

[Man Chain illustration](#) by [Frits Ahlefeldt-Laurvig](#) | [CC BY-ND](#)

## loops Exercises

Programming (TICT-V1PROG-22)  
HBO-ICT propedeuse blok 1

# Python Loops



Python has two primitive loop commands:

- **while** loops
- **for** loops



Real Python

# Python Loops: While

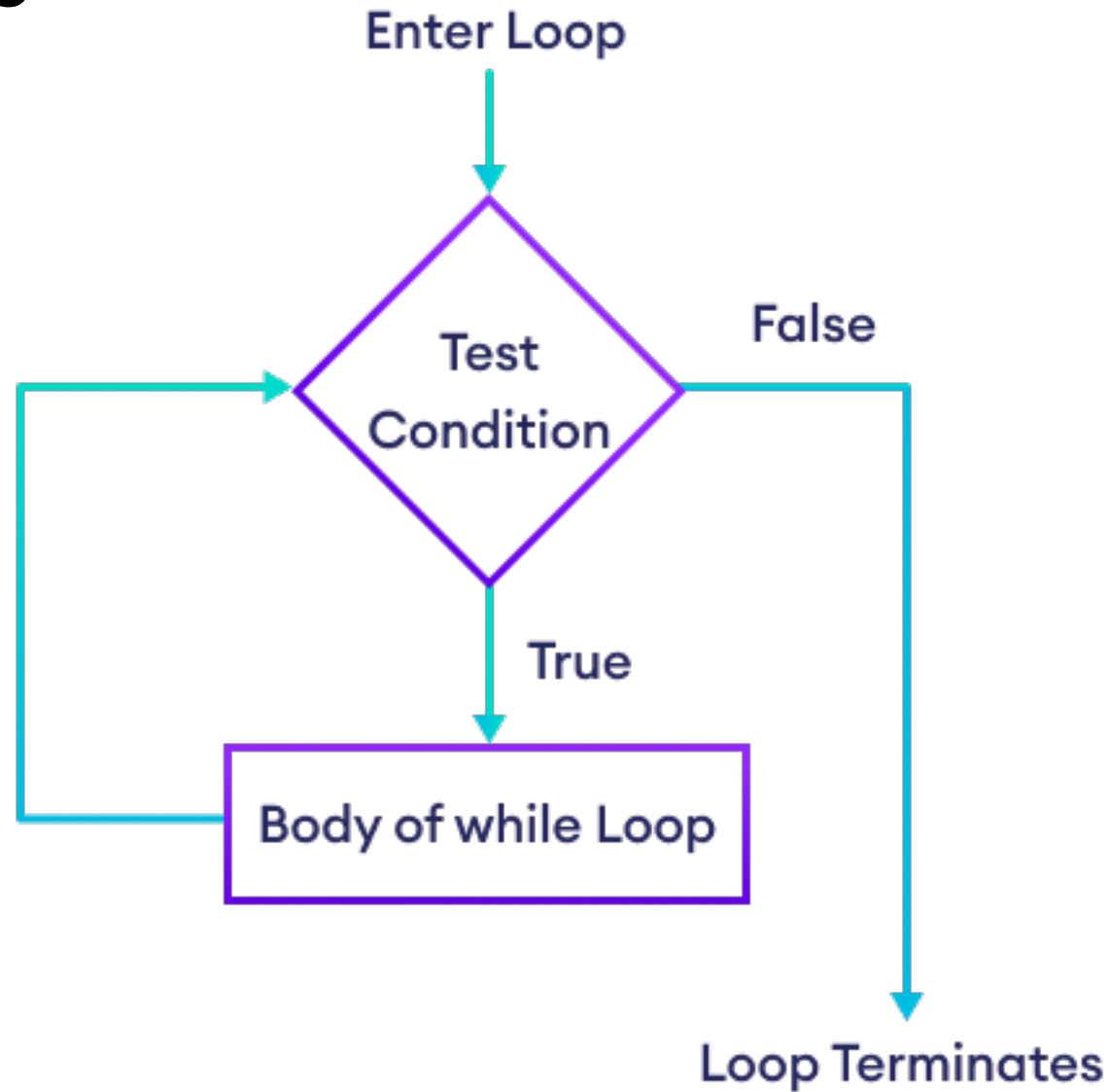
Met de while loop kunnen we een reeks instructies uitvoeren zolang een voorwaarde waar is.

