

OOP :01 Aufgabenblatt Eclipse, Hello World

A01.1:

Windows-System

Nach dem Installieren der Anwendungen ist es nun erforderlich den Javapath anzulegen. In Windows 10 geht es wie folgt:

1. Gehen Sie in die Suche und suchen Sie nach System (Systemsteuerung)
2. Klicken Sie dort links auf den untersten Link "Erweiterte Systemeinstellungen".
3. Weiter auf den Link "Umgebungsvariablen".
4. Klicken Sie dort auf bearbeiten bzw. neu, wenn "Path" noch nicht vorhanden ist.
5. Geben Sie nun den Wert der Variable ein. **C:\Program Files\java\jdk1.8.0_102\bin;**

Mac OS X

1. Downloaden der Java JDK
2. Ausführen des .dmg Files
3. Öffnen von Eclipse
4. Ein neues Java-Projekt anlegen
5. Projekt speichern

A.01.2:

Unter Windows

Als erstes starten wir die Konsole als Administrator mit *cmd*. Mit dem Befehl "**cd**" können wir uns innerhalb der Ordner hin und her bewegen, um in dass gewünschte Verzeichniss zu gelangen. Um in die vorherigen Ordner zu gelangen, muss "**cd ..**" eingegeben werden und um in den gewünschten Ordner zu gelangen "**cd [Ordnername]**". Mit "**dir**" erhalten wir eine Übersicht aller Ordner und Dateien im aktuellen Verzeichnis in dem wir uns befinden. Im gewünschten Verzeichnis angekommen kompilieren wir den Quellcode nun mit *Javac*.

In meinem Falle befinden wir uns in **C:\users\[Benutzername]\workspace\Praktikum1\src**. Wir führen nun "**javac oop\HelloWorld.java**" aus. Wenn keine Fehlermeldung erscheint, ist die Datei kompiliert und eine *.class* Datei mit dem Namen *HelloWorld.class* erstellt worden. Wir können nun das Programm mit "**java oop.HelloWorld**" starten und es erscheint *HelloWorld* in der Konsole.

Unter OS X

1. Öffnen der Konsole
2. Navigieren zum Java-Programm (/HelloWorld/src/oop/)
3. Kompilieren via „javac oop/HelloWorld.java“
4. Öffnen des Programms mit „java oop.Helloworld“
5. Ausgabe kontrollieren.

A01.3:

Java-Compiler:

Der Compiler wandelt Java Quellcode in einen Javabytcode um. Der Bytecode kann nun in den entsprechenden Laufzeitumgebungen ausgeführt werden.

Java Laufzeitumgebung (JRE):

Damit kann man die in Java programmierten und übersetzten Programme ausführen. Die Java Laufzeitumgebung besteht aus der Java Virtual Maschine(JVM), eine Programmschnittstelle(API) und weiteren Programmbibliotheken.

Java VM ist für die Ausführung der Programme verantwortlich, diese muss mit dem API abgestimmt sein und wird deshalb zur JRE zusammengefasst.

Jedes ausgeführte Java-Programm wird in ihrer eigenen virtuellen Maschine gestartet.

Entwicklungsumgebung:

Eine Entwicklungsumgebung kann alles, vom einfachen Editor bis hin zum Umfangreichen Tool mit integrierten Compilern sein. Für kleinere Projekte reicht oftmals ein Tool wie Notepad++, während größere Projekte oft sehr viel Komplexer und Umfassender sind.

Im optimalsten Fall besteht eine Entwicklungsumgebung aus einem Editor, Compiler bzw. Interpreter, Linker und einem Debugger. Diese werden in einer IDE (Integrated Development Environment) vereint. Die IDE besteht aus dem oben genannten Programmen und bringt eine grafische Benutzeroberfläche (GUI) mit. Damit wird das entwickeln deutlich vereinfacht, da der Programmierer viele wichtige und nützliche Tools direkt zur Hand hat und Informationen übersichtlich angezeigt bekommt.