

Start to Program: Python, Les 1

Intro: Wat is Python?



Python is een programmeertaal.
Ze is er al sinds de begin jaren
90. Ontworpen door de Nederlander Guido van Rossum.
En ja, het heeft zijn naam te danken aan Monty Python.

Het is een open-source taal. Dit wil zeggen dat er de vrijheid is voor iedereen om het project mee te verbeteren of aanpassingen aan te brengen. Python3, versie 3.7.4, is de huidige versie. Hier ga je deze cursus mee aan de slag.

De bedoeling was om een programmeertaal te ontwikkelen die voor iedereen toegankelijker was en tegelijk veelzijdig is. De nadruk ligt dus op de leesbaarheid van de code. Daarom laat python een aantal dingen achterwege die je wel in andere programmeertalen tegenkomt. Een lijn code afsluiten gebeurt niet met een punt-komma. Er wordt ook geen gebruik gemaakt van accolades. In plaats hiervan werkt men met insprongen en gewoonweg een nieuwe regel te starten. Hierdoor worden spaties, tabs en nieuwe regels wel iets belangrijk om op te letten.

Python is the
easier language
to learn.
No brackets,
no main.



You get errors
for writing an
extra space



INSTA: RATHAN.CAGE

Waarom Python?

Buiten zijn gemakkelijk gebruik zijn er nog een aantal andere troeven.

Python programmeert cross-platform. Een stuk code moet niet eerst 'compilen'(samen gevoegd worden). Python doet de berekeningen meteen. Hierdoor is het iets trager, maar kan het wel op zowat alle besturingssystemen, Windows, los,... draaien.

Python heeft een grote standaard bibliotheek. Dit wil zeggen dat je niet alles zelf zal moeten programmeren vanaf nul. In de bibliotheek(of library) zijn al vele functies reeds geprogrammeerd. Er zijn ook nog vele extra toe te voegen bibliotheken die je extra voor een specifiek doel kan toevoegen.

Wat kan je nu in Python programmeren? Het antwoord is: veel.
Hier zijn een aantal hoofdgebruiken van Python.

Webontwikkeling

Web back-end frameworks kunnen gemaakt worden met Python. Django en Flask zijn frameworks die werken met Python. Je kan er HTML mee genereren, databanken aansturen,...

Python wordt in vele webapplicaties gebruikt. Yahoo gebruikt het voor hun versie van maps. Google was origineel geschreven in Python. Dropbox is bijna volledig met Python. De eerste bitcoin torrents werkten met Python.

Zelfs de CIA gebruikt het om hun sites te beheren.

COMPANIES USING PYTHON



DataScience

Met behulp van databanken data analyseren en visualiseren.

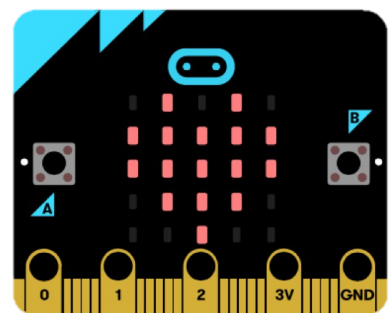
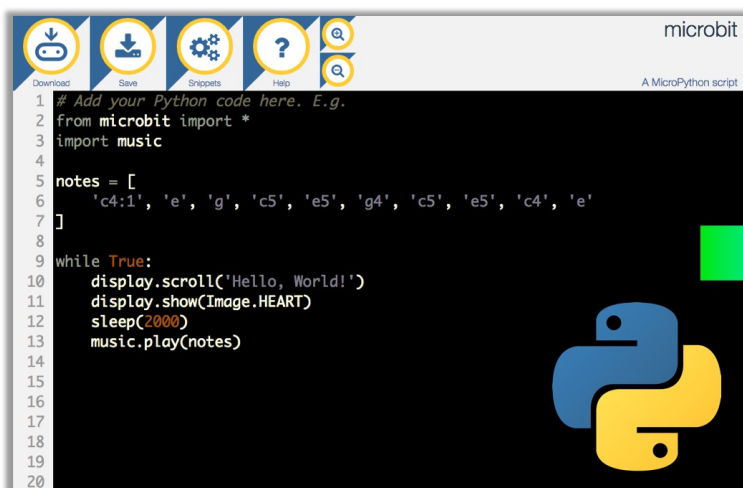
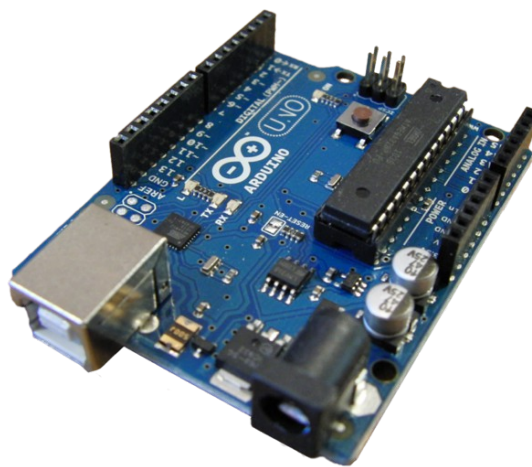
Dit kan helpen om verkoopdata van een bedrijf te visualiseren en analyseren, maar ook data uit beelden om patronen te herkennen. Dit is dichter gelinkt met machinaal leren of artificiële intelligentie.