Gebruik:

- Voor de while loop kunnen we vooraf een indexeringsvariabele instellen
- Vergeet dan niet om de waarde te verhogen, anders zal de loop voor altijd doorgaan (infinite loop)
- While loop wordt vaak gebruikt wanneer het aantal iteraties onbekend is

```
while condition:  
    # body of while loop
```

# Python Loops



# Python Loops: While

## Break

Met de break-instructie kunnen we de loop stoppen

## Continue

Met de continue-instructie kunnen we de huidige iteratie stoppen en doorgaan met de volgende.

## Else

Met de else-opdracht kunnen we een codeblok één keer uitvoeren als de voorwaarde niet langer waar is

# Oefening 1



Druk de eerste 10 natuurlijke getallen af met een while-lus

# Demo while

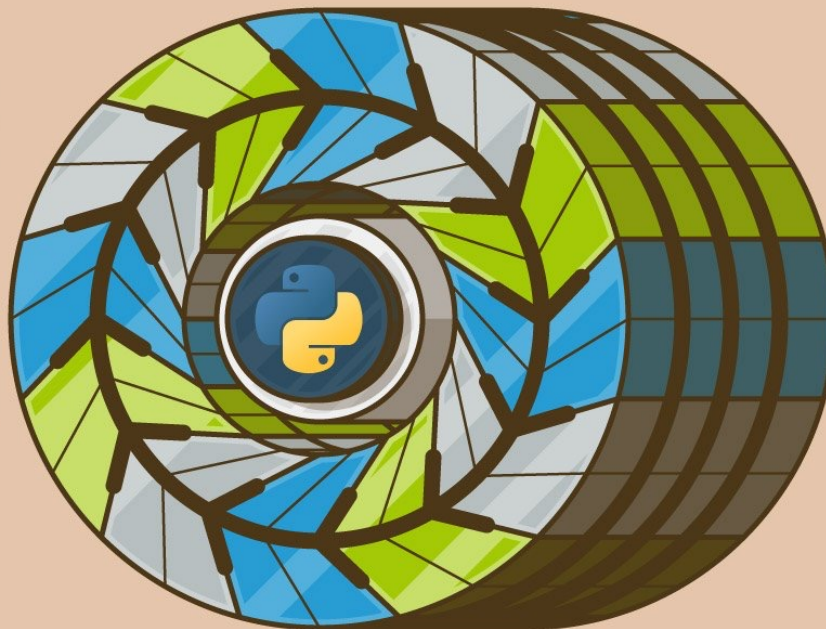


Aantal voorbeelden klassikaal behandelen



**DEFINITE**

**FOR**



**1 to 10**

**BREAK**

**CONTINUE**

**1... 2... 3... 4... 5... 6... .. 10**

**1**  
**2**  
**3**  
**4**  
**5**  
**6**  
**7**  
**8**  
**9**  
**10**

**END**

*Real Python*

# Python Loops: for

Een for loop wordt gebruikt voor het herhalen van een reeks:

- Een list,
- een tuple,
- een dictionary,
- een set (geen onderdeel van de herkansing)
- een string.

Met de for loop kunnen we een reeks instructies uitvoeren, één keer voor elk item in een lijst, tuple, set enz.

Voor de for-lus is het niet nodig om vooraf een indexeringsvariabele in te stellen.

The for loop wordt meestal gebruikt wanneer het aantal iteraties bekend is. Denk aan de lengte van een lijst.

# Python Loops: for ... in ...

Loop door een lijst

Je kunt door de list items 'lopen' met behulp van een for-lus:

```
vruchten = ["appel", "banaan", "kers"]
```

```
for vrucht in vruchten:  
    print(vrucht)
```

```
C:\Users\hans.bastiaan\.virtualenvs\python_PROG_lesse-1jMXSSoa\S  
    Utrecht\Development\python_PROG_lesse\loops\uitleg-for-in.py"  
appel  
banaan  
kers  
  
Process finished with exit code 0
```

# Python Loops: for ... in ...

Loop door een string

Aangezien strings arrays zijn, kunnen we de tekens in een string doorlopen met een for loop.

```
naam = "Hans Bastiaan"
```

```
for letter in naam:  
    print(letter)
```

