
Python printen en variabelen

Aan het einde van dit werkblad kun jij:

- voorspellen wat een stukje Pythoncode doet met `prints()` en variabelen erin
- goede en foute codes van elkaar onderscheiden

Spelregels:

- Denk niet te diep na over de codes, het gaat er om dat je het snel gaat kunnen
- Schrijf de codes op als daarom gevraagd wordt, juist door opschrijven oefen je goed
- Spieken/overschrijven is zinloos, het gaat juist om het oefenen. Snap je iets niet, vraag het dan aan ons!

###`print()` en variabele uitleg

Met een `print()` kun je code naar het scherm schrijven. Als je een woord wilt printen moet dat woord tussen dubbele of enkele aanhalingstekens staan, maar in de voorbeelden gebruiken wij altijd enkele.

```
print('Hallo!')
print('Welkom in Leiden.')
```

Je kunt ook variabelen gebruiken, die stel je in met een enkele is. Je kunt de variabelen ook printen

```
naam = 'Feliene'
print('Welkom in Leiden.')
print(naam)
```

Je mag een `print()` ook met meerdere *argumenten* gebruiken. Dan moet er een komma tussen de argumenten.

```
naam = 'Harry'
print('Hallo!')
print('Welkom in Leiden', naam)
```

Een variabele mag ook een getal zijn, dat gaat in principe hetzelfde als met een tekst, alleen dan wel zonder aanhalingstekens:

```
temperatuur = 17
print('Hallo!')
print('Het is vandaag wel', temperatuur, 'graden')
```

Ten slotte hebben we ook nog lijsten. In een lijst kun je meerdere dingen opslaan. Je maakt een lijst met een `is` en rechte haken. Als je een element wilt aanwijzen gebruik je ook rechte haken met ertussen een getal, de *index*.

```
namen = ['Felienne', 'Harry', 'Henk', 'Elma']
print('Hallo!')
print('Welkom in Leiden', namen[1])
```

Let op deze dingen:

- Tellen begint bij 0!
 - Dus namen[0] is het eerste element van de lijst namen, de de code hierboven print “Welkom in Leiden Harry”
- Als iets tussen rechte haken staat wordt het precies zo geprint als het er staat
- Code achter een # is commentaar, dat wordt door Python overgeslagen en is alleen voor mensen om te lezen.

oefeningen

1) Wat print Python als deze codes worden uitgevoerd?

Als er een fout in de code zit, schrijf dan FOUT!

```
1. doelgroep = '5 en 6 VW0-ers'
print('Hallo', doelgroep)
```

```
2. naam = 'Jansen'
#print('Hallo', 'meneer', naam)
```

```
3. eten = 'stamppot'
print('We', 'eten', eten)
```

```
4. #print('Goedemorgen')
print('Klas', '6c')
```

```
5. klas = 'c'
print('Hallo', '6c')
```

```
6. #klas = 6c
print('Leerlingen', 'uit', klas)
```

```
7. tijd = 'half 9'
print('Het', 'is', half, 9)
```

-
- 2) Hieronder staan een aantal incomplete opdrachten om lijsten te vullen. Vul de juiste tekens in op de stippeltjes.

```
1. dieren = ['konijn' ... 'biggetje']
```

```
2. hobbies ... [ ... dansen ... , ... voetballen ...]
```

```
3. kleuren ... .. groen ... , ... geel ... .. blauw ... ..
```

- 3) Schrijf nu de code voor deze lijsten zelf:

- lijst namen met daarin de woorden: Jan, Merel, Samir

.....

- lijst vakken met daarin de woorden: Informatica, Aardrijkskunde, Frans

.....

- lijst Engelse_woorden. Verzin zelf 3 woorden om in je lijst op te slaan

.....

- 4) Hier staan steeds lijsten in code, en een aanwijzer.

Wat wordt er uitgeprint? Soms is de code fout! Schrijf dan FOUT.

```
dieren = ['konijn', 'biggetje', 'vleermuis']  
print(dieren[3])
```

```
kleuren = ['blauw', 'geel', 'groen']  
print('De', 'deur', 'is', kleuren[0])
```

```
hobbies = ['turnen', 'voetbal', 'hockey']  
print('Ik', 'zit', 'op', hobbies[2])
```

```
kleuren = ['blauw', 'geel', 'groen']  
print('kleuren[1]')
```

```
talen = ['Python', 'JavaScript', 'HTML']  
print('De', 'beste', 'programmeertaal', 'is', taal[1])
```

```
snacks = ['chips', 'taco', 'drop']  
print('Ik', 'hou', 'van', snacks(2))
```

```
straatnamen = ['Witte Hertstraat', 'Takstraat', 'Coolsingel']  
print('Ik', 'woon', 'in', 'de', 'straatnamen[2]')
```

```
kleuren = ['blauw', 'geel', 'groen', 'paars', 'roze']  
trui', 'is', kleuren[5])
```

```
print('Mijn', 'trui', 'is', kleuren[5])
```

```
vakken = ['Jan', 'Robin', 'Samir']  
print('Ik', 'vind', vakken[0], 'leuk')
```