JAVA SE 8

Java SE 8 Programmer I
Bzw.
ORACLE CERTIFIED ASSOCIATE, JAVA SE 8 PROGRAMMER (OCAJP 8)

Christian Schirmer

Development

- javac und java
  - *Bp. I – mehrere Java Dateien in einem Verzeichnis*
  - *Bp. II – mehrere Java Dateien in unterschiedlichen Paketen (Option -classpath)*
- System Eigenschaften
- Kommandozeilen Argumente
- Jar Dateien
  - *Bp. I – Erzeugung von jar -Files*

Christian Schirmer

javac und java – Beispiel 1

- Mehrere Dateien in einem Ordner

```
E:\Comcave\demo>dir
 Datenträger in Laufwerk E: ist Elements 2TB
 Volumeseriennummer: 5839-C2ED

 Verzeichnis von E:\Comcave\demo

28.10.2018  05:10    <DIR>          .
28.10.2018  05:10    <DIR>          ..
28.10.2018  05:10               109 Bar.java
28.10.2018  05:10               109 Foo.java
               2 Datei(en),            218 Bytes
               2 Verzeichnis(se), 1.738.868.621.312 Bytes frei
```

```java
public class Bar{

    public static void main(String[] args){
        System.out.println("This is Bar");
    }

}
```

```java
public class Foo{

    public static void main(String[] args){
        System.out.println("This is Foo");
    }

}
```

Christian Schirmer

javac und java – Beispiel 1

- Mehrere Dateien in einem Ordner

```
E:\Comcave\demo>dir
 Datenträger in Laufwerk E: ist Elements 2TB
 Volumeseriennummer: 5839-C2ED

 Verzeichnis von E:\Comcave\demo

28.10.2018  05:10    <DIR>          .
28.10.2018  05:10    <DIR>          ..
28.10.2018  05:10               109 Bar.java
28.10.2018  05:10               109 Foo.java
               2 Datei(en),            218 Bytes
               2 Verzeichnis(se), 1.738.868.621.312 Bytes frei

E:\Comcave\demo>javac Bar.java

E:\Comcave\demo>javac Foo.java

E:\Comcave\demo>dir
 Datenträger in Laufwerk E: ist Elements 2TB
 Volumeseriennummer: 5839-C2ED

 Verzeichnis von E:\Comcave\demo

28.10.2018  05:18    <DIR>          .
28.10.2018  05:18    <DIR>          ..
28.10.2018  05:18               411 Bar.class
28.10.2018  05:10               109 Bar.java
28.10.2018  05:18               411 Foo.class
28.10.2018  05:10               109 Foo.java
               4 Datei(en),          1.040 Bytes
               2 Verzeichnis(se), 1.738.868.568.064 Bytes frei
```

Nach dem Kompilieren mit javac erhalten wir 2 weitere Dateien in unseren Ordner.

javac und java – Beispiel 1

```
E:\Comcave\demo>dir
 Datenträger in Laufwerk E: ist Elements 2TB
 Volumeseriennummer: 5839-C2ED

 Verzeichnis von E:\Comcave\demo

28.10.2018  05:18    <DIR>          .
28.10.2018  05:18    <DIR>          ..
28.10.2018  05:18               411 Bar.class
28.10.2018  05:10               109 Bar.java
28.10.2018  05:18               411 Foo.class
28.10.2018  05:10               109 Foo.java
               4 Datei(en),          1.040 Bytes
               2 Verzeichnis(se), 1.738.868.568.064 Bytes frei

E:\Comcave\demo>java Bar
This is Bar

E:\Comcave\demo>java Foo
This is Foo

E:\Comcave\demo>
```

Mit dem Befehl:
java Bar
bzw.
java Foo
Können diese Dateien ausgeführt werden.

javac und java – Beispiel 1

- Mehrere Dateien in einem Ordner

  !!! Beachten sie, das die Endung .class nicht angegeben wurde. Wenn Sie diesen befehl mit der Dateiendung .class ausführen, bekommen Sie eine Fehlermeldung.

