

JAVACREAM

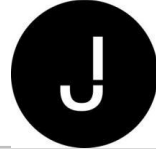
*Training
Consulting
Projectmanagement*

Groovy - Eine Einführung

Organisatorisches

Dies ist eine interaktive Veranstaltung!

- Akute Fragen/Anmerkungen einfach durch Aktivieren des eigenen Mikrofons und Unterbrechen des Vortrags
 - “Handheben” etc. nicht nötig
- Nutzen Sie den Chat, um nicht ganz dringende Fragen/Anmerkungen zu schreiben
- Schreiben Sie eine Mail an training@rainer-sawitzki.de, um z.B. komplexere Fragen mit Anhang etc. zu senden
 - Gerne auch zwischen den Online-Sessions
 - Bitte auch bei technischen Problemen als “Notfall-Kommunikation” nutzen



- Name, Rolle im Unternehmen
- Themenbezogene Vorkenntnisse
- Aktuelle Problemstellung
- Konkrete individuelle Zielsetzung

Aufsetzen der Trainings-Umgebung

Anmeldung: Informationen und Zuordnung der Teilnehmenden



JAVACREAM

Training
Consulting
Projectmanagement

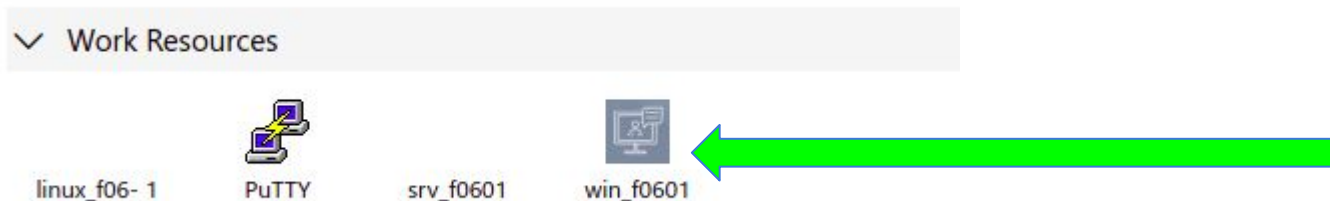
2=Denchev
3=Rudolph
4=El Karsi
5=Hartlap
6=Helbig
7=Ronge
8=Bolz

student_f0602@integrata-cegos.de	Pa\$\$w0rd_f0602	win_f0602	W8Admin	Pa\$\$w0rd
student_f0603@integrata-cegos.de	Pa\$\$w0rd_f0603	win_f0603	W8Admin	Pa\$\$w0rd
student_f0604@integrata-cegos.de	Pa\$\$w0rd_f0604	win_f0604	W8Admin	Pa\$\$w0rd
student_f0605@integrata-cegos.de	Pa\$\$w0rd_f0605	win_f0605	W8Admin	Pa\$\$w0rd
student_f0606@integrata-cegos.de	Pa\$\$w0rd_f0606	win_f0606	W8Admin	Pa\$\$w0rd
student_f0607@integrata-cegos.de	Pa\$\$w0rd_f0607	win_f0607	W8Admin	Pa\$\$w0rd
student_f0608@integrata-cegos.de	Pa\$\$w0rd_f0608	win_f0608	W8Admin	Pa\$\$w0rd
student_f0609@integrata-cegos.de	Pa\$\$w0rd_f0609	win_f0609	W8Admin	Pa\$\$w0rd
student_f0610@integrata-cegos.de	Pa\$\$w0rd_f0610	win_f0610	W8Admin	Pa\$\$w0rd
student_f0611@integrata-cegos.de	Pa\$\$w0rd_f0611	win_f0611	W8Admin	Pa\$\$w0rd
student_f0612@integrata-cegos.de	Pa\$\$w0rd_f0612	win_f0612	W8Admin	Pa\$\$w0rd
student_f0613@integrata-cegos.de	Pa\$\$w0rd_f0613	win_f0613	W8Admin	Pa\$\$w0rd

