

Beispiele für Verstöße gegen das LSP in der Java API

In der Java API gibt es einige Beispiele, bei denen das Liskov Substitution Principle (LSP) verletzt wurde. Diese Verletzungen führen oft zu Verwirrung und Problemen bei der Verwendung der betreffenden Klassen. Hier sind ein paar bekannte Beispiele:

1. `java.util.Stack` erbt von `java.util.Vector`: Die Klasse `Stack` ist in Java als eine Subklasse von `Vector` implementiert. Dies verstößt gegen das LSP, da `Vector` eine Liste repräsentiert, die Zugriff auf jedes Element an jeder Position erlaubt, während ein `Stack` eine Last-In-First-Out (LIFO) Datenstruktur ist, die im Normalfall keinen solchen Zugriff bieten sollte. Die Vererbung von `Vector` erlaubt Operationen auf einem `Stack`, die nicht mit der grundlegenden Stack-Semantik übereinstimmen.

2. `java.util.Properties` erbt von `java.util.Hashtable`: Ähnlich wie bei `Stack` und `Vector`, hat die Klasse `Properties` Methoden von `Hashtable` geerbt, die nicht unbedingt für den Zweck von `Properties` geeignet sind. `Properties` ist für Schlüssel-Wert-Paare von Strings gedacht, aber durch die Vererbung von `Hashtable` können nicht-String-Objekte als Schlüssel oder Werte hinzugefügt werden, was gegen das LSP verstößt.

3. `java.sql.Timestamp` erbt von `java.util.Date`: `Timestamp` erbt von `Date`, aber das Überschreiben bestimmter Methoden führt zu unterschiedlichem Verhalten. Zum Beispiel ignoriert `Timestamp.equals(Object obj)` die nanosekundengenauen Unterschiede, während `Date.equals(Object obj)` dies nicht tut. Dies kann zu unerwartetem Verhalten führen, wenn `Timestamp`-Objekte als `Date`-Objekte verwendet werden.

Diese Beispiele zeigen, wie Vererbung in der Java API manchmal nicht im Einklang mit dem LSP steht. Es ist wichtig, diese Fälle zu kennen, um Fallstricke bei der Verwendung dieser Klassen zu vermeiden. In neueren Java-Versionen gibt es Bestrebungen, solche Design-Probleme durch alternative Klassen und Methoden zu beheben, aber aus Gründen der Abwärtskompatibilität bleiben die ursprünglichen Klassen oft unverändert.