GO! BUITENGEWOON ONDERWIJS
opleidingsvorm 4
Campus Impuls

Informaticaweten-schappen

Campus Impuls (2025-2026)

Dhr. Renaud Leroy – Mevr. Carmen Van Belle
CAMPUS IMPULS | WOLFPUTSTRAAT 42, 9041 GENT

Campus Impuls (2025-2026)

Campus Impuls (2025-2026)

Campus Impuls (2025-2026)

Campus Impuls (2025-2026)

Introductie

Digitale systemen en algoritmes zijn in onze maatschappij nauwelijks nog weg te denken. Van het eten dat op jouw bord komt, tot de games die je speelt, in elk aspect van ons dagelijks leven is informatica aanwezig. Soms op de achtergrond, zonder dat je erbij stil staat.

In deze cursus zal je een inleiding krijgen tot programmeren en oplossen van algoritmes. Bij het programmeren kiezen we voor Python, aangezien dit één van de meest gebruikte programmeertalen is, zowel in het professionele leven als later in het hoger onderwijs. Algoritmes zijn (vaste) stappenplannen die een computer kan uitvoeren om een taak uit te voeren. Dit kan héél simpel het berekenen van een getal zijn, tot het sorteren van data of het vinden van routes in een doolhof.

Voor het programmeren, gaan we gebruik maken van Visual Studio Code (hierna: VS Code). Dit is een populaire IDE (ontwikkelomgeving), waarin je eenvoudig extra stukjes software kan toevoegen. Ook gaan we werken met GitHub. Dit is een platform waar je code kan opladen en delen met anderen. Dit is geïntegreerd met VS Code, zodat je héél snel en zonder al te veel tussenstappen je code kan krijgen. O.a. de oplossingen op de oefeningen in deze cursus zullen via GitHub worden aangeleverd. Meer info hierover verder in de cursus.

Campus Impuls (2025-2026)

Python essentials: datatypes en variabelen

WAT IS PYTHON

Python is een programmeertaal, net als bijvoorbeeld C, C++, C#, Java en JavaScript. Met een programmeertaal kun je **instructies** geven aan een computer. Deze instructies worden uiteindelijk omgezet in machinetaal, zodat de computer ze kan uitvoeren. Dit noemen we *interpreteren*.

Één groot voordeel van Python is de **toegankelijkheid**. Andere programmeertalen bevatten vaak veel *syntax*, specifieke tekens die je moet gebruiken om code te schrijven. Python leest en schrijft vaak als een Engelstalige tekst. Neem bijvoorbeeld onderstaand stukje code:

```
1 |leeftijd = 15
2 |if leeftijd < 18:
3 |      print(“Je bent minderjarig”)
```

Als je deze code leest, zou je al een idee moeten hebben wat er gebeurd. We kunnen dit iets meer concreet uitschrijven: De leeftijd is 15 jaar. Als de leeftijd kleiner is dan 18 jaar, dan moet je printen dat de persoon minderjarig is. Maar geen zorgen, alles wordt stap voor stap uitgelegd in de volgende hoofdstukken.

Een ander voordeel van Python is de **toepassingen**. Veel informaticasystemen draaien op Python code, of kunnen d.m.v. Python worden aangestuurd. In de beginjaren van de digitale revolutie zijn er veel programmeertalen ontstaan (en verdwenen), maar sinds enkele jaren lijkt de strijd gestreden te zijn en is Python sowieso een blijver.

VARIABELEN

Elke programmeertaal, en dus ook Python, werkt met variabelen. Dit zijn stukjes computergeheugen die worden vrijgehouden om data naartoe te kunnen schrijven. Bij het programmeren onthoud je dan de locatie (adres) van het dit deeltje computergeheugen, zodat je dit kan uitlezen of iets naartoe kan schrijven.

Campus Impuls (2025-2026)

Een voorbeeld van een geheugenadres is bijvoorbeeld *0x7ffe5367e044*. Good luck! Dit is natuurlijk niet te onthouden, en dus ook niet werkbaar. Daarom kan je met variabelen werken. Dit zijn verwijzingen naar een geheugenlocatie, maar je kan die een eigen naam geven. Als we het bovenstaande voorbeeld nemen, is *leeftijd* hier een variabele. Dit verwijst naar een geheugenlocatie waarin het getal 15 zit opgeslagen.

Bij de naamgeving van een variabele, zijn er wel enkele voorwaarden / informele regels:

- Begint met een letter of underscore (_)
- Bevat enkel letters, cijfers en underscores
- Zijn hoofdlettergevoelig

In het onderstaande voorbeeld zie je een paar voorbeelden van correcte en foute variabelen.

```
1enaam = “Marcel” begint niet met letter of underscore
voor.naam = “Marcel” bevat geen speciaal teken
leeftijd = 15
Leeftijd = 18
     opgelet: leeftijd en Leeftijd zijn aparte variabelen
```

Tenslotte heb je ook nog verschillende manieren om namen van variabelen die uit meerdere woorden bestaan, leesbaar te kunnen opstellen. Bij *camelCase* en *PascalCase* schrijf je alle woorden aan elkaar, maar laat je ieder woord met een hoofdletter beginnen (*camelCase* begint met kleine letter). Bij *snake_case* schrijf je de woorden van elkaar, maar gebruik je een underscore om ze aan elkaar te koppelen. Je bent vrij om eender welke stijl te gebruiken, maar gelieve deze niet te combineren.

```
camelCase
snake_case
PascalCase
```

UITVOER (PRINT)

Om variabelen op het scherm af te drukken, gebruik je de functie *print*. Net zoals bij functies in MS Excel gebruik je haakjes bij functies om *parameters* mee te geven. Parameters zijn gegevens die een functie nodig heeft om te kunnen werken. De print functie schrijft iets op het scherm. Die heeft dus één parameter, nl. dat iets dat op het scherm moet komen.

Campus Impuls (2025-2026)

```
1 |leeftijd = 15
2 |print(leeftijd)
3 |print(3)
```

INVOER

Hierboven hebben we gezien hoe je variabelen kan aanmaken en kan afdrukken op het scherm. We kunnen echter ook tekst inlezen. Dit kan met de functie *input*. Deze functie heeft, met als *print*, haakjes met hiertussen een parameter. Dit keer is de parameter welke vraag op het scherm moet verschijnen alvorens de gebruiker tekst kan intypen. Dit is echter optioneel. Dit wil zeggen dat je ook niets kan typen (dan verschijnt er ook geen vraag). De *input* leest hetgeen de gebruiker intypt in tot er op enter wordt geduwd en geeft dit terug.

```
1 |naam = input('Wat is jouw naam? ')
2 |print(naam)
```

DATA-TYPES

Bij het programmeren werken we veel met data. Maar er zijn verschillende soorten data. Bekijk hieronder de voorstelling van een persoon:

Hallo, ik ben jullie leerkracht, *Renaud Leroy*. Ik ben *32* jaar, ben *1,70* meter groot en heb *3* kinderen. Mijn hobby's zijn: *lezen, voetbal, gamen, fietsen en muziekspelen*. Ik speel *wel* dwarsfluit maar *geen* piano. Mijn lievelingsfilm is *The Lord of the Rings: The Fellowship of the Ring*.

In *cursief* stonden alle stukjes data die interessant zouden kunnen zijn voor een algoritme. Maar niet alle data is dus van hetzelfde type.

Data	Soort data	Datatype
Renaud Leroy, LOTR: The Fellowship of the Ring	Tekst	string
32, 3	Gehele getallen	int(eger)

Campus Impuls (2025-2026)

1,70	Kommagetallen	float
Lezen, voetbal, games, fietsen, muziekspelen	Lijst	array
wel, geen	Waar/Vals	boolean

Zo heb je bijvoorbeeld stukjes tekst, zoals een naam of de titel van een film. Je hebt ook getallen, zoals een leeftijd, een aantal kinderen of een grootte, maar dit kunnen dus gehele getallen of kommagetallen zijn. Je kan ook wel of niet iets hebben, en al deze soort data kan je ook nog eens in lijsten steken. Hieronder overlopen we per datatype wat je ermee kan doen.

STRING

Een string is een stuk tekst. Eigenlijk is dit een lijst (array) van tekens (char (van character)), maar omdat tekst zoveel voorkomt bij het programmeren beschouwen we dit als een geheel.

Belangrijk is dat tekst steeds tussen aanhalingstekens moet worden geplaatst. Dit kunnen dubbele of enkele aanhalingstekens zijn.

```
1 | eerste_tekst = "Hallo, ik ben tekst"
2 | tweede_tekst = 'En ik ben ook tekst'
```

Maar je mag ze niet met elkaar combineren om een string te openen en af te sluiten.

```
1 | tekst = 'Hallo, ik ben tekst"
```

Je kan ze wel met elkaar gebruiken, maar dan om deze als deel van de tekst te gebruiken:

```
1 | zin = 'Ik zei: "Hallo!" '
2 | tekst = "Ik ga 'direct' mijn huiswerk maken."
```

Als je ze allebei wil gebruiken, kan je gebruik maken van een \. Deze plaats je voor het aanhalingsteken, zodat dit niet als het einde van de string wordt beschouwd maar als een teken van de tekst.

Campus Impuls (2025-2026)

```
1 | zin = "Ik zei: \"Ik ga 'direct' mijn huiswerk maken.\""
```

Je kan ook teksten met elkaar combineren met een +-teken.

```
1 | naam = "Obama"
2 | voornaam = "Barack"
3 | president = voornaam + naam            ("BarackObama")
```

Het +-teken heeft echter het nadeel dat dit bij andere datatypes niet compatibel is. Beter is om een *format* te gebruiken. Hiervoor typ je een *f* voor de een string en gebruik je accolades als placeholder om variabelen in de string in te voegen.

```
1 | leeftijd = 64
2 | voornaam = "Barack"
3 | naam = "Obama"
4 | president = f"{voornaam} {naam} is {leeftijd} jaar."
```

BASISOEFENINGEN

1. Laat de gebruiker een naam intypen ("Naam: ") en toon deze op het scherm.
2. Laat de gebruiker een kleur intypen ("Kleur: ") en toon deze op het scherm.
3. Laat de gebruiker een naam intypen ("Naam: ") en toon op het scherm "Hallo ", gevolgd door de naam.
4. Laat de gebruiker een titel van een film intypen ("Film: ") en toon op het scherm "Ik kijk graag naar ", gevolgd door de titel van de film en tenslotte nog een uitroepingsteken.

BASISOEFENINGEN: OPLOSSING

1. Laat de gebruiker een naam intypen ("Naam: ") en toon deze op het scherm.

Campus Impuls (2025-2026)

```
1 |naam = input("Naam: ")
2 |print(naam)
```

2. Laat de gebruiker een kleur intypen ("Kleur: ") en toon deze op het scherm.

```
1 |naam = input("Kleur: ")
2 |print(naam)
```

3. Laat de gebruiker een naam intypen ("Naam: ") en toon op het scherm "Hallo ", gevolgd door de naam.

```
1 |naam = input("Naam: ")
2 |print(f'Hallo {naam}')
```

4. Laat de gebruiker een titel van een film intypen ("Film: ") en toon op het scherm "Ik kijk graag naar ", gevolgd door de titel van de film en tenslotte nog een uitroepingsteken.

```
1 |titel = input("Film: ")
2 |print(f'Ik kijk graag naar {titel}!')
```

OEFENINGEN: STRING

a. Schrijf de tekst "Hello World!" op het scherm.

b. Maak een variabele (*naam)* en schrijf hier "John" in weg. Schrijf vervolgens op het scherm: "Hallo *John*"

c. Vraag aan de gebruiker diens naam (*"Wat is jouw naam? "*) en de titel van diens lievelingsgame (*"Wat is jouw favoriete game? "*). Schrijf vervolgens op het scherm: *naam* speelt graag *titel*!

Campus Impuls (2025-2026)

d. Vraag naar een favoriete quote uit een film (*"Favoriete quote: "*) en schrijf op het scherm: Mijn favoriete quote: "*qoute*"

e. Schrijf de volgende tekst op het scherm, zonder gebruik te maken van variabelen: *Ik zei tegen hem: "Ik bel je 's avonds"*

f. Vraag aan de gebruiker de naam van diens favoriete huisdier ("Wat is jouw favoriete huisdier? ") en schrijf: Ik hou zo van *huisdier.*

g. Vraag aan de gebruikers eerst diens naam (*"Naam: "*), daarna diens favoriete film (*"Favoriete film: "*) en toon het volgende op het scherm: Hallo *naam*, ik kijk ook graag naar *film*.

