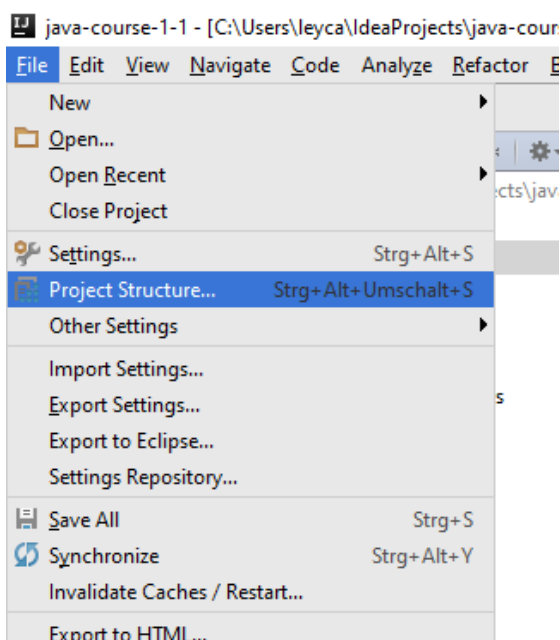

A2. Programme auf der Konsole ausführen

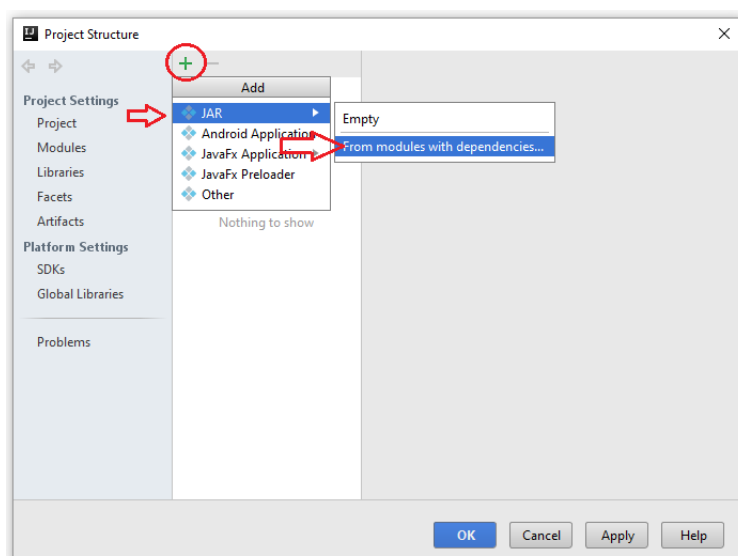
1. Erstellen einer JAR-Datei mit IntelliJ

Um ein Java Programm ohne Entwicklungsumgebung zu starten, muss es zunächst mit der **JDK** – dem **Java Development Kit** kompiliert werden. Das Ergebnis ist (zum Beispiel) eine JAR-Datei, die dann mit der **JRE** – also die **Java Runtime Environment** und damit der **JVM** – **Java Virtual Machine** ausgeführt werden kann.



