

Überblick

- Hier wird beschrieben, wie die Ausgabe, vor allem von Zahlen, übersichtlicher gestaltet werden kann.
 - Bisherige Ausgabe: `out.print` oder `out.println`
 - Alternative Ausgabe: `out.printf` oder `out.format`

Eigenschaften von println

- Bei der Ausgabe mit `print` bzw. `println` werden so viele Stellen ausgegeben, wie für die Darstellung des Werts nötig sind.
- Nachteil: kann sehr unübersichtlich werden; die Ausgabe entspricht immer den Regeln des englischen Sprachraums z.B. Punkt statt Komma bei Zahlen.
- Beispiel:

```
double x = 1;
System.out.println ( " sin ( " + x + " ) = " + Math. sin ( x ) );
Ausgabe:
sin(1.0) = 0.8414709848078965
```

System.out.format, System.out.printf

- Die Methode `format` gibt es in zwei Varianten:
 - `static String format(String format, Object... args)`
 - `static String format(Locale l, String format, Object... args)`
 - Die Methode basiert auf den variablen Parameterlisten
 - `args` steht für eine durch Komma getrennte Liste von Ausdrücken beliebigen Typs, die ausgegeben werden sollen – z.B. Variablen oder auch Literale
- Der Formatstring besteht aus Formatangaben und festem Text. Formatangaben beginnen mit einem `%` und enden mit einem Konvertierungszeichen. Dazwischen können weitere Optionen enthalten sein.
Syntax:
 - `% [Argument-Index$][Flags][Width][.Precision]Conversion`

Beispiel mit formatierter Ausgabe

```
double x = 1;
System.out.printf(" sin (%.2f) = %.4f", x, Math.sin(x));
Ausgabe:
sin (1,00) = 0,8415
```

- `x` wird mit 2 Stellen der `sin(x)` mit 4 Stellen hinter dem Komma ausgegeben. `sin(x)` wird gerundet
- Es wird tatsächlich ein Komma verwendet. Das ist der lokale (also Deutsche) Standard, das aus dem Betriebssystemkontext abgeleitet wird.
- Z.B. USA muss hier explizit angegeben werden:

```
double x = 1;
System.out.printf(Locale.US, " sin (%.2f) = %.4f", x, Math.sin(x));
Ausgabe:
sin (1.00) = 0.8415
```

- `format` bzw. `printf` macht keinen Zeilenumbruch (wie `print`). Um den gleichen Effekt wie bei `println` zu erzielen muss am Ende des Format-Strings eine entsprechende Escape-Sequenz stehen ("`\n`") z.B. :
`" sin (%.2f) = %.4f\n"`

Ausgabeformattierung

- Möglichkeiten, elementare Datentypen formatiert auszugeben

```
// Boolesche Werte
System.out.format("%b %b %2$b %1$b%n", true, false);
// Ganzzahlen
System.out.printf("%d%n", 2517);
System.out.format("%7d%n", 2517);
System.out.format("[%7d]%n", -2517);
System.out.format("[(7d)]%n", -2517);
System.out.format("%07d%n", 2517);
System.out.format("%,7d%n", 2517);
System.out.format("%1$d %<o %<x %<X%n", 127);
// Fließkommazahlen
System.out.format("%f%n", 0.000314);
System.out.format("%1$6.2f %1$6.2e %1$6.2E %1$6.2G%n", 3.141592);
System.out.format("%,8.2f%n", 31415.92);
System.out.format(Locale.ENGLISH, "%,8.2f%n", 31415.92);
// Zeichen und Strings
System.out.format("%c%c%c\n", 97, 64, 98);
System.out.format("%s nein\n", "ja");
// Datum/Uhrzeit
Calendar now = Calendar.getInstance();
System.out.format("%1$td.%1$tm.%1$ty %1$th:%1$tm:%1$ts%n", now);
System.out.format("%tF%n", now);
System.out.format("%tc%n", now);
```

```
// Boolesche Werte
true false false true
// Ganzzahlen
2517
2517
[-2517 ]
[ (2517)]
0002517
.2517
127 177 7f 7F
// Fließkommazahlen
0,000314
3,14 3.14e+00 3.14E+00 3.1
31.415,92
31,415.92
// Zeichen und Strings
a@b
ja nein
// Datum/Uhrzeit
27.02.2007 17:18:18
2007-02-27
Di Feb 27 17:18:18 CET 2007
```

Die wichtigsten Konvertierungszeichen

- | | |
|---|--|
| • b - Boolescher Wert | Mit Präfix t: |
| • c - Einzelnes Zeichen | ❖ H - Stunde, zweistellig, im 24-Stunden-Format |
| • d - Ganzzahl in Dezimaldarstellung | ❖ M - Minute, zweistellig |
| • o - Ganzzahl in Oktalдарstellung | ❖ S - Sekunde, zweistellig |
| • x - Ganzzahl in Hexadezimaldarstellung | ❖ d - Tag, zweistellig |
| • X - Dito, mit großen Buchstaben | ❖ m - Monat, zweistellig, 1..12 |
| • f - Flieskommazahl | ❖ Y - Jahr, vierstellig |
| • e - Flieskommazahl mit Exponent | ❖ F - Datum, formatiert nach ISO 8601 (YYYY-MM-DD) |
| • E - Dito, mit großem "E" | ❖ c - Kompletter Datums-/Zeitstring inkl. Zeitzone |
| • g - Flieskommazahl in gemischter Schreibweise | |
| • G - Dito, ggfs. mit großem "E" | |
| • t - Prefix für Datums-/Zeitangaben | |
| • s - Strings und andere Objekte | |

Mit "Argument-Index\$" kann angegeben werden, auf welchen Parameter sich die Formatangabe beziehen soll
mit "<" kann angegeben werden, dass dasselbe Argument verwendet werden soll