Einbinden von Klassen in Groovy

In verschiedenen Situationen kommt es vor, dass bei einem Groovy Programm auf eine bereits vorhandene Klasse zurückgegriffen werden muss. Oft wurde diese Klasse in eine Datei geschrieben, die den selben Namen hat, wie die Klasse. Dieses ist bei Java-Klassen, die auch in Groovy genutzt werden können, immer der Fall. Um nicht die Programmdatei deutlich zu vergrößern und damit unübersichtlicher zu machen, kann man diese Klassen auch auf andere Art einbinden. Wenn es sich dabei um eine Java-Datei handelt muss diese zuerst kompiliert werden. Dieses geschieht durch den Aufruf des Java-Kompilers mit Angabe der zu übersetzenden Java-Datei:

```
javac Klassenname.java
```

Alternativ kann auch die Java-Datei in GroovyConsole geladen werden und dann dort kompiliert werden (Tastenkürzel Strg+L). Klassen, die sich in einer Groovy-Datei befinden müssen nicht zuvor kompiliert werden.

Bei der Benutzung von GroovyConsole ist zu beachten, dass es entweder von dem Verzeichnis aus gestartet wird, in dem sich die Klassendateien befinden. Ist dieses nicht der Fall gibt es zwei Möglichkeiten: Man startet GroovyConsole mit einem Parameter, der den Klassenpfad angibt:

```
groovyConsole -cp <Verzeichnispfad>
```

Diese Möglichkeit kann auch genutzt werden, wenn die GroovyShell genutzt wird. Die zweite Möglichkeit besteht darin, dass man in der Oberfläche von GroovyConsole den Klassenpfad manuell wählt.

Es kann vorkommen, dass eine Klasse eingebunden werden soll, die den gleichen Namen trägt wie eine Standardklasse von Java. In diesem Fall muss zusätzlich noch direkt am Anfang der Groovy-Datei folgendes angegeben werden:

```
import <Klassennamen>|
```

Dieses kommt z. B. dann vor, wenn fertige Implementierungen der Klassen List oder Queue genutzt werden sollen. Dieses kommt bei den Klassen für das Zentralabitur in Nordrhein-Westfalen vor.