

Softwarequalität und Dokumentation

1. Regeln für guten Java-Code

Für guten Java-Code gibt es festgelegte Regeln, die es beim Programmieren zu beachten gilt, um Übersichtlichkeit und Lesbarkeit zu gewährleisten:

- **Verbalisierung:** aussagekräftige, selbsterklärende Namensgebung sowie geeignete Kommentare.
- **Adäquate Datenhaltung:** Nutzung vom umfangreichen Repertoire an Basistypen, geeigneten Typkonstruktoren sowie benutzerdefinierbaren Typen.
- **Verfeinerung:** Durch die Benutzung von Abstraktionsebenen wird der Quellcode in einer Art Hierarchie verfeinert.
- **Integrierte Dokumentation:** Kurzbeschreibung des Programms (empfehlenswert ist auch ein Programmvorspann, welcher den Programmnamen, die jeweilige Aufgabe, Zeit- und Speicherkomplexität, Name des Programmautors sowie Versionsnummer und Datum enthält), Verwaltungsinformationen und Kommentierung des Quellcodes.
- Blockklammern sollten jeweils für sich in einer neuen Zeile stehen. Des öfteren werden sie gebraucht, um einzeilige Kommentare dahinter zu vermerken.
- Zur besseren Lesbarkeit bei den Parameterklammern (,) sollte man nach der öffnenden und vor der schließenden Klammer ein Leerzeichen einfügen.
- Man sollte anstatt der Tabs 4 Leerzeichen verwenden (Tabs werden bei manchen Übersetzungen von Code nicht beachtet).
- Es sollten Linienflüsse innerhalb von zusammenhängenden Namen anstatt von Leerzeichen verwendet werden.
- Jede Klasse sollte in die automatisierte Javadokumentation eingebunden werden. Dies dient zur leichteren Einarbeitung von fremden Programmierern in den derzeitigen Code und man sieht leicht, was man vergessen hat.
- Javadockommentare für Variablen sollten nicht mehr als eine einzige Zeile umfassen.
- Importstatements müssen voll ausgeschrieben und nicht mit einem Sternchen abgekürzt werden.
- Wenn viele Imports benötigt werden, müssen diese alphabetisch und nach Produkten geordnet sein.
- Der IF-Teil einer IF-Anweisung sollte nicht zum Testen eines Wahrheitswertes einer Variable verwendet werden.
- Wenn man eine Methode einer fremden Klasse braucht, sollte man den Pfad zu dieser Methode bei den Imports hinzufügen, anstatt ihn mitten im Code auszuschreiben.
- Iteratoren, die in Schleifen genutzt werden, sollten auch im Schleifenkopf initialisiert werden.

2. Java-Doc – automatisierte Dokumentation in Java

Verfahren:

Javadoc ist ein automatisiertes Dokumentationskonzept, welches Informationen über den geschriebenen Code sammelt. Mit Hilfe von Javadoc-Kommentaren, welche mit `/**` beginnen und mit `*/` enden, die der Implementierer schreiben muss, werden diese Informationen bereitgestellt. Javadoc sucht sich dann diese Kommentare und weißt die Informationen denen in Tags angegebenen Themen zu (z.B. `@author: Name`).

Tags:

Tags sind Themen von Javadoc. Javadoc gibt die Themen samt Informationen normalerweise in HTML und angebrachter Form heraus. Es gibt auch Formen wo man sie in PDF und anderen Dateitypen anzeigen kann. In den Javadoc-Kommentaren werden Tags mit einem `@` angegeben. Dann folgt der Name des Themas und dann die Informationen dazu. Die Liste der Themen kann man auch selber erweitern.