

# STRING

F. FRÖHLICH UND A. FEINEIS

# Überblick

1. Was ist eine Klasse?
2. Was ist ein String?
3. Was ist eine Verkettung?
4. String Methoden
5. Quellenangaben

# Was ist eine Klasse?

- Java ist in Klassen aufgeteilt
- werden im Gegensatz zu einfachen Variablen mit Großbuchstaben begonnen. Ihre Methoden können durch einen Punkt aneinander "gekettet" werden

Merkhilfe:

„Klassen sind Baupläne für Objekte.“

## Beispiele Klassen

- `java.lang.math`
- `.System`
- `.string`

# Was ist ein String?

- ist eine Zeichenkette
- über Wertemenge aller Zeichenketten (unicode)
- " " leiten einen String ein
- sind Objekte
- sind unveränderlich
- komplexer Datentyp

## Beispiele für Strings

- "abba"
- "Hallo Welt!"
- "@%#&\$"
- "x"

Aber auch

- "" alleine

# Was ist eine Verkettung?

- Strings können mit dem Operator + verkettet werden
- Verkettung auch mit primitiven Datentypen, diese werden damit zu Strings

## Beispiel Verkettung von Strings

```
String firstName = "John";  
String lastName = "Doe";  
System.out.println(firstName + " " + lastName);
```

# String Methoden

1. `"equals(String x)"`
2. `"length(String x)"`
3. `"substring"`
4. `"int compareTo(String x)"`
5. `"int indexOf(String x)"`
6. `"String builder"`
7. `"String buffer"`

## equals(String x)

-> prüft ob zwei Strings gleich sind

## length(String x)

-> liest die Länge eines Strings aus und gibt ihn zurück

## Substring()

-> ruft eine Teilzeichenfolge dieser Instanz ab

### Beispiele string equals

```
public boolean equals(Object anotherObject)
```

Gibt zurück: boolean Wert

überschreibt: equals in Klasse Objekt

### Beispiel string length

```
public int length()
```

Gibt zurück: int Wert

### Beispiel substring:

```
public String substring (int startIndex);
```

Gibt zurück: String Wert

## int compareTo(String x)

-> vergleicht diese Instanz mit einem angegebenen String und gibt an

## indexOf(String x)

-> die Position, an der String x im String vorkommt

### Beispiele compareTo()

```
public int compareTo(String string2)
public int compareTo(Object object)
```

Gibt zurück: int Wert

0 wenn gleich

<0 wenn kleiner als string

>0 wenn größer als string

### Beispiel in indexOf()

```
public int indexOf(String str)
public int indexOf(String str, int fromIndex)
public int indexOf(int char)
public int indexOf(int char, int fromIndex)
```

Gibt zurück: int Wert

-1 wenn nicht vorhanden



# String builder

- Veränderbare Zeichenfolge
- eine Klasse

# String buffer

-> stellt eine Character-Sequenz bereit, die beliebig verändert werden kann

# String Literal

-> ist ein Objekt (der Klasse String) Compiler merkt sich Literale und verwendet sie wieder

# Quellen

[https://www.w3schools.com/java/ref\\_string\\_equals.asp](https://www.w3schools.com/java/ref_string_equals.asp)

Letzter Zugriff 25.10.2021 17:23 Uhr

<https://docs.microsoft.com/de-de/dotnet/api/system.string.substring?view=net-5.0>

Letzter Zugriff 25.10.2021 17:34 Uhr

<https://gocoding.org/de/working-with-stringbuilder-class-in-java/>)

Letzter Zugriff 25.10.2021 17:18 Uhr

<https://www.inf.fh-flensburg.de/lang/prog/string.htm>

Letzter Zugriff 25.10.2021 17:14 Uhr

Java ist auch eine Insel – Einführung, Ausbildung, Praxis  
Christian Ullenboom, Rheinwerk Verlag GmbH 2017, 12.  
Auflage

Vielen Dank für die  
Aufmerksamkeit!!