Les 1: Python - Introductie, installatie & operatoren

Python

Introductie - Wat is Python?

Wat is Python?

Python is een object-georiënteerde programmeertaal die begin jaren 90 ontworpen en ontwikkeld werd door Guido van Rossum.
Hoewel Python tegenwoordig veel wordt gebruikt als back-end programmeertaal voor de Data Science, kan het ook gebruikt worden als front-end taal.

Iedereen denkt dat Python alleen gebruikt kan worden voor webdevelopment. Toch is het ook mogelijk om Python te gebruiken bij het ontwikkelen van desktop applicaties en spelletjes.

Fun fact: Python heeft zijn naam te danken aan het favoriete televisieprogramma van Guido van Rossum, namelijk: Monty Python's Flying Circus **[2].**

Installeer Python 3

Tijdens de lessen gebruiken wij Windows. Uiteraard zijn andere Operating Systems toegestaan, maar het kan zijn dat de docent niet (altijd) kan helpen met het troubleshooten van problemen die zich voordoen.

Hieronder staat hoe je Python 3 kunt installeren op verschillende Operating Systems.

Windows:

Stap	Toelichting
Download Python 3	Ga naar https://www.python.org/downloads/ en selecteer Python-3.8.1. Er wordt een download gestart, klik op het bestand om Python 3 te installeren.

Ubuntu:
Open de Konsole **[3]**, deze kan via de Software Explorer of via de keyboard shortcut: `CTRL + ALT + T` .

Stap	Toelichting
Installeer Python 3	Volg de stappen op de website https://linuxize.com/post/how-to-install-python-3-7-on-ubuntu-18-04/ om Python 3 te installeren.

Mac OS:

Stap	Toelichting
Installeer Python 3	Volg de stappen op de website https://installpython3.com/mac/ om Python 3 te installeren.

Operatoren

Net als andere programmeertalen, maakt ook Python gebruik van operatoren (operators).
In python kunnen we deze operatoren in drie categorieën indelen, namelijk:

- Vergelijkende operatoren
- Rekenkundige operatoren
- Logische operatoren

Alle bovenstaande categorieën zullen we hieronder behandelen.

Vergelijkende operatoren

Vergelijkende operatoren worden gebruikt om waardes met elkaar te vergelijken en returnen een True/False. Deze waardes zijn boolean **[B3]** waardes. Het onderstaande tabel - Tabel 1 - beschrijft vergelijkende operatoren in Python:

Operator	Beschrijving
==	Gelijk aan
!=	Niet gelijk aan
<	Kleiner dan
<=	Kleiner dan of gelijk aan
>	Groter dan
>=	Groter dan of gelijk aan

Tabel 1: Overzicht van vergelijkende operatoren

Rekenkundige operatoren

De rekenkundige operatoren kunnen gebruikt worden voor berekeningen. Voorbeelden van deze berekeningen zijn bijvoorbeeld optellen, aftrekken, vermenigvuldigen, delen, rest berekeningen en exponenten.

Tabel 1 toont verschillende rekenkundige operatoren die je kunt gebruiken wanneer je programmeert in Python. Er zijn twee getallen als voorbeeld genomen om de syntax **[B1]** van een operator uit te leggen, de getallen 4 en 2.

Operator	Beschrijving	Syntax
+	Plus: telt twee waarden bij elkaar op.	4 + 2
-	Min: trekt twee waarden van elkaar af.	4 - 2
*	Vermenigvuldig: telt twee waarden herhaaldelijk bij elkaar op.	4 * 2
/	Delen: trekt twee waarden herhaaldelijk van elkaar af.	4 / 2
%	Modulo: berekend de restwaarde van een deelsom.	4 % 2
//	Delen met afronden naar beneden.	4 // 2
**	Exponent: herhaaldelijk vermenigvuldigen van twee waarden.	4 ** 2

Tabel 2: Overzicht van rekenkundige operatoren

Lesopdracht: vergelijkende - en rekenkundige operatoren:

Lesopdracht 1: Wat is de uitkomst van `5 % 2` ?

Lesopdracht 2: Wat is de uitkomst van `2 + 4 > 10 % 2` ?

Lesopdracht 3: Wat is de betekenis van het woord "exponent" en hoe kun je rekenen met exponenten in Python?

Lesopdracht 4: Klopt deze vergelijking `"1" == 1` ? Leg uit waarom wel/niet.

Lesopdracht 5: Een "product" is een wiskundige term **[B2]**. Leg uit hoe je een product kunt uitrekenen in Python en laat zien hoe je het product van 2 en 8 hebt uitgerekend in je Python shell.

Lesopdracht 6: We hebben de volgende vergelijking `4 + 25 * 2 = 58`. Klopt deze berekening? Leg uit waarom wel/niet.

Lesopdracht 7: Wat is de uitkomst van `5 * 5 * 5` en hoe kun je dit anders berekenen in Python?
Tip: `5 * 5 * 5 = 5 ^ 3`.

Lesopdracht 8: Wat is de uitkomst van `16 + 4 * 8` ? Leg uit welke volgorde wordt aangehouden tijdens deze berekening.

Lesopdracht 9: De volgorde waarin rekenkundige operatoren worden toegepast kan beïnvloed worden door haakjes. Hoe zorg je er in je Python shell voor dat de uitkomst van `16 + 4 * 8` gelijk is aan 160?

Logische operatoren

In Python zijn er 3 logische operatoren. Deze zijn:

- and (en)
- or (of)
- not (niet)

De bovenstaande - logische operatoren - worden gebruikt voor vergelijken van boolean waarden. Afhankelijk van de operator geeft deze een True/False terug. Hieronder zijn drie tabellen die de logische operatoren uitwerken.

Logische operator - AND

De uitkomst van een vergelijking met de logische operator "and" geeft alleen True als beide kanten waar zijn. Voor alle andere vergelijkingen returned het False:

Vergelijking	Uitkomst
True *and* True	True

True *and* False	False
False *and* True	False
False *and* False	False

Tabel 3: Vergelijking en uitkomst van logische operator "and".

Logische operator - OR

De uitkomst van een vergelijking met de logische operator "or" geeft altijd True als minstens een kant van de vergelijking waar is:

Vergelijking	Uitkomst
True *or* True	True
True *or* False	True
False *or* True	True
False *or* False	False

Tabel 3: Vergelijking en uitkomst van logische operator "or".

Logische operator - NOT

De logische operator "not" returned het tegenovergestelde. Dit betekend dat not True altijd False wordt en visa versa:

Vergelijking	Uitkomst
not True	False
not False	True

Tabel 4: Vergelijking en uitkomst van logische operator "not"

Lesopdrachten: logische operatoren

Lesopdracht 10: Wat is de uitkomst van `10 < 4 and 5 > 3`? Licht je antwoord toe.

Lesopdracht 11: Wat is de uitkomst van `5 < 4 or 4 < 2`? Licht je antwoord toe.

Lesopdrachten en huiswerk

> De lesopdrachten en het huiswerk kun je vinden op Microsoft Teams: Team "Python-OITAOO8A".
> De deadline voor het huiswerk staat in de additionele informatie van het genoemde team.

Begrippen

[B1] Syntax = De manier waarop iets, bijvoorbeeld een operator, gebruikt/geschreven wordt.
[B2] Term = Woord; Benaming.
[B2] Boolean = True of False.

Bronnen

[1] Bronvermelding: https://docs.python.org/2/faq/general.html#why-is-it-called-python
[2] De Ubuntu terminal wordt "Konsole" genoemd.

Revision #16
Created 2020-01-28 10:21:54 UTC
Updated 2020-02-05 21:15:54 UTC