

# Einschub: Autoboxing

Können wir `List<int>` verwenden?

Nein, denn `int` ist keine Klasse.

Java ist keine *reine* objektorientierte Sprache: Nicht *alles* ist ein Objekt.

Grund: Effizienz.

Lösung in Java: Wrapper-Klassen.

Primitiver Typ	Wrapper-Klasse
<code>int</code>	<code>Integer</code>
<code>double</code>	<code>Double</code>
<code>boolean</code>	<code>Boolean</code>

# Autoboxing

```
Set<Integer> zahlen = new HashSet<Integer>();  
zahlen.add(34);  
zahlen.add(23);  
zahlen.add(117);  
zahlen.add(3);  
zahlen.add(-345);  
zahlen.add(23);  
zahlen.add(35);  
for(int k : zahlen) {  
    System.out.println(k * k);  
}
```



# Autoboxing

```
Map<Short, String> map;  
map = new HashMap<Short, String>();  
short n = 17;  
map.put(n, "Siebzehn");  
System.out.println(map.get(17));
```



Vorsicht! Es gibt merkwürdige Effekte und schwer zu findende Fehler.

Tip: Nur *Integer*, *Double*, *Boolean* verwenden.

Nur mit Vorsicht *Short*, *Long*, etc.

# Container-Klassen

Klassen die viele Elemente zusammenfassen, sind sehr nützlich.

Die wichtigsten solchen Klassen modellieren Mengen, Listen, Arrays und Abbildungen.

Interfaces in der Java-Bibliothek:

- *Set* $\langle E \rangle$ , Mengen
- *List* $\langle E \rangle$ , Listen und Arrays
- *Map* $\langle K, E \rangle$ , Abbildungen und Wörterbücher

Um auf sie zuzugreifen, müssen wir *java.util.Set* schreiben, oder am Anfang der Datei:

```
import java.util.Set; // importiert Set
import java.util.*; // importiert alles aus java.util
```

# Set

*Set* $\langle E \rangle$  ist nur ein Interface.

Implementierungen davon sind z.B. *HashSet* $\langle E \rangle$  und *TreeSet* $\langle E \rangle$ .

Um *Set* $\langle E \rangle$  korrekt verwenden zu können, muß die Klasse *E* gewisse Anforderungen erfüllen.

- **boolean** *equals*(*Object* *o*), ist *o* das gleiche?
- **int** *hashCode*(), berechnet einen Hashwert.
- **int** *compareTo*(*E* *e*), vergleiche und gib  $<0$ ,  $0$ ,  $>0$  zurück

Jede Klasse erbt *equals* und *hashCode* von *Object*.

*compareTo* ist im Interface *Comparable* $\langle E \rangle$  definiert.