OEFENINGEN: STRING (OPLOSSINGEN)

1. Schrijf de tekst "Hello World!" op het scherm.

```
1 | print("Hello World!")
```

2. Maak een variabele (*naam)* en schrijf hier "John" in weg. Schrijf vervolgens op het scherm: "Hallo *John*"

```
1 | naam = "John"
2 | print(naam)
```

3. Vraag de gebruiker om een naam in te geven ("Naam: "). Vervolgens schrijf je de naam weg op het scherm.

```
1 | naam = input("Naam: ")
2 | print(naam)
```

Campus Impuls (2025-2026)

4. Vraag aan de gebruiker diens naam (*"Wat is jouw naam? "*) en de titel van diens lievelingsgame (*"Wat is jouw favoriete game? "*). Schrijf vervolgens op het scherm: *naam* speelt graag *titel*!

```
naam = input("Wat is jouw naam? ")
titel = input("Wat is jouw favoriete game? ")
print(f"{naam} speelt graag {titel}!")
```

5. Vraag naar een favoriete quote uit een film (*"Favoriete quote: "*) en schrijf op het scherm: Mijn favoriete quote: "*quote*"

```
quote = input("Favoriete quote: ")
print(f'Mijn favoriete quote: "{quote}"')
```

6. Schrijf de volgende tekst op het scherm, zonder gebruik te maken van variabelen: *Ik zei tegen hem: "Ik bel je 's avonds"*

```
print('Ik zei tegen hem: "Ik bel je \'s avonds"')
```

7. Vraag aan de gebruiker de naam van diens favoriete huisdier ("Wat is jouw favoriete huisdier? ") en schrijf: Ik hou zo van *huisdier.*

```
huisdier = input("Wat is jouw favoriete huisdier? ")
print(f"Ik hou zo van {huisdier}.")
```

Campus Impuls (2025-2026)

8. Vraag aan de gebruikers eerst diens naam (*"Naam: "*), daarna diens favoriete film (*"Favoriete film: "*) en toon het volgende op het scherm: Hallo *naam*, ik kijk ook graag naar *film*.

```
naam = input("Naam: ")
film = input("Favoriete film: ")
print(f"Hallo {naam}, ik kijk ook graag naar {film}.")
```

TOETSVRAGEN

1. Maak een programma dat aan de gebruiker diens naam en leeftijd vraagt. Vervolgens schrijf je op het scherm: Naam is leeftijd jaar oud.
2. Maak een programma dat aan de gebruiker diens naam en computerspel vraagt. Vervolgens schrijf je op het scherm: Naam speelt graag computerspel!
3. Schrijf op het scherm, met één commando (dus zonder variabelen), het volgende: Ik zei ‘s ochtends tegen mijn ouders: “Ik vertrek nu naar school!”
4. Schrijf op het scherm, met één commando (dus zonder variabelen), het volgende: ‘s Avonds kijk ik op tv graag naar “Blokken”.

TOETSVRAGEN (OPLOSSINGEN)

1. Maak een programma dat aan de gebruiker diens naam en leeftijd vraagt. Vervolgens schrijf je op het scherm: Naam is leeftijd jaar oud.

```
naam = input("Naam: ")
leeftijd = input("Leeftijd: ")
print(f'{naam} is {leeftijd} jaar oud')
```

2. Maak een programma dat aan de gebruiker diens naam en computerspel vraagt. Vervolgens schrijf je op het scherm: Naam speelt graag computerspel!

```
naam = input("Naam: ")
spel = input("Computerspel: ")
print(f'{naam} speelt graag {spel}!')
```

Campus Impuls (2025-2026)

3. Schrijf op het scherm, met één commando (dus zonder variabelen), het volgende: Ik zei ‘s ochtends tegen mijn ouders: “Ik vertrek nu naar school!”

```
1 | print('Ik zei \'s ochtends tegen mijn ouders: "Ik vertrek nu
  naar school!"')
```

4. Schrijf op het scherm, met één commando (dus zonder variabelen), het volgende: ‘s Avonds kijk ik op tv graag naar “Blokken”.

```
1. print("'s Avonds kijk ik op tv graag naar \"Blokken\".")
```

INT

Net als bij andere programeertalen is er een verschil tussen een kommagetal en een geheel getal. Een *int* is een geheel getal. Hiermee kan je de standaardbewerkingen +, -, * (maal) en / doen. Let op: er zijn 2 soorten delingen. Met één enkele / voer je een gewone deling uit. Als je echter 2 / gebruikt, dan voer je een deling uit waarbij je afrond naar het dichtste gehele getal naar onder.

```
1 | a = 3
2 | b = a + 1
3 | c = 9 / b
4 | d = 9 // b
```

Hierboven is de waarde van b dus 4 (3 + 1), c is dan 2.25 en d is dan gewoon 2.

Je kan uiteraard ook haakjes gebruiken om voorrang te geven aan bepaalde bewerkingen.

Handig is ook het gebruik van **. Hiermee kan je namelijk een macht uitrekenen. Ervoor schrijf je dan het grondtal, erachter de exponent. Zo is de waarde van a in het onderstaande 9.

Campus Impuls (2025-2026)

```
1 |a = 3 ** 2
```

Bij wiskunde heb je echter ook gezien hoe je wortels (bv. vierkantswortel) kan uitrekenen d.m.v. een macht. Door de inverse van de exponent te nemen, kan je dus ook n^{de}-machtswortels berekenen.

```
1 |a = 9 ** (1/2)
```

Tenslotte is er ook nog de modulo. Dit is de restberekening van een deling tussen twee getallen. Zo zie je in het onderstaande voorbeeld dat 5 % 3 de restwaarde is van 5 te delen door 3. 3 past 1 keer in 5, dan hou je nog 2 over, dus 5 % 3 is dus gelijk aan 2.

```
1 |var restwaarde = 5 % 3;
```

FLOAT

Float is het representeren van een rationaal getal. Let op: zelfs ook bijvoorbeeld 3,0 is een kommagetal. De reden waarom *float* en *int* dan nog apart bestaan, heeft te maken met de geheugenallocatie van deze variabelen. Beiden gebruiken 32 bits. Dit wil zeggen dat wanneer je een variabele aanmaakt van het type *int*, deze 32 bits computergeheugen reserveren. Ook al gebruiken ze niet alle bits om hun getal te onthouden, toch worden deze 32 bits vastgelegd en kunnen die niet door een andere variabele worden gebruikt. Zo bestaat het getal 5 uit 3 bits (101), maar toch zullen de andere 29 bits (allemaal 0), toch niet beschikbaar blijven voor andere variabelen.

Zo heeft een *int* met de 32 beschikbare bits een bereik van -2 147 483 648 tot 2 147 483 647, en heeft *float* 'maar' een bereik van ±16 777 216. Uiteraard kan deze wel meer significant worden op het gedeelte na de komma, afhankelijk van hoeveel bits daarvoor resteren.

Let op: in Python is Engels de voortaal, dus schrijf je decimale getallen met een punt (bv. 3.0)

Campus Impuls (2025-2026)

```
1 |a = float("1,2")
           ^^^^^^^^^^^^
2 |ValueError: could not convert string to float: '1,2'
```

Met de functie *round()* kan je tenslotte ook getallen afronden. Je geeft hier 2 parameters mee. Eerst het getal dat je wenst af te ronden, tenslotte het aantal decimalen. Tip: je kan bij het aantal decimalen ook een negatief aantal gebruiken om zo op tientallen, hondertallen, etc. af te ronden.

```
1 |getal = 172.238
2 |getal_172_komma_24 = round(getal,2)
3 |getal_200 = round(getal,-2)
```

CONVERTIES

Misschien had je er al op gelet, maar *input()* geeft een *string* terug (tekst). Dit is echter een probleem voor getallen, want de *string* (lees: tekst) 5 – 3 (lees: getal 3), heeft geen enkele betekenis. Daarom krijg je ook een foutmelding als je het volgende probeert, met als invoer "3":

```
1 |a = input("Geef een getal: ")
2 |b = a + 1
3 |
4 |b = a + 1
5 |    ~~^~~
6 |TypeError: unsupported operand type(s) for +: 'str' and 'int'
```

Je moet dus de tekst (bv. "3") omzetten naar het getal (bv. 3). Dit kan met de functie *int()* (en *float()* voor kommagetallen). Wat er gebeurt wanneer we een ongeldige waarde ingeven, bv. "abc", zien we later.

Je kan ervoor kiezen om de conversie in één of twee stappen te doen. Ofwel schrijf je eerst de *string*-waarde van *input()* in een variabele weg, en wijs je in een tweede stap aan de variabele een nieuwe waarde toe, nl. het resultaat van de conversie-functie, die als enige parameter de oude waarde krijgt. Of je doet de twee stappen tegelijk,

Campus Impuls (2025-2026)

waarbij je de uitvoer van *input()* rechtstreeks naar de conversie-functie schrijft, die dan aan een variabele wordt toegewezen.

```
1 |a = int(input("a = "))
2 |b = input("b = ")
3 |b = int(b)
4 |c_tekst = input("c = ")
5 |c = int(c_tekst)
```

BASISOEFENINGEN: INT

1. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), tel hier 1 bij op en schrijf dit op het scherm.
2. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), trek hier 2 van af en schrijf dit op het scherm.
3. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), vermenigvuldig dit met 3 en schrijf dit op het scherm.
4. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), deel dit door 2 en schrijf dit op het scherm.
5. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), bereken het kwadraat en schrijf dit op het scherm.
6. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), bereken de vierkantswortel en schrijf dit op het scherm. Tip: een vierkantswortel is de machtsverheffing met exponent 0.5 (of 1/2). Als je math.sqrt gebruikt, krijg je een ander resultaat bij de test

BASISOEFENINGEN: INT (OPLOSSINGEN)

a. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), tel hier 1 bij op en schrijf dit op het scherm.

Campus Impuls (2025-2026)

```
getal_tekst = input("Getal: ")
getal = int(getal_tekst)
getal = getal + 1
print(getal)
```

b. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), trek hier 2 van af en schrijf dit op het scherm.

```
getal_tekst = input("Getal: ")
getal = int(getal_tekst)
getal = getal - 2
print(getal)
```

c. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), vermenigvuldig dit met 3 en schrijf dit op het scherm.

```
getal = int(input("Getal: "))
getal = getal * 3
print(getal)
```

d. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), deel dit door 2 en schrijf dit op het scherm.

```
getal = int(input("Getal: "))
getal = getal / 2
print(getal)
```

e. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), bereken het kwadraat en schrijf dit op het scherm.

```
getal = int(input("Getal: "))
getal = getal ** 2
print(getal)
```

f. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), bereken de vierkantswortel en schrijf dit op het scherm. Tip: een

Campus Impuls (2025-2026)

vierkantswortel is de machtsverheffing met exponent 0.5 (of 1/2). Als je math.sqrt gebruikt, krijg je een ander resultaat bij de test

```
getal = int(input("Getal: "))
getal = getal ** (1/2)
print(getal)
```

BASISOEFENINGEN: FLOAT

1. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (int), tel hier 1.5 bij op en schrijf dit op het scherm.
2. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (float), trek hier 2.3 van af en schrijf dit op het scherm.
3. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (float), vermenigvuldig dit met 3 en schrijf dit op het scherm.
4. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (float), deel dit door 2 en schrijf dit op het scherm.
5. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (float), bereken het kwadraat en schrijf dit op het scherm.
6. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), bereken de vierkantswortel en schrijf dit op het scherm. Tip: een vierkantswortel is de machtsverheffing met exponent 0.5 (of 1/2). Als je math.sqrt gebruikt, krijg je een ander resultaat bij de test.

BASISOEFENINGEN: FLOAT (OPLOSSINGEN)

1. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (int), tel hier 1.5 bij op en schrijf dit op het scherm.

```
getal = float(input("Getal: "))
getal = getal + 1.5
print(getal)
```

Campus Impuls (2025-2026)

2. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (float), trek hier 2.3 van af en schrijf dit op het scherm.

```
1 |getal = float(input("Getal: "))
2 |getal = getal - 2.3
3 |print(getal)
```

3. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (float), vermenigvuldig dit met 3 en schrijf dit op het scherm.

```
1 |getal = float(input("Getal: "))
2 |getal = getal * 3
3 |print(getal)
```

4. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (float), deel dit door 2 en schrijf dit op het scherm.

```
1 |getal = float(input("Getal: "))
2 |getal = getal / 2
3 |print(getal)
```

5. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een kommagetal (float), bereken het kwadraat en schrijf dit op het scherm.

```
1 |getal = float(input("Getal: "))
2 |getal = getal ** 2
3 |print(getal)
```

6. Laat de gebruiker een getal intypen ("Getal: "). Vorm dit om naar een geheel getal (int), bereken de vierkantswortel en schrijf dit op het scherm. Tip: een vierkantswortel is de machtsverheffing met exponent 0.5 (of 1/2). Als je math.sqrt gebruikt, krijg je een ander resultaat bij de test.

```
1 |getal = float(input("Getal: "))
2 |getal = getal ** (1/2)
3 |print(getal)
```

Campus Impuls (2025-2026)

BASISOEFENINGEN: OPERATOREN (%, //)

1. Vraag een geheel getal a ("a: ") en een geheel getal b ("b: "). Vervolgens bereken je de rest na deling van getal a gedeeld door getal b.
2. Vraag een de gebruiker hoeveel pizza's ("Aantal pizza's: ") hij besteld heeft en hoeveel mensen er willen mee eten ("Aantal eters: "). Een pizza wordt in 8 stukken verdeeld. Bereken je hoeveel stukken pizza er over zullen zijn als iedereen evenveel stukken eet.
3. Vraag een geheel getal a ("a: ") en een geheel getal b ("b: "). Vervolgens bereken je de deling van getal a door getal b, afgerond naar onder (dus zonder kommagedeelte).
4. Vraag een de gebruiker hoeveel pizza's ("Aantal pizza's: ") hij besteld heeft en hoeveel mensen er willen mee eten ("Aantal eters: "). Een pizza wordt in 8 stukken verdeeld. Bereken je hoeveel stukken pizza er elk kunnen worden gegeten als iedereen evenveel stukken eet.

