

Tuples

Een tuple is in Python een manier om meerdere waardes te combineren. Je kan het beschouwen als een constante lijst; eens aangemaakt kan je er niets meer aan wijzigen. Verder kan je wel de individuele elementen opvragen met `[]`, maar je kan ze niet veranderen. Een tuple maak je aan met ronde haken "(" en ")" in plaats van vierkante haken bij lijsten:

```
In [1]: t=(1,"a",7)
        print(t,t[0],t[-2],t[0:2])
```

```
(1, 'a', 7) 1 a (1, 'a')
```

```
In [2]: t[1]=5 # geeft error!
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-2-306aef3316bd> in <module>
----> 1 t[1]=5 # geeft error!
```

```
TypeError: 'tuple' object does not support item assignment
```

Tuples worden vaak gebruikt om meerdere waarden tegelijk terug te geven in een functie. Kijk bijvoorbeeld naar volgende code om gemiddelde en standaardafwijking terug te geven:

```
In [3]: import math

# "getallen" is een lijst getallen.
# De functie geeft (m,s) terug waarbij m het gemiddelde van L is en s de stand
aardafwijking van de steekproef
# Voor uitleg over de code, zie de notebook over lijsten.
def statistieken(getallen):
    som=0.0
    for x in getallen:
        som=som+x
    gemiddelde=som/len(getallen)

    ssom=0.0
    for x in getallen:
        ssom=ssom+(x-gemiddelde)**2 # ** is machtsverheffing
    sigma=math.sqrt(ssom/(len(getallen)-1))

    return (gemiddelde,sigma)

L=[2,5,7,12,8]
t=statistieken(L)
print(t)
print("m=",t[0],"s=",t[1])

(6.8, 3.7013511046643495)
m= 6.8 s= 3.7013511046643495
```

Python voorziet ook volgende handige manier om de waarden in een tuple "uit te pakken":

```
In [4]: (m,s)=statistieken(L)
print("m=",m,"s=",s)

m= 6.8 s= 3.7013511046643495
```

Extra Je vraagt je misschien af waarom in Python het tuple type voorzien is terwijl we ook lijsten hebben die strikt krachtiger zijn; waarom is het nuttig om een constant type te hebben? De reden komt later nog aan bod en heeft te maken met het feit dat we tuples kunnen gebruiken als een zoek sleutel om te zoeken in datatypes zoals de *dictionary* die we in de volgende notebook bespreken. Die datastructuren bewaren de data bijvoorbeeld gesorteerd op sleutel om snel zoeken mogelijk te maken. Bij zoek sleutels is het echter belangrijk dat de waarden constant zijn, zodat de zoek sleutel *in* de datastructuur niet per ongeluk veranderd kan worden waardoor de datastructuur intern corrupt wordt.