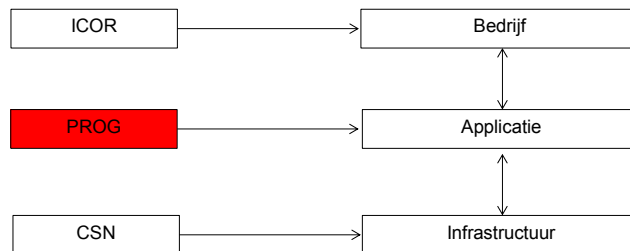


Sheet 1: onderlinge samenhang



De samenhang in de hele studie, de vier richtingen e.d. worden in de introductiebijeenkomst van donderdag of vrijdag behandeld.

Algemene onderwerpen zijn toegelicht op sharepoint dmv filmpjes.



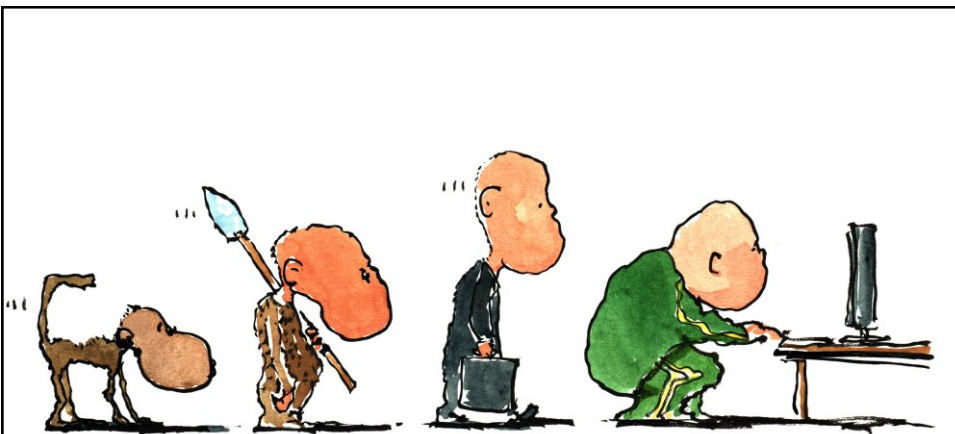
Sheet 2: kennismaken

1. Dit ben ik
2. In de rij
3. De Pitch
4. Bingo
5. Alphabet



Sheet 3. De studiehandleiding

- Zijn er nog vragen over de studiehandleiding?



Man Chain illustration by Frits Ahlefeldt-Laurvig | CC BY-ND

Week 1 | Les 1: Starten met Python

Programming (TICT-V1PROG-15)
HBO-ICT propedeuse periode 1 2015-2016



Vraag: Wie is er vandaag met de trein gekomen? Hebben jullie ingecheckt?



[NS 186 013 met Intercity Direct Breda, Hoofddorp](#) by [Roel Hemkes](#) | CC BY

Week 1: 5



Nederlandse Spoorwegen

- Jaarlijks miljoenen reizigers
- Helemaal HBO-ICT:
 - Kaartautomaat draait op software
 - Chipkaartlezer werkt op hardware
 - Het herontwerp van kaartverkoopproces bij invoering van de OV-chipkaart
 - Alle services moeten beschikbaar zijn.



CC
SOME RIGHTS RESERVED
Spoorjan

6



Taken en Rollen

Taken	Rollen
Kaartautomaat moet de klant door het bestelproces leiden	Software & Information Engineer
Chipkaartlezer moet ov-chipkaart uitlezen	Technische Informaticus
Het herontwerp van het kaartverkoopproces bij invoering van de OV-chipkaart	Business & IT Manager
De software moeten "altijd" beschikbaar zijn.	System & Network Engineer



Wat leer je daarvoor bij Programming? (1)

“Kaartautomaat moet de klant door het bestelproces leiden”:

- gestructureerd programmeren
 - Taak opdelen in subtaken
- input en output verwerken
 - Verwerken van de keuze “tweede klas”
- condities
 - als er dalkorting van toepassing is, dan krijg de klant 40% korting op de ritprijs



Wat leer je daarvoor bij Programming? (2)

“Chipkaartlezer moet de ov-chipkaart uitlezen”:

- gestructureerd programmeren
 - Taak opdelen in subtaken
- input en output verwerken
 - De ov-chipkaart moet uitgelezen worden
- condities
 - als de ov-chipkaart bekend is bij de NS, ga dan verder
- testen
 - kun je ook met een bankpas inchecken?



Wat leer je daarvoor bij Programming? (3)

“Het herontwerp van het kaartverkoopproces”:

- taak vertalen in een ontwerp
 - Welke stappen doorloopt het systeem?
- ontwerp vertalen naar programmastructuur
 - Om bijv. de kosten in te schatten
- condities
 - Als klant “tweede klas” kiest, wat moet het systeem dan doen?
- testen
 - Is er precies geprogrammeerd volgens het ontwerp?