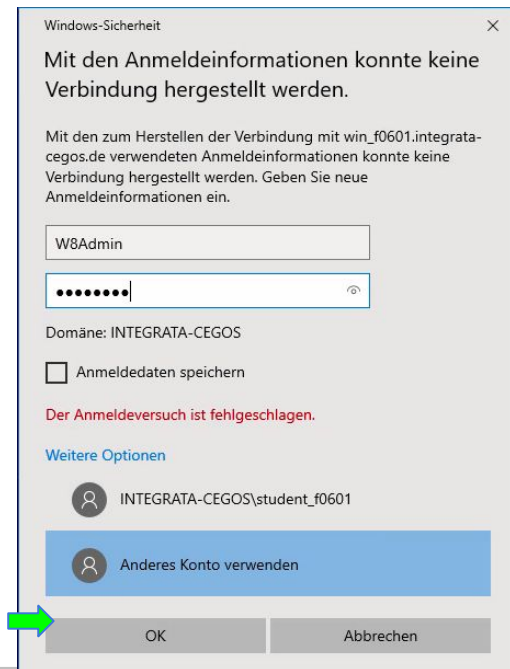
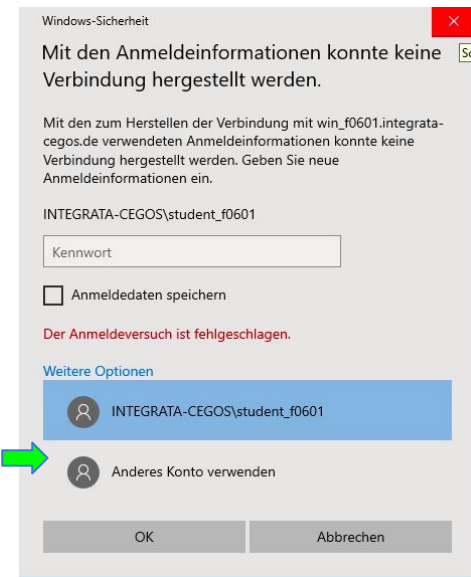
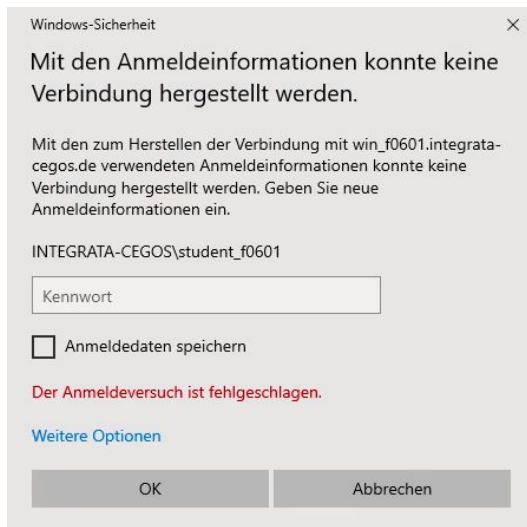
RDWEB-Link:

<https://rdweb.integrata-cegos.de/rdweb/webclient>

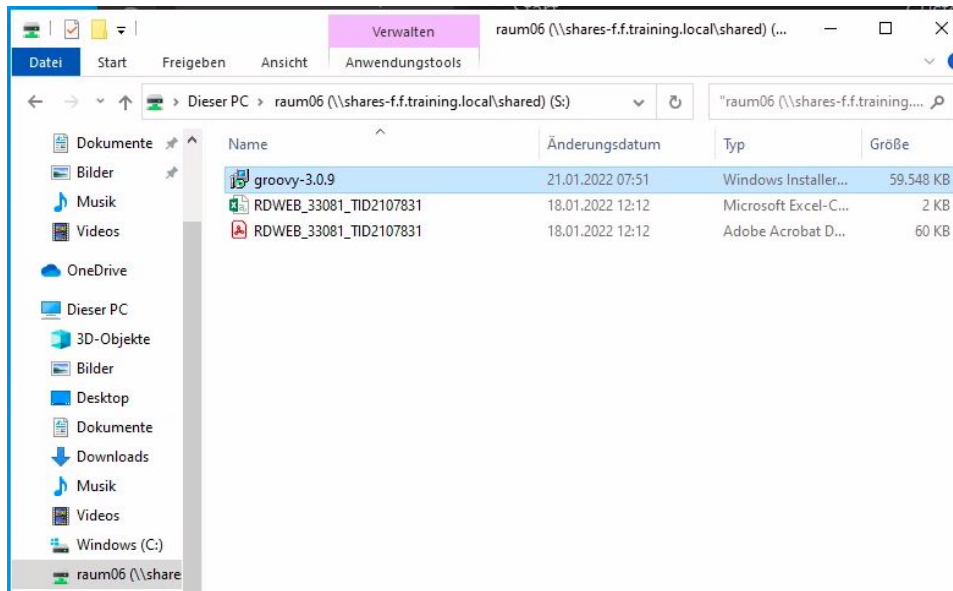
- Öffnen der URL <https://rdweb.integrata-cegos.de/rdweb/webclient/>
- Eingabe des ersten Usernamens (Pseudo-Mail-Adresse) und Passwort
- Auswahl der Windows-Ressource
-



