

# Programmieren mit Python

## Print und Variablen

---

Erstellen Sie über **File**→**New**→**Python File** eine neue Datei „**Ausgabe**“ und schreiben hier **print(123)** – starten Sie mit der rechten Maustaste „**Run Ausgabe**“. Print ist eine **Funktion** daher auch die Klammern.

Fügen Sie eine weitere Zeile ein **print("Ihr Name")**.

Texte (sogenannte „Strings“) werden in einfachen oder doppelten Anführungszeichen geschrieben, Zahlen ohne.

Programmierung lebt davon, dass Werte gespeichert werden können in sogenannten Variablen, die Sie als Behälter auffassen können. Dabei verwendet man den sogenannten **Zuweisungsoperator =**.

Für Variablennamen (und später auch Funktionen, Klassen usw. - allgemein Bezeichner) gelten folgende Regeln:

1. Bezeichner können eine Kombination aus Buchstaben in Kleinbuchstaben (**a bis z**), Großbuchstaben (**A bis Z**), Ziffern (**0 bis 9**) oder ein Unterstrich sein **\_**. Namen wie **dmyClass**, **vars\_1** and **ausgabe\_auf\_dem\_bildschirm**, alle sind gültige Beispiele.
2. Ein Bezeichner kann nicht mit einer Ziffer beginnen. **2variabel** ist ungültig, aber **variabel2** ist ein gültiger Name.
3. Sie können keine Sonderzeichen verwenden wie **!, @, #, \$, %** usw.
4. Schlüsselwörter können nicht als Bezeichner verwendet werden.

**Schlüsselwörter** sind reservierte Wörter in Python.

Sie werden verwendet, um die Syntax und Struktur der Sprache Python zu definieren.

In Python wird bei Schlüsselwörtern zwischen Groß- und Kleinschreibung unterschieden.

Es gibt 35 Schlüsselwörter in Python 3.11. Diese Anzahl kann im Laufe der Zeit leicht variieren.

Alle Schlüsselwörter außer **True**, **False** und **None** werden klein geschrieben und müssen so geschrieben werden.

and	as	assert	async	await	break	class	continue
def	del	elif	else	except	False	finally	for
from	global	if	import	in	is	lambda	None
nonlocal	not	or	pass	raise	return	True	try
while	with	yield					

Geben Sie den Bezeichnern immer einen sinnvollen Namen. **d = 10** ist ein gültiger Name, das Schreiben von **zahlen = 10** würde mehr Sinn machen und es wäre einfacher herauszufinden, was es repräsentiert, wenn Sie sich Ihren Code nach einer längeren Pause ansehen.

Mehrere Wörter können durch einen Unterstrich getrennt werden, wie z.B.

**dies\_ist\_eine\_lange\_variable**.

In Python müssen Variablen nicht deklariert werden und können sich auch zur Laufzeit ändern.

Beispiel:

```
variable = 12
print(variable)
variable = "Text"
print(variable)
```