Unter IntelliJ navigieren Sie dazu zur

„File“ -> „Projektstruktur“



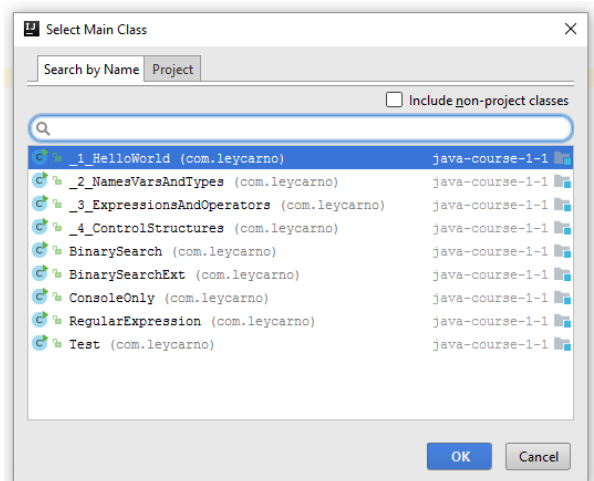
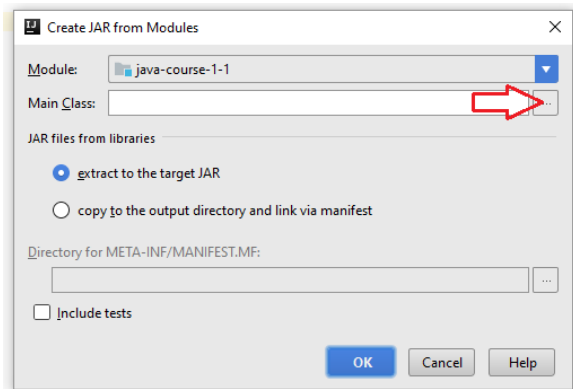
Dort Wählen Sie mit einem Klick auf das **grüne Plus**

„JAR“ ->

„From modules with dependencies“

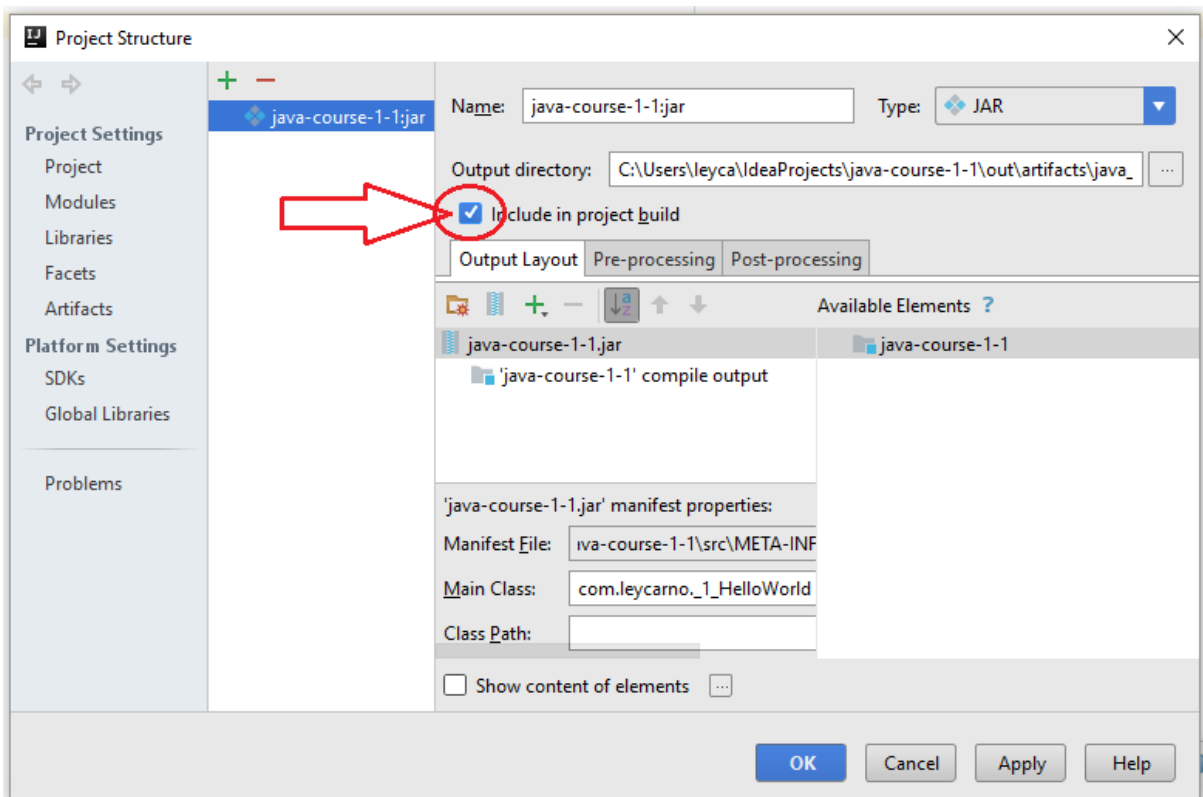
aus.