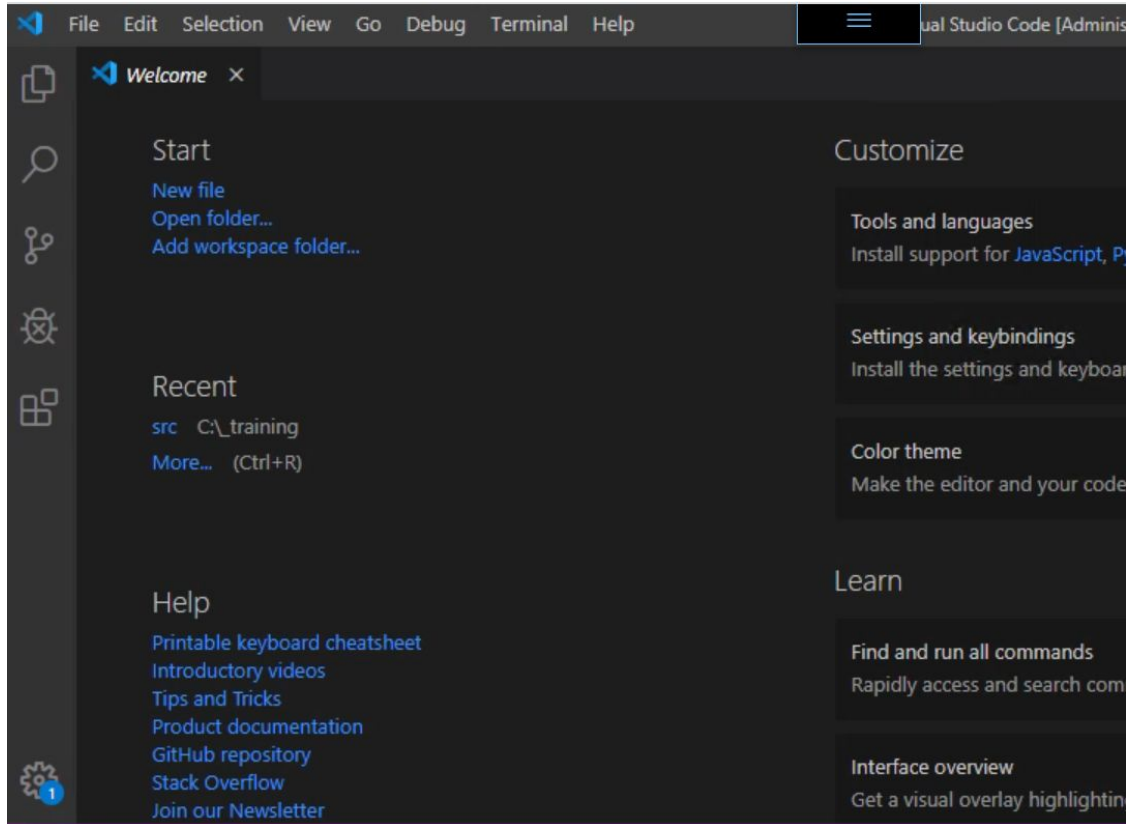
- Anmeldung schlägt fehl, dann die Option “Als Anderer Benutzer anmelden” wählen und als Benutzer W8Admin eingeben



- Öffnen Sie einen Dateiexplorer
- Installieren Sie vom Laufwerk raum06 groovy-3.0.9
 - Akzeptieren Sie die Lizenz sowie alle Vorschläge



Start Visual Studio Code

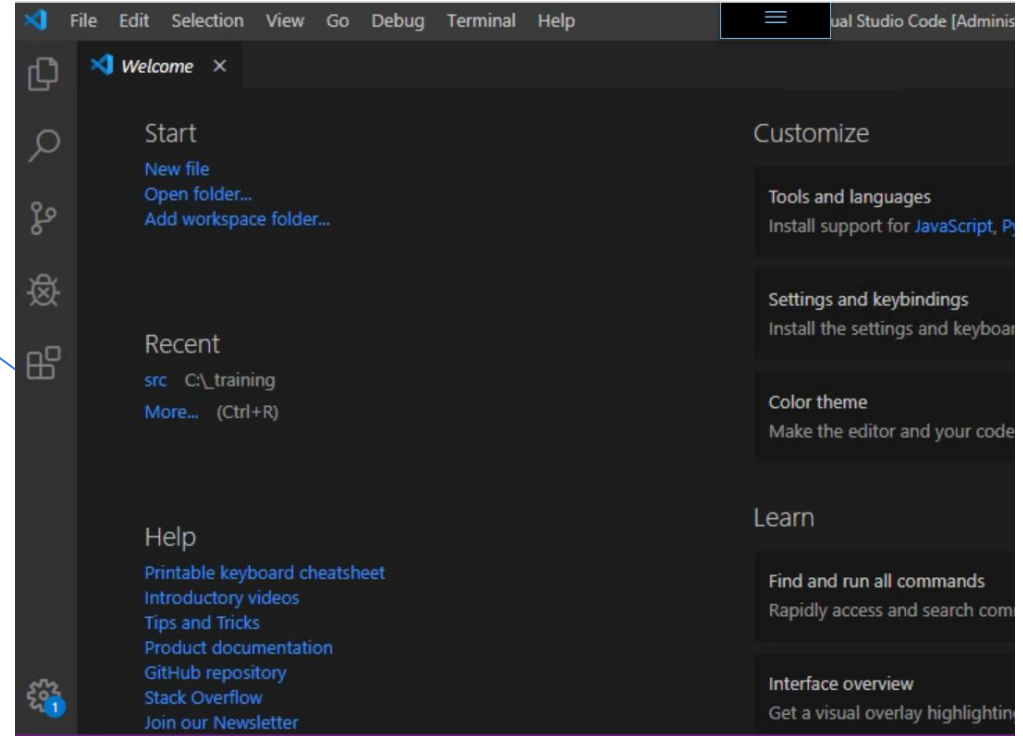
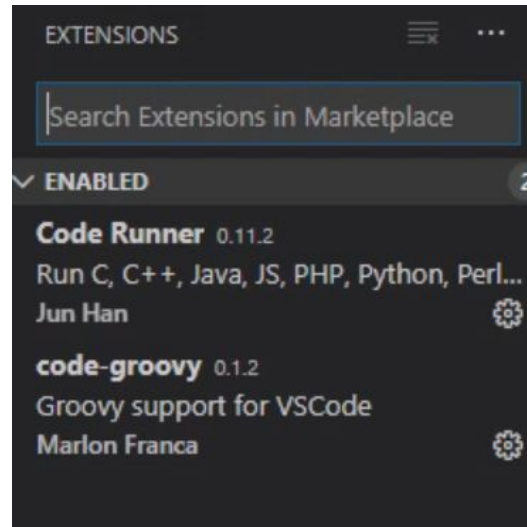


