

Handout: Zeiger und Referenzen in Java

Einführung

In Java werden Zeiger und Referenzen nicht explizit vom Programmierer verwaltet, wie es in Sprachen wie C oder C++ der Fall ist. Java nutzt Referenzen, um auf Objekte im Speicher zu zeigen, und der **Garbage Collector** übernimmt die Speicherverwaltung automatisch.

Zeiger und Referenzen in Java

Unterschiede zwischen Zeigern und Referenzen

Eigenschaft	Zeiger (C/C++)	Referenzen (Java)
Direkter Zugriff	Zeigt direkt auf Speicheradressen	Verweist auf ein Objekt, keine Adressmanipulation
Speicherverwaltung	Manuell, durch den Entwickler	Automatisch, durch den Garbage Collector
Null-Sicherheit	Kann auf null oder ungültige Adressen zeigen	Kann <code>null</code> sein, aber ungültige Adressen sind nicht möglich
Gefährlichkeit	Kann Speicherprobleme oder Sicherheitslücken verursachen	Sicherer, da keine Adressarithmetik möglich ist

Referenzen in Java

Referenzen in Java sind "Zeiger-ähnlich", erlauben aber keine direkten Speicheroperationen. Ein Referenztyp speichert die Adresse eines Objekts im Speicher, auf das verwiesen wird.

Beispiel: Referenzen

```
public class ReferenzBeispiel {
    public static void main(String[] args) {
        String original = "Hallo, Welt!";
        String kopie = original;

        System.out.println("Original: " + original);
        System.out.println("Kopie: " + kopie);

        // Ändere Referenz
        kopie = "Neue Nachricht!";
        System.out.println("Nach der Änderung:");
        System.out.println("Original: " + original);
        System.out.println("Kopie: " + kopie);
    }
}
```

Erklärung:

- Beide Variablen `original` und `kopie` zeigen anfangs auf dasselbe Objekt.
- Nach der Änderung verweist `kopie` auf ein neues Objekt, während `original` unverändert bleibt.