Im dritten Dialog müssen Sie die Klasse auswählen, in der sich die main-Methode befindet, mit der das kompilierte Programm starten soll:

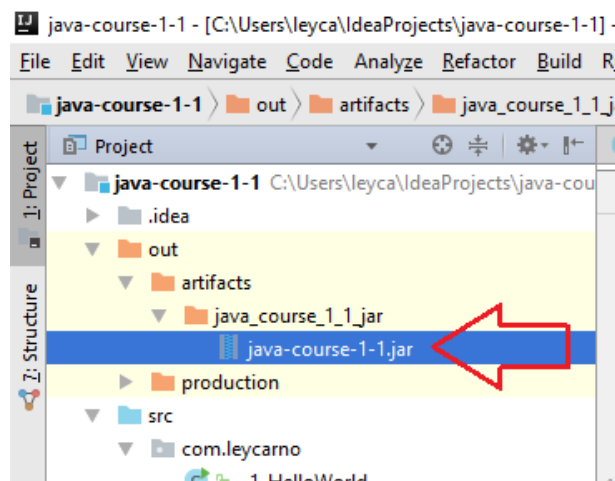
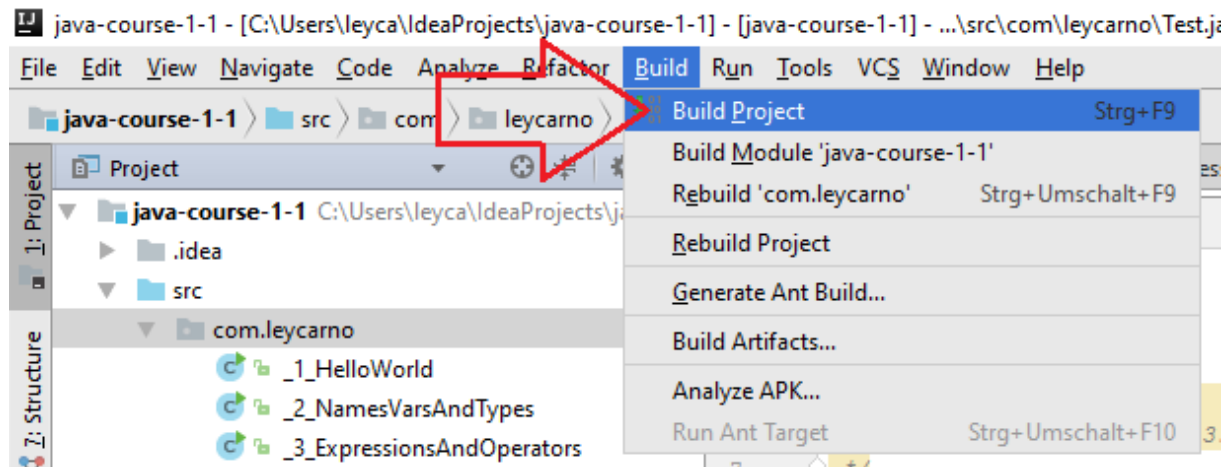


Hier bei diesem Beispiel wäre das „Hello World“.

Bestätigen Sie mit „OK“ und machen Sie einen Haken bei „**Include in project build**“. Dadurch wird eine JAR-Datei immer automatisch mit erzeugt, wenn Sie das Programm mit IntelliJ erfolgreich starten und ist immer auf dem neuesten Stand:



Sie können den Vorgang des Kompilierens nun auch durchführen, ohne das Programm zu starten, indem Sie es „bauen“. Navigieren Sie hierfür zu „**Build**“ -> „**Build Project**“.



Anschließend haben Sie im „out“ Ordner unter „artifacts“ in etwa die dargestellte Struktur.