Wat leer je daarvoor bij Programming? (4)

“De software moeten "altijd" beschikbaar zijn”:

- condities
 - Als er veel klantvragen zijn, schakel dan extra servers bij (gescripte deploy).
- taak vertalen naar ontwerp
 - Automatisch update proces ontwerpen.
- input en output
 - Rapporteer over de capaciteit en beschikbaarheid.
- testen
 - Zijn alle systemen nog in de lucht (monitoring)?

Week 1: 11



Python

- Python is een programmeertaal die begin jaren 90 ontworpen en ontwikkeld werd door Guido van Rossum
- Python is Open Source



Waarom Python?

- Snel programmeren
- Concepten zijn bruikbaar als je een andere taal leert
- Veel ingebouwde modules en uitbreidbaar
- Platformonafhankelijk
- Netjes programmeren (o.a. *indentation*)
- Makkelijk te leren, lijkt op natuurlijke taal
- Wordt o.a. gebruikt door Google, Dropbox, Pixar, NASA
- Reacties ouderejaars: “Python? Had ik dat maar gehad”

13



Onderwerpen per week

Unit	Week
1. Starten met Python	1
2. Control flow / Recursie	2
3. Exceptie / Doctest / Datastructure	3
4. I/O / CSV / SQLite	4
5. API / XML / tkinter	5
6. Objects/classes en Source control	6

14



Weekplanning

- Voorbereiding: 2 uur
- Inloopspreekuur: 1 uur
- (Werk) college 1: 2 uur
- Voorbereiding: 2 uur
- (Werk) college 2: 2 uur
- Opdracht: 4 uur
- Online spreekuur: 1 uur
- Tentamen voorbereiding



Calendar* by Dafne Cholet | CC BY

groen = face-to-face

15

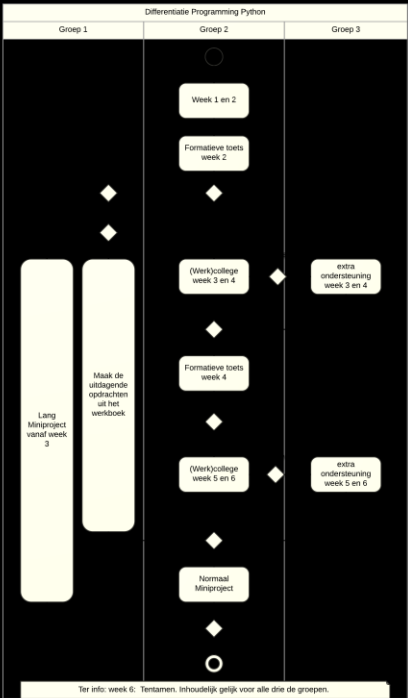


week 1 en 2

week 3 en 4

week 5 en 6

week 7, 8 of 9



Tar info: week 6: Tentamen. Inhoudelijk gelijk voor alle drie de groepen.

Week 1: 16



Beoordeling

- MC-tentamen
 - Week 41 (do of vrij)
 - Weging: 50%
 - Minimaal 5,5
- Miniproject
 - Week 42, 44 of 45
 - Weging: 50%
 - Minimaal 5,5



[Ran across this pic while doing some homework. This dude hit all 122 boy scout merit badges. Go you!](#) by [dennis.crowley](#) | CC-BY

17



Materialen en bronnen

- Sharepoint:
 - Python-Werkboek met opdrachten.
 - Slides (handout).
 - Overig.
- Boek (aanbevolen):
 - Perkovic, L. (2015). *Introduction to Computing Using Python: An Application Development Focus*. NJ: Wiley & Sons.
- Byte of Python (optioneel):
 - http://files.swaroopch.com/python/byte_of_python.pdf

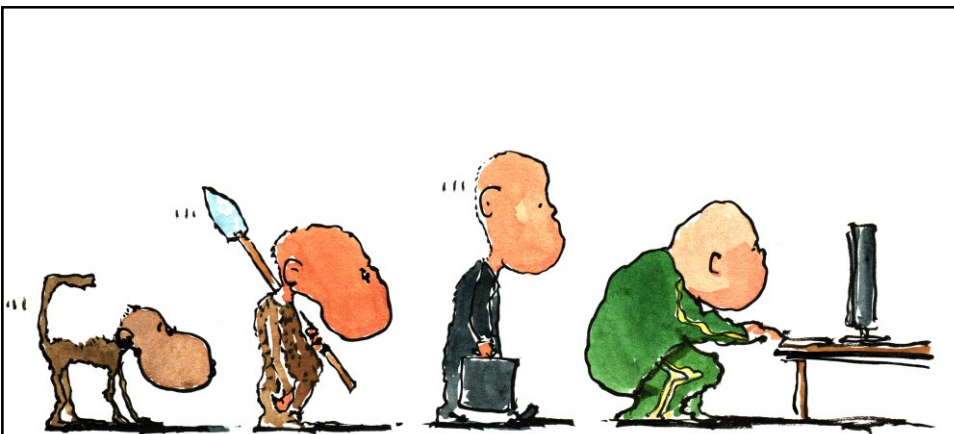
18



Leerdoelen Week 1

- IDE inrichten
- Datatypes: number, string, list
- Operators
- Van taak naar ontwerp

