



Deep Dive Java

openHPI-Java-Team
Hasso-Plattner-Institut



```
1  class Story{
2      String title = "Finde Paco";
3      public static void main(String[] args){
4          Robot ourLittleRobot = new Robot();
5      }
6  }
```

Konventionen

- Klassennamen werden groß geschrieben
- Objekte, Attribute und Methodennamen werden klein geschrieben (Bezeichner)
- CamelCase
- Sinnvolle Bezeichner wählen



print() vs println()

```
1 class HelloPaco{
2     public static void main(String[] args){
3         System.out.print("Hallo ");
4         System.out.print("Paco");
5         System.out.println();
6     }
7 }
```

print()

- Kein impliziter Zeilenumbruch

println()

- Impliziter Zeilenumbruch

print() vs println()



```
1 class HelloPaco{
2     public static void main(String[] args){
3         System.out.print("Hallo ");
4         System.out.print("Paco");
5         System.out.println();
6     }
7 }
```

Ausgabe:

Hallo Paco



```
1 class FooBar{  
2     String text = "Hallo Paco";  
3     char cedille = 'ç';  
4     //methods here  
5 }
```

Anführungszeichen

- chars werden in einfache Anführungszeichen (' ') geschrieben
- Strings werden in doppelte Anführungszeichen (" ") geschrieben
- **Achtung!** bei Copy und Paste von Anführungszeichen



Name	Art	Beispiele	Komplex/ primitiv
String	Text, Zeichenkette	String text; String name = "Robin";	komplex
char	Buchstabe, Zeichen	char country = 'd'; char cedille = 'ç';	primitiv
double	Kommazahl	double speed = 98.7;	primitiv
int	Ganzzahl	int age = 2; int truth = -42;	primitiv
Robot	Roboter	Robot robin = new Robot(); Robot robby = new Robot();	komplex



```
1 void stringManipulation(){
2     String output = "Hallo Paco";
3     System.out.println(output.toUpperCase());
4     System.out.println(output.charAt(7));
5     System.out.println(output.reverse());
6 }
```

- `toUpperCase()` übersetzt einen String komplett in Großbuchstaben
- `charAt(x)` gibt einem einen char an der Position x zurück
 - Das erste Zeichen der Zeichenkette wird mit 0 indexiert

Ausgabe:

HALLO PACO

a

???



```
1 void calculate(String[] args) {  
2     double x = 42.1;  
3     double y = Math.floor(x);  
4     System.out.println(y);  
5 }
```

- Math stellt Mathematik-Funktionen bereit
 - z.B. floor(), ceil(), pow(2)
- Andere Bibliotheken muss man importieren



- public
- static
- void
- class
- new
- return
- int
- char
- float
- u.v.m.



Ausgabe von Text
auf dem Bildschirm

OOP

Klassen

Objekte

Methoden

Attribute

Variablen