Scripting

Je kan python gebruiken om scripts te schrijven. Dit zijn stukken code die een eenvoudige taak uitvoeren die veel herhaald moet worden. Bijvoorbeeld het sorteren van e-mails op sleutelwoorden.

Embedded applications

Kleine programma's die een extern stuk hardware aansturen. Zoals bijvoorbeeld een arduino of een microbit. Zo kan je eigen domotica ontwikkelen.



Games

Er bestaan frameworks om games te ontwikkelen in Python. Toch zijn C# en Unity hier meer voor geschikt. Dit geldt ook voor losstaande programma's. Omdat deze gecompileerd worden werken ze efficiënter.

Installatie

Om Python code te kunnen schrijven op je pc moet je Python eerst installeren. Hiervoor heb je een aantal dingen nodig.

Python

Je kan de laatste versie van Python gratis downloaden op hun website.

<https://www.python.org/downloads/>



Normaal krijg je afhankelijk van het besturingssysteem dat je gebruikt meteen de juiste download voorgeschoteld. Dit installeert een aantal tools en de Python bibliotheek op je computer.

Na de installatie ga je controleren welke versie van Python geïnstalleerd is. Open hiervoor de **command prompt** in **Windows** of de **terminal** in **los**.

Je kan deze vinden door:

Bij Windows beneden in het startmenu *cmd* in te typen.

Bij los klik je op het vergrootglas rechts boven in je scherm, type *terminal* in.

Hier kan je rechtstreeks info over je computer en zijn files opvragen. Je kan er ook programma's in schrijven en uitvoeren.

Je geeft het volgende commando in:

```
python --version
```

Dit zou *Python 3.7.4* moeten opleveren.

In de terminal bij los krijg je *Python 2.7.10* Toch is de nieuwste versie geïnstalleerd. Probeer nog eens hetzelfde met het commando

```
python3 --version
```

Nu krijg je wel de juiste versie te zien. Dit is meteen het grootste verschil tussen Windows en los. Voor los moet je steeds het commando `python3` gebruiken. Verder werkt alles grotendeels hetzelfde.

In de command prompt kunnen we nu rechtstreeks programmeren in Python.

Eerst start je hiervoor Python op.
Dit doe je door het commando:

```
python
```

Nu zit je in de programmeer module van Python.

Er zijn een beperkt aantal acties die je hier kan uitvoeren. Python doet zelf een aantal suggesties.

```
Python 3.7.4 (v3.7.4:e09359112e, Jul  8 2019, 14:54:52)
[Clang 6.0 (clang-600.0.57)] on darwin
Type "help", "copyright", "credits" or "license" for more
information.
```

Probeer ze allemaal even uit. *help* is hierbij de nuttigste. Hiermee kan je hulp oproepen over verschillende elementen van Python.