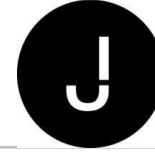
Installation der PlugIns

Add-Ons auswählen

Installation von

- Code Runner
- code-groovy



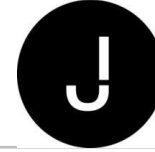


Einführung in Groovy

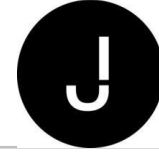
- Eine ganz normale Programmiersprache
 - Operatoren
 - Mathematisch
 - Logisch
 - Variablen
 - Funktionen
- Kontrollstrukturen
 - Schleifen
 - Abfragen
- “Groovy ist eine ganz normale Programmiersprache”

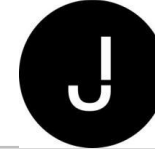
- Eine Objekt-orientierte Sprache
 - Klassen-Hierarchien, Objekt-Erzeugung
 - “Zum Glück ist OOP relativ menschlich”
- Eine funktionale Sprache
 - Eine etwas schwierige formale Syntax
- Groovy ist interoperabel mit der Programmiersprache Java
 - alle Java-Bibliotheken sind in Groovy sofort verfügbar

- Skript-Programmierung
 - Features 1 und nur Teile der Features 2
- Vollständige Anwendungsprogrammierung (z.B. RESTful Webservice mit Datenbankzugriff)
 - Features1 + 2
- Unser Seminar
 - Einsatz in JasperReports durch Verwendung von Expressions
 - Expressions sind Skript-Fragmente



Groovy Programmierung



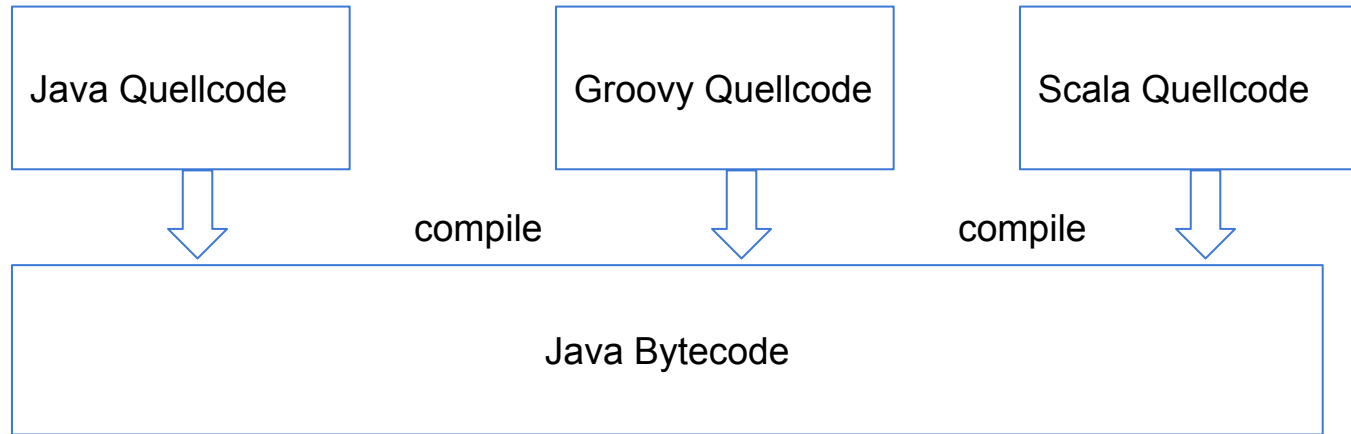


Groovy: Zusammenspiel mit Java



- Programmiersprache
 - Objekt-orientiert
 - Statisch typisiert
 - Ein Java-Compiler übersetzt den Java-Quellcode in den Java Bytecode
- Ausführung dieses ByteCodes übernimmt die so genannte Java Virtual Machine (JVM)
- Java ist eine Spezifikation von Oracle
 - Lang Jahre war die Weiterentwicklung von Java-Features recht langwierig
 - Release-Zyklen 2006 (Java 5), Java 8 2013