19

HOGESCHOOL
UTRECHT

Man Chain illustration by [Frits Ahlefeldt-Laurvig](#) | CC BY-ND

Numbers

Programming (TICT-V1PROG-15)
HBO-ICT propedeuse periode 1 2015-2016

HOGESCHOOL
UTRECHT

HOGESCHOOL
UTRECHT

Numbers: twee types

- Hele getallen (integer)
 - 1
 - 4
 - -179
 - 0
- Decimale getallen (float)
 - 0.0
 - 1.8797
 - -673.9038

126

1

punt...
geen
komma!

-568

3.1415

-13.78



Numbers: variabelen

variabele

```
leeftijd = 23
print("Mijn leeftijd is:", leeftijd)
```

Tips:

- Variabelen: letters / cijfers / underscore
- Niet beginnen met: `_` / cijfer
- Variabele is hoofdlettergevoelig:
- Meerdere woorden? Liever `user_id` dan `UserId` of `userid`

```
>>> a=3
>>> A=6
>>> print(a,A)
3 6
```



Advanced: Number-functies

- `abs(x)` geeft de absolute waarde van een getal
 - `abs(-2)` geeft 2
- `math.trunc(x)` rondt op een heel getal naar beneden af
 - `math.trunc(10.287)` geeft 10
- `round(x,[,n])` rondt x af op n decimalen. n is standaard 0
 - `round(3.1415, 2)` geeft 3.14

23




Meer over numbers

- Perkovic blz. 16-18
- <http://learnpythonthehardway.org/book/ex3.html>
- http://www.learnpython.org/en/Variables_and_Types
- <https://youtu.be/hnxIRVZ0EyU>

24





[Man Chain illustration](#) by [Frits Ahlefeldt-Laurvig](#) | CC BY-ND

Operators

Programming (TICT-V1PROG-15)
HBO-ICT propedeuse periode 1 2015-2016

HU HOGESCHOOL UTRECHT

Operators

- Doen iets (“operate”) met variabelen
- Arithmetic operators:
 - rekenen met variabelen
 - Bijv. $1 + 3 = 4$
 - Bijv. $3 * 8 = 24$
- Comparison operators
 - vergelijken een variabele met een waarde of een andere variabele
 - Bijv. $1 == 3$
- Assignment operators
 - wijzen een waarde toe aan een variabele
 - Bijv. `var = 4`

Arithmetic operators: // Floor division

- **"Floor division"**: $a // b$ geeft het grootste hele getal dat je krijgt als je a door b deelt:

$$14 // 3 = 4$$

Als je 14 wilt delen door 3, dan kan dat maximaal 4 keer

27



Arithmetic operators: % Modulus

- $a \% b$ geeft juist de restwaarde nadat a door b gedeeld wordt (*remainder*):

$$14 \% 3 = 2$$

- Als je 14 wilt delen door 3, dan kan dat maximaal 4 keer.
- Dan heb je $14 - 12 = 2$ over.

```
>>> 14%3
2
>>> 14//3
4
```

28



Arithmetic operators: ** Machtsverheffen

- Om een macht uit te rekenen, bijv.:
 - 410^2 (= 410 tot de tweede macht)

- `a ** b` geeft “a tot de macht b”

`10 ** 3 = 1000`

`2 ** 10 = 1024`

29



Comparison operators

- Wordt gebruikt om te kijken of iets waar of niet waar is, bijv.:
 - 15 is gelijk aan 14
 - Utrecht is niet gelijk aan Nieuwegein
- “Evalueren” naar **True** of **False**

30



Comparison operators voorbeeld

<code>a == b</code>	Gelijk aan
<code>a != b</code>	Niet gelijk aan
<code>a <> b</code>	Niet gelijk aan
<code>a < b</code>	Kleiner dan
<code>a > b</code>	Groter dan
<code>a <= b</code>	Kleiner of gelijk aan
<code>a >= b</code>	Groter of gelijk aan



