

Formatierte Ausgabe mit printf

Im einfachsten Fall wird ein fester Text auf dem Bildschirm ausgegeben:

```
printf("Dies ist ein einfaches Beispiel");
```

Der Text kann auch Sonderzeichen (z. B. Zeilenumbrüche) enthalten:

```
printf("Hier werden \n zwei Zeilen ausgegeben!");
```

Die Zeichensequenz `\n` bewirkt einen Sprung an den Anfang der folgenden Bildschirmzeile. Weitere gebräuchliche Sonderzeichen sind:

<code>\n</code>	Sprung an den Anfang der folgenden Bildschirmzeile
<code>\b</code>	Gehe ein Zeichen zurück
<code>\a</code>	Akustisches Signal
<code>\r</code>	Sprung an den Anfang der aktuellen Bildschirmzeile
<code>\\</code>	Ausgabe des Gegenschrägstrichs "\" (Backslash)
<code>%%</code>	Ausgabe des Prozent-Zeichens "%"
<code>\"</code>	Ausgabe eines doppelten Anführungszeichens
<code>\t</code>	Sprung zur nächsten Tabulatorposition

Sollen aktuelle Variablenwerte ausgegeben werden, werden in den Aufruf der Funktion `printf` entsprechende „Platzhalter“ eingefügt:

```
int x = 10;  
printf("Der Wert %d wurde der Variablen x zugewiesen.", x);
```

Auf dem Bildschirm erfolgt die Ausgabe „Der Wert 10 wurde der Variablen x zugewiesen“, es wird also der Platzhalter `%d` durch den aktuellen Wert der Variablen `x` ersetzt. Es ist möglich, mehrere Variablen zugleich auszugeben:

```
int x = 123, y = 234;  
printf("x = %d und y = %d", x, y);
```

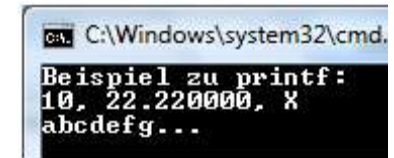
Die Ausgabe lautet in diesem Fall „x = 123 und y = 234“. Für jede Variable ist ein eigener Platzhalter (hier: `%d`) notwendig. Die auszugebenden Variablen werden durch Kommas getrennt aufgelistet. Für jeden Platzhalter muss dabei eine Variable angegeben werden (hier: `x, y`).

Es ist zu beachten, dass der Platzhalter zum Typ der auszugebenden Variablen passt (z. B. dient `%d` zur Ausgabe einer Variablen des Typs `int`, `short` oder `long`). Weitere Platzhalter sind:

<code>%d, %i</code>	<code>int, short, long</code>	Ganze Zahl
<code>%x, %X</code>	<code>int, short, long</code>	Ganze Zahl, Ausgabe als Hexadezimalzahl
<code>%f</code>	<code>float, double</code>	Fließkommazahl
<code>%e, %E</code>	<code>float, double</code>	Fließkommazahl, Ausgabe im Exponentialformat
<code>%c</code>	<code>char</code>	Einzelnes Zeichen (Buchstabe, Ziffer, ...)
<code>%s</code>	<code>char*</code>	Zeichenkette („String“)

Beispiel:

```
int i = 10; double d = 22.22; char c = 'X';  
char* str = "abcdefg...";  
printf("Beispiel zu printf:\n");  
printf("%d, %f, %c\n", i, d, c);  
printf("%s", str);
```



Es kann die Breite des Ausgabebereichs angegeben werden. So wird mit `%10d` eine ganze Zahl rechtsbündig in einem Bereich von 10 Zeichen Länge ausgegeben:

```
int i = 123;  
printf("->%d<\n", i);  
printf("->%4d<\n", i);  
printf("->%5d<\n", i);
```



Bei der Ausgabe von Fließkommazahlen kann zusätzlich zur Länge des Ausgabebereichs die Anzahl der Nachkommastellen eingestellt werden:

```
float f = 123.625;  
printf("->%f<\n", f);  
printf("->%.2f<\n", f);  
printf("->%10.0f<\n", f);  
printf("->%10.1f<\n", f);  
printf("->%10.2f<\n", f);  
printf("->%10.3f<\n", f);  
printf("->%10.4f<\n", f);
```

