



Primitive Arrays

openHPI-Java-Team
Hasso-Plattner-Institut



Arrays - Erklärung

- Container für gleichartige „Dinge“ → Werte gleichen Datentyps
- Gekennzeichnet durch []
- `int[] numbers` ist ein Array von Integer → nur Integer können dort „reingepackt“ werden
- Garage (im weitesten Sinne) Array für Autos
 - Stellplätze durchnummeriert
 - Zuweisung und Zugriff über eindeutige Stellplatznummer
 - Definierte Anzahl von Stellplätzen (nicht unendlich groß)

Arrays: Deklarieren und initialisieren (1/2)



```
1  int[] numbers;  
2  numbers = new int[5];  
3  
4  
5  
6
```

- Deklaration: „Da gibt es jetzt dieses Objekt mit diesem Namen“
- Initialisierung: „Das Objekt kriegt jetzt auch einen Wert“
- Kann auch gleichzeitig geschehen:

```
int[] numbers = new int[5];
```
- `<Datentyp>[] <arrayname> = new <gleicher Datentyp>[<Größe>];`

Arrays: Deklarieren und initialisieren (2/2)



```
1  int[] numbers = new int[5];  
2  numbers[0] = 6;  
3  numbers[1] = 33;  
4  numbers[2] = 9;  
5  numbers[3] = 0;  
6  numbers[4] = 503;
```

- Kann auch gleichzeitig mit Deklaration geschehen
`int[] numbers = {6,33,9,0,503};`
- Initialisierung mit `new` setzt alle Felder des Arrays auf Defaultwerte
- **Achtung!** In Java fängt man mit der 0 an zu zählen
- **Achtung!** In Java können primitive Arrays **nicht** ihre Größe verändern
→ nicht dynamisch weitere Elemente anfügen



```
           0           1           2
1  String[] words = {"Hallo", "Huhu", "Na?"};
2
3  for( int i = 0; i < words.length; i++){
4      System.out.println(words[i]);
5  }
```

- Anzahl der Elemente eines Arrays mit `<Arrayname>.length` abfragen

Ausgabe:

Hallo ← Ausgabe für i = 0
Huhu ← Ausgabe für i = 1
Na? ← Ausgabe für i = 2