Tijd om een eerste lijn code te schrijven.

We doen dit aan de hand van een klassieker in de programmeerwereld. => Hello World
Met Python draaiende type je het volgende in de command line.

```
print('Hello World')
```

Druk nadien op Enter.

Proficiat! Je hebt een eerste stap gezet in de digitale wereld van het programmeren!

Opdracht:

Probeer een aantal dingen te “printen” op je scherm. Bijvoorbeeld je naam.

Probeer een aantal fouten te maken.

Zijn de quotes echt nodig?

Hoe zit het met getallen?

Kan je berekeningen maken?

Om Python te sluiten geef je het commando: `exit()`

Je kan nu een enkele lijn code ingeven en kijken wat het resultaat zou zijn. Goed om commando's uit te proberen dus.

Aan deze manier van werken zijn een aantal duidelijke nadelen verbonden.

- Je kan maar 1 lijn tegelijk testen.
- Je kan je code niet bewaren.

Editor (ide = integrated development environment)

Om meerdere lijnen code te schrijven en je programma op te slaan ga je een editor moeten gebruiken.

Python heeft tijdens de installatie al een eenvoudige editor geïnstalleerd, **idle**.

Je kan deze vinden in de Python folder.

Start het programma **idle**.

Als eerste scherm krijgen wel een venster dat sterk lijkt op de cmd. Hier kan je niet veel meer mee dan de cmd. Achter >>> kan je een enkelvoudig commando ingeven.

```
Python 3.7.4 (v3.7.4:e09359112e, Jul  8 2019, 14:54:52)
[Clang 6.0 (clang-600.0.57)] on darwin
Type "help", "copyright", "credits" or "license()" for more information.
>>> print('hello world')
hello world
>>> |
```

Op het eerste zicht niet veel nieuw.

Je kan hier echter wel een nieuw bestand openen. Daar kan je dan meerdere lijnen code ingeven.

Klik op *bestand->nieuw bestand*.

Type hier enkele nieuwe lijnen code. Gebruik de `print()` functie.

Let op! Je moet de lijn niet afsluiten, maar gewoon een nieuwe lijn beginnen.

Voor je je code kan testen moet je deze bewaren.

Maak op de pc een nieuwe map aan met je naam.

Vervolgens zet je hierin een map met `les1`.

Hier komen al je scripts die je geschreven hebt per les.

Nu kan je het pas geschreven stuk code hier bewaren.

In **idle** klik op *bestand->opslaan als*.

Geef je bestanden steeds een duidelijke naam. Zo weet je wat er juist instaat.

Een python bestand heeft een `.py` extensie.

Bv.: `intro.py`

Nu je code bewaart is kan je deze testen.

In **idle** met het codevenster geselecteerd klik op *run->run module* of druk op F5.

In de shell van **idle** zie je verschillende lijnen code in 1keer uitgevoerd.

Je kan dit programma nu ook testen in de cmd.

In de command prompt type je eerst `python`.

Druk nog niet op enter, maar geef eerst het pad naar je python file in.

Dit kan door die file in de cmd te slepen. Druk vervolgens op enter.

Nu voert de cmd het programma 1 keer uit.

Atom

Na het gebruik van idle merk je dat dit al veel makkelijker werkt. De verschillende syntax krijgt een kleurencode.

Nu werkt deze perfect als je in het begin Python onder de knie wil krijgen. Als je echter grotere projecten op poten wil zetten is een gespecialiseerde ide handiger.



Bij de cursussen webontwikkeling en webdesign maak je gebruik van **Brackets**. Je kan deze perfect gebruiken, maar deze cursus ga je vooral werken met **Atom**. Dit is een uitgebreidere editor die veel gebruikt wordt.

