

Unter einer **Klasse** (auch *Objektyp* genannt) versteht man in der objektorientierten Programmierung ein abstraktes Modell bzw. einen *Bauplan* für eine Reihe von ähnlichen Objekten. Die Klasse dient als Bauplan für die Abbildung von realen Objekten in Softwareobjekte und beschreibt Attribute (Eigenschaften) und Methoden (Verhaltensweisen) der zu erzeugenden Objekte. Verallgemeinernd sagt man, dass eine Klasse dem Datentyp eines Objekts entspricht.

Quelle: [https://de.wikipedia.org/wiki/Klasse\\_\(Objektorientierung\)](https://de.wikipedia.org/wiki/Klasse_(Objektorientierung))

### Beschreibung:

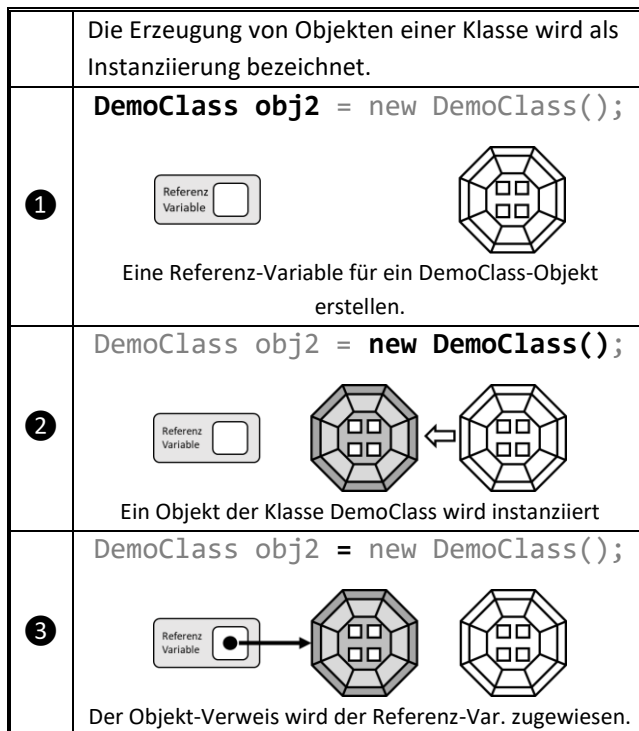
- ③ und ④ deklarieren die Attribute der Klasse.
- von ⑦ bis ⑩ wird eine Methode der Klasse definiert.
- Mit ⑤ wird eine Referenz-Variable, welche auf ein **DemoClass**-Objekt verweisen kann, deklariert.
- In ⑥ wird das Objekt instanziiert und der Verweis darauf in der Referenz-Variable gespeichert.
- ⑧ führt diesen Vorgang für ein weiteres Objekt in einem Schritt (Deklaration, Erzeugung und Zuweisung) aus.

### Hinweise:

- Attribute mit primitiven Datentypen werden standardmäßig initialisiert (int=0, double=0.0, boolean=false und char=' ').
- Für jede Klasse sollte eine eigene Datei (Dateiname = Klassenname) erzeugt werden.

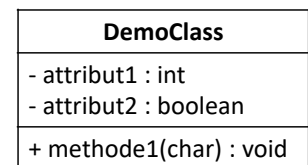
exemplarischer Quellcode einer Klassendatei	
①	<code>public class DemoClass {</code>
②	
③	<code>private Datentyp attribut1;</code>
④	<code>private Datentyp attribut2;</code>
⑤	<code>:</code>
⑥	<code>public Datentyp methode1( ... ){</code>
⑦	<code>// Anweisungen</code>
⑧	<code>return Datentyp;</code>
⑨	<code>}</code>
⑩	<code>:</code>
⑪	<code>}</code>
	DemoClass.java

exemplarischer Quellcode einer Start-Klasse	
①	<code>public class StartClass {</code>
②	
③	<code>public static void main(String[] args){</code>
④	
⑤	<code>DemoClass obj1;</code>
⑥	<code>obj1 = new DemoClass();</code>
⑦	
⑧	<code>DemoClass obj2 = new DemoClass();</code>
⑨	<code>:</code>
⑩	<code>}</code>
⑪	<code>}</code>
	StartClass.java

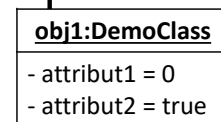


### Die Darstellung in den UML-Klassen / Objekt Diagramm würde folgendermaßen aussehen.

Klassendiagramm mit Attributen und Methoden der Klasse



Objektdiagramm des obj1 der Klasse DemoClass



Methoden werden im Objekt Diagramm dargestellt. Attributs-Werte sind beschrieben

