

# Java Funktionsreferenz

## Grundstruktur

```
public class HalloWelt {  
    public static void main(String[] args) {  
        System.out.println("Hallo Welt!");  
    }  
}
```

## Variablen & Datentypen

```
int ganzezahl = 5;  
float fliesskomma = 2.3f;  
double doppeltgenaue_fliesskomma = 2.3333;  
char buchstabe = 'a';  
String text = "Dies ist ein Text!";  
boolean fertig = false;
```

## Ausgabe

```
System.out.println("Text mit Zeilenumbruch");  
System.out.print("Text ohne Umbruch");  
  
String EinWort = "Text";  
System.out.printf("Formatierter %s ohne Umbruch", EinWort);
```

## Eingabe

Erfassen der Eingabe mit Scanner

```
import java.util.Scanner;  
  
Scanner sc = new Scanner(System.in);  
  
System.out.print("Name: ");  
String name = sc.nextLine();  
System.out.println("Hallo, " + name);
```

# Arrays

```
int[] zahlen = {1, 2, 3, 4};  
System.out.println(zahlen[2]); // Ausgabe: 3
```

# Kontrollstruktur

## If/Else

```
if (zahl > 5) {  
    System.out.println("Groesser als 5");  
} else {  
    System.out.println("Kleiner oder gleich 5");  
}
```

## Switch/Case

```
switch (tag) {  
    case 1:  
        System.out.println("Montag");  
        break;  
  
    case 2:  
        System.out.println("Dienstag");  
        break;  
  
    default:  
        System.out.println("Anderer Tag");  
}
```

# Schleifen

## for

```
for (int i = 0; i < 5; i++) {  
    System.out.println(i);  
}
```

## while

```
while (x < 10) {  
    x++;  
}
```

## do-while

```
do {  
    System.out.println("Mindestens einmal!");  
} while (x < 5);
```

## Eigene Methoden/ Funktionen/ Prozeduren

```
public static int verdoppeln(int x) {  
    return x * 2;  
}  
System.out.println(verdoppeln(5)); // Ausgabe: 10
```

## Klassen & Objekte

### Eine Klasse definieren

```
public class Auto {  
    String marke;  
    int baujahr;  
  
    void hupen() {  
        System.out.println("Huuup!");  
    }  
}
```

### Ein Objekt erstellen

```
public class Main {  
    public static void main(String[] args) {  
        Auto meinAuto = new Auto(); // neues Objekt  
  
        meinAuto.marke = "VW";  
        meinAuto.baujahr = 2010;
```

```

        System.out.println(meinAuto.marke);

        meinAuto.hupen();
    }
}

```

## Konstrukturen

```

public class Auto {
    String marke;
    int baujahr;

    // Konstruktor
    public Auto(String m, int b) {
        marke = m;
        baujahr = b;
    }
}

public class Main {
    public static void main(String[] args) {
        Auto a1 = new Auto("BMW", 2020);
        System.out.println(a1.marke + " " + a1.baujahr);
    }
}

```

## Methoden

```

public class Rechner {
    int addieren(int x, int y) {
        return x + y;
    }
}

this
public class Auto {
    String marke;
    int baujahr;

    public Auto(String marke, int baujahr) {
        this.marke = marke;
        this.baujahr = baujahr;
    }
}

```

## static vs. Objektmethoden

```
public class Mathe {  
    static int quadrat(int x) {  
        return x * x;  
    }  
}  
  
System.out.println(Mathe.quadrat(5)); // geht ohne Objekt
```

## Vererbung

```
class Tier {  
    void essen() { System.out.println("Tier isst"); }  
}  
  
class Hund extends Tier {  
    void bellen() { System.out.println("Wuff!"); }  
}  
  
Hund h = new Hund();  
h.essen(); // von Tier geerbt  
h.bellen(); // eigene Methode
```