Installatie

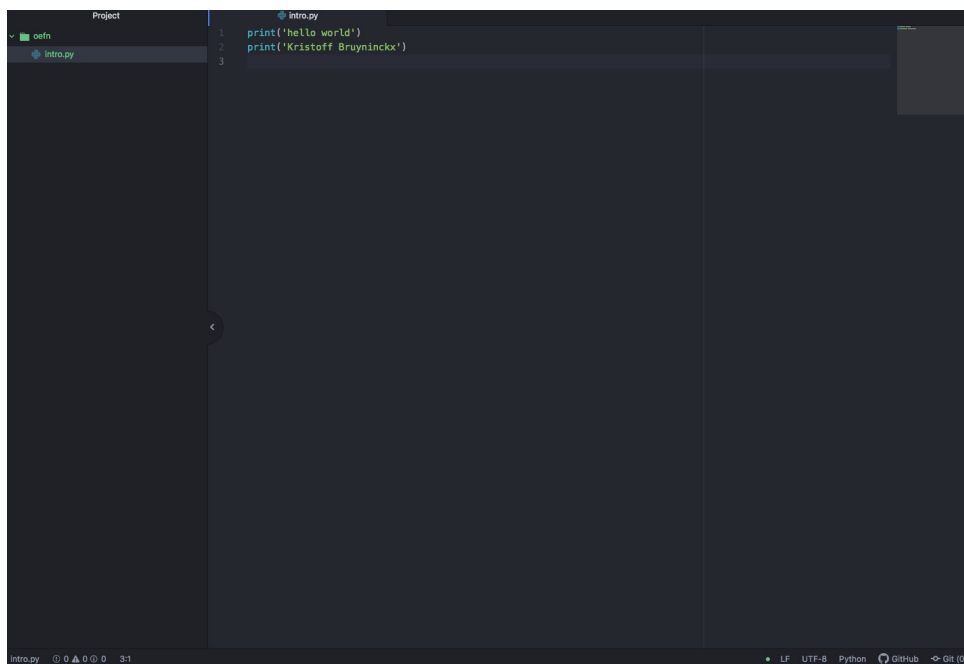
Atom is een gratis editor. Je kan deze downloaden op hun website:

<https://atom.io/>

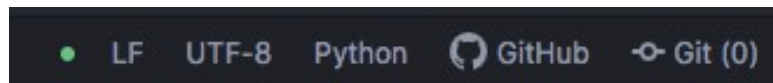
Als deze geïnstalleerd is krijg je een welkom-scherm. Hier kan je afvinken om de welkompagina telkens te tonen bij het opstarten van Atom.

Je kan nu je **intro.py** file of je **les 1** map openen in **Atom**.

Je ziet dat er nu verschillende delen zijn in de editor. Langs de ene kant tabbladen waar je je code kan terugvinden. Langs de andere kant een bestandsboom met je mappenstructuur. Als deze er niet staat kan je hem aanzetten onder het menu: **view ▢ toggle tree view**

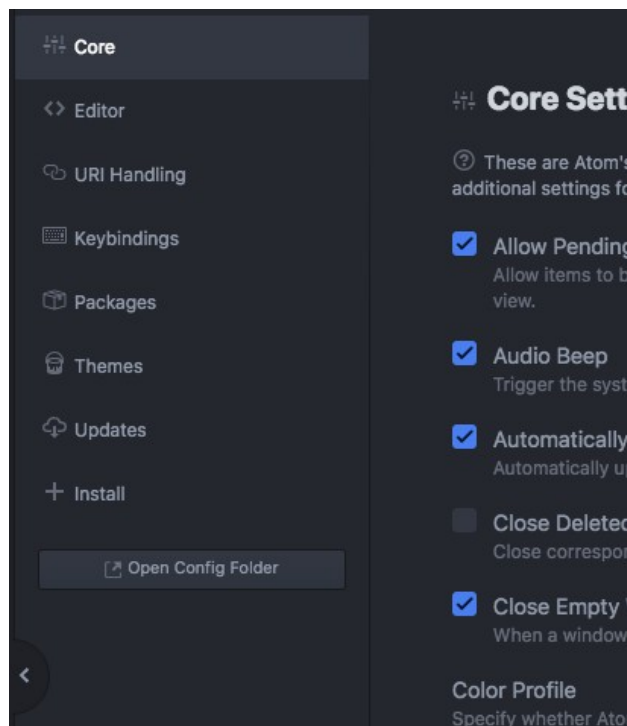


Als je nu je *intro.py* bekijkt zie je een soortgelijke kleurencode als in *idle*. Atom herkent ook meteen dat je met Python aan het werken bent. Dat kan je rechts onderaan ook zien.

