

Spickzettel Java

Blöcke	Blöcke werden in geschweiften Klammern notiert. <pre>{ //... }</pre>
Kommentare	// einzeliger Kommentar /* mehrzeiliger Kommentar */
Primitive Datentypen	boolean: Wahrheitswert int: Ganzzahl float: Kommazahl (notiert in der Form 1.0f) double: Kommazahl char: Zeichen, z. B. 'C' Weitere: byte, short, long
Zeichenketten	String: Zeichenkette, z. B. "Hallo"
Vereinbarung und Zuweisung bei primitiven Datentypen	int x; x = 6;
Klasse mit Deklaration von Attributen und Konstruktor	class Haus { String strasse; int hausnummer; public Haus(String strasseNeu, int hNrNeu) { strasse = strasseNeu; hausnummer = hNrNeu; } }
Vererbung	class Reihenhaus extends Haus Der Aufruf des Oberklassenkonstruktors erfolgt mit super(...), anderer Methoden der Oberklasse mit super.Methodenname(...)
Methode überschreiben	@Override void AktionAusführen(){ //... Dabei muss der Methodenkopf identisch wie in der Oberklasse sein.
Deklaration eines Referenzattributs mit Objekterzeugung	Haus h1; h1 = new Haus("Schulstr.", 8);
Referenz auf das ausführende Objekt	this
Leere Referenz	null
Methode ohne Rückgabewert	void RückwärtsGehen(int schritte) { Gehen(-schritte); }

Methode mit Rückgabewert	<pre>int Quadrieren(int zahl) { return zahl*zahl; }</pre>
Vergleichsoperatoren	<pre>== ist gleich != ungleich > größer >= größer gleich < kleiner <= kleiner gleich</pre>
Logische Operatoren	<pre>! nicht && und oder</pre>
Bedingte Anweisung (bei einseitiger bedingter Anweisung entfällt der else-Teil)	<pre>if(farbe == "rot") { anzahlRot = anzahlRot + 1; } else { //...</pre>
Mehrfachauswahl	<pre>switch(farbe) { case "rot": anzahlRot = anzahlRot + 1; break; case "blau": anzahlBlau = anzahlBlau + 1; break; default: System.out.println("Farbe unbekannt"); }</pre>
Zählwiederholung	<pre>for(int i = 0; i < anzahlRechtecke; i=i+1) { //zu wiederholende Sequenz }</pre>
Wiederholung mit Anfangsbedingung	<pre>while(!Berührt()) { Gehen(10); }</pre>
Konsolenausgabe	<pre>System.out.println(...)</pre>