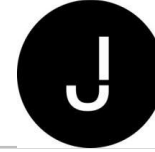
- Groovy-Programme werden ebenfalls als Java ByteCode in der JVM ausgeführt



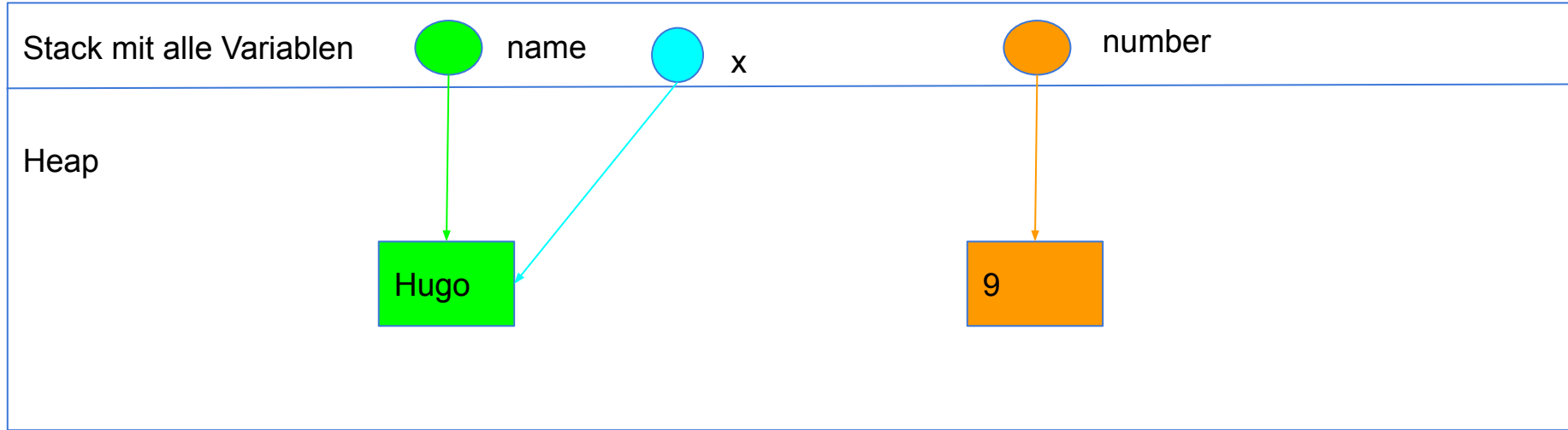
- Aus Sicht der JVM ist die Programmiersprache vollkommen belanglos
- Aus Sicht der Programmiersprachen kann jeder ByteCode auch aus anderen Sprachen jederzeit benutzt werden

- Entstammt der Zeit der langen Java-Release-Zyklen
- Groovy war als agile Spielwiese für Java-Entwickler gedacht
 - Heutzutage spielt das keine sonderlich große Rolle mehr
- Auch heute ist Groovy immer noch im Einsatz, wird weiterentwickelt,
...
 - Verantwortung liegt bei der Apache Community
 - Aktuelle Version 3.0.9
- Heute ein Nischen-Produkt insbesondere als Skript-Sprache in diversen Java-basierten Server-Anwendungen
 - Job-Definitionen in Jenkins
 - Expressions in JasperReports

- Besteht aus drei Teilen
 - Die eigentliche Groovy-Dokumentation
 - Java-Dokumentation
 - Groovy-Erweiterungen von Standard-Java-Bibliotheken



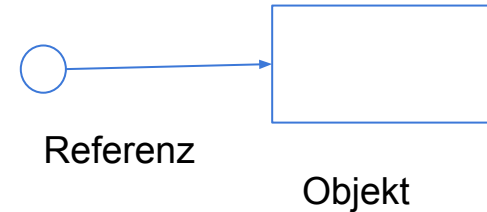
Kurzeinführung OOP



```
def name = "Hugo"
```

```
def number = 9
```

```
def x = name
```