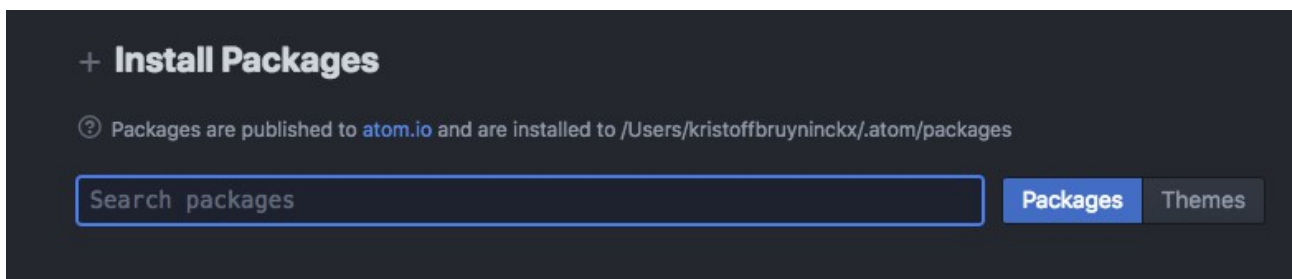


Zoals je merkt zijn er nog wel een aantal verschillen met de editor hier afgebeeld en een vers geïnstalleerde versie. Dit komt omdat je met Atom verschillende extensies(packages) kan toevoegen om je werkruimte te personaliseren.

Dit doe je in de preferences of settings. (Respectievelijk mac of windows)
Dan opent Atom volgend tabblad.



Daar ga je naar het menuonderdeel *install*. Dan krijg je dit.



Hier kan je de extra packages downloaden en installeren om je extra te helpen.
OF onder themes kan je andere kleurenthema's downloaden om de editor in een ander jasje te steken. Dit kan je zo meteen zelf eens uitproberen.

Bij deze packages gaan we op zoek naar het package **script**.
Deze gaat ervoor zorgen dat je code van Python rechteerks in Atom kan testen.

Ga nu naar het menu **packages** ▢ **script** ▢ **run script**

Je code zal onderaan in een apart venster worden uitgevoerd.

Hier zijn een aantal handige shortcuts:

Command	macOS	Linux/Windows	Notes
Script: Run	<code>cmd-i</code>	<code>shift-ctrl-b</code>	If text is selected a "Selection Based" is used instead of a "File Based" run
Script: Run by Line Number	<code>shift-cmd-j</code>	<code>shift-ctrl-j</code>	If text is selected the line number will be the last
Script: Run Options	<code>shift-cmd-i</code>	<code>shift-ctrl-alt-o</code>	Runs the selection or whole file with the given options
Script: Run with profile	<code>shift-cmd-k</code>	<code>shift-ctrl-alt-b</code>	Runs the selection or whole file with the specified profile
Script: Close View	<code>esc</code> or <code>ctrl-w</code>	<code>esc</code>	Closes the script view window
Script: Kill Process	<code>ctrl-c</code>	<code>ctrl-q</code>	Kills the current script process

Ga op zoek naar de volgende packages en installeer ze:

- auto complete python
- file icon
- minimap

Tip! Deze extensies zijn vrij uit te brengen dus kijken naar het aantal downloads is altijd een goede manier om de betere eruit te halen.

Nu je programmeer-omgeving ingesteld is kan je aan de slag.

OPDRACHT:

Maak een nieuw python bestand aan.

Schrijf een programma waar je jezelf bij voorsteld.

Bv: naam, achternaam, job, hobby's, cursussen die je volgt.

Run dit programma zowel in de cmd als in de Atom.

Laad dit programma op naar de ELO, je gaat hier volgende keer verder mee aan de slag.