this.initial = false;
    this.unit = "Kelvin: ";
    this.converted = convert + 273.15;
}

public void fahrenheitToKelvin() {
    this.initial = false;
    this.unit = "Kelvin: ";
    this.converted = ((convert - 32) / 1.8) + 273.15;
}

public void kelvinToCelsius() {
    this.initial = false;
    this.unit = "Celsius: ";
    this.converted = convert - 273.15;
}

public void kelvinToFahrenheit() {
    this.initial = false;
    this.unit = "Fahrenheit: ";
    this.converted = ((convert - 273.15) * 1.8) + 32;
}
```

4.3 SCHRITT 3

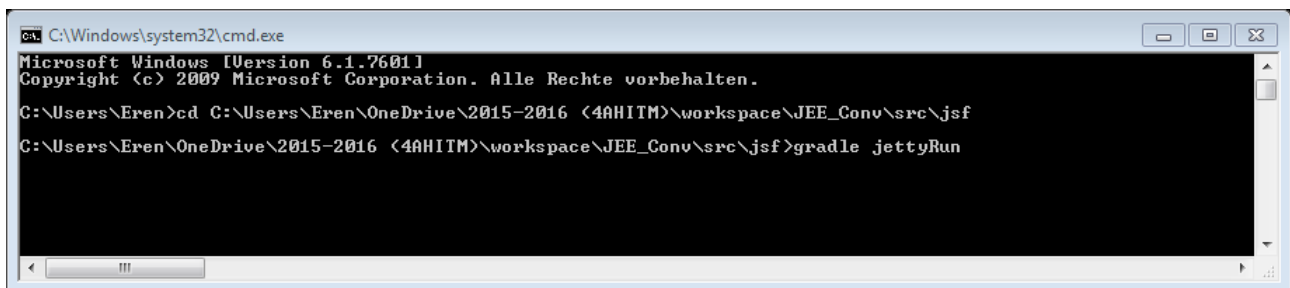
Natürlich sollen die Methoden auch verwendet werden. Damit sie auch über Buttons funktionieren (wie Celsius -> Fahrenheit und Fahrenheit -> Celsius) muss die `convertor.xhtml` Datei bearbeitet werden. Folgende Zeilen kommen hinzu / wurden bearbeitet:

```
</p:panelGrid>
<br />
<p:commandButton value=" Reset " ajax="false"
    action="#{temperatureConvertor.reset}" />
<br />
<br />
<p:commandButton value=" C -> F " ajax="false"
    action="#{temperatureConvertor.celsiusToFahrenheit}" />
<p:commandButton value=" C -> K " ajax="false"
    action="#{temperatureConvertor.celsiusToKelvin}" />
<br />
<br />
<p:commandButton value=" F -> C " ajax="false"
    action="#{temperatureConvertor.fahrenheitToCelsius}" />
```

```
<p:commandButton value=" F -> K " ajax="false"
    action="#{temperatureConvertor.fahrenheitToKelvin}" />
<br />
<br />
<p:commandButton value=" K -> C " ajax="false"
    action="#{temperatureConvertor.kelvinToCelsius}" />
<p:commandButton value=" K -> F " ajax="false"
    action="#{temperatureConvertor.kelvinToFahrenheit}" />
</p:panel>
```

4.4 SCHRITT 4

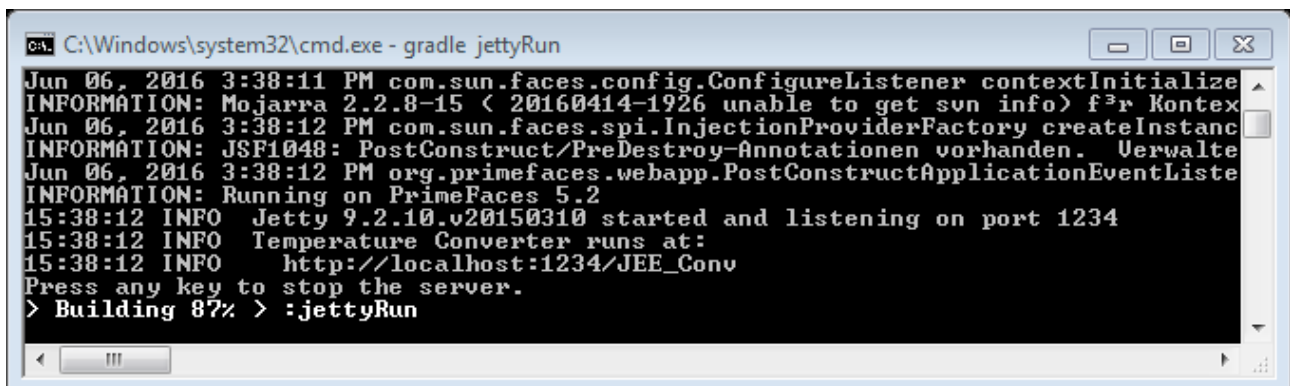
Mit `gradle jettyRun` kann nun das Projekt gebuildet werden.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\Eren>cd C:\Users\Eren\OneDrive\2015-2016 <4AHITM>\workspace\JEE_Conv\src\jsf
C:\Users\Eren\OneDrive\2015-2016 <4AHITM>\workspace\JEE_Conv\src\jsf>gradle jettyRun
```

Unter `localhost:1234/JEE_Conv` kann nun das erweiterte Projekt angesehen werden.



```
C:\Windows\system32\cmd.exe - gradle jettyRun
Jun 06, 2016 3:38:11 PM com.sun.faces.config.ConfigureListener contextInitialize
INFORMATION: Mojarra 2.2.8-15 < 20160414-1926 unable to get sun info> f3r Kontex
Jun 06, 2016 3:38:12 PM com.sun.faces.spi.InjectionProviderFactory createInstanc
INFORMATION: JSF1048: PostConstruct/PreDestroy-Annotationen vorhanden. Uerwalte
Jun 06, 2016 3:38:12 PM org.primefaces.webapp.PostConstructApplicationEventListe
INFORMATION: Running on PrimeFaces 5.2
15:38:12 INFO Jetty 9.2.10.v20150310 started and listening on port 1234
15:38:12 INFO Temperature Converter runs at:
15:38:12 INFO http://localhost:1234/JEE_Conv
Press any key to stop the server.
> Building 87% > :jettyRun
```

4.5 SCHRITT 6

So sieht das Ergebnis nun aus:



Eingegeben: „0.0“

Button: „C -> K“

Ergebnis: „Kelvin 273.15“

5 ZEITAUFZEICHNUNG

Arbeitspaket	tatsächlicher Aufwand
Anleitung befolgen und Temperatur-Konverter zum Laufen bringen	15 min
Aufgabe erweitern	30 min
Protokoll	45 min
Summe	1 h 30 min

6 LITERATURVERZEICHNIS

- [1] *eLearning - Sew Kurs 4xHIT 15/16 - Anleitung TempConv.* (2016)
 Abgerufen am 06. 06. 2016 von
<https://elearning.tgm.ac.at/mod/resource/view.php?id=50344>