

Python GGH Workshop 1 (09 April 2021)

Opgave 0:

Start de Python 3 IDLE programmeeromgeving. Ga naar File → New File. Nu heb je de tekst editor gestart waarmee je Python scripts kunt schrijven. Typ het volgende in:

```
print("Hello, World!")
```

Klik op Run -> Run Module. Alternatief kun je gewoon F5 indrukken. Als het goed is heeft je programma in de console Hello, World! Afgedrukt.

Gefeliciteerd, je hebt je eerste programma gerund!

Opgave 1: Berekeningen `print()` en

Gebruik `print()` om het resultaat van de volgende berekeningen af te drukken naar het scherm:

```
2*5
5/2
5//2
4**2
0.5**2
3.8/2
2+5
2.5+5
int(5/2)
int(2.5+5)
```

Wat zie je? Welke resultaten had je verwacht, en welke niet? Wat doen de operatoren `*`, `/`, `//`, `**`, `+` respectievelijk? Wat doet de `int()` functie?

Opgave 2: Datatypes met `type()`

Sla in hetzelfde programma elk van de bovenstaande berekeningen op in een variabele, voordat je hem output. Bijvoorbeeld:

```
var1 = 2*5
print(var1)
```

In Python heeft elke variabele een 'type': het soort data dat is opgeslagen. De belangrijkste types zijn: `int` (gehele getallen), `float` (kommagetallen), `string` (tekst), `bool` (waar/niet waar). Controleer het type van elke variabele. Dat kun je als volgt doen met de `type()` functie:

```
print("mijnvariabele heeft type", type(mijnvariabele))
```

Opgave 3: Tekst met een `string`

Nu gaan we strings, dat zijn tekstvariabelen, maken. Met strings kun je letters, woorden en zinnen opslaan. Strings staan altijd tussen dubbele (""), of enkele aanhalingstekens ("`'` ").

Maak een nieuwe file aan. Hierin maak je een variabele die een string bevat als volgt:

```
mijnstring = "<je tekst>"
```

Print `mijnstring` naar het scherm en controleer ook het type, net als hiervoor, met `type()`. Wat is volgens Python het type van je string?

Maak een tweede string `mijnstring2` aan. Je kunt twee strings aan elkaar plakken met de `+` operator:

```
mijnstring3 = mijnstring + mijnstring2
```

Als je wil kun je een spatie in het midden toevoegen:

```
mijnstring3 = mijnstring + ' ' + mijnstring2
```

Controleer het resultaat met een `print()` statement.

Een string met een getal: "`3.0`" is nog steeds geen getal, en je kunt er niet mee rekenen. Om er een `int` of `float` variabele van te maken van te maken, gebruik je de `int()`, `float()` functies. Het veranderen van het type heet 'type casting'. Maak een bestand aan met de volgende regels:

```
mijnstring1 = "8"  
mijnstring2 = "3.5"  
var3 = int(mijnstring1)  
var4 = float(mijnstring2)  
var5 = var4 * 8
```

Controleer de waarders van `var3` t/m `var5` met `print()`. Wat zie je? Wat gebeurt er als je `var5 = float(mijnstring2)` verandert naar `var5 = int(mijnstring2)` of `var5 = mijnstring2`? De type van een variabele bepaalt dus hoe een operator erop werkt.

Opgave 4: `Input()` van de gebruiker

Met de `input()` functie kan de gebruiker een string invoeren naar het programma. Wanneer het programma `input()` aanroept, wacht Python totdat de gebruiker iets intypt en 'Enter' drukt. Dan geeft `input()` de input van de gebruiker terug als een string.

Je kunt tussen de haakjes van `input` ook een vraag stellen aan de gebruiker: `input("Typ alsjeblieft")`. Maak en run het volgende programma maken.

```
mijninput = input("Geeft u invoer: ")
print("Dankuwel voor uw invoer.")
print("De gebruiker heeft:", mijninput, "ingevoerd!")
```

Controleer het type van `mijninput` met `type()`. Verandert het type als je als invoer een `int` bijv. 8 of een `float` bijv. 4.8 geeft?

Om te kunnen rekenen met invoer, moet je het perse eerst 'type casten' met de `int()` of `float()` functies. Maak een programma dat twee verschillende getallen opvraagt van de gebruiker, en deze opslaat in twee variabelen als een getal. Druk vervolgens het resultaat van de vermenigvuldiging van deze getallen af naar het scherm.

Zo zie je, je kunt met Python een rekenmachine maken!

Huiswerk Opgave 1: Kleuren

Door zogenaamde "ANSI escape codes" te gebruiken in je strings, kun je bij het afdrukken de console vertellen om overstappen op een andere stijl, kleur, of achtergrond kleur. Dit staat hier beschreven: <https://ozzmaker.com/add-colour-to-text-in-python/>.

Probeer het volgende voorbeeld uit:

```
print("Ik houd van \033[0;31mrood\033[0m. De kleur van je  
\033[2;30;41mlippen\033[0m.")
```

Hierin duidt `\033[0` de ANSI escape code aan. `0` duidt normale tekststijl aan. `;` geeft aan dat nog een eigenschap volgt. `31` geef de kleur rood aan. `m` geeft het einde van de escape code aan. `\033[0m` zegt weer: "gebruik normale tekst, zonder bijzondere kleur", anders is alles straks rood!

Experimenteer hiermee. Maak iets moois. Versier bijvoorbeeld een gedicht met kleuren. Brownie points voor wie het gedicht "Rozen zijn Rood" het mooist kan maken!

Tip: Sla een ANSI escape code op in een variabele:

```
rodekleur = "\033[0;31m"  
normaal = "\033[0m"
```

Dan kan je hiermee eenvoudig strings verbinden en je formatting toevoegen:

```
print(rodekleur+"Dit is ROOD."+normaal+" Maar dit niet.")
```

Huiswerk Opgave 1: Rekenmachine

Maak een python script dat twee getallen in neemt, de volgende operaties toepast: +, -, *, /, //, **. Vervolgens `print()` het dit terug naar de gebruiker. Een rekenmachine dus!

Uitdaging 1: Wat doet de `%` operator? Deze kun je het best op twee `ints` toepassen.

Uitdaging 2: Lees hier: <https://cscircles.cemc.uwaterloo.ca/06-if/> hoe de `if`-statement werkt. Hiermee kun je je programma iets laten doen als een bepaalde voorwaarde waar is. Het volgende print bijvoorbeeld alleen hoi als de input `*` is:

```
if input()=="*":  
    print("Hoi.")
```

Kun je hiermee een programma maken dat aan de gebruiker vraagt of hij een +, -, *, /, //, ** berekening wil doen, en dan alleen dat resultaat output? Dan heb je een echte rekenmachine te pakken.

Huiswerk Opgave 3: Blijf experimenteren

Als je tijd over hebt, blijf experimenteren. Ga lekker puzzelen en maak iets moois met wat je deze les hebt geleerd. Je kan hier meer vinden als je nieuwsgierig bent:

Nuttige online cursus voor leerlingen als referentie en oefening:

<https://cscircles.cemc.uwaterloo.ca/nl/>

Officiële Python Tutorial (Zeer uitgebreid): <https://docs.python.org/3/tutorial/index.html>

Je Favoriete Youtube Kanaal.

Officiële Python Documentatie: <https://docs.python.org/>

Algemene vragen: <https://stackoverflow.com/>

Online Python Interpreter: <https://replit.com/languages/python3>