Thursday, 24. October 2024

Proxy Design-Pattern in Java

Charakterisierung

Proxy ist ein strukturelles Entwurfsmuster, das ein Objekt bereitstellt, das als Ersatz für ein echtes, von einem Client verwendetes Dienstobjekt fungiert.
Das Proxy-Objekt ist eine **Referenz** zum tatsächlichen Objekt, nicht eine Kopie.
Es agiert als Stellvertreter oder Platzhalter, der den Zugriff auf das Originalobjekt kontrolliert. Der Proxy leitet Aufrufe an das echte Objekt weiter oder führt zusätzliche Logik aus, ohne das echte Objekt zu duplizieren.
Das bedeutet, dass Änderungen, die am Originalobjekt vorgenommen werden, über den Pr oxy sichtbar sind und umgekehrt.

Nutzen

- Kontrolliert den Zugriff auf das ursprüngliche Objekt.
- Ermöglicht die Steuerung des Zugriffs auf das Objekt basierend auf bestimmten Bedingu ngen.
- Führt zusätzliche Funktionen aus z.B. Lazy Initialization
  - Lazy Loading kann die Instanziierung von „teuren“ Objekten verzögern, bis sie tatsächlich benötigt werden.

Arten von Proxies

- **Virtual Proxy**: Verzögert die Instanziierung des „teuren“ Objekts, bis es benötigt wird
- **Remote Proxy**: Verwaltet den Zugriff auf ein Objekt, das sich in einem anderen Adress raum befindet
- **Protection Proxy**: Kontrolliert den Zugriff basierend auf den Berechtigungen des aufrufenden Codes
- **Smart Proxy**: Führt zusätzliche Aktionen aus, wenn ein Objekt referenziert wird

Quellen

Oracle Java Documentation

https://refactoring.guru/

https://entwickler.de/java/proxy-oder-delegator-das-ist-hier-die-frage#:~:text=Ein%20Proxy%20ist%20ein%20Surrogat,um%20die%20Funktionalit%C3%A4t%20zu%20erweitern

GeeksforGeeks und Baeldung