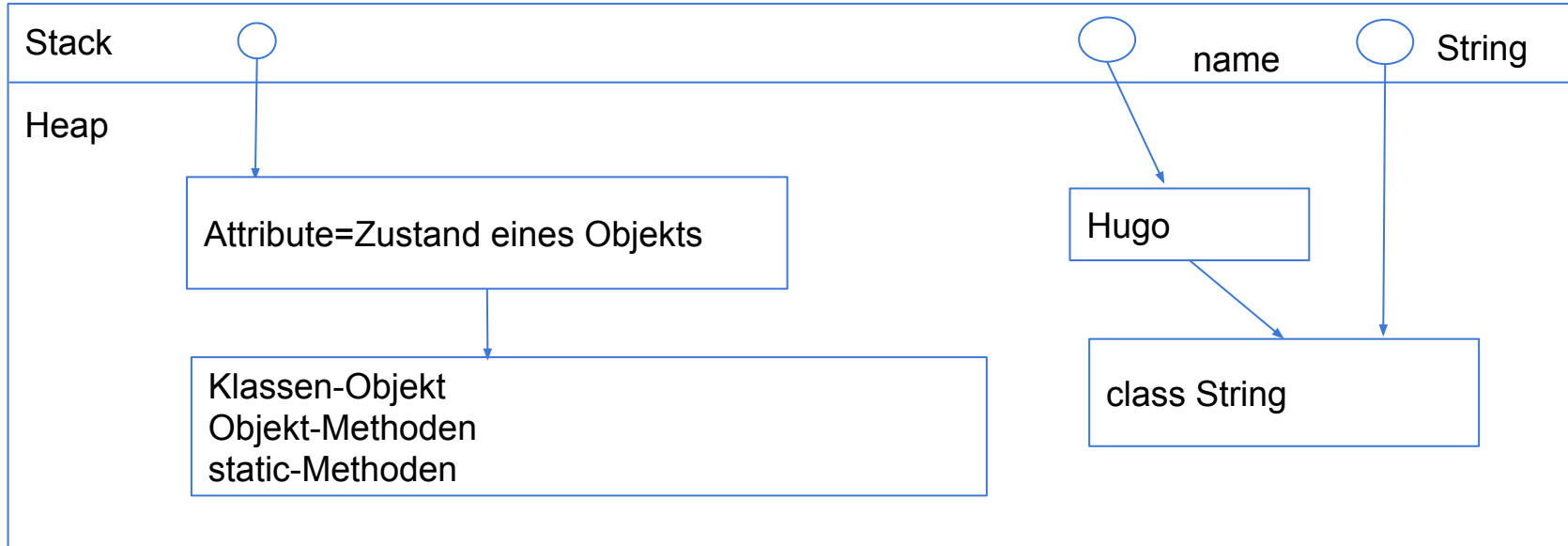


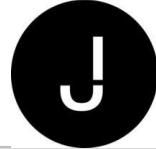
- liegen im Heap
- Können Referenzen auf andere Objekte im Heap enthalten
 - Attribute
 - Funktionen -> Methoden
- Zum Verfolgen einer Referenz wird der Punkt-Operator genutzt
 - `name.length()`
 - `name.startsWith("H")`
- Objekte haben zur Laufzeit immer einen Typ
 - `"Hugo"` -> String
 - `9` -> BigInteger
 - `true, false` -> Boolean
 - `4.2` -> BigDecimal

- Zeichenkettenverarbeitung mit Hilfe der Klasse String
- Mathematische Operationen BigDecimal bzw. BigInteger
- Konversionen
 - Integer.parseInt(string)
 - Method toString()
 - Vorsicht: Hüten Sie sich vor impliziten Konversionen
 - $5 + 2 + \text{"Hugo"} + 42 \rightarrow 7\text{Hugo}42$
- Abfrage

```
if (Bedingung){  
}  
else{  
}
```

- Schreiben Sie eine Anwendung, die mit einer Zeichenkette arbeitet
 - Bestimmung der Länge
 - Ist eine bestimmte Zeichenkette enthalten(contains)
 - ...
- Umwandlung von Zeichenketten in Zahlen
 - `Integer.parseInt("42")`
 - `Integer.parseInt("Hugo")`
- Hinweis:
 - Noch nicht behandelt sind Array, also z.B. "Zerlege eine Zeichenkette"





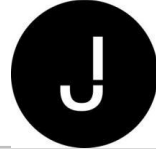
- `def name = "Hugo"`
 - Bisher: `"..."` ist ein Objekt-Literal, hier: String-Literal
- Arbeitsweise eine Literals
 - Es wird ein neues Objekt vom Typ `"String"` erzeugt und im Heap abgelegt
 - Der Rückgabe-Wert ist die Referenz
- Neu: Der Aufruf eines Constructors
 - `def name = new String("Hugo")`
 - `def number = new Integer(9)`

- Bisher: String
 - Problem: Mögliche Kollision
- Nun: Voll qualifizierte Klassennamen
 - `package.Classname`
 - `java.lang.String`
 - “Die Klasse String befindet sich im Package java.lang”
 - Voll qualifizierte Klassennamen müssen in Groovy für einen Teil der Bibliothek nicht angegeben werden
 - `java.lang`
 - `java.util`
 - `java.math`
 - Alle anderen müssen entweder voll qualifiziert angegeben werden oder sie müssen den Klassen-Namen “importieren”
 - `import java.sql.Connection`