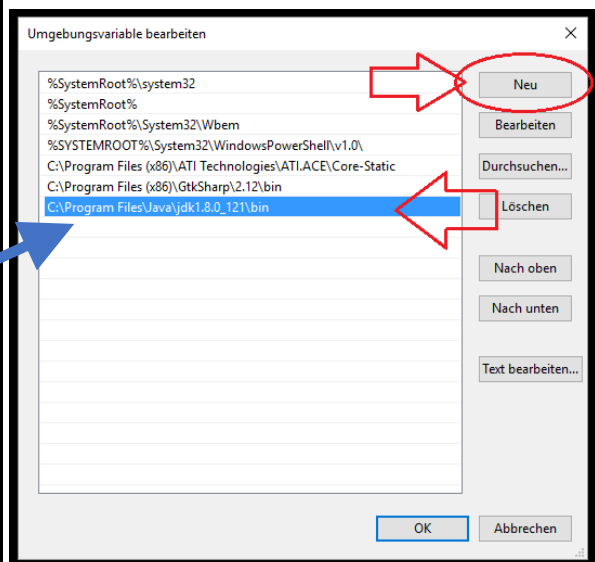
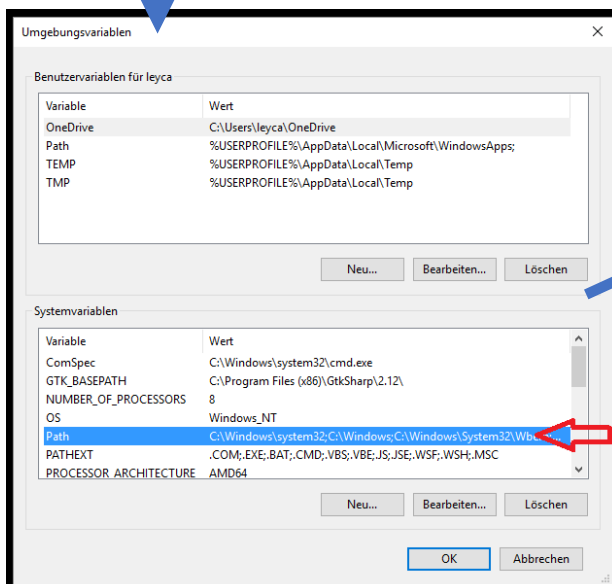
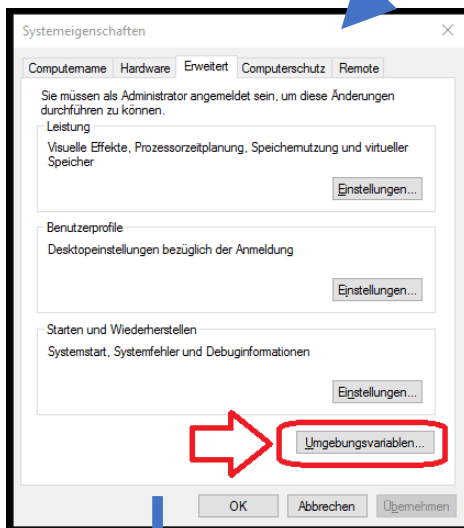
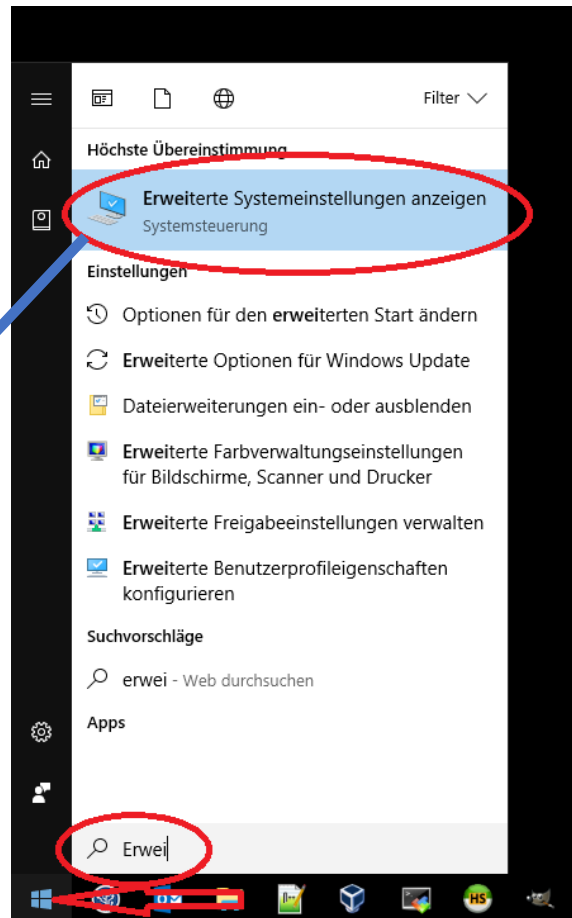
Diese JAR-Datei kann nun auf jedem Rechner ausgeführt werden auf dem zumindest die passende JRE, also das „normale“ Java installiert ist.