BASISOEFENINGEN: OPERATOREN (OPLOSSINGEN)

1. Vraag een geheel getal a ("a: ") en een geheel getal b ("b: "). Vervolgens bereken je de rest na deling van getal a gedeeld door getal b.

```
1 |a = int(input("a: "))
2 |b = int(input("b: "))
3 |rest = a % b
4 |print(rest)
```

2. Vraag een de gebruiker hoeveel pizza's ("Aantal pizza's: ") hij besteld heeft en hoeveel mensen er willen mee eten ("Aantal eters: "). Een pizza wordt in 8 stukken verdeeld. Bereken je hoeveel stukken pizza er over zullen zijn als iedereen evenveel stukken eet.

Campus Impuls (2025-2026)

```
pizzas = int(input("Aantal pizza's: "))
eters = int(input("Aantal eters: "))
rest = pizzas*8 % eters
print(f'{rest} stukken')
```

3. Vraag een geheel getal a ("a: ") en een geheel getal b ("b: "). Vervolgens bereken je de deling van getal a door getal b, afgerond naar onder (dus zonder kommagedeelte).

```
a = int(input("a: "))
b = int(input("b: "))
rest = a // b
print(rest)
```

4. Vraag een de gebruiker hoeveel pizza's ("Aantal pizza's: ") hij besteld heeft en hoeveel mensen er willen mee eten ("Aantal eters: "). Een pizza wordt in 8 stukken verdeeld. Bereken je hoeveel stukken pizza er elk kunnen worden gegeten als iedereen evenveel stukken eet.

```
pizzas = int(input("Aantal pizza's: "))
eters = int(input("Aantal eters: "))
rest = pizzas*8 // eters
print(f'{rest} stukken')
```

OEFENINGEN

a. Maak een programma dat 2 getallen aan de gebruiker vraagt ("a = ", "b = ") en van deze getallen het volgende toont: som, verschil, product, quotiënt, macht, wortel.

b. Maak een programma dat een invoer 'x' bevraagt ("x = ") en vervolgens de functiewaarde van de volgende functies berekent:

- $x^2 + 2x - 3$
- $x^4 - 3x^3 + \frac{1}{4}x - 9$

Campus Impuls (2025-2026)

- $x^5 - x^4$

c. Maak een programma dat de lengte ("Lengte (m): ") en het gewicht ("Gewicht (kg): ") van iemand bevraagd. Bereken vervolgens het BMI van deze persoon ("BMI = "). Rond af tot een geheel getal. De berekening hiervoor is $\frac{gewicht}{lengte \cdot lengte}$.

d. Maak een programma dat aan de gebruiker een aantal pizza's vraag die hij wenst te bestellen ("Aantal pizza's: "). Een pizza kost € 14,50 euro, en je hebt een speciale deal: 4 + 1 gratis! Laat de gebruiker het aantal pizza's ingeven en bereken de totale prijs.

e. Toon van een getal enkel het kommagedeelte. Bv. bij 14,245 toon je 0,245.

OEFENINGEN (OPLOSSINGEN)

a. Maak een programma dat 2 getallen aan de gebruiker vraagt ("a = ", "b = ") en van deze getallen het volgende toont: som, verschil, product, quotiënt, macht, wortel.

```
1 | a = int(input("a = "))
2 | b = int(input("b = "))
3 | print(f"som = {a + b}")
4 | print(f"verschil = {a - b}")
5 | print(f"product = {a * b}")
6 | print(f"quotiënt = {a / b}")
7 | print(f"macht = {a ** b}")
8 | print(f"wortel = {a ** (1/b)}")
```

b. Maak een programma dat een invoer 'x' bevraagt ("x = ") en vervolgens de functiewaarde van de volgende functies berekent:

- $x^2 + 2x - 3$

Campus Impuls (2025-2026)

```
x = int(input("x = "))
y = x**2 + 2*x - 3
print(f"x²+2x-3 met x = {x} is {y}")
```

- $x^4 - 3x^3 + \frac{1}{4}x - 9$

```
x = int(input("x = "))
y = x**2 + 2*x - 3
print(f"x²+2x-3 met x = {x} is {y}")
```

- $x^5 - x^4$

```
x = int(input("x = "))
y = x**5 - x**4
print(f"x^5 - x^4 met x = {x} is {y}")
```

c. Maak een programma dat de lengte ("Lengte (m): ") en het gewicht ("Gewicht (kg): ") van iemand bevraagd. Bereken vervolgens het BMI van deze persoon ("BMI = "). Rond af tot een geheel getal. De berekening hiervoor is $\frac{gewicht}{lengte \cdot lengte}$.

```
l = float(input('Lengte (m): '))
g = float(input('Gewicht (kg): '))
bmi = g/(l*l)
print(f'BMI = {bmi}')
```

d. Maak een programma dat aan de gebruiker een aantal pizza's vraag die hij wenst te bestellen ("Aantal pizza's: "). Een pizza kost € 14,50 euro, en je hebt een speciale deal: 4 + 1 gratis! Laat de gebruiker het aantal pizza's ingeven en bereken de totale prijs.

Campus Impuls (2025-2026)

```
aantal = int(input("Aantal pizza's: "))
prijs = 14.5
pizzas_per_deal = 4 + 1
prijs_per_deal = 4 * prijs
aantal_keer_deal = aantal // pizzas_per_deal
aantal_pizzas_buiten_deal = aantal % pizzas_per_deal
# of aantal_pizzas_buiten_deal = aantal - (aantal_keer_deal * pizzas_per_deal)
totaal = (aantal_keer_deal * prijs_per_deal) + (aantal_pizzas_buiten_deal * prijs)
print(f'Prijs: {totaal} euro')
```

e. Toon van een getal enkel het kommagedeelte. Bv. bij 14,245 toon je 0,245.

```
getal = float(input("Getal: "))
kommagedeelte = getal % 1
print(kommagedeelte)
```

TOETSVRAGEN

1. Maak 2 variabelen (gehele getallen), a en b, en geef deze zelf waarden (moet dus niet door de gebruiker worden ingegeven) en bereken het volgende: $\sqrt{\frac{a \cdot b}{a/b}}$ Toon vervolgens het resultaat op het scherm.
2. Maak 2 variabelen (gehele getallen), a en b, en geef deze zelf waarden (moet dus niet door de gebruiker worden ingegeven) en bereken het volgende: $\left(\frac{a+b}{a-b}\right)^2$ Toon vervolgens het resultaat op het scherm.
3. De lokale supermarkt geeft een speciale korting: 3+1 op alle frisdranken. Een blikje cola kost 1,2 euro. Je koopt er 10. Reken uit hoeveel je betaalt door deze 10 in een variabele te steken.
4. Op Steam is er in de categorie van games voor € 9,99 een promotie: 4 + 1 gratis. Zo betaal je in plaats van € 49,95 maar € 39,96. Bereken van een variabele aantal, de totaalprijs. De promotie kan meerdere keren worden

Campus Impuls (2025-2026)

toegekend. Koop je bijvoorbeeld 13 games, dan heb je 2 keer de korting van 4 + 1 en betaal je de laatste 3 games nog aan 'volle prijs' (€ 9,99).

TOETSVRAGEN (OPLOSSING)

1. Maak 2 variabelen (gehele getallen), a en b, en geef deze zelf waarden (moet dus niet door de gebruiker worden ingegeven) en bereken het volgende: $\sqrt{\frac{a \cdot b}{a/b}}$
   Toon vervolgens het resultaat op het scherm.

```
1 |a = 4
2 |b = 3
3 |resultaat = ((a*b)/(a/b))** (1/2)
4 |print(resultaat)
```

2. Maak 2 variabelen (gehele getallen), a en b, en geef deze zelf waarden (moet dus niet door de gebruiker worden ingegeven) en bereken het volgende: $\left(\frac{a+b}{a-b}\right)^2$
   Toon vervolgens het resultaat op het scherm.

```
1 |a = 4
2 |b = 3
3 |resultaat = ((a+b)/(a-b))**2
4 |print(resultaat)
```

3. De lokale supermarkt geeft een speciale korting: 3+1 op alle frisdranken. Een blikje cola kost 1,2 euro. Je koopt er 10. Reken uit hoeveel je betaalt door deze 10 in een variabele te steken.

Campus Impuls (2025-2026)

```
1 |aantal = 10
2 |aantal_korting = 3+1
3 |prijs = 1.2
4 |prijs_korting = 3*prijs
5 |totaal = (aantal // aantal_korting)*prijs_korting + (aantal %
  aantal_korting)*prijs
6 |print(totaal)
```

4. Op Steam is er in de categorie van games voor € 9,99 een promotie: 4 + 1 gratis. Zo betaal je in plaats van € 49,95 maar € 39,96. Bereken van een variabele aantal, de totaalprijs. De promotie kan meerdere keren worden toegekend. Koop je bijvoorbeeld 13 games, dan heb je 2 keer de korting van 4 + 1 en betaal je de laatste 3 games nog aan 'volle prijs' (€ 9,99).

```
1 |prijs = 9.99
2 |promotie = 4+1
3 |prijs_promotie = 4*prijs
4 |aantal = 12
5 |totaal = (aantal // promotie)*prijs_promotie + (aantal %
  promotie)*prijs
6 |print(totaal)
```

Campus Impuls (2025-2026)

List

Een variabele kan ook een *list* zijn. Een list is een datatype dat een lijst van verschillende waarden uitmaakt en wordt als volgt opgesteld:

```
1 | nummers = []
```

Let op: lijsten zijn **zero-based**,
dat wil zeggen dat ze vanaf 0 beginnen te tellen.
nummers[0] wijst dus naar de eerste plaats in de lijst.

Enkele interessante functie om te beginnen met *List* zijn:

- *append(element)*: voegt een nieuw element toe aan het einde van de lijst.
- *insert(positie,element)*: voegt een nieuw element toe op een specifieke positie. Ook hier verwijst de positie 0 naar de voorste (eerste) positie in de lijst.
- *remove(element):* verwijdert een bepaalde waarde uit de lijst. Let op: *Remove* gaat de lijst van voor naar achter af en verwijdert het eerste element dat hij tegen komt. Komt het meegegeven element meerdere keren voor, dan wordt het slecht een eerste keer verwijderd.
- *del naam_lijst[positie]*: verwijdert een bepaalde waarde op een bepaalde plaats uit de lijst.

Hieronder vind je een voorbeeld van een lijst met getallen. Er worden 3 getallen (nl. 1, 2, 3) toegevoegd en vervolgens het eerste getal (1) getoond op het scherm. Dan wordt eerst getal 1 verwijderd het tweede getal (3) verwijderd.

```
1 | nummers = []
2 | nummers.append(1)
3 | nummers.append(2)
4 | nummers.append(3)
5 | print(nummers[0])
6 | nummers.remove(1)
7 | del nummers[1]
```

FOREACH

Om een collectie te overlopen, kan je een for each loop gebruiken. Deze gaat een bepaald stuk code voor elk element in de collectie uitvoeren. Zie onderstaand voorbeeld waarbij voor elk getal in de collectie deze op het scherm wordt getoond. De

Campus Impuls (2025-2026)

naam voor de variabele bij het overlopen van de variabelen (hier: nummer) mag je vrij kiezen.

```
1 | nummers = [1,2,3]
2 | for nummer in nummers:
3 |     print(nummer)
```

FOR

Een *for-lus* wordt gebruikt om een bepaalde *int* (hieronder bv. *index*) vanaf een bepaalde startwaarde naar een eindwaarde te laten gaan, met een bepaalde verhoging/verlaging per keer. Hiervoor wordt de functie *range* gebruikt. Deze geeft een lijst van getallen terug die je dan over uit 3 delen:

1. Je stelt een variabele in met een startwaarde
2. Je bepaalt tot wanneer de herhaling wordt uitgevoerd (excl.)
3. Je verhoogt/verlaagt de variabele met een bepaalde waarde

In het onderstaande voorbeeld, start de variabele *i* met de waarde 1, blijft de herhaling duren zolang *i* kleiner is dan 5 en verhoogt *i* per herhaling met *1*. De uitvoer is dus 1, 2, 3 en 4.

```
1 |for i in range(1,5,1):
2 |     print(i)
```

In het onderstaande voorbeeld, start de variabele *i* met de waarde 5, blijft de herhaling duren zolang *i* groter is dan 1 en verlaagt *i* per herhaling met *1*. De uitvoer is dus 5, 4, 3, 2

```
1 |for i in range(5,1,-1):
2 |     print(i)
```

De start en eindpositie kan je dus vrij kiezen, al moet je natuurlijk ervoor zorgen dat de eindpositie ooit kan worden bereikt. In het onderstaande scenario verhoogt de variabele steeds meer, terwijl de eindpositie lager ligt (0) dan de startpositie (3).

```
1 |for i in range(3,0,1):
2 |     print(i)
```

Campus Impuls (2025-2026)

Een *for-lus* wordt vaak gebruikt in combinatie met de lengte van een array/list. Het nadeel van *foreach* is dat je, zolang je de *array/list* overloopt, moeilijk wijzigingen hierop kan uitvoeren.

Neem het onderstaande voorbeeld:

```
1 |nummers = [1,2,3]
2 |for nummer in nummers:
3 |    nummers.remove(nummer)
4 |    print(nummers)
5 |print("Alles verwijderd")
6 |print(nummers)
```

Hier gaat het getal 2 'onverwachts in blijven staan'. Dit komt omdat er alvorens er wordt overlopen een iterator wordt opgesteld, die wijzigingen aan de lijst niet actief meeneemt.