  Dieser Fehler beruht auf das Classpath System von Java.

```
E:\Comcave\demo>java Foo.class
Fehler: Hauptklasse Foo.class konnte nicht gefunden oder geladen werden
```

javac und java - classpath

Beide Programme verwenden den gleichen Suchalgorithmus:

- Sie schauen in den gleichen Ordnern nach den Klassen
- Sofern sie die Klasse gefunden haben, die sie gerade suchen, beenden sie ihre suche
- Für den Fall, das ihre Suchliste mehr als eine Datei beinhaltet, die den Gleichen Namen haben, wird die zuerst gefundene Datei verwendet.

javac und java - classpath

- Zuerst wird in den Verzeichnissen der Standart Java SE Bibliothek gesucht
- Danach werden die Verzeichnisse des definierten Classpath durchsucht.
- Classpaths sollten als "Klassensuchpfade" betrachtet werden. Das sind Listen von Verzeichnissen, in denen Klassen gefunden werden können

Christian Schirmer

javac und java - classpath

- Es gibt zwei Optionen wie Klassenpfade deklariert werden können:
  - *Ein Klassenpfad kann als Betriebssystemumgebungsvariable deklariert werden. Das hier deklarierte Klassenverzeichnis wird standardmäßig verwendet, wenn java oder javac aufgerufen wird.*
  - *Ein Klassenpfad kann für java oder javac als Befehlszeilenoption deklariert werden. Klassenpfade, die als Befehlszeilenoptionen deklariert werden, überschreiben das als Umgebungsvariable deklarierte Klassenverzeichnis, bleiben jedoch nur für die Dauer des Aufrufs bestehen.*

javac und java - classpath

- Klassenpfade bestehen aus einer variablen Anzahl von Verzeichnispositionen, die durch Trennzeichen getrennt sind.
- Bei Unix-basierten Betriebssystemen werden Schrägstriche zum Erstellen von Verzeichnispositionen verwendet, und das Trennzeichen ist der Doppelpunkt (:).
- Wenn Sie ein Windows-Benutzer sind, werden Ihre Verzeichnisse mit umgekehrten Schrägstrichen (\) deklariert und das Trennzeichen, das Sie verwenden, ist ein Semikolon (;).

javac und java - classpath

- Als beispiel: -classpath /com/foo/acct:/com/foo
- Gibt zwei Verzeichnisse an, in denen Klassen gefunden werden können: /com/foo/acct und /com/foo
- In beiden Fällen sind diese Verzeichnisse absolut an das Wurzelverzeichnis des Dateisystems gebunden, der durch den vorangestellten Schrägstrich angegeben wird.
- Beachten Sie, dass Sie bei Angabe eines Unterverzeichnisses NICHT die darüber liegenden Verzeichnisse angeben.
- Zum Beispiel wird im obigen Beispiel das Verzeichnis /com NICHT durchsucht.

javac und java - classpath

- Bei der Suche nach Klassendateien durchsuchen die Befehle java und javac standardmäßig nicht das aktuelle Verzeichnis. Sie müssen ihnen sagen, dass sie dort suchen sollen.
- Um javac oder java im aktuellen Verzeichnis suchen zu lassen, fügen Sie dem Klassenpfad einen Punkt (.) hinzu:
- -classpath /com/foo/acct:/com/foo:.
- Wenn Sie javac mitteilen, welche .java-Datei kompiliert werden soll, sucht javac standardmäßig im aktuellen Verzeichnis.

javac und java - classpath

- Beachten Sie auch, dass Klassenpfade von **links nach rechts** durchsucht werden.
- Daher werden in einer Situation, in der Klassen mit doppelten Namen sich in mehreren verschiedenen Verzeichnissen in den folgenden Klassenpfaden befinden, unterschiedliche Ergebnisse auftreten:
- -classpath /com:/foo:.
- Ist nicht identisch mit
- -classpath .:/foo:/com

Christian Schirmer

javac und java - classpath

- Für den Befehl java können Sie -classpath mit -cp abkürzen.
- Für javac ist die Dokumentation zur Abkürzung inkonsistent. Bei den meisten Systemen funktioniert es, aber es gibt keine Garantien.

javac und java - cp – Beispiel 2

- Mehrere Java Dateien in unterschiedlichen Ordnern(Paketen)

```
E:\COMCAVE\DEV02
\---pkg01
    |   Bar.java
    |
    \---pgk02
            Foo.java
```

```java
package pkg01;

public class Bar{

    public static void main(String[] args){
        System.out.println("This is Bar");
```

```java
import pkg01.Bar;

public class Foo{

    public static void main(String[] args){
        System.out.println("This is Foo");
        Bar bar = new Bar();
```