Wurde das Programm also z.B. mit Java 7 erstellt muss das ausführende System dieses Minimum erfüllen.

2. Java global im System verwenden (Beispiel Windows 10)

Um Java Programme in der Konsole auszuführen muss es entweder direkt dort ausgeführt werden, wo es sich befindet, oder man „erzählt“ dem System wo es zu finden ist.

Daher muss es im Fall von Windows bei den Umgebungsvariablen hinterlegt werden. Danach kann man das die Java.exe von „überall“ aus aufrufen, um jar-Dateien auszuführen.



3. Programm über die Konsole starten

Starten Sie die Eingabeaufforderung CMD (unter Windows – unter anderen Systemen kann dies abweichen) und geben sie einfach mal:

java -version

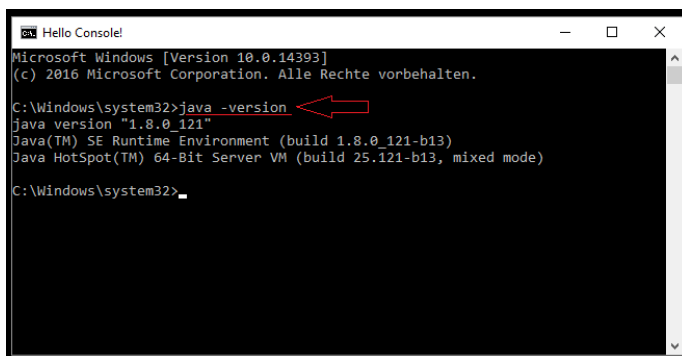
ein. Wenn die installierte Version angezeigt ist, können Sie davon ausgehen, dass Schritt 2 dieser Anleitung funktioniert hat.

Anschließend können Sie die JAR-Dateien über den Befehl:

java -jar [NAME DES JAR.DATEI]

ausführen.