Beter is dus:

```
1 |nummers = [1,2,3]
2 |for index in range(len(nummers)):
3 |    del nummers[index]
4 |print("Alles verwijderd")
5 |print(nummers)
```

SLICING

'Slicing' is een manier om een deel uit een lijst te halen. Dit kan d.m.v. de volgende syntax:

naam_lijst[start_index (incl.) **:** eind_index (excl.) **:** stapgrootte **]**

Hier zal dus een stuk uit de lijst gehaald worden vanaf de beginpositie (vanaf 0) tot (niet tot en met) de eindpositie. Deze eindpositie kan buiten het bereik van de lijst liggen zonder hierop foutmeldingen te krijgen. De stapgrootte bepaald om de hoeveel elementen er elementen worden genomen. In de onderstaande voorbeelden gaan we de volgende lijst gebruiken:

Campus Impuls (2025-2026)

```
1 |letters = [a, b, c, d, e, f, g, h, i, j]
```

Soms ga je zien dat een bepaald getal niet is ingevuld. Dit is omdat deze optioneel zijn. Is de beginpositie niet ingevuld, dan begint men bij positie 0. Is de eindpositie niet ingevuld, dan wordt de lengte van de lijst genomen (tenzij stap negatief). Is de stapgrootte niet ingevuld, dan wordt een 1 als stapgrootte gekozen.

Hieronder zie je een aantal voorbeelden van ‘slicing’, met telkens de elementen die worden geselecteerd.

Campus Impuls (2025-2026)

	`letters[0:2]` *geen stap, dus stap = 1, vanaf 0 (begin) tot 2*	a, b
	`letters[1:5]` *van 1 (begin = 0) tot 5*	b, c, d, e
	`letters[:2]` *geef beginpositie dus vanaf 0*	a, b
	`letters[7:]` *geef eindpositie dus tot einde*	h, i, j
	`letters[0:5:2]` *indexen 0, 2 en 4 (want tot 4)*	a, c, e
	`letters[0:4:2]` *indexen 0 en 2 (want tot 4, niet tot en met 4)*	a, c
	`letters[:4:2]` *geef startpositie dus vanaf 0*	a, c
	`letters[2:0:-1]` *stap -1 dus van achter naar voor*	c, b

Campus Impuls (2025-2026)

	`letters[2:-1:-1]` *leeg want eindpositie moet tot bereik behoren*	
	`letters[2::-1]`	c, b, a
	`letters[-3::]` *negatieve index betekenen de positie vanaf het einde gezien (hier de op 2 na laatste positie)*	h, i, j

BASISOEFENINGEN: LIST

1. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: 1, 2, 3, 4 en print de lijst op het scherm (print(naam_lijst)).
2. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: Appel, Banaan, Citroen en print de lijst op het scherm (print(naam_lijst)).
3. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: Gent, Antwerpen, Brussel, Brugge en print de lijst op het scherm.
4. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: Spaghetti, Frietjes, Soep en print de lijst op het scherm.
5. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: 3.5, 1.2, 2.4, 9.0, 8.7 en print de lijst op het scherm.

Campus Impuls (2025-2026)

BASISOEFENINGEN: LIST (OPLOSSING)

1. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: 1, 2, 3, 4 en print de lijst op het scherm (print(naam_lijst)).

```
getallen = []
getallen.append(1)
getallen.append(2)
getallen.append(3)
getallen.append(4)
print(getallen)
```

2. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: Appel, Banaan, Citroen en print de lijst op het scherm (print(naam_lijst)).

```
fruit = []
fruit.append('Appel')
fruit.append('Banaan')
fruit.append('Citroen')
print(fruit)
```

3. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: Gent, Antwerpen, Brussel, Brugge en print de lijst op het scherm.

```
steden = []
steden.append('Gent')
steden.append('Antwerpen')
steden.append('Brussel')
steden.append('Brugge')
print(steden)
```

4. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: Spaghetti, Frietjes, Soep en print de lijst op het scherm.

Campus Impuls (2025-2026)

```
gerechten = []
gerechten.append('Spaghetti')
gerechten.append('Frietjes')
gerechten.append('Soep')
print(gerechten)
```

5. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: 3.5, 1.2, 2.4, 9.0, 8.7 en print de lijst op het scherm.

```
getallen = []
getallen.append(3.5)
getallen.append(1.2)
getallen.append(2.4)
getallen.append(9.0)
getallen.append(8.7)
print(getallen)
```

BASISOEFENINGEN: INSERT

1. Maak een lijst en steek steeds vooraan in de lijst de volgende waarden:1, 2, 3 en print deze lijst op het scherm.
2. Maak een lijst en steeds vooraan in de lijst de volgende waarden: Blauw, Geel, Zwart, Rood en print deze lijst op het scherm.
3. Maak een lijst en steeds vooraan in de lijst de volgende waarden: 3.2, 4.5, 9.0, 1.3 en print deze lijst op het scherm.
4. Maak een lijst en steeds vooraan in de lijst de volgende waarden: 3, 4, 5, 6 en print deze lijst op het scherm.

BASISOEFENINGEN: INSERT (OPLOSSINGEN)

1. Maak een lijst en steek steeds vooraan in de lijst de volgende waarden:1, 2, 3 en print deze lijst op het scherm.

Campus Impuls (2025-2026)

```
getallen = []
getallen.insert(0, 1)
getallen.insert(0, 2)
getallen.insert(0, 3)
print(getallen)
```

2. Maak een lijst en steek steeds vooraan in de lijst de volgende waarden: Blauw, Geel, Zwart, Rood en print deze lijst op het scherm.

```
kleuren = []
kleuren.insert(0,"Blauw")
kleuren.insert(0,"Geel")
kleuren.insert(0,"Zwart")
kleuren.insert(0,"Rood")
print(kleuren)
```

3. Maak een lijst en steek steeds vooraan in de lijst de volgende waarden: 3.2, 4.5, 9.0, 1.3 en print deze lijst op het scherm.

```
getallen = []
getallen.insert(0, 3.2)
getallen.insert(0, 4.5)
getallen.insert(0, 9.0)
getallen.insert(0, 1.3)
print(getallen)
```

4. Maak een lijst en steek steeds vooraan in de lijst de volgende waarden: 3, 4, 5, 6 en print deze lijst op het scherm.

Campus Impuls (2025-2026)

```
getallen = []
getallen.insert(0, 3)
getallen.insert(0, 4)
getallen.insert(0, 5)
getallen.insert(0, 6)
print(getallen)
```

BASISOEFENINGEN: FOR

1. Maak een lijst met de getallen 1, 2, 3, 4 en toon elk van deze getallen apart op het scherm.
2. Maak een lijst met de namen Anna, Bert en Charlie en toon elk van deze namen apart op het scherm.
3. Maak een lijst met de getallen 3.0, 1.4, 5.6, 9.0 en toon elk van deze getallen apart op het scherm.
4. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: Spaghetti, Frietjes, Soep en toon elk van deze waarden apart op het scherm.
5. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: 3.5, 1.2, 2.4, 9.0, 8.7 en vervolgens toon je deze getallen met 'Getal: ' ervoor.

BASISOEFENINGEN: FOR (OPLOSSINGEN)

1. Maak een lijst met de getallen 1, 2, 3, 4 en toon elk van deze getallen apart op het scherm.

```
getallen = [1,2,3,4]
for getal in getallen:
    print(getal)
```

2. Maak een lijst met de namen Anna, Bert en Charlie en toon elk van deze namen apart op het scherm.

Campus Impuls (2025-2026)

```
namen = ["Anna", "Bert", "Charlie"]
for naam in namen:
    print(naam)
```

3. Maak een lijst met de getallen 3.0, 1.4, 5.6, 9.0 en toon elk van deze getallen apart op het scherm.

```
getallen = [3.0,1.4,5.6,9.0]
for getal in getallen:
    print(getal)
```

4. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: Spaghetti, Frietjes, Soep en toon elk van deze waarden apart op het scherm.

```
gerechten = []
gerechten.append('Spaghetti')
gerechten.append('Frietjes')
gerechten.append('Soep')
for gerecht in gerechten:
    print(gerecht)
```

Campus Impuls (2025-2026)

5. Maak een lijst en steek hier achtereenvolgend de volgende waarden in: 3.5, 1.2, 2.4, 9.0, 8.7 en vervolgens toon je deze getallen met 'Getal: ' ervoor.

```
getallen = []
getallen.append(3.5)
getallen.append(1.2)
getallen.append(2.4)
getallen.append(9.0)
getallen.append(8.7)
for getal in getallen:
    print(f'Getal: {getal}')
```

BASISOEFENINGEN: RANGE

1. Toon alle getallen van 0 tot 5 apart op het scherm.
2. Toon alle getallen van 1 tot 6 apart op het scherm.
3. Toon alle getallen van 1 t.e.m. 3 apart op het scherm.
4. Toon alle oneven getallen van 1 t.e.m. 9 apart op het scherm.
5. Toon alle even getallen van 2 t.e.m. 8 apart op het scherm.
6. Toon alle getallen van -2 t.e.m. 4 apart op het scherm.
7. Toon 20 keer "Ik ga graag naar school" onder elkaar.
8. Vraag 3 keer aan de gebruiker een getal ("Getal: ") en toon deze getallen op het scherm

BASISOEFENINGEN: RANGE (OPLOSSINGEN)

1. Toon alle getallen van 0 tot 5 apart op het scherm.

```
for getal in range(5):
    print(getal)
```

2. Toon alle getallen van 1 tot 6 apart op het scherm.

Campus Impuls (2025-2026)

```
1 |for getal in range(1,6):
2 |    print(getal)
```

3. Toon alle getallen van 1 t.e.m. 3 apart op het scherm.

```
1 |for getal in range(1,4):
2 |    print(getal)
```

4. Toon alle oneven getallen van 1 t.e.m. 9 apart op het scherm.

```
1 |for getal in range(1,10,2):
2 |    print(getal)
```

5. Toon alle even getallen van 2 t.e.m. 8 apart op het scherm.

```
1 |for getal in range(2,9,2):
2 |    print(getal)
```

6. Toon alle getallen van -2 t.e.m. 4 apart op het scherm.

```
1 |for getal in range(-2,5):
2 |    print(getal)
```

7. Toon 20 keer "Ik ga graag naar school" onder elkaar.

```
1 |for getal in range(20):
2 |    print("Ik ga graag naar school")
```

8. Vraag 3 keer aan de gebruiker een getal ("Getal: ") en toon deze getallen op het scherm

Campus Impuls (2025-2026)

```
for getal in range(3):
    getal = int(input("Getal: "))
    print(getal)
```

OEFENINGEN

1. Maak een lijst en steek hier 5 getallen in. Overloop deze lijst en toon de getallen op het scherm.
2. Maak een lijst en steek hier 3 getallen in. Overloop deze lijst en tel alle getallen bij elkaar op.
3. Maak een lijst en steek hier 3 namen in van huisdieren ("Geef een naam van jouw huisdier op: "). Overloop deze lijst en toon deze strings op het scherm.
4. Maak een lijst van getallen. Laat automatisch hier alle getallen van 278 t.e.m. 300 in steken. Overloop vervolgens deze lijst en tel deze getallen bij elkaar op.
5. Laat de gebruiker een lijst van 3 ingrediënten opgeven. Overloop vervolgens deze lijst en plaats hier overal het nummer van de lijst voor. Bijvoorbeeld: 1. Eieren 2. Melk 3. Bloem
6. Laat de gebruiker een getal ingeven waarvan hij de tafel van 10 wil hebben ("Geef een getal in: "). Steek vervolgens in een lijst alle vermenigvuldigingen van 1 t.e.m. 10 in. Toon deze getallen op het scherm en vervang vervolgens deze getallen door hun kwadraat. Bijvoorbeeld: bij het ingeven van het getal 6, vul je eerst de lijst op met 6, 12, 18, … Vervolgens toon je 6, 12, 18, … op het scherm en vervang je de getallen in deze lijst met de getallen 36, 144, 324, …
7. Vraag de gebruiker hoeveel titels van films hij wil ingeven ("Aantal films: "). Vervolgens vraag je het aantal keer dat werd opgegeven naar een filmtitel, van slechtste naar beste ("Titel film (van slecht naar goed): "). Vervolgens toon je de films van goed naar slecht, met ervoor een nummer (1, 2, 3, …).

   Bijvoorbeeld: Films: "Super slechte film", "Oké film", "Best film ooit".

   Uitvoer:
   1. Beste film ooit
   2. Oké film
   3. Super slechte film

Campus Impuls (2025-2026)

8.

