



Unit 3 : Beginselen van programmeren in Python

Oefenblad 1 : Recursieve functie

In deze Unit gaan we wat dieper in op programmeren in Python.
Lijsten in Python.

Doelen :

- Lijsten maken
- Verschillende lijst-opdrachten gebruiken.

Een lijst wordt in Python genoteerd als een opsomming van elementen tussen rechte haken waarbij de elementen door een komma gescheiden worden. Zo is bijvoorbeeld [7,3,5] een lijst met 3 elementen (in dit geval getallen). Elementen kunnen ook van een ander type zijn bijvoorbeeld tekst of zelf weer een lijst.

Open een nieuw Python programma en maak een lijst a met de getallen 1 t/m 10. Elk element in de lijst heeft een rangnummer waarbij het eerste element nummer 0 heeft. Met a[3] krijg je het element met rangnummer 3.

Voeg een regel aan het programma toe waarbij je het derde element afdruckt en ga na dat dit element de waarde 4 heeft.

Je kunt elementen toevoegen aan een rij met **lijst.append()**. Dit kun je vinden bij F1(Fns...) en dan List.

Als bijvoorbeeld a = [1,2,3], dan is a.append(5) de lijst a = [1,2,3,5]

We gaan dit gebruiken om een list van Fibonacci getallen te maken.

De rij van Fibonacci is: 1, 1, 2, 3, 5, 8, 13, ... waarbij steeds het volgende getal de som is van de twee vorige getallen.

Begin met de lijst L = [1,1].

Dan moeten we de eerste keer toevoegen L[0] + L[1]. Er geldt dan: L = [1,1,2]

De tweede keer voegen we L[1]+L[2] toe. Dit herhalen we steeds weer opnieuw.

Maak een programma dat de rij van de eerste 9 Fibonacci getallen maakt en gebruik hiervoor een for-loop.

Een mogelijk programma staat hiernaast (de lijst heeft hierin de naam lst).

De loop wordt 7 keer doorlopen omdat we al met twee getallen beginnen.

Op deze manier kun je makkelijk een lijst van 100 Fibonacci getallen maken.

```
EDITOR: TEST
PROGRAM LINE 0002
a=[1,2,3,4,5,6,7,8,9,10]
```

```
EDITOR: TEST
PROGRAM LINE 0004
a=[1,2,3,4,5,6,7,8,9,10]
print(a[3])
```

```
EDITOR: FIB
PROGRAM LINE 0005
lst=[1,1]
for i in range(7):
    lst.append(lst[i]+lst[i+1])
print(lst)
```



Een lijst kan ook bestaan uit elementen die zelf een lijst zijn.

Dit is een twee dimensionale lijst of matrix.

We maken nu zo'n lijst waarbij elk element een lijst met Fibonacci getallen is, te beginnen met [1,1].

Verander het vorige programma eerst zo dat de functie f(n) een lijst geeft met Fibonacci getallen.

Definieer nu een lege lijst. (bijvoorbeeld a = [])

Maak dan een for-loop waarbij steeds de lijst f(i) wordt toegevoegd.

Zie hiernaast.

```
EDITOR: FIB
PROGRAM LINE 0006
def f(n):
  lst=[1,1]
  for i in range(n):
    lst.append(lst[i]+lst[i+1])
  return lst
```

```
EDITOR: FIB
PROGRAM LINE 0011
def f(n):
  lst=[1,1]
  for i in range(n):
    lst.append(lst[i]+lst[i+1])
  return lst

a=[]
for i in range(4):
  a.append(f(i))
print(a)
```

De lijst a uit dit programma bestaat uit lijsten.

Als we die lijsten onder elkaar willen afdrukken dan kan dat met een for-lus.

We gebruiken hiervoor de optie **for i in list**. (dit kun je vinden met F1(Fns...) en dan Ctl).

In deze lus doorloopt i alle elementen van de lijst. (Dus i is steeds zelf een lijst)

Breid het programma uit met zo'n loop en druk alle lijsten uit a af.

```
EDITOR: FIB
PROGRAM LINE 0012
lst=[1,1]
for i in range(n):
  lst.append(lst[i]+lst[i+1])
return lst

a=[]
for i in range(4):
  a.append(f(i))

for i in a:
  print(i)
```