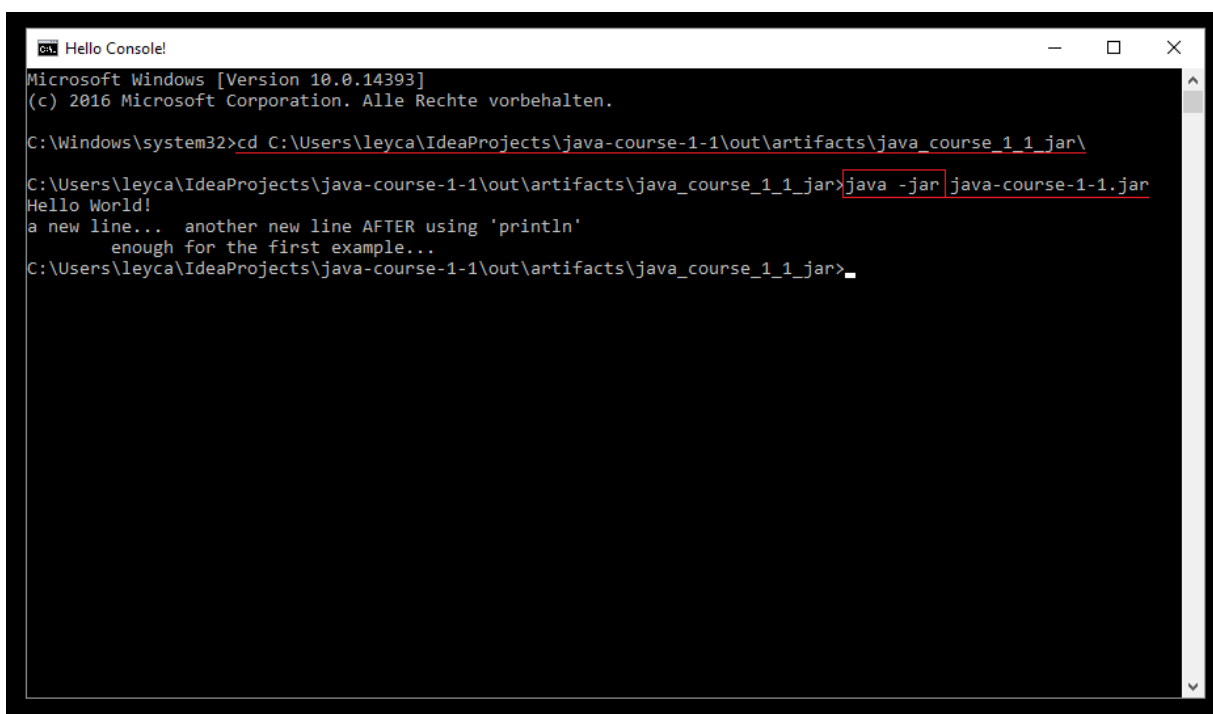
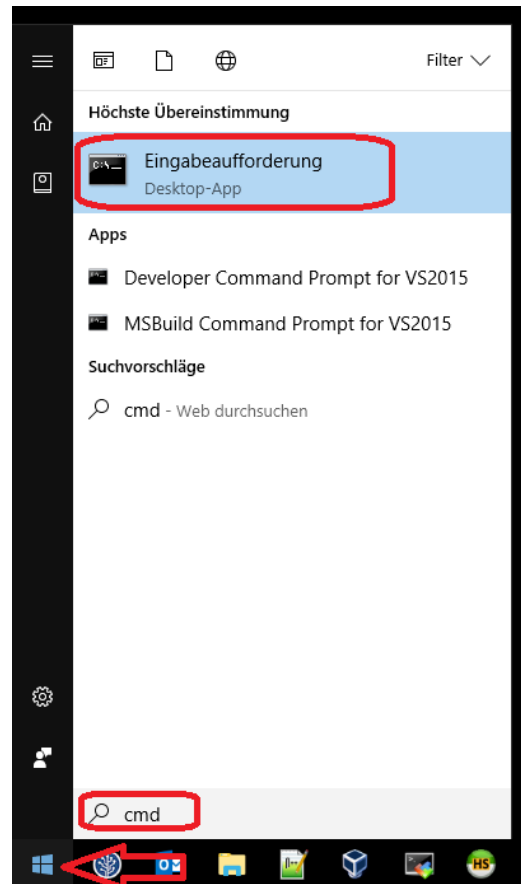
ANMERKUNG: Es ist auch möglich jar-Dateien umzubenennen und Sie an eine andere Stelle zu kopieren...



```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Windows\system32>java -version
java version "1.8.0_121"
Java(TM) SE Runtime Environment (build 1.8.0_121-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.121-b13, mixed mode)

C:\Windows\system32>
```



```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Windows\system32>cd C:\Users\leyca\IdeaProjects\java-course-1-1\out\artifacts\java_course_1_1_jar\
C:\Users\leyca\IdeaProjects\java-course-1-1\out\artifacts\java_course_1_1_jar>java -jar java-course-1-1.jar
Hello World!
a new line... another new line AFTER using 'println'
enough for the first example...
C:\Users\leyca\IdeaProjects\java-course-1-1\out\artifacts\java_course_1_1_jar>
```