9. Laat een gebruiker 20 nummers ingeven. Bepaal vervolgens een boxplot, door de volgende data te tonen:

   cijfers = [4, 5, 5.5, 6, 6, 6.5, 7, 7, 7.5, 8, 8, 9, 9.5, 10]

   4|--|6--7--8|--|10

OEFENINGEN (OPLOSSINGEN)

1. Maak een lijst en steek hier 5 getallen in. Overloop deze lijst en toon de getallen op het scherm.

```
1 |getallen = [5,6,7,8,9]
2 |for getal in getallen:
3 |    print(getal)
```

2. Maak een lijst en steek hier 3 getallen in. Overloop deze lijst en tel alle getallen bij elkaar op.

```
1 |getallen = [1,2,3]
2 |som = 0
3 |for getal in getallen:
4 |    som = som + getal
5 |print(som)
```

3. Maak een lijst en steek hier 3 namen in van huisdieren ("Geef een naam van jouw huisdier op: "). Overloop deze lijst en toon deze strings op het scherm.

```
1 |huisdieren = []
2 |for keer in range(3):
3 |    huisdier = input("Geef een naam van jouw huisdier op: ")
4 |    huisdieren.append(huisdier)
5 |for naam in huisdieren:
6 |    print(f'Ik hou zo van {naam}')
```

Campus Impuls (2025-2026)

4. Maak een lijst van getallen. Laat automatisch hier alle getallen van 278 t.e.m. 300 in steken. Overloop vervolgens deze lijst en tel deze getallen bij elkaar op.

```
getallen = []
som = 0
for getal in range(278,301):
    getallen.append(getal)
    som = som + getal
print(getallen)
print(som)
```

5. Laat de gebruiker een lijst van 3 ingrediënten opgeven. Overloop vervolgens deze lijst en plaats hier overal het nummer van de lijst voor. Bijvoorbeeld: 1. Eieren 2. Melk 3. Bloem

```
ingredienten = []
for keer in range(1,4):
    ingr = input("Ingrediënt: ")
    ingredienten.append(ingr)
    #ingredienten.append(f'{keer}. {ingr}')
for teller in range(3):
    ingredienten[teller] = f'{teller+1}.{ingredienten[teller]}'
for ingr in ingredienten:
    print(ingr)
```

6. Laat de gebruiker een getal ingeven waarvan hij de tafel van 10 wil hebben ("Geef een getal in: "). Steek vervolgens in een lijst alle vermenigvuldigingen van 1 t.e.m. 10 in. Toon deze getallen op het scherm en vervang vervolgens deze getallen door hun kwadraat. Bijvoorbeeld: bij het ingeven van het getal 6,

Campus Impuls (2025-2026)

vul je eerst de lijst op met 6, 12, 18, … Vervolgens toon je 6, 12, 18, … op het scherm en vervang je de getallen in deze lijst met de getallen 36, 144, 324, …

```
getal = int(input("Geef een getal in: "))
tafels_van_10 = []
for factor in range(1,11):
    tafels_van_10.append(factor*getal)
for tafel in tafels_van_10:
    print(tafel)
for positie in range(len(tafels_van_10)):
    tafels_van_10[positie] = tafels_van_10[positie]**2
print(tafels_van_10)
```

7. Vraag de gebruiker hoeveel titels van films hij wil ingeven ("Aantal films: "). Vervolgens vraag je het aantal keer dat werd opgegeven naar een filmtitel, van slechtste naar beste ("Titel film (van slecht naar goed): "). Vervolgens toon je de films van goed naar slecht, met ervoor een nummer (1, 2, 3, …).

   Bijvoorbeeld: Films: "Super slechte film", "Oké film", "Best film ooit".

   Uitvoer:

   1. Beste film ooit
   2. Oké film
   3. Super slechte film

```
aantal = int(input("Aantal films: "))
films = []
for i in range(aantal):
    film = input("Titel film (van slecht naar goed): ")
    films.insert(0,film)
for i in range(len(films)):
    print(f'{i+1}. {films[i]}')
```

Campus Impuls (2025-2026)

8. Laat een gebruiker 20 nummers ingeven. Bepaal vervolgens een boxplot, door de volgende data te tonen:

   cijfers = [4, 5, 5.5, 6, 6, 6.5, 7, 7, 7.5, 8, 8, 9, 9.5, 10]

```
1 |cijfers = [4, 5, 5.5, 6, 6, 6.5, 7, 7, 7.5, 8, 8, 9, 9.5, 10]
2 |print(f'{cijfers[0]}|--|{cijfers[len(cijfers)//4]}--
  {cijfers[len(cijfers)//2]}--{cijfers[len(cijfers)//4*3]}|--
  |{cijfers[len(cijfers)-1]}')
```

TOETSVRAGEN

1. Maak een programma dat 1000 keer op het scherm schrijft: "Python is cool!"
2. Maak een programma dat alle getallen van -100 t.e.m. 100 op het scherm toont.
3. Maak een programma dat alle getallen toont tussen 1 en 100 (1, 2, 3, ... , 97, 98, 99, 100)
4. Maak een programma dat aan de gebruiker 10 keer vraagt om een toetsresultaat op te geven (kommagetal op 10). Vervolgens bereken je het rapportcijfer door het gemiddelde hiervan te nemen.
5. Maak een programma dat aan de gebruiker hoeveel examens hij heeft. Vervolgens vraag je de namen van de vakken. Dus als de gebruiker bij de vraag "Hoeveel examens heb je?" een 7 invult, dan verschijnt er 7 keer "Vak: " waarbij de gebruiker een vak kan intypen.
6. Je maakt een programma voor het kiezen van babynamen. Je vraagt aan de gebruiker hoeveel namen die wenst op te geven ("Hoeveel namen wil je opgeven?"). Vervolgens vraag je het opgegeven aantal keer een babynaam ("Wat is de naam? ") en steek je deze namen in een lijst. Hierna overloop je deze lijst en plaats je voor de even namen (2e naam, 4e naam, ...) 3 spaties in de lijst. Je toont deze ook op het scherm.

   Bv. Boeleke
   Petoetje
   Prutsje
   Zoetje

Campus Impuls (2025-2026)

7. Maak een lijst en steek hier 5 willekeurige getallen (je mag zelf kiezen) achter elkaar in. Vervolgens overloop je deze lijst en toon je van deze getallen hun derdemachtsverheffing (x^3).
8. Vraag aan de gebruiker hoeveel getallen hij wil ingeven. Vervolgens vraag je het opgegeven aantal keer een getal en toon je deze lijst. Hierna vervang je alle getallen in de lijst naar hun tegengestelde (3 wordt -3 en -5 wordt 5).

TOETSVRAGEN (OPLOSSING)

1. Maak een programma dat 1000 keer op het scherm schrijft: "Python is cool!"

```
1 |for keer in range(1000):
2 |    print("Python is cool!")
```

2. Maak een programma dat alle getallen van -100 t.e.m. 100 op het scherm toont.

```
1 |for getal in range(-100, 101):
2 |    print(getal)
```

3. Maak een programma dat alle getallen toont tussen 1 en 100 (1, 2, 3, ... , 97, 98, 99, 100)

```
1 |for getal in range(1, 101):
2 |    print(getal)
```

4. Maak een programma dat aan de gebruiker 10 keer vraagt om een toetsresultaat op te geven (kommagetal op 10). Vervolgens bereken je het rapportcijfer door het gemiddelde hiervan te nemen.

Campus Impuls (2025-2026)

```
totaal = 0.0
for keer in range(10):
    toets = float(input("Toets (op 10): "))
    totaal = totaal + toets
gemiddelde = totaal / 10
print(gemiddelde)
```

5. Maak een programma dat aan de gebruiker hoeveel examens hij heeft. Vervolgens vraag je de namen van de vakken en steek je deze in een lijst. Dus als de gebruiker bij de vraag "Hoeveel examens heb je?" een 7 invult, dan verschijnt er 7 keer "Vak: " waarbij de gebruiker een vak kan intypen.

```
examenvakken = []
aantal_examens = int(input("Hoeveel examens heb je? "))
for keer in range(aantal_examens):
    examen = input("Vak: ")
    examenvakken.append(examen)
```

6. Je maakt een programma voor het kiezen van babynamen. Je vraagt aan de gebruiker hoeveel namen die wenst op te geven ("Hoeveel namen wil je opgeven?"). Vervolgens vraag je het opgegeven aantal keer een babynaam ("Wat is de naam? ") en steek je deze namen in een lijst. Hierna overloop je deze lijst en plaats je voor de even namen (2e naam, 4e naam, ...) 3 spaties in de lijst. Je toont deze ook op het scherm.

Bv. Boeleke
  Petoetje
Prutsje
  Zoetje

Campus Impuls (2025-2026)

```
babynamen = []
aantal_namen = int(input("Hoeveel babynamen wil je opgeven? "))
for keer in range(aantal_namen):
    babynaam = input("Wat is de naam? ")
    babynamen.append(babynaam)
for positie in range(0,aantal_namen, 2):
    babynamen[positie] = f'   {babynamen[positie]}'
for babynaam in babynamen:
    print(babynaam)
```

7. Maak een lijst en steek hier 5 willekeurige getallen (je mag zelf kiezen) achter elkaar in. Vervolgens overloop je deze lijst en toon je van deze getallen hun derdemachtsverheffing (x^3).

```
getallen = [-3.5, -1.2, 0, 4, 5.6]
for getal in getallen:
    print(getal**3)
```

8. Vraag aan de gebruiker hoeveel getallen hij wil ingeven. Vervolgens vraag je het opgegeven aantal keer een getal en toon je deze lijst. Hierna vervang je alle getallen in de lijst naar hun tegengestelde (3 wordt -3 en -5 wordt 5).

```
getallen = []
aantal_getallen = int(input("Hoeveel getallen wil je opgeven?
"))
for keer in range(aantal_getallen):
    getal = int(input("Getal: "))
    getallen.append(getal)
for positie in range(len(getallen)):
    getallen[positie] = getallen[positie]*-1
for getal in getallen:
    print(getal)
```

Campus Impuls (2025-2026)

Herhaling

OEFENING INPUT, OUTPUT, STRINGS

Basis

1. Maak een programma dat aan de gebruiker diens naam vraagt en deze naam dan op het scherm schrijft.
2. Maak een programma dat aan de gebruiker een stad vraagt en deze op het scherm schrijft.
3. Maak een programma dat aan de gebruiker vraagt wat z'n lievelingsvak is. Toon dit dan op het scherm.
4. Maak een programma dat aan de gebruiker vraagt wat z'n lievelingssnack is. Vervolgens toon je dit nogmaals op het scherm.

Gevorderd

1. Maak een programma dat aan de gebruiker diens naam en geboortestad vraagt. Tenslotte zegt het programma: "Hallo *naam*, ik ben ook geboren in *stad*!"
2. Maak een programma dat aan de gebruiker vraagt wat z'n lievelingsvak is. Vervolgens toon je "Toevallig, ik doe ook graag *vak*!"
3. Maak een programma dat aan de gebruiker vraagt wat zijn favoriete gerecht is. Laat het programma dan zeggen: "Ik eet ook graag *gerecht*!"
4. Schrijf de volgende zin op het scherm, zonder variabelen: Ik zei: "Ik sta niet graag 's ochtends op!"
5. Schrijf de volgende zin op het scherm, zonder variabelen: 's Avonds zeg ik altijd "Slaapwel" tegen mijn knuffeltje.

OEFENINGEN INT, FLOAT, WISKUNDIGE OPERATOREN

Basis

1. Vraag aan de gebruiker 2 gehele getallen en print het verschil op het scherm.
2. Vraag aan de gebruiker 2 gehele getallen en print de som op het scherm.
3. Vraag aan de gebruiker 2 gehele getallen en print het product op het scherm.
4. Vraag aan de gebruiker 2 gehele getallen en print het quotiënt op het scherm.
5. Vraag aan de gebruiker 2 gehele getallen en print de rest na deling van het eerste gedeeld door het tweede getal op het scherm.
6. Vraag aan de gebruiker 2 gehele getallen en print de gehele deling van het eerste door het tweede getal op het scherm.
7. Vraag aan de gebruiker 2 gehele getallen en print het verschil op het scherm.
8. Vraag aan de gebruiker 2 gehele getallen en print de macht van het eerste tot het tweede getal op het scherm.

Campus Impuls (2025-2026)

9. Vraag aan de gebruiker 2 gehele getallen en print de machtswortel van het eerste tot het tweede getal op het scherm. Vraag aan de gebruiker 2 gehele getallen en print het verschil op het scherm.
10. Vraag aan de gebruiker 2 kommagetallen en print de som op het scherm.
11. Vraag aan de gebruiker 2 kommagetallen en print het product op het scherm.
12. Vraag aan de gebruiker 2 kommagetallen en print het quotiënt op het scherm.
13. Vraag aan de gebruiker 2 kommagetallen en print de rest na deling van het eerste gedeeld door het tweede getal op het scherm.
14. Vraag aan de gebruiker 2 kommagetallen en print de gehele deling van het eerste door het tweede getal op het scherm.
15. Vraag aan de gebruiker 2 kommagetallen en print het verschil op het scherm.
16. Vraag aan de gebruiker 2 kommagetallen en print de macht van het eerste tot het tweede getal op het scherm.
17. Vraag aan de gebruiker 2 kommagetallen en print de machtswortel van het eerste tot het tweede getal op het scherm.

Gevorderd

1. Vraag aan het gebruiker om een getal x in te vullen en bereken het resultaat van de volgende functie: $x^3 + \left(\frac{5-x}{3}\right) \cdot \sqrt{x}$
2. Vraag aan het gebruiker om een getal x in te vullen en bereken het resultaat van de volgende functie: $x^{-2} - \left(\frac{4}{x-\sqrt{x}}\right) \cdot 3x$
3. Vraag aan het gebruiker om een getal x in te vullen en bereken het resultaat van de volgende functie: $\sqrt{3^x - (\frac{3-x}{3+x}) \cdot x^x}$
4. Vraag aan het gebruiker om een getal x in te vullen en bereken het resultaat van de volgende functie: $\sqrt{\frac{\frac{1}{x}}{x}^x}$
5. Vraag aan het gebruiker om een getal x in te vullen en bereken het resultaat van de volgende functie: $\frac{\sqrt{x}}{\sqrt{\frac{x^x}{\frac{\sqrt{x}}{x}}}}$
6. Vraag aan het gebruiker om een getal x in te vullen en bereken het resultaat van de volgende functie: $\left(\frac{x-x^3}{x^{\sqrt{x}}-x}\right)^{\sqrt{x}}$

% en //

1. Een toegangsticket kost €11. Voor elke 5 personen betaalt er één gratis. Vraag aan de gebruiker met hoeveel personen ze komen en bereken de prijs.
2. Een doos bevat 6 koekjes. Vraag aan de gebruiker hoeveel koekjes hij wil kopen.
   Bereken hoeveel volledige dozen nodig zijn en hoeveel losse koekjes overblijven.

