

Zusammenfassung: Arrays & for-Schleifen

◇ 1. Arrays – Sammlungen fester Grösse

- Ein **Array** speichert mehrere Werte gleicher Art.
- Die Grösse wird beim Erzeugen festgelegt und kann später nicht mehr geändert werden.
- Besonders geeignet, wenn:
 - die Anzahl Elemente vorher bekannt ist
 - viele Werte von **primitiven Typen** gespeichert werden sollen
 - maximale Effizienz wichtig ist

Beispiel:

```
int[] zahlen = new int[5];    // Array mit 5 Elementen
zahlen[0] = 10;
zahlen[1] = 20;
```

◇ 2. Zugriff über Indices

- **Array**-Indizes beginnen bei 0.
- Man greift mit **array[index]** auf Elemente zu.

Beispiel:

```
int x = zahlen[1]; // ergibt 20
```

◇ 3. for-Schleife (klassische **for**-Schleife)

Ideal, wenn man:

- einen **Index** braucht,
- über ein **Array** iteriert,
- eine bestimmte Anzahl Durchläufe hat.

Beispiel:

```
for (int i = 0; i < zahlen.length; i++) {  
    System.out.println(zahlen[i]);  
}
```

◇ 4. Konditionaloperator **?:**

Eine Kurzform von **if-else**, auch **Ternary Operator** genannt.

Syntax:

```
bedingung ? wertWennTrue : wertWennFalse
```

Beispiel:

```
int a = 5;  
int b = (a > 3) ? 10 : 0;    // b = 10
```

◇ 5. Zuordnungstabelle (Mapping mit **Arrays**)

Arrays können wie einfache **Tabellen** oder **Maps** genutzt werden, wenn die Zuordnung **indexbasiert** ist.

Beispiel:

Ein Array, das für jede Note eine Beschreibung speichert:

```
String[] notenText = { "ungenuegend", "genuegend", "gut", "sehr gut" };  
System.out.println(notenText[2]);    // "gut"
```

📌 Komplettes Lernbeispiel: Arbeiten mit Arrays

```
public class Beispiel {  
    public static void main(String[] args) {  
        int[] werte = { 3, 5, 7, 9 };  
  
        // Ausgabe mit for-Schleife  
        for (int i = 0; i < werte.length; i++) {  
            System.out.println("Index " + i + ": " + werte[i]);  
        }  
  
        // Konditionaloperator  
        int max = (werte[0] > werte[1]) ? werte[0] : werte[1];  
        System.out.println("Max: " + max);  
    }  
}
```

Dieses Beispiel zeigt:

- Array-Erstellung
- Zugriff per Index
- klassische for-Schleife
- Ternary Operator ?:

🧠 Merksätze

Ein Array hat immer eine fixe Grösse.

Indizes starten bei 0.

Die klassische for-Schleife ist der Standard für Arrays.

Viele Aufgaben lassen sich heute auch mit ArrayList lösen – ein Array lohnt sich vor allem bei festen Grössen oder **primitiven Typen**.

Der Konditionaloperator ist eine Kurzform fürs if-else.