Christian Schirmer

javac und java - cp – Beispiel 2

- Mehrere Java Dateien in unterschiedlichen Ordnern(Paketen)

```
E:\COMCAVE\DEV02
\---pkg01
    |   Bar.java
    |
    \---pgk02
            Foo.java


E:\Comcave\dev02>javac -classpath . pkg01/pkg02/Foo.java

E:\COMCAVE\DEV02
\---pkg01
    |   Bar.class
    |   Bar.java
    |
    \---pkg02
            Foo.class
            Foo.java
```

javac und java - cp – Beispiel 2

- Mehrere Java Dateien in unterschiedlichen Ordnern(Paketen)

```
E:\COMCAVE\DEV02
\---pkg01
    |   Bar.class
    |   Bar.java
    |
    \---pkg02
            Foo.class
            Foo.java

E:\Comcave\dev02>java pkg01/pkg02/Foo
This is Foo

E:\Comcave\dev02>java -classpath . pkg01/pkg02/Foo
This is Foo

E:\Comcave\dev02>java -classpath . pkg01.pkg02.Foo
This is Foo
```

javac und java - d – Beispiel 3

- .class Dateien in separates Verzeichnis erstellen lassen.

```
E:\COMCAVE\DEV03
+---classes
\---source
        MyClass.java
```

```java
public class MyClass{

    public static void main(String[] args){
        System.out.println("This is MyClass");
    }

}
```

Christian Schirmer

javac und java - d – Beispiel 3

- .class Dateien in separates Verzeichnis erstellen lassen.

```
E:\COMCAVE\DEV03
+---classes
\---source
        MyClass.java


E:\Comcave\dev03>javac -d classes source/MyClass.java

E:\COMCAVE\DEV03
+---classes
|       MyClass.class
|
\---source
        MyClass.java

E:\Comcave\dev03>java -classpath ./classes MyClass
This is MyClass

E:\Comcave\dev03>java -cp ./classes MyClass
This is MyClass
```

System Eigenschaften auslesen

```
import java.util.Properties;

public class TestProps {

    public static void main(String[] args) {
        Properties p = System.getProperties();
        p.setProperty("myProp", "MyValue");
        p.list(System.out);
    }

}
```

System Eigenschaften auslesen

```
E:\Develop\demo>java TestProps
-- listing properties --
java.runtime.name=Java(TM) SE Runtime Environment
sun.boot.library.path=C:\Java\jre1.8.0_191\bin
java.vm.version=25.191-b12
java.vm.vendor=Oracle Corporation
java.vendor.url=http://java.oracle.com/
path.separator=;
java.vm.name=Java HotSpot(TM) 64-Bit Server VM
file.encoding.pkg=sun.io
user.script=
user.country=DE
sun.java.launcher=SUN_STANDARD
sun.os.patch.level=
java.vm.specification.name=Java Virtual Machine Specification
user.dir=E:\Develop\demo
java.runtime.version=1.8.0_191-b12
java.awt.graphicsenv=sun.awt.Win32GraphicsEnvironment
java.endorsed.dirs=C:\Java\jre1.8.0_191\lib\endorsed
os.arch=amd64
java.io.tmpdir=C:\Users\[redacted]AppData\Local\Temp\
line.separator=

java.vm.specification.vendor=Oracle Corporation
user.variant=
os.name=Windows 10
myProp=MyValue
```

Kommandozeilen Argumente

```
public class CmdArgs {

    public static void main(String[] args) {
        int x = 0;
        for (String s : args){
            System.out.println(x++ + " element = " + s);
        }
    }
}
```

Christian Schirmer

Kommandozeilen Argumente

```
E:\Develop\demo>javac CmdArgs.java

E:\Develop\demo>java CmdArgs a
0 element = a

E:\Develop\demo>java CmdArgs a 1
0 element = a
1 element = 1

E:\Develop\demo>java CmdArgs a 1 b
0 element = a
1 element = 1
2 element = b

E:\Develop\demo>java CmdArgs a 1 b 3
0 element = a
1 element = 1
2 element = b
3 element = 3
```

Java –jar – Beispiel 1

- Jar-Dateien (Java-Archiv) bilden ein Archivformat
  - *Die Dateien sind gebündelt, jedoch nicht komprimiert.*