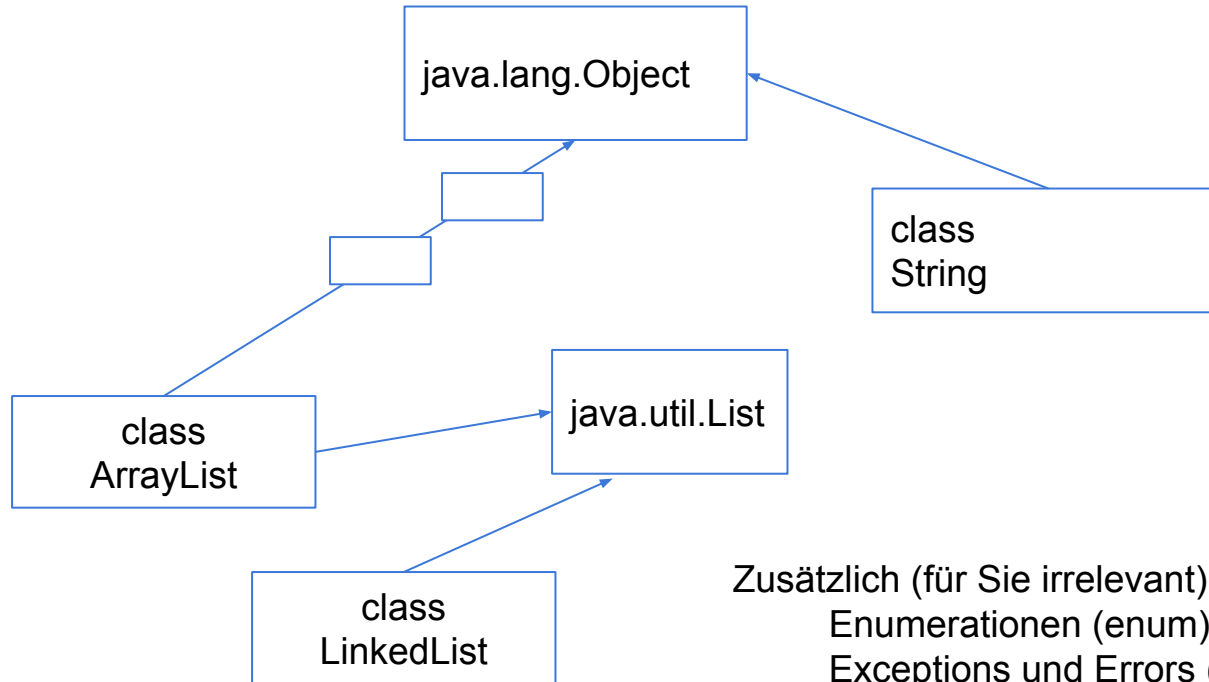


- “”
 - `java.lang.String`
- `true` | `false`
 - `java.lang.Boolean`
- `9`
 - `java.math.BigInteger`
- `4.2`
 - `java.math.BigDecimal`
- `["A", "B", "C"]`
 - `java.util.ArrayList` -> `def names = new ArrayList(); names.add("A")`
- `[81373: "München", 30000: "Berlin"]`
 - `java.util.HashMap` -> `def plz = new HashMap(); plz.put(81373, "München")`

In Groovy eine Erleichterung: Probleme mit Wertebereichen und Rundungsfehler werden damit vermieden



- Verzetteln Sie sich nicht in der Klassen-Dokumentation
 - java.lang
 - String
 - Boolean
 - java.util
 - java.text
 - Formatierte Darstellung von Texten

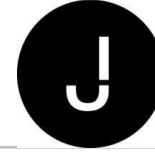


Zusätzlich (für Sie irrelevant)
Enumerationen (enum)
Exceptions und Errors (Fehler-Strukturen)
Annotations

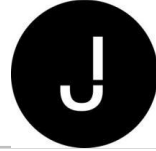


- Bessere, saubere Alternative zu Groovy-Skripten

- Schreiben Sie eine SimpleUtility-Klasse (SimpleUtility.groovy)
 - checkStringContainsSubstring(def string) : Boolean
 - numberOfOccurrences(def string, def substring) : Zahl
 - ...
 - checkListContainsElement(def list, def element) : Boolean
 - numberOfOccurrences(def list, def element) : Zahl
 - ...
- Ergänzen Sie die CollectionsApplication um eine Methode setDemo
 - Ein Set ist eine Menge, die selbstständig beim add auf Duplikate prüft
 - Implementierende Klasse eines set heißt "HashSet"
 - Erzeugen Sie ein Set und prüfen durch hinzufügen von Elementen, dass das Set tatsächlich Duplikate erkennt



