

Aufgabe 5: Packages (Programmorganisation) – Lösung

- a) Weshalb wird für die Anweisung `System.out.println("Hallo");` keine `import`-Anweisung benötigt?
Die Klasse "System" gehört zum Package "java.lang". Dieses Package wird automatisch, auch ohne `import`-Anweisung, importiert.
- b) Was muss geändert werden, wenn das Programm von Übung 2, Aufgabe 1 ("TextEinAusgabe") ohne `import`-Anweisung auskommen soll?
An Stelle von "BufferedReader" ist jeweils "java.io.BufferedReader" zu schreiben. Entsprechend bei "InputStreamReader" (neu: "java.io.InputStreamReader") und "IOException" (→ "java.io.IOException").
- c) Wie lautet die `import`-Anweisung, um in der Klasse "August" (aus dem Package "aaa") die Klasse 'Xaver' aus dem Default-Package zu verwenden?
Eine solche `import`-Anweisung gibt es nicht. Es ist nicht möglich Klassen im Default-Package in Klassen zu verwenden, die nicht im selben Default-Package definiert sind!
- d) Was kann über die Filestruktur (Verzeichnisse) ausgesagt werden, wenn die Datei "Julius.java" die Anweisung "package romeo.julia;" enthält?
Die Datei "Julius.java" ist im Unterverzeichnis "julia" untergebracht, und dieses Unterverzeichnis ist Teil des Verzeichnisses "romeo" (Pfad: "romeo/julia/Julius.java").

Aufgabe 7: Packages (Programmorganisation) – Lösung

Gegeben ist das folgende UML-Klassendiagramm, welches die für die Programmorganisation bei Java wichtigen Begriffe darstellt.