Java –jar – Beispiel 1

- Die wichtigsten Formen für das Kommandozeilenprogramms jar sind:
- Anlegen: jar c[Optionen] Jar-Datei Eingabedateien
- Aktualisieren: jar u[Optionen] Jar-Datei Eingabedateien
- Auspacken: jar x[Optionen] Jar-Datei
- Inhalt anzeigen: jar t[Optionen] Jar-Datei
- Indexdatei INDEX.LIST erzeugen: jar i Jar-Datei

https://docs.oracle.com/javase/tutorial/deployment/jar/build.html

Christian Schirmer

Java –jar – Beispiel 1

- Die wichtigsten Formen für das Kommandozeilenprogramms jar sind:
- c: create
- f: file
- e: entrypoint (seit Java 1.6)
  ermöglicht die direkte Angabe der ausführbaren Klasse in der Manifest-Datei des Java-Archivs

Christian Schirmer

Java –jar – Beispiel 1

- Erzeugen einer jar - Datei

```java
public class HelloWorld {
    public static void main(String... args) {
        System.out.println("Hallo Welt");
        System.out.println("Hello World");
    }
}
```

```
E:\Develop\dev04>javac HelloWorld.java

E:\Develop\dev04>jar -cf HelloWorld.jar HelloWorld.class

E:\Develop\dev04>java -cp HelloWorld.jar HelloWorld
Hallo Welt
Hello World
```

Java –jar – Beispiel 1

- Erzeugen einer ausführbaren jar - Datei

```
public class HelloWorld {
    public static void main(String... args) {
        System.out.println("Hallo Welt");
        System.out.println("Hello World");
    }
}
```

```
E:\Develop\dev04>jar cfe HelloWorld.jar HelloWorld HelloWorld.class

E:\Develop\dev04>java -jar HelloWorld.jar
Hallo Welt
Hello World

E:\Develop\dev04>
```

Christian Schirmer

Java –jar – Beispiel 1

- jar – Datei (nicht ausführbar)

```
E:\Develop\dev04>jar -cf HelloWorld.jar HelloWorld.class

E:\Develop\dev04>dir
 Verzeichnis von E:\Develop\dev04

28.10.2018  07:13    <DIR>          .
28.10.2018  07:13    <DIR>          ..
28.10.2018  07:12               453 HelloWorld.class
28.10.2018  07:13               773 HelloWorld.jar
28.10.2018  07:01               153 HelloWorld.java
               3 Datei(en),          1.379 Bytes
               2 Verzeichnis(se), 1.738.867.245.056 Bytes frei

E:\Develop\dev04>jar -tf HelloWorld.jar
META-INF/
META-INF/MANIFEST.MF
HelloWorld.class

E:\Develop\dev04>java -jar HelloWorld.jar
kein Hauptmanifestattribut, in HelloWorld.jar
```

```
1  Manifest-Version: 1.0
2  Created-By: 1.8.0_191 (Oracle Corporation)
```

Christian Schirmer

Java –jar – Beispiel 1

- jar – Datei (ausführbar)

```
E:\Develop\dev04>jar -cfe HelloWorld.jar HelloWorld HelloWorld.class

E:\Develop\dev04>jar -tf HelloWorld.jar
META-INF/
META-INF/MANIFEST.MF
HelloWorld.class

E:\Develop\dev04>java -jar HelloWorld.jar
Hallo Welt
Hello World
```

```
1  Manifest-Version: 1.0
2  Created-By: 1.8.0_191 (Oracle Corporation)
3  Main-Class: HelloWorld
```

Java –jar – Beispiel 2

```
E:\DEVELOP\DEV05
\---dirAI
        ClassA.java
        ClassABC.java
```

```java
package dirAI;

public class ClassABC {
    public static void main(String[] args) {
        ClassA cA = new ClassA();
        cA.methodA();
```

```java
package dirAI;

public class ClassA {
    public void methodA(){
        System.out.println("methodA");
```

Christian Schirmer

Java –jar – Beispiel 2

```
E:\Develop\dev05>javac -classpath . dirAI/ClassABC.java

E:\Develop\dev05>java dirAI.ClassABC
methodA

E:\Develop\dev05>jar -cfe ClassABC.jar dirAI/ClassABC dirAI/

E:\Develop\dev05>java -jar ClassABC.jar
methodA
```

Java –jar – Beispiel 3
Ausführbare Jar Files mit Eclipse erstellen

- File, Export – Java , Runnable JAR file
- Launch Configuration: Hello World – Hello World
- JAR file Zielpfad angeben – Finish klicken
- Die Datei um Dateisystem suchen: Pfad in die Zwischenablage kopieren
- Windows-Kommandozeile: Windows, Start, cmd
- Cd <Pfad aus der Zwischenablage>
- Java –jar HelloWorld.jar

Christian Schirmer