Assignment operators

operator	beschrijving	voorbeeld
<code>a = b</code>	Wijst b toe aan a	<code>a = 3</code>
<code>a += b</code>	Wijst <code>a + b</code> toe aan a	<code>a += 3</code> wordt 6
<code>a -= b</code>	Wijst <code>a - b</code> toe aan a	<code>a -= 4</code> wordt 2

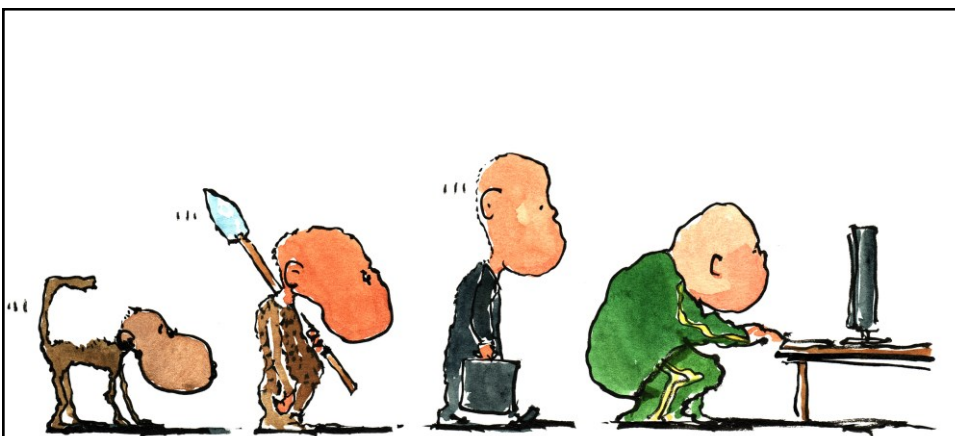
Dit is gelijk aan:
`a = a - b`

Kan ook met `*`, `/`, `%`, `**` en `//`



Meer over operators

- Perkovic Hoofdstuk 2
- http://www.tutorialspoint.com/python/python_basic_operators.htm
- http://en.wikibooks.org/wiki/Python_Programming/Operators
- <http://www.afterhoursprogramming.com/tutorial/Python/Operators/>



Man Chain illustration by Frits Ahlefeldt-Laurvig | CC BY-ND

Strings

Programming (TICT-V1PROG-15)
HBO-ICT propedeuse periode 1 2015-2016



HOGESCHOOL UTRECHT

Strings

- tekst `s="Hello World"`
- bijzondere tekst `s="Hello\nWorld"`
- letters `s="a"`
- nummers `s="5"`
opgeslagen als tekst, dus je kunt er niet mee rekenen
- Gebruik:
Tekst tussen ' ' of " " plaatsen
`'Hello World'`
`"Hello World"`
`"Jan's huis"` of `'Jan zei: "hoi!"'`
`"345"`

35



Strings: Subscripting



```
s = "hello world"
s[0]
s[2]
s[-1]
```

Dit is positie 0

36

Strings: Slicing



```
s = "hello world"
s[1:]
s[:4]
s[0:10:2]
```

37

Advanced: String

- Strings zijn *immutable*, je kunt ze dus niet wijzigen
 - Als je dat probeert, dan krijg je een foutmelding
- Bij *slicing* kun je ook karakters overslaan
 - tekst = 'Hello World!'
 - print(tekst[6:12:2]) geeft **Wrld**
- Bij *splitting* kun je het resultaat opvangen in variabelen
 - tijd = '16:30'
 - uren, minuten = tijd.split(':')
 - print(uren) geeft 16, print(minuten) geeft 30

38



Advanced: String

- `string.replace()` vervangt een karakter door een ander karakter
 - `t = 'Hello Kitty'`
 - `t2 = t.replace('e','a')`
 - `print(t2)` geeft Hallo Kitty

39



Meer over Strings

- Perkovic blz. 23-26
- <http://www.afterhoursprogramming.com/tutorial/Python/Strings/>
- http://www.tutorialspoint.com/python/python_strings.htm
- <https://developers.google.com/edu/python/strings>
- <https://developers.google.com/edu/python/strings>

40



Tentamenvraag 2

- Wat is de uitvoer van de volgende code?

```
s = 'This is a long sentence'
print(s[:6]+s[-2])
```

- A: This nce
- B: This ice
- C: This ic
- D: ntencec

41



Wat hebben we geleerd?

- **Datatypes:**
 - Numbers
 - Strings
- **Operators:**
 - Arithmetic
 - Comparison
 - Assignment

42



Volgende les

- Voorbereiding:
 - Lees Perkovic hoofdstuk 3
 - Of bekijk <https://youtu.be/1RVZ6d2KPcc>
- Inhoud college:
 - Van taak naar ontwerp
 - Practicumopdracht

43



Opdracht

Maak thuis de opdrachten die horen bij:

- Week 1: Starten met Python | College 1

44

