



Unit 2 : Beginselen van programmeren in Python

Oefenblad 2 : De for lus

In deze les leren we hoe je opdrachten makkelijk kunt herhalen

Doelen :

- Hoe gebruik je de opdracht **for** .
- Wat is de syntax voor deze opdracht in Python

De kracht van een computerprogramma is de mogelijkheid om een serie opdrachten eventueel met een kleine wijziging te herhalen.

Een van de opdrachten hiervoor is de **for ... lus**

We maken een programma dat van een rij getallen het kwadraat en de derde macht afdrukt.

Open een nieuw Python programma en geef het een naam.

Kies F1(Fns...) en dan **Ctl** en kies nu optie 4.

```

EDITOR: TABEL
Func Ctl Ops List Type I/O Modul
1:if ..
2:if .. else ..
3:if .. elif .. else
4:for i in range(size):
5:for i in range(start,stop):
6:for i in range(strt,stp,step):
7:for i in list:
8:while condition:
9:elif :
0:else:
Esc
    
```

Tussen de haakjes bij range() komt het aantal herhaling en te staan.

```

EDITOR: TABEL
PROGRAM LINE 0001
for i in range(_:
**
**
Fns... a A # Tools Run F
    
```

De lus werkt als volgt:

De variabele i krijgt eerst de waarde 0. Als de lus eenmaal is doorlopen wordt i met 1 verhoogd (dus dan is i=1) De volgende keer

```

EDITOR: TABEL
PROGRAM LINE 0004
for i in range(5):
* a=i**2
* b=i**3
* print(i,a,b)
Fns... a A # Tools Run F
    
```



wordt i
opnieuw
met 1
verhoogd
en dan is
 $i=2$.
Dit wordt
zo vaak
herhaald
als
aangeven
bij range.
Stel we
willen van
de eerste
vijf gehele
getallen
het
kwadraat
en de
derde
macht
afdrukken
, dan is
het aantal
herhaling
en dus 5.
Het
programm
a kan er
nu uitzien
zoals
hiernaast.

Je ziet
weer
duidelijk
de
structuur.
Alles wat
even ver
is
ingespron
gen wordt
5 keer
herhaald.
Merk op



dat i loopt
van 0 t/m
4.

Verander het programma zo dat i de waarden 10 t/m 18 aanneemt.
Tussen de haakjes van **range()** kun je een beginwaarde en een eindwaarde intypen. Dus: `range(.. , ..)`
Merk op dat de eindwaarde 19 is.
In `range(a, b)` neemt i de waarden a t/m b-1 aan.

```

EDITOR: TABEL
PROGRAM LINE 0004
for i in range(10,19):
  a=i**2
  b=i**3
  print(i,a,b)
  
```

We gaan het programma wijzigen zodat de uitkomsten in een lijst worden opgeslagen.
Een lijst is een rij gegevens. In Python noteer je een lijst met vierkante haken. Bijvoorbeeld: `[3,5,7,9]` is een lijst met vier getallen.
Een lijst kan ook leeg zijn (notatie: `[]`).
Begin met een lege lijst. In het voorbeeld noemen we deze: `lst`
Voeg nu 8 keer een nieuwe waarde toe aan de bestaande lijst.
Append() kun je vinden in het Lijst-menu (**F1(Fns...)**, **List**, **append.**)
Append is een Pythonmethode om een lijst uit te breiden met nieuwe elementen.

```

EDITOR: TABEL
PROGRAM LINE 0008
lst=[]
for i in range(8):
  a=i**2
  lst.append(a)
print(lst)
  
```

Voer het programma uit met Run.
Je ziet nu een lijst met de eerste acht kwadraten.
De variabele nu een lijst.
Dat kun je ook zien door in de shell op de **var**-toets te drukken. Dan zie je `lst` in het scherm en als je die kiest en op enter drukt krijg je opnieuw de lijst in beeld.

```

PYTHON SHELL
>>> # Shell Reinitialized
>>> # Running TABEL
>>> from TABEL import *
[0, 1, 4, 9, 16, 25, 36, 49]
>>> |
  
```

Tip voor de docent: De printopdracht staat in I/O-menu. (Kies **F1(Fns...)** en dan I/O en kies daar `print()`)