Java Bibliotheken

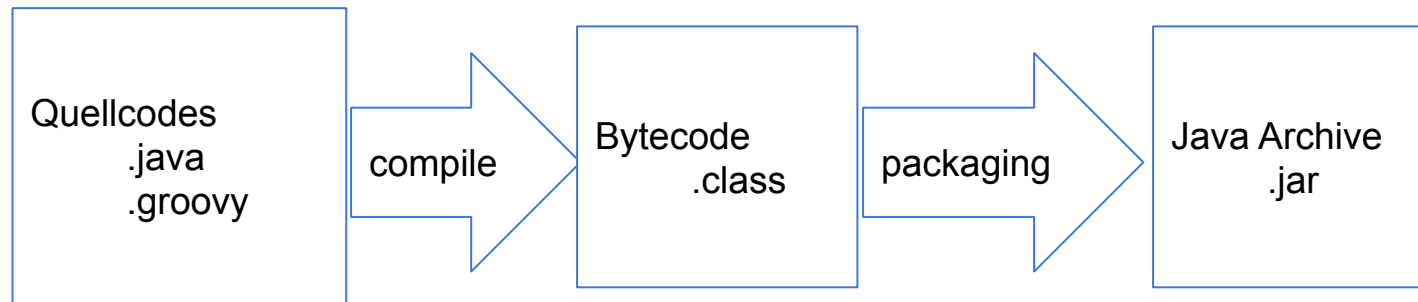


- Das Binärformat, das von einem Compiler erzeugt wird
- Jede Klassendefinition bekommt eine eigene ByteCode-Datei
- Klassen-Verzeichnisse enthalten ByteCode-Dateien
- Eine Menge Klassen-Verzeichnissen: CLASSPATH

- Angabe eines packages
 - `package my.package`
 - Namenskonvention: Umgekehrte Domäne des Unternehmens
 - z.B. `javacream.org` -> `org.javacream`



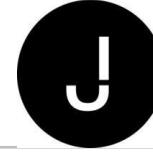
- Gezippte Klassen-Verzeichnisse
 - .jar-Dateien
- Erzeugung
 - Hilfsprogramm des JDK (Java Development Kit)
 - jar
 - Normales ZIP-Programm tut es auch



- Dieser Prozess wird für große Anwendungen durch ein Build-Werkzeug (Apache ANT, Maven, Gradle) durchgeführt
- Eine automatisiertes Bauen erfolgt mit einer Build-Machine, z.B. Jenkins
- Für uns: Eher oversized...

- Ordnen Sie ihre Klassen einem Paket zu
- `groovyc -d classes *.groovy`
- Mit einem ZIP-Programm ihrer Wahl wird daraus eine JAR erzeugt
 - CHECK
 - Der Inhalt der JAR-Datei muss mit dem ersten package übereinstimmen
- Ziel
 - Benutzerdefinierte Logik in JasperReports wird als JAR realisiert

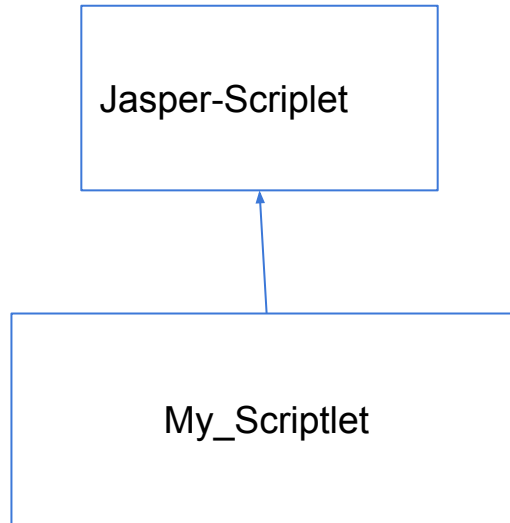
```
java -classpath classes;"C:\Program Files  
(x86)\Groovy\lib\groovy-4.0.0-rc-2.jar" org.javacream.util.SimpleUtility
```



JasperReports

- 3 Sprach-Alternativen
 - Java
 - JavaScript
 - Groovy
- Beispiel
 - `$P{A} + $P{B}`
 - `$P{...}` -> Zugriff auf einen deklarierten Report-Parameter
- Expressions sind immer als kleine Einzeiler aufzufassen

- Eine Klasse, die für den Report eine Reihe von Methoden/Funktionen zur Verfügung stellt



- Erstellen Sie die Jar-Datei mit der Utility
- Öffnen von Jasper Studio
- Projekt neu oder auswählen
- Neuer Report
 - Sprache groovy
 - Import auf ihre Klasse
 - Hinzufügen eines simplen Text-Feldes
 - Expression unter Benutzung einer Methode der Utility
- Optional
 - Definieren Sie Parameter, z.B. name und substring
 - Text-Feld: Anzeige “Wie oft ist substring im name vorhanden?”