Campus Impuls (2025-2026)

3. Bij een schoenenwinkel geldt een actie: koop 3 paar, betaal er 2.
   Een paar schoenen kost €59.
   Vraag hoeveel paren de klant wil kopen en bereken de prijs.
4. Vraag aan de gebruiker een aantal seconden.
   Bereken hoeveel uren, minuten en seconden dat zijn.
5. Een doos bevat 10 eieren.
   Vraag aan de gebruiker hoeveel eieren hij heeft.
   Toon hoeveel volle dozen hij kan vullen en hoeveel eieren overblijven.
6. Je wil zakjes vullen met 15 snoepjes per zakje.
   Vraag aan de gebruiker hoeveel snoepjes hij heeft.
   Bereken het aantal volledige zakjes en hoeveel snoepjes te weinig zijn voor een volgend zakje.

OEFENINGEN LIJSTEN

Append

1. Maak een lijst en steek hier de volgende waarden in: 1, 2, 3, 4, 5. Overloop deze lijst en toon de waarden op het scherm (1 per lijn).
2. Maak een lijst en steek hier de volgende waarden in: Kaas, hesp, zalm. Overloop deze lijst en toon de waarden op het scherm (1 per lijn).
3. Maak een lijst en steek hier de volgende waarden in: Wiskunde, Frans, Engels. Overloop deze lijst en toon de waarden op het scherm (1 per lijn).
4. Maak een lijst en steek hier de volgende waarden in: Fiets, Auto, Bus. Overloop deze lijst en toon de waarden op het scherm (1 per lijn).

Insert

1. Maak een lijst en steek steeds de volgende waarden vooraan: 3, 2, 1
2. Maak een lijst en steek steeds de volgende waarden vooraan: Citroen, Banaan, Appel
3. Maak een lijst en steek steeds de volgende waarden vooraan: Cedric, Bert, Achraf
4. Maak een lijst en steek steeds de volgende waarden vooraan: 3.4, 2.1 ,1.5

Range

1. Vraag aan de gebruiker een getal en toon de tafel van 10 van dit getal.
2. Vraag aan de gebruiker een getal A en een getal B en toon de tafel van getal B van getal A. (bv. A = 4, B = 5 wordt de tafel van 5 van 4: 4, 8, 12, 16, 20
3. Vraag aan de gebruiker een getal en toon alle even, natuurlijke getallen kleiner dan dit getal.
4. Vraag aan de gebruiker een getal en toon alle machtsverheffingen van 2 tot dit getal (bv. getal 4 wordt 2, 4, 8, 16)

Campus Impuls (2025-2026)

Voorwaardelijke uitvoer

Tot nu toe heb je geleerd dat er bij Python de ene lijn na de andere wordt uitgevoerd. Je kan ook bepaalde lijnen herhaald uitvoeren (for).

Nu is het echter zo dat je soms ook gaat willen dat bepaalde lijnen enkel maar worden uitgevoerd in bepaalde omstandigheden. Bijvoorbeeld: je maakt een programma die de prijs voor een online bestelling moet doen. Er worden geen verzendkosten (3 euro) aangerekend als het aankoopbedrag hoger ligt dan 20 euro. Je zal dus de lijn waarbij je de kost met 3 verhoogt enkel maar willen uitvoeren, afhankelijk van de waarde van het aankoopbedrag.

Ook het herhaald uitvoeren van lijnen kan aan voorwaarden worden gekoppeld. Stel dat je een lijst van namen wil laten bevragen tot de gebruiker 'STOP' intypt, dan wil je een herhaalde *input* en *append* enkel maar uitvoeren tot de gebruiker 'STOP' intypte.

Hoe dit werkt, zien we hieronder. Eerst moeten we wel nog bespreken wat een **boolean** is.

BOOLEAN

Een *boolean* is in het computergeheugen eigenlijk niet meer dan 1 bit (0 of 1). Dit is een datatype, net zoals een int (gehele getallen), float (kommagetallen), string (tekst) en list (lijst). Deze 0 of 1 worden respectievelijk dan *waar* en *niet waar* genoemd, ofwel in Python *True* en *False*. Dit zijn de ook de enige 2 waarden die een boolean kan aannemen.

Voor de benaming van een boolean variabele, is het best om een naam te kiezen die een vraag of stelling poneert waarop het antwoord *waar* of *niet waar* zou zijn. Stel dat ik een variabele wil waarbij ik wil bijhouden dat een gekozen product een zwarte kleur heeft.. Een goede naam zou bijvoorbeeld 'is_zwart' of 'is_zwart_product'. Als je dit op *True* zet, is het duidelijk wat hiermee wordt bedoeld. Een 'slechte' naam zou bijvoorbeeld 'kleur' of 'gekozen_kleur'.

Voorbeeld:

Campus Impuls (2025-2026)

```
1 |is_zwart = True
2 |heeft_korting = False
3 |is_mannelijk = True
```

Merk op dat hierboven een variabele naam 'mannelijk' wordt gekozen. Dit is véél duidelijker dan bijvoorbeeld 'geslacht', aangezien je dan niet weet of *True* net mannelijk of vrouwelijk betekent.

VERGELIJKENDE OPERATOREN

Het inlezen van een boolean, wordt niet vaak gedaan. Er bestaat een *bool*-functie, maar vaak ga je een waarde willen instellen op basis van een andere variabele. Hiervoor is het belangrijk om de vergelijkende operatoren te leren kennen.

Een operator is een teken dat in Python wordt gebruikt om een bepaalde waarde te geven. Een simpel voorbeeld is *. Dit is een operator die de getallen ervoor en erna met elkaar vermenigvuldigt (als dit kan) en het resultaat teruggeeft. Of =, die wordt gebruikt om een waarde op te slaan in een variabele.

De eerste twee operatoren die we gaan zien, gaan na of twee waarden (bv. variabelen, uitkomsten van functies/operatoren, …) wel of niet **gelijk zijn** aan elkaar. Om te kijken of twee waarden *wel* gelijk zijn aan elkaar, gebruiken we **==**. Let op: dit zijn 2 gelijkheidstekens, niet 1. Dit komt omdat 1 gelijkheidsteken al wordt gebruikt om een waarde op te slaan in de variabelen. Stel dat je toch per ongeluk maar één gelijkheidsteken schrijft, krijg je vaak de volgende melding:

SyntaxError: invalid syntax. Maybe you meant '==' or ':=' instead of '='?

De IDE helpt jou in dit geval. Hij vraagt of je 2 gelijkheidstekens bedoelde in plaats van 1.

Voorbeeld: je wil een variabele 'wachtwoord_klopt' die wil kijken of een ingegeven wachtwoord van de gebruiker overeenkomt met een bepaald wachtwoord (in dit voorbeeld 'password123').

Dan kan dit op de volgende manier:

Campus Impuls (2025-2026)

```
1 |wachtwoord = input("Geef het wachtwoord: ")
2 |wachtwoord_klopt = wachtwoord == "password123"
```

Om na te gaan of twee waarden *niet* gelijk zijn aan elkaar, kan je gebruik maken van **!=**.

Je kan dan ook een variabele 'wachtwoord_is_fout' als volgt hebben:

```
1 |wachtwoord = input("Geef het wachtwoord: ")
2 |wachtwoord_is_fout = wachtwoord != "password123"
```

Vier andere veel voorkomende operatoren zijn degene die kijken of twee waarden kleiner/groter en/of gelijk zijn aan elkaar. Deze zijn logischerwijs **<**, **>**, **<=** en **>=**. Let op: het gelijkheidsteken moet na het groter of kleiner dan teken.

Enkele voorbeelden:

```
1 |prijs = float(input("Prijs: "))
2 |prijs_kleiner_dan_20 = prijs < 20
3 |prijs_groter_dan_20 = prijs > 20
4 |prijs_minstens_20 = prijs >= 20
5 |prijs_hoogstens_20 = prijs <= 20
```

```
1 |leeftijd = int(input("Leeftijd: "))
2 |is_meerderjarig = leeftijd >= 18
```

```
1 |lengte_cm = int(input("Lengte (in cm): "))
2 |korter_dan_1_meter = lengte_cm < 100
```

```
1 |namen = ["Ann", "Bert", "Charlie", "David", "Ezra", "Faoud"]
2 |minstens_3_namen = len(namen) >= 3
```

Campus Impuls (2025-2026)

```
1 |prijs_1 = float(input("Prijs 1 (in euro): "))
2 |prijs_2 = float(input("Prijs 2 (in euro): "))
3 |totaal_minder_dan_100 = prijs_1 + prijs_2 < 100
```

Je kan deze ook combineren met elkaar om te kijken of een getal tussen bepaalde grenzen ligt. Hiervoor schrijf je rond (dus aan beide kanten van het getal/variabele) een grensvoorwaarde op. Het onderstaande voorbeeld geeft een *true* als getal groter is dan 0 en kleiner is dan 11. Let op: je kan hier dus ook <= of >= gebruiken.

```
1 |getal = int(input("Getal: "))
2 |tussen_1_en_10 = 0 < getal < 11
3 |tussen_1_en_10 = 1 <= getal <= 10
```

IF

Nu hebben we wel gezien hoe je kan weten of aan een voorwaarde al dan niet is voldaan, daarmee kan je nog niet bepaalde code wel/niet uitvoeren. Dit kan in de eerste plaats met een **if-structuur**. Dit is een structuur (zoals for) die bepaalde lijnen code gaan samennemen door ze vooraf te laten gaan met een inspringing. Na de hoofding van deze structuur, schrijf je ook een dubbelpunt. De if-structuur gaat de ingesprongen lijnen code enkel uitvoeren als de boolean die in de hoofding staat **waar** (True) is.

Voorbeeld:

```
1 |leeftijd = int(input("Leeftijd: "))
2 |if leeftijd >= 18:
3 |    print("Meerderjarig")
```

Het bovenstaande stukje code zal de uitvoer 'Meerderjarig' (lijn 3) enkel en alleen uitvoeren als de ingegeven leeftijd groter of gelijk is aan 18.

Een ander voorbeeld, waarbij een verzendkost (3 euro) wordt toegekend als het aankoopbedrag kleiner is dan 20 euro.

Campus Impuls (2025-2026)

```
1 |aankoopbedrag = float(input("Aankoopbedrag (in euro): "))
2 |if aankoopbedrag < 20:
3 |      aankoopbedrag = aankoopbedrag + 3
```

Merk op dat hier *leeftijd* >= *18* en *aankoopbedrag* < *20* op zichzelf booleans zijn. Het volgende is dit ook een mogelijkheid.

```
1 |leeftijd = int(input("Leeftijd: "))
2 |meerderjarig = leeftijd >= 18
3 |if meerderjarig:
4 |      print("Meerderjarig)
```

Of

```
1 |aankoopbedrag = float(input("Aankoopbedrag (in euro): "))
2 |minder_dan_20 = aankoopbedrag < 20
3 |if minder_dan_20:
4 |      aankoopbedrag = aankoopbedrag + 3
```

BASISOEFENING IF:

1. Vraag aan de gebruiker een geheel getal. Als het getal groter is dan 10, toon je dit getal nogmaals op het scherm.
2. Vraag aan de gebruiker een naam. Als de naam gelijk is aan John, toon je "Hallo John" op het scherm.
3. Vraag aan de gebruiker een kommagetal. Als het kommagetal kleiner is dan 0, dan toon je "Negatief" op het scherm.
4. Vraag aan de gebruiker een geheel getal. Als het getal niet gelijk is aan 0, dan toon je dit getal nogmaals op het scherm.
5. Vraag aan de gebruiker twee kommagetallen. Als de deling van het ene door het andere getal kleiner is dan 8, dan toon je "Kleiner dan 8" op het scherm.

Campus Impuls (2025-2026)

BASISOEFENING IF (OPLOSSINGEN):

1. Vraag aan de gebruiker een geheel getal. Als het getal groter is dan 10, toon je dit getal nogmaals op het scherm.

```
1 |getal = int(input("Getal: "))
2 |if getal > 10:
3 |    print(getal)
```

2. Vraag aan de gebruiker een naam. Als de naam gelijk is aan John, toon je "Hallo John" op het scherm.

```
1 |naam = input("Naam: ")
2 |if naam == "John":
3 |    print("Hallo John")
```

3. Vraag aan de gebruiker een kommagetal. Als het kommagetal kleiner is dan 0, dan toon je "Negatief" op het scherm.

```
1 |getal = float(input("Getal: "))
2 |if getal < 0:
3 |    print("Negatief")
```

4. Vraag aan de gebruiker een geheel getal. Als het getal niet gelijk is aan 0, dan toon je dit getal nogmaals op het scherm.

```
1 |getal = int(input("Getal: "))
2 |if getal != 0:
3 |    print(getal)
```

5. Vraag aan de gebruiker twee kommagetallen. Als de deling van het ene door het andere getal kleiner is dan 8, dan toon je "Kleiner dan 8" op het scherm.