# Python Loops: for ... in ...

Loop door een dictionary (key, value pairs)

```
auto = {
    "merk": "Ford",
    "model": "Mustang",
    "bouwjaar": 1964
}
```

```
for key in auto:
    print(auto[key])
```

```
Ford
Mustang
1964
```

```
for value in auto.values():
    print(value)
```

```
Ford
Mustang
1964
```

```
for key in auto.keys():
    print(key)
```

```
merk
model
bouwjaar
```

# Python Loops: range()

De functie range() retourneert een reeks getallen  
Standaard beginnend bij 0, oplopend met 1 (standaard) en stopt voor  
een opgegeven getal.

range(start, **stop**, stap)

range(8) => start bij en stop bij 7

range(4,8) => start bij 4 en stop bij 7

range(7, -1, -1 => start bij 7 en stop bij 0 met een stap van -1

Parameter	Beschrijving
start	Optioneel. Een geheel getal dat aangeeft op welke positie moet worden gestart. Standaard is 0
stop	Vereist. Een geheel getal dat aangeeft op welke positie moet worden gestopt (niet inbegrepen).
step	Optioneel. Een geheel getal dat de ophoging aangeeft. Standaard is 1

# Opdrachten



[https://github.com/RoosCocolien/python\\_bijles/](https://github.com/RoosCocolien/python_bijles/)

# Oefening 1 – Vervolg while

Druk de eerste 10 natuurlijke getallen af met een while loop

Verlaat de loop wanneer i gelijk is aan 3

Ga door naar de volgende iteratie  
wanneer i gelijk is aan 3

Druk een bericht af zodra de voorwaarde onwaar is



# Oefening 2



Schrijf een programma om een getallenpatroon af te drukken met behulp van een for .. in ... - loop.

Vraag aan de gebruiker hoeveel rijen er afgedrukt moeten worden.

Tip: Maak gebruik van een dubbele loop en de range() functie

```
Geef een getal: 5
```

```
Nummer patroon
```

```
1
```

```
1 2
```

```
1 2 3
```

```
1 2 3 4
```

```
1 2 3 4 5
```

# Oefening 3

Bereken de som van alle getallen van 1 tot een bepaald getal

Voorbeeld:

Als de gebruiker bijvoorbeeld 10 heeft ingevoerd, moet de uitvoer 55 zijn  
(1+2+3+4+5+6+7+8+9+10)

Verwachte output:

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 = 55$$

# Oefening 4

Gegeven is de volgende list: `numbers = [12, 75, 150, 180, 145, 525, 50]`

Schrijf een programma die alleen nummers weergeeft die voldoen aan de volgende voorwaarden:

- Het getal moet deelbaar zijn door 5
- Als het getal groter is dan 150, sla het dan over en ga naar het volgende getal
- Als het aantal groter is dan 500, stop dan de lus
- Tip: Maak gebruik van break en continue

75

150

145

# Oefening 5

Bereken het kwadraat van elk nummer van de lijst.

Voorbeeld lijst = [1, 2, 3, 4, 5]

Verwachte output:

Kwadraat van 1 is: 1

Kwadraat van 2 is: 4

Kwadraat van 3 is: 9

Kwadraat van 4 is: 16

Kwadraat van 5 is: 25

**\*\*** exponent operator

# Oefening 6

Bereken het gemiddelde van een lijst met getallen.

Voorbeeld lijst: nummers = [10, 20, 30, 40, 50]

Verwachte output:

Het gemiddelde is: 30.0

## — Oefening 7

Tel het totale aantal 'm' in een gegeven string.

Voorbeeld naam = "Marie Antoinette van Oostenrijk"

Verwachte output:

In de naam Marie Antoinette van Oostenrijk komt de letter 'm' 1x voor

- Variaties:
  - Vraag de gebruiker om een naam
  - Vraag naar de letter die gezocht wordt in de naam
  - Als het niet voorkomt verander dan de output:
    - De letter 'p' komt niet voor in de naam"



# Oefening 8



We willen dat de gebruiker een getal tussen 100 en 500 invoert.

We blijven de gebruiker vragen om een correcte invoer in te voeren totdat hij/zij het getal binnen een bepaald bereik invoert.

Tip: gebruik een while loop

Voorbeeld:

Voer een getal tussen 100 en 500 in: 99

Onjuist nummer, vul a.u.b. een geldig nummer in:

Voer een getal tussen 100 en 500 in: 101

Het gegeven nummer 101 valt binnen het bereik.

# Quiz



<https://pynative.com/python-if-else-and-for-loop-quiz/>