

FreeFlower

Informatik: Programmieren



Kapitel 3: **Variablen erstellen und verändern**

Python-Programmierung: Variablen erstellen und verändern

Was macht dieser Code?

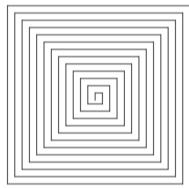
```
import turtle as t

t.forward(10)
t.right(90)
t.forward(15)
t.right(90)
t.forward(20)
t.right(90)
t.forward(25)
t.right(90)
...
t.forward(255)
t.right(90)

t.done()
```

- ▶ Die Turtle beginnt in der Mitte.
- ▶ Jede Seite ist um 5 länger als die vorherige.
- ▶ Das Programm ist lang und aufwendig.
- ▶ Eine `for _ in range(...)`-Schleife scheint nicht möglich, weil immer eine andere Zahl addiert wird (5, 10, 15, ..., 245).

Python-Programmierung: Variablen erstellen und verändern



Variable	Wert
seite	10

```
import turtle as t
```

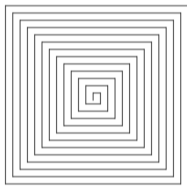
```
t.forward(10)
t.right(90)
t.forward(15)
t.right(90)
t.forward(20)
t.right(90)
t.forward(25)
t.right(90)
...
t.forward(255)
t.right(90)

t.done()
```

- ▶ Die Turtle beginnt in der Mitte.
- ▶ Jede Seite ist um 5 länger als die vorherige.
- ▶ Das Programm ist lang und aufwendig.
- ▶ Eine `for _ in range(...)`-Schleife scheint nicht möglich, weil immer eine andere Zahl addiert wird (5, 10, 15, ..., 245).

Beispiel 3.9: Zweites Programm

Mit **Variablen**



```
import turtle as t

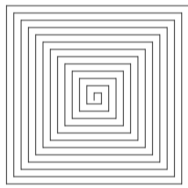
seite = 10
for _ in range(50):
    t.forward(seite)
    seite += 5 # erhöhe Seite um 5
    t.right(90)

t.done()
```

- ▶ Der Befehl `seite += 5` erhöht den in `seite` gespeicherten Wert um 5.

Beispiel 3.9: Zweites Programm

Mit **Variablen**



```
import turtle as t

seite = 10
for _ in range(50):
    t.forward(seite)
    seite += 5 # erhöhe Seite um 5
    t.right(90)

t.done()
```

- ▶ Der Befehl `seite += 5` erhöht den in `seite` gespeicherten Wert um 5.
- ▶ Zeittabelle für die Variable `seite`:

	Nach Aufruf spirale(10)	Nach 1. repeat	Nach 2. repeat	...	Nach 50. repeat
Wert von <code>seite</code>	10	15	20	...	245

Variablen verändern

```
import turtle as t

seite = 10
for _ in range(50):
    t.forward(seite)
    t.right(90)
    seite = seite + 5

t.done()
```

```
import turtle as t

seite = 10
for _ in range(50):
    t.forward(seite)
    t.right(90)
    seite += 5

t.done()
```

- ▶ Das Gleichzeichen („=“) funktioniert in Python anders als in der Mathematik: der Ausdruck oder Wert rechts vom „=“ wird ausgewertet und in die Variable links vom „=“ gespeichert

$a = 10+3$

Wir werten also $10+3 = 13$ aus und speichern den Wert 13 in der Variable `a`.

Variablen verändern

```
import turtle as t

seite = 10
for _ in range(50):
    t.forward(seite)
    t.right(90)
    seite = seite + 5

t.done()
```

```
import turtle as t

seite = 10
for _ in range(50):
    t.forward(seite)
    t.right(90)
    seite += 5

t.done()
```

► Folgende Befehle bewirken dasselbe:

seite+=5 (1)

seite = seite + 5 (2)



Variablen verändern

```
import turtle as t

seite = 10
for _ in range(50):
    t.forward(seite)
    t.right(90)
    seite = seite + 5

t.done()
```

```
import turtle as t

seite = 10
for _ in range(50):
    t.forward(seite)
    t.right(90)
    seite += 5

t.done()
```

- ▶ Folgende Befehle bewirken dasselbe:

seite+=5 (1)

seite = seite + 5 (2)



- ▶ seite+=5 ist also eine abgekürzte Schreibweise für
seite = seite + 5

Unterschiedliche Ausdrücke

- ▶ Was ist der Unterschied zwischen den folgenden Zeilen:
 - ▶ `seite + 5`
 - ▶ `seite += 5`

Befehl	Beschreibung
<code>x+=y</code>	Erhöht den Wert von <code>x</code> um <code>y</code>
<code>x-=y</code>	Verringert den Wert von <code>x</code> um <code>y</code>
<code>x*=y</code>	Multipliziert den Wert von <code>x</code> mit <code>y</code> (und speichert das Resultat in <code>x</code>)
<code>x/=y</code>	Dividiert den Wert von <code>x</code> durch <code>y</code> (und speichert das Resultat in <code>x</code>)

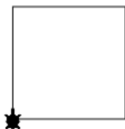
Neue Variablen erstellen

```
import turtle as t

for _ in range(4):
    t.forward(100)
    t.right(90)
```

```
import turtle as t
```

```
distanz = 100
for _ in range(4):
    t.forward(distanz)
    t.right(90)
```

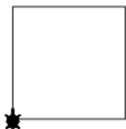


Nur weil man Variablen verwenden *kann*, *muss* man sie nicht immer verwenden.

Neue Variablen erstellen

```
import turtle as t

for _ in range(4):
    t.forward(100)
    t.right(90)
```



```
import turtle as t

distanz = 100

for _ in range(4):
    t.forward(distanz)
    t.right(90)
```

→ = Variable!

Nur weil man Variablen verwenden *kann*, *muss* man sie nicht immer verwenden.

Fazit

- ▶ Variablen sind Namen für Speicherplätze.
- ▶ Wir können jederzeit neue Variablen erstellen, z.B. $y=5$ oder $x=10*3$
- ▶ Der im Speicherplatz abgelegte Wert kann verändert werden.
- ▶ Die Anweisung $x+=add$ erhöht den Wert der Variable x um den Wert von add .
- ▶ Die Anweisungen: $--$, $*=$ und $/=$ funktionieren analog dazu.

Übungen

Allgemein: speichern Sie ihre Lösungen jeweils ab



Aufgaben in Kapitel 3



Abgabe (Moodle): Aufgaben zu Kapitel 3



Challenge: Moodle: ASCII, Schachbrett, Zeit