Campus Impuls (2025-2026)

```
a = float(input("Getal a: "))
b = float(input("Getal b: "))
if a/b < 8:
    print("Kleiner dan 8")
```

OEFENING IF:

1. Een webshop rekent verzendkosten op basis van gewicht. Standaard levering: € 5 + € 1,50 per kg. Express levering: € 12 + € 0,80 per kg. Vraag aan de gebruiker hoeveel zijn pakketje weegt en toon welke levering het goedkoopste is ("Standaard levering" of "Express levering"). Als er kostprijs voor beide leveringen hetzelfde is, kies je voor de express levering.
2. Een bankrekening heeft een saldo. Bij een saldo kleiner dan 0 wordt een waarschuwing gegeven ("Negatief saldo"). Bij een saldo groter dan 10.000 krijgt de klant een VIP-melding ("VIP-klant"). Vraag aan de gebruiker het saldo en laat een melding verschijnen als deze van toepassing is.
3. Een gebruiker wil weten of een getal een priemgetal is. Maak een programma dat aan de gebruiker een getal vraagt en vervolgens zegt of dit wel of geen priemgetal is.
4. Een leerling wil weten of hij geslaagd is voor zijn rapport voor wiskunde. Daarom geeft hij eerst of hoeveel toetsen hij heeft gemaakt ("Hoeveel toetsen heb je gemaakt?"). Vervolgens geeft hij per toets de behaalde resultaat is ("Behaald: ") en op hoeveel de toets stond ("Totaal: "). Tenslotte bereken je of de leerling minstens 50% haalt. Je toont vervolgens of de leerling wel ("Geslaagd") of niet ("Gebuisd") geslaagd is.
5. Je maakt een programma voor een nieuwsdienst. Zij willen weten of er een hittegolf is. Hiervoor vraag je de maximumtemperaturen van de voorbije 7 dagen. Je mag van een hittegolf spreken als er minimaal 5 zomerse dagen (maximumtemperatuur 25,0 °C of hoger) waren, waarvan er minimaal drie tropische dagen (maximumtemperatuur 30,0 °C of hoger).

   Tip: Je kan een if-structuur in een andere steken, zolang je ook het aantal inspringingen mee verandert.
6. Je schrijft een programma voor een drankfabriek. Om te kijken of de volmachine nog voldoende goed werkt, wordt er steekproefsgewijs gekeken hoeveel flesjes

Campus Impuls (2025-2026)

er teveel of te weinig drank hadden. Het zijn flesjes van 33 cl, en er wordt een marge aanvaard van 5 %. Laat de gebruiker 15 inhouden van flesjes ingeven. Als er meer dan 3 boven/onder de marge zitten, schrijf je op het scherm "Controleer de machine!".

OEFENING IF (OPLOSSINGEN):

1. Een webshop rekent verzendkosten op basis van gewicht. Standaard levering: € 5 + € 1,50 per kg. Express levering: € 12 + € 0,80 per kg. Vraag aan de gebruiker hoeveel zijn pakketje weegt en toon welke levering het goedkoopste is ("Standaard levering" of "Express levering").

```
gewicht = float(input("Hoeveel weegt uw pakket? "))
standaard_levering = 5 + (gewicht * 1.5)
express_levering = 12 + (gewicht * 0.8)

if standaard_levering < express_levering:
    print("Standaard levering")
if express_levering <= standaard_levering:
    print("Express levering")
```

2. Een bankrekening heeft een saldo. Bij een saldo kleiner dan 0 wordt een waarschuwing gegeven ("Negatief saldo"). Bij een saldo groter dan 10.000 krijgt de klant een VIP-melding ("VIP-klant"). Vraag aan de gebruiker het saldo en laat een melding verschijnen als deze van toepassing is.

```
saldo = float(input("Saldo: "))

if saldo < 0:
    print("Negatief saldo")
if saldo > 10000:
    print("VIP-klant")
```

Campus Impuls (2025-2026)

3. Een gebruiker wil weten of een getal een priemgetal is. Maak een programma dat aan de gebruiker een getal vraagt en vervolgens zegt of dit wel of geen priemgetal is.Bijvoorbeeld: "3 is een priemgetal." of "4 is geen priemgetal."

```
1 | getal = int(input("Getal: "))
2 | origineel = getal
3 | if getal < 0:
4 |     getal = getal * -1
5 |
6 | priemgetal = True
7 | for deler in range(2,(getal//2)+1):
8 |     if getal % deler == 0:
9 |         priemgetal = False
10 |if priemgetal:
11 |    print(f'{origineel} is een priemgetal.')
12 |if not priemgetal:
13 |    print(f'{origineel} is geen priemgetal.')
```

Campus Impuls (2025-2026)

4. Een leerling wil weten of hij geslaagd is voor zijn rapport voor wiskunde. Daarom geeft hij eerst of hoeveel toetsen hij heeft gemaakt ("Hoeveel toetsen heb je gemaakt?"). Vervolgens geeft hij per toets de behaalde resultaat is ("Behaald: ") en op hoeveel de toets stond ("Totaal: "). Tenslotte bereken je of de leerling minstens 50% haalt. Je toont vervolgens of de leerling wel ("Geslaagd") of niet ("Gebuisd") geslaagd is.

```
aantal = int(input("Hoeveel toetsen heb je gemaakt? "))

punten = 0
totaal = 0

for keer in range(aantal):
    punten = punten + float(input("Behaald: "))
    totaal = totaal + float(input("Totaal: "))

    percentage = punten / totaal

    if percentage >= 0.5:
        print("Geslaagd")
    if percentage < 0.5:
        print("Gebuisd")
```

Campus Impuls (2025-2026)

5. Je maakt een programma voor een nieuwsdienst. Zij willen weten of er een hittegolf is. Hiervoor vraag je de maximumtemperaturen van de voorbije 7 dagen. Je mag van een hittegolf spreken als er minimaal 5 zomerse dagen (maximumtemperatuur 25,0 °C of hoger) waren, waarvan er minimaal drie tropische dagen (maximumtemperatuur 30,0 °C of hoger).

   Tip: Je kan een if-structuur in een andere steken, zolang je ook het aantal inspringingen mee verandert.

```
zomerse_dagen = 0
tropische_dagen = 0

for dag in range(7):
    temperatuur = float(input("Temperatuur: "))
    if temperatuur >= 25:
        zomerse_dagen = zomerse_dagen + 1
    if temperatuur >= 30:
        tropische_dagen = tropische_dagen + 1

    if zomerse_dagen >= 5:
        if tropische_dagen >= 3:
            print("Hittegolf")
```

Campus Impuls (2025-2026)

6. Je schrijft een programma voor een drankfabriek. Om te kijken of de volmachine nog voldoende goed werkt, wordt er steekproefsgewijs gekeken hoeveel flesjes er teveel of te weinig drank hadden. Het zijn flesjes van 33 cl, en er wordt een marge aanvaard van 5 %. Laat de gebruiker 15 inhouden van flesjes ingeven. Als er meer dan 3 boven/onder de marge zitten, schrijf je op het scherm "Controleer de machine!".

```
1 |buiten_marge = 0
2 |standaard_inhoud = 33
3 |marge = standaard_inhoud*0.05
4 |
5 |for flesje in range(15):
6 |    inhoud = float(input("Inhoud: "))
7 |    if inhoud < standaard_inhoud - marge:
8 |        buiten_marge = buiten_marge + 1
9 |    if inhoud > standaard_inhoud + marge:
10 |            buiten_marge = buiten_marge + 1
11 |
12 |    if buiten_marge > 3:
13 |        print("Controleer de machine!")
```

IF-ELSE

Met de if-structuur kunnen we code uitvoeren onder bepaalde voorwaarden. Het zal soms echter ook gebeuren dat je in bepaalde situaties ofwel de **ene**, ofwel de **andere** code wil uitvoeren. Hiervoor kan je gebruik maken van de if-else-structuur. Deze is een uitbreiding van de if-structuur, met het bijkomende dat er na de if-structuur nog een else-structuur kan worden geschreven. Hierbij is er dan geen voorwaarde nodig, want deze code zal worden uitgevoerd als de stelling bij de if-structuur **niet** waar was. Let wel op dat je de else-structuur op hetzelfde niveau (inspringen) als de if-structuur schrijft.

Campus Impuls (2025-2026)

Hieronder een voorbeeld van een programma waarbij je ze leeftijd van iemand vraagt. Is de persoon 18 jaar of ouder, dan toon je "Meerderjarig". Anders toon je "Minderjarig".

```
1 |leeftijd = int(input("Leeftijd: "))
2 |if leeftijd >= 18:
3 |      print("Meerderjarig")
4 |else:
5 |      print("Minderjarig")
```

Let dus vooral op dat je jouw else-structuur niet in jouw if-structuur steekt:

```
1 |leeftijd = int(input("Leeftijd: "))
2 |if leeftijd >= 18:
3 |      print("Meerderjarig")
4 |      else:
5 |           print("Minderjarig")
```

Als je een waarde wil toewijzen aan een variabele met een if-else structuur, kan dit ook verkort met het volgende: *variabele = waarde_1 if voorwaarde else waarde_2*. Bijvoorbeeld:

```
1 |aankoopbedrag = float(input("Aankoopbedrag: "))
2 |verzendkost = 0
3 |if aankoopbedrag >= 20:
4 |      verzendkost = 1
5 |else:
6 |      verzendkost = 3
```

kan korter als

Campus Impuls (2025-2026)

```
1 |aankoopbedrag = float(input("Aankoopbedrag: "))
2 |verzendkost = 3
3 |if aankoopbedrag >= 20:
4 |      verzendkost = 1
```

maar kan dus nog korter als:

```
1 |aankoopbedrag = float(input("Aankoopbedrag: "))
2 |verzendkost = 1 if aankoopbedrag >= 20 else 3
```

BASISOEFENING IF-ELSE:

1. Vraag aan de gebruiker een geheel getal. Als het getal groter of gelijk aan 0, toon je "Positief" op het scherm. Anders toon je "Negatief" op het scherm.
2. Vraag aan de gebruiker een naam. Als de naam gelijk is aan John, toon je "Hallo John" op het scherm. Anders toon je "Hallo vreemde".
3. Vraag aan de gebruiker een kommagetal. Als het kommagetal kleiner is dan 10, toon je "Kleiner dan 10". Anders toon je "Minstens groter dan 10".
4. Vraag aan de gebruiker een geheel getal. Als het getal gelijk is aan 0, dan toon op het scherm "Je hebt 0 ingegeven". Anders toon je "Je hebt geen 0 ingegeven".
5. Vraag aan de gebruiker twee gehele getallen. Toon het grootste getal.

BASISOEFENING IF-ELSE (OPLOSSINGEN):

1. Vraag aan de gebruiker een geheel getal. Als het getal groter of gelijk aan 0, toon je "Positief" op het scherm. Anders toon je "Negatief" op het scherm.

Campus Impuls (2025-2026)

```
1 |getal = int(input("Getal: "))
2 |
3 |if getal >= 0:
4 |    print("Positief")
5 |else:
6 |    print("Negatief")
```

2. Vraag aan de gebruiker een naam. Als de naam gelijk is aan John, toon je "Hallo John" op het scherm. Anders toon je "Hallo vreemde".

```
1 |naam = input("Naam: ")
2 |
3 |if naam == "John":
4 |    print("Hallo John")
5 |else:
6 |    print("Hallo vreemde")
```

3. Vraag aan de gebruiker een kommagetal. Als het kommagetal kleiner is dan 10, toon je "Kleiner dan 10". Anders toon je "Minstens groter dan 10".

```
1 |getal = float(input("Getal: "))
2 |
3 |if getal < 10:
4 |    print("Kleiner dan 10")
5 |else:
6 |    print("Minstens groter dan 10")
```

4. Vraag aan de gebruiker een geheel getal. Als het getal gelijk is aan 0, dan toon op het scherm "Je hebt 0 ingegeven". Anders toon je "Je hebt geen 0 ingegeven".

Campus Impuls (2025-2026)

```
getal = int(input("Getal: "))

if getal == 0:
    print("Je hebt 0 ingegeven")
else:
    print("Je hebt geen 0 ingegeven")
```

5. Vraag aan de gebruiker twee gehele getallen. Toon het grootste getal.

```
a = int(input("Getal a: "))
b = int(input("Getal b: "))

if a > b:
    print(a)
else:
    print(b)
```

OEFENING IF-ELSE:

1. In een ziekenhuis is er een triage-systeem. Die bepaald of een patiënt onmiddellijk moet worden geholpen, of dat deze nog even kan wachten. Hiervoor gelden de volgende criteria:
   - Is de patiënt bewusteloos? "Directe hulp!"
   - Zo niet, heeft de patiënt een pijnscore hoger dan 8 op 10? "Dringende hulp".
   - Zo niet: "Patiënt kan wachten".

   Maak een programma voor een ziekenhuis dat de onderstaande zaken bevraagd en een boodschap toont op het scherm. Let op: als de patiënt bewusteloos is, kan (en hoeft) er geen pijnscore te worden afgenomen.
2. Bij de verzekeringsmaatschappij heb je 2 tarieven m.b.t. de jaarlijkse verzekering van jouw brommer. Ben je minder dan 20 jaar, betaal je 800 euro.

Campus Impuls (2025-2026)

Indien je 20 jaar of ouder bent, betaal je slechts 350 euro. Maak een programma zodat een gebruiker te weten kan komen hoeveel hij jaarlijkse zal moeten betalen.

