

# #KLOOIENMETCOMPUTERS

ARNOUT VAN KEMPEN OVER ROMMELEN IN EEN DIGITALE WERELD

**A**n het begin van deze serie heb ik al eens geadviseerd een Raspberry Pi aan te schaffen. Op dat moment was de Pi 4B, met Raspbian als besturingssysteem, de voor de hand liggende keuze. Inmiddels is ook de Pi 400 te koop, waarbij je in feite een complete computer koopt, met toetsenbordje, muis en meegeleverd besturingssysteem. Wie nog geen Pi heeft en wel wil mee-klooien, zou daar zeker naar moeten kijken.

Een tweede ontwikkeling is dat het besturingssysteem langzaam verandert en inmiddels ook een nieuwe naam heeft gekregen: Pi OS. Inhoudelijk maakt dat voor het gebruik nauwelijks verschil, zoals ook de 4B en de 400 voor deze serie volledig inwisselbaar zijn. Ik zal de typenummers niet meer noemen hierna als ze niet relevant zijn. Voor wie op internet gaat zoeken is het dus wel goed te weten dat de aanduidingen inwisselbaar zijn.

Goed, nu naar Python op de Pi. De simpelste manier om met Python te beginnen op de Pi is vanuit de GUI, de grafische shell ofwel de desktop. Linksboven in de desktop zit de menuknop, daaronder de optie Programming en daaronder zouden zowel Python 2 als Python 3 zichtbaar moeten zijn.

Helaas is Python 3 niet *backward compatible* met Python 2 en dat betekent dat we een keuze moeten maken. Het voordeel van versie 2 is dat veel informatie beschikbaar is op internet en dat ook **veel** bestaande code in versie 2 is geschreven. Voordeel van versie 3 is dat deze langer zal worden ondersteund. Aangezien ik blanco begin, kies ik voor versie 3 en kan meteen de helft van mijn Python-boekenkast weggooien, omdat die op versie 2 betrekking blijkt te hebben. Gouden tip: eerst kijken voor welke versie iets is, dan pas aanschaffen en gebruiken.

Na het starten van Python 3 opent een window met als titel zoets als 'Python 3.x.x. Shell'. Bovenin het window wat inleidende tekst, dan een prompt en een cursor, verder is het window leeg. Het simpelste gebruik hier is het invoeren van expressies die direct door Python worden uitgevoerd. Tik 2+2 gevolgd door enter en Python zal correct vertellen dat dat 4 is.

Om al iets meer het gevoel te krijgen dat je met Python bezig bent kun je tikken Print ('Het is wat'), gevolgd door enter. Nog niks hoogstaands, maar toch een eerste echt stukje programmacode.

Als het goed is past de Python Shell kleurcoderingen toe, die

er aan bij moeten dragen dat programmacode beter leesbaar is. Je kan dit in de opties aanpassen, maar voorlopig zou ik het zo laten. Er is over nagedacht en het beter weten kan altijd nog.

Dat wat bij het printcommando tussen haakjes staat wordt weergegeven. Staat iets tussen aanhalingstekens, dan wordt dat gezien als een string, een reeks tekens. Zonder de aanhalingstekens ziet Python een expressie die het probeert uit te werken. Dus: Print("3+4") levert 3+4 Print(3+4) levert 7

```
En
a=3
b=4
Print (a+b) levert ook 7
```

In Python geldt een soort afspraak dat 'mooier is beter'. Een manier om je code mooi te maken is het gebruik van duidelijke namen voor variabelen. Losse letters zeggen meestal weinig. Gebruik dus liever namen die iets zeggen over waar je de variabele voor gebruikt. Liever niet r, of t, maar rente of tijd. ←