

Python: For/While

D. Leeuw

14 mei 2025

1.0.0

© 2025 Dennis Leeuw



Dit werk is uitgegeven onder de Creative Commons BY-NC-SA Licentie en laat anderen toe het werk te kopiëren, distribueren, vertonen, op te voeren, en om afgeleid materiaal te maken, zolang de auteurs en uitgever worden vermeld als maker van het werk, het werk niet commercieel gebruikt wordt en afgeleide werken onder identieke voorwaarden worden verspreid.

1 Wanneer een for- en wanneer een while-loop?

Zowel de for-loop als de while-loop zijn manieren om bepaalde handelingen in een programma te herhalen onder bepaalde voorwaarden. Waarom zijn er twee manieren om dat te doen?

1.1 De verschillen

Er zijn een aantal redenen om waarom we in de ene situatie een for-loop gebruiken en in de andere een while-loop. Dit heeft te maken van de constructie van de twee verschillende loops. Bij een for-loop geef je op welke waarden doorlopen moeten worden, bij de while-loop geef je een conditie op die bereikt moet worden.

1.2 For kenmerken

- Als je van te voren weet hoe vaak je door een loop moet lopen, dan gebruik je de for-loop
- De for-loop volgt een sequentie (list, tuple, string, range) en voert het code block uit voor elk item in de sequentie
- De loop variable is beschikbaar in de loop
- Een for-loop zal doorgaan totdat alle elementen doorlopen zijn.
- Een for-loop stopt als het laatste item is bereikt.

1.3 While kenmerken

- De while-loop wordt gebruikt als je niet van te voren weet hoe vaak je een block code moet doorlopen, maar je weet aan welke conditie voldaan moet worden om de loop te laten eindigen.
- Bij een while-loop is het van belang dat de gegeven eind conditie ooit bereikt wordt, anders heb je een oneindige loop.
- Een while-loop zal doorgaan tot een bepaalde conditie is bereikt.
- Een while-loop stopt zodra de opgegeven conditie niet meer geldig is.

1.4 Samenvatting

Het belangrijkste verschil tussen een for- en while-loop is dat de for-loop kan worden gebruikt als het aantal iteraties bekend is en de while-loop kan worden gebruikt als het aantal iteraties niet bekend is.

	for-loop	while-loop
Loop variabele	Gedefinieerd in de loop aan het begin	Gedefinieerd buiten de loop, moet expliciet gedaan worden
Conditie	Controle voor elke iteratie	Controle voor elke iteratie
Update	Gedaan na elke iteratie	Gedaan in de loop, moet expliciet gedaan worden
Scope	Wordt bepaald door de loop body (voor gedefinieerd)	Moet expliciet gedaan worden
Gebruik	Als het aantal iteraties bekend is	Als het aantal iteraties niet bekend is, of als er aan een bepaalde conditie voldaan moet worden

2 Break: ontsnappen aan de loop

De for- en de while-loop kunnen voortijdig afgebroken worden door het break-statement. Een for-loop hoeft dan bijvoorbeeld niet alle items te doorlopen en een while-loop kan gestopt worden als bijvoorbeeld de loop een aantal keer doorlopen is.

Voorbeeld

```
for i in range(10):
    if i == 5:
        break
    print(i)
```

3 Continue: stappen overslaan

Het kan ook voorkomen dat je uit een reeks 1 bepaalde waarde niet mee wil nemen in je for-loop. Om door te gaan naar de volgende waarde in de reeks gebruiken we continue.

Voorbeeld

```
for i in range(10):
```

```
if i == 5:
    continue
print(i)
```

4 Vragen en opdrachten

4.1 Vragen

1. Achmed heeft een script geschreven om zijn servers automatisch te controleren. Via de verschillende IP-adressen kan hij via `ping` zien of de servers online zijn. De lijst met servers is gegevens als:

```
server_ips = [192.168.10.1, 192.168.10.4, 192.168.10.5,
              192.168.10.10, 192.168.10.55, 192.168.10.56 ]
```

- (a) Wat zou achmed moeten gebruiken om alle IP-adressen via `ping` te controleren een `for`- of een `while`-loop?
- (b) Zou Achmed de lijst ook kunnen doorlopen met een `range(57)`? Waarom wel/niet?

4.2 Opdrachten

1. Schrijf een script dat door de IP-adressen van Achmed loopt en een `ping` uitvoert naar elke server. Als de server aanwezig is geeft het script `UP` terug en als de server niet gevonden kan worden dan wordt `DOWN` teruggegeven.
2. Neem de volgende code over in je editor, sla de code op:

```
while num in range(55):
    print(f"Nummer is {num}")
```

run de code, welke foutmelding krijg je? En hoe los je deze op?