

SOURCE CODE 1. Beispiele: Ausgabe und Formatierung von Strings.

```
# Ein string (deutsch: Zeichenkette oder Buchstabenfolge) ist in der Regel
# ein Wort oder Satz. In Python gibt man ihn zwischen ', " oder """ ein, z.B.:
s1 = 'hello, world!'
s2 = "hello, world!"
s3 = """hello, world!"""
# Die drei Strings wurden zwar unterschiedlich definiert, sind aber
# dennoch identisch:
s1 == s2 and s2 == s3
# Wenn man ein Zeichen ', " oder """ _in_ einem String haben will,
# kann man so vorgehen:
s1 = '\'hello world!\'', he said'
s2 = "\"hello world!\\"", he said"
s3 = """\"""hello world!\"""", he said"""
print(s1,s2,s3)
# In Strings kann man Zahlenwerte oder andere Ausdrücke (wenn sie eine
# "string representation", also eine Darstellung als string haben) einsetzen,
# die von Python-Funktionen (eingebauten oder selbst programmierten) erzeugt
# werden: Es gibt mehrere Möglichkeiten, solche "formatted strings"
# (formatierte Zeichenketten) zu erzeugen; wir werden hier immer
# folgende Formatierung von Strings verwenden:
a = 1.0/7
print('Hier wird eine Zahl vom Datentyp float eingefügt')
print(f'1/7 ist ungefähr gleich {a}.')
print(f"""Dies ist ein String, der eine von Python berechnete ganze Zahl
ausgibt, nämlich  $2^{**} 1 + 2^{**} 3 + 2^{**} 5 = \{2^{**} 1 + 2^{**} 3 + 2^{**} 5\}$ """)
# Strings, die Python-Befehlen entsprechen, können auch "als solche
# verwendet werden":
python_string = "2 ** 1 + 2 ** 3 + 2 ** 5"
print(f"""Der Ausdruck "eval({python_string})" wertet den Python-Code
in "{python_string}" aus; Ergebnis: {eval(python_string)}."""")
```