3. Een universiteit geeft studieadvies aan leerlingen uit het secundair onderwijs. Hiervoor wordt in de eerste plaats naar het aantal onvoldoende scores voor vakken gekeken. Als dit meer is dan 3, is er een negatief advies. Anders is er een voorwaardelijk advies. Enkel als je op het totaalgemiddelde van alle vakken geslaagd bent en minder dan 2 onvoldoendes hebt, krijg je een positief advies. Maak een programma waarbij een leerling al zijn scores voor de verschillende vakken kan ingeven en vervolgens een advies kan krijgen.
4. Laat een leerling een behaalde score invullen van een toets en een maximumscore invullen. Vervolgens toon het programma een kleur (bv. "Groen") volgens de volgende tabel:

Onvoldoende (rood)	**Onvoldoende** (oranje)	**Voldoende** (geel)	**Voldoende** (groen)	**Blinkt uit** (blauw)
0%-29,5%	30%-49,9%	50%-69,9%	69.9%-89,9%	90%-100%

IF-ELIF-ELSE

De if-elif-else structuur is een uitbreiding van de if-else structuur. *Elif* staat voor *else if*, en laat jou toe om, als de voorwaarde bij *if false* was, een andere voorwaarde op te stellen. Als deze voorwaarde wel waar is, dan wordt de code hierin uitgevoerd. Is deze voorwaarde ook niet waar, dan wordt er tenslotte naar de *else* gegaan. De *else* wordt bij wijze van spreken dan eigenlijk een soort van *'als al het andere niet waar was'*.

Onderstaand voorbeeld kijkt of een getal positief, negatief of exact 0 is.

Campus Impuls 2025-2026

```
getal = int(input("Getal: "))
if getal > 0:
    print("Positief")
elif getal < 0:
    print("Negatief")
else:
    print("Exact 0")
```

Je kan ook meerdere *elif*-voorwaarden na elkaar gebruiken.

```
maand = int(input("Geef de maand in (getal): "))
if maand == 1:
    print("Januari")
elif maand == 2:
    print("Februari")
elif maand == 3:
    print("Maart")
else:
    print("April-December")
```

OEFENING IF-ELIF-ELSE:

1. Laat een leerling een behaalde score invullen van een toets en een maximumscore invullen. Vervolgens toon het programma een kleur (bv. "Groen") volgens de volgende tabel:

Onvoldoende (rood)	**Onvoldoende** (oranje)	**Voldoende** (geel)	**Voldoende** (groen)	**Blinkt uit** (blauw)
0%-29,5%	30%-49,9%	50%-69,9%	69.9%-89,9%	90%-100%

2. Je maakt een domotica-systeem die automatisch kleren voor jou klaar legt en een transportmiddel klaar maakt. Om dit aan te sturen, gaat die af op de voorspelde gemiddelde temperatuur van die dag. Als het 0°C of kouder wordt, zegt het systeem "Slipgevaar: warm auto op. Kledij: muts, sjaal, jas". Is het warmer dan 0°C maar nog steeds onder de 10°C, dan vermeld het systeem:

Campus Impuls (2025-2026)

"Ontgrendel fiets. Kledij: muts, jas". Is het nog warmer maar nog steeds onder de 18°C, dan vermeld het systeem: "Ontgrendel fiets. Kledij: jas". Is het nog warmer maar minder warm dan 35°C, dan krijg je de melding: "Ontgrendel fiets. Kledij: shirt en short". Vanaf 35°C zegt het systeem: "Hittegolf: zet airco in auto aan. Kledij: shirt en short + sandalen".

3. Maak een programma waarbij je de naam van de dag van een datum berekend. Hiervoor geef je het dagnummer van de eerste maandag van de maand op (bv. "5" voor januari 2026) en het dagnummer van welke dag je de dagnaam wil weten. Bijvoorbeeld: "13" > "Dinsdag", "22 > Donderdag", "1" > "Donderdag".

AND, OR, NOT

Binnen de wiskunde heb je normaal gezien al booleaanse algebra gebruikt (logica). Hierbij kon je uitspraken combineren door ofwel ∧ (of), ∨ (en) en ¬. In Python kan je deze logica ook gebruiken, met respectivelijk de operatoren *and, or* en *not*.

And

Met *and* kan je twee uitspraken aan elkaar verbinden en hier enkel een *true* uit krijgen als beide uitspraken *true* waren. Neem onderstaand voorbeeld. Het getal 0 zal enkel op het scherm verschijnen als het ingegeven getal groter is dan -1 EN kleiner is dan 1.

```
1 |getal = int(input("Getal: "))
2 |if getal > -1 and getal < 1:
3 |      print(f"{getal} is gelijk aan 0.")
```

Or

Met *or* kan je ook twee uitspraken aan elkaar verbinden, maar dan zal er minstens 1 van de voorwaarden *true* moeten zijn. Handig om te weten is ook dat de tweede voorwaarde enkel zal worden geëvalueerd als de eerste voorwaarde *false* was (zie later).

Onderstaand voorbeeld toont aan of een getal niet gelijk is aan 0, door te kijken of het groter of gelijk is aan 1 of kleine of gelijk is aan -1.

Campus Impuls (2025-2026)

```
1 |getal = int(input("Getal: "))
2 |if getal >= 1 or getal <= -1:
3 |      print(f"{getal} is niet gelijk aan 0.")
```

Not

Tenslotte kan je met de *not* een boolean omdraaien van waarde; *true* wordt *false* en *false* wordt *true*.

```
1 |getal = int(input("Getal: "))
2 |if not getal == 0:
3 |      print(f"{getal} is niet gelijk aan 0.")
```

Haakjes

Als je meerdere logica operatoren met elkaar combineert, kan het nogal moeilijk zijn om te onthouden welke operator eerst wordt toegepast. De volgorde is eerst de vergelijkende operatoren (==, !=, >, >=, <, <=), dan *not*, vervolgens *and* en tenslotte *or*. De volledige volgorde vind je hier: https://docs.python.org/3/reference/expressions.html#operator-precedence .

Daarom kan het nuttig zijn om haakjes te gebruiken. Net als bij wiskunde hebben ze voorrang op elke andere operatie, zodat je wat meer structuur en leesbaarheid in jouw voorwaarden kan steken.

In het onderstaande voorbeeld zal het getal enkel worden getoond als:

- Het kleiner is dan 10 en niet gelijk is aan 0
- Of het deelbaar is door 2.

```
1 |getal = int(input("Getal: "))
2 |if (getal < 10 and getal != 0) or getal % 2 == 0 :
3 |      print(getal)
```

BASISOEFENING AND, OR, NOT:

1. Vraag aan de gebruiker een geheel getal. Als het getal een positief (groter dan 0) veelvoud is van 2 (rest na deling door 2 is 0), schrijf je "*getal* is een positief, even getal".

Campus Impuls (2025-2026)

2. Vraag aan de gebruiker een naam. Als de naam gelijk is aan John of aan Jane, dan toon je "Dag John/Jane".
3. Vraag aan de gebruiker een kommagetal. Als het getal niet kleiner is dan 0, schrijf je "*kommagetal* is een niet-strikt, positief getal".
4. Vraag aan de gebruiker een geheel getal. Toon aan de gebruiker "Yes!" of het een strikt positief (groter dan 0), even getal (deelbaar door 2) is, tenzij het -100.

BASISOEFENING AND, OR, NOT (OPLOSSINGEN):

1. Vraag aan de gebruiker een geheel getal. Als het getal een positief (groter dan 0) veelvoud is van 2 (rest na deling door 2 is 0), schrijf je "*getal* is een positief, even getal".

```
1 |getal = int(input("Getal: "))
2 |if getal > 0 and getal % 2 == 0:
3 |    print(f"{getal} is een strikt positief, even ssgetal")
```

2. Vraag aan de gebruiker een naam. Als de naam gelijk is aan John of aan Jane, dan toon je "Dag John/Jane".

```
1 |naam = input("Naam: ")
2 |if naam == "John" or naam == "Jane":
3 |    print("Dag John/Jane")
```

3. Vraag aan de gebruiker een kommagetal. Als het getal niet kleiner is dan 0, schrijf je "*kommagetal* is een niet-strikt, positief getal".

```
1 |getal = float(input("Getal: "))
2 |if not getal < 0:
3 |    print(f"{getal} is een niet-strikt, positief getal")
```

4. Vraag aan de gebruiker een geheel getal. Toon aan de gebruiker "Yes!" of het een strikt positief (groter dan 0), even getal (deelbaar door 2) is, tenzij het -100.

Campus Impuls (2025-2026)

```
4 |getal = int(input("Getal: "))
5 |if (getal >= 0 and getal % 2 == 0) or getal == -100:
6 |    print("Yes!")
```

OEFENINGEN OP AND, OR, NOT , ()

1. Maak een programma die bepaalt of een jaar een schrikkeljaar is. Een schrikkeljaar is een jaar waarbij het jaartal deelbaar is door 4 maar niet door 100. Maar als het deelbaar is door 400, is het wel terug een schrikkeljaar. Bijvoorbeeld: "2000 is een schrikkeljaar", "2004 is een schrikkeljaar", "2003 is geen schrikkeljaar", "1900 is geen schrikkeljaar"
2. Een pizzeria heeft een speciale korting: -20% op alle pizza's voor wie jonger is dan 18 of wie student is (maar niet ouder is 25 jaar). Bevraag de totaalprijs ("Totaal: "), de leeftijd ("Leeftijd: ") en of de persoon nog een student is ("Bent u nog student (ja/nee): "). Je mag ervan uitgaan dat de persoon de laatste vraag ofwel "ja" of "nee" antwoordt.
3. Je maakt een programma waarbij de gebruiker een nieuw wachtwoord kan instellen. Hiervoor moet de gebruiker eerst 2 x zijn vorig wachtwoord ingeven ("Wachtwoord: ") en vervolgens zijn nieuw wachtwoord ("Nieuw wachtwoord: "). Als er 2 keer hetzelfde wachtwoord is ingegeven en het een nieuw wachtwoord is niet hetzelfde als het vorige wachtwoord, toon je op het scherm ("Nieuw wachtwoord is ingesteld!"). Anders toon je ("Nieuw wachtwoord niet ingesteld!").
4. Om te controleren of een leerling naar buiten mag over de middag, schrijf je een programma. Een leerling mag naar buiten in het jaar dat die 16 jaar wordt (vraag "Geboortejaar: " aan de gebruiker). Tenzij ze een uitdrukkelijke afwijzing hebben van hun ouders ("Mag uw zoon/dochter het school wel/niet verlaten: "). Leerlingen vanaf het jaar dat ze 18 jaar worden, mogen sowieso naar buiten. Bijvoorbeeld (in het jaar 2026):

"Geboortejaar: "	("Mag uw zoon/dochter het school wel/niet verlaten: "	Output
2008	Wel	Toegestaan
2008	Niet	Toegestaan

Campus Impuls (2025-2026)

2010	Wel	Toegestaan
2010	Niet	Geweigerd
2011	Wel	Geweigerd
2011	Niet	Geweigerd

Campus Impuls (2025-2026)

Hulp bij programmeren/algoritmes

Soms kan het moeilijk zijn om aan een oefening te beginnen, omdat je niet exact weet wat je eerst moet doen of welke stappen je allemaal moet zetten. Vaak helpt het maken van kleine oefeningen hierin omdat je dit proces dan traint.

Als dit toch onvoldoende lukt, kan je proberen om op papier neer te schrijven hoe jij het probleem zou oplossen als je het aan iemand anders zou oplossen. Je schrijft dan lijn per lijn wat die andere persoon zou moeten doen, en probeert dit dan achteraf te vertalen naar een programmeertaal (hier Python).

Bijvoorbeeld: maak een programma die de nulwaarden bepaalt van een tweedegraadsvergelijking ($ax^2 + bx + c$).

1. Je weet dat je hiervoor de formule van de discriminant nodig hebt ($D = b^2 - 4ac$).
2. Voor je dit kan uitrekenen, moet je eerst weten wat de waarde is van a, b en c.
3. Je schrijft dus als eerste op:
    a. **Vraag getal a aan de gebruiker**
    b. **Vraag getal b aan de gebruiker**
    c. **Vraag getal c aan de gebruiker**
4. Vervolgens moet je de discriminant berekenen. Je schrijft dus:
    d. **Bereken discriminant.**
5. Tenslotte moet je kijken of je discriminant groter, kleiner of gelijk is aan 0.
    e. **Controleer of D > 0**
6. Als dit waar is, bereken je de twee nulwaarden: $\frac{-b \pm \sqrt{D}}{2a}$
    f. **Als D > 0**
    g. **Bereken nulwaarde 1 en toon**
    h. **Bereken nulwaarde 2 en toon**
7. Als dit niet waar is, maar D is wel gelijk aan 0, schrijf je:
    i. **Als D = 0**
    j. **Bereken nulwaarde en toon**
8. Als dit ook niet waar was, dan zijn er geen oplossingen. Dan schrijf je dus:
    k. **Als D < 0**
    l. **Toon dat er geen oplossingen zijn.**

Dit vertaal je dan stap-voor-stap naar het volgende programma:

Campus Impuls (2025-2026)

```
1 |  a = float(input("a = "))    (a)
2 |  b = float(input("b = "))    (b)
3 |  c = float(input("c = "))    (c)
4 |  D = (b**2)-(4*a*c)          (d)
5 |  if D > 0:                   (e)
6 |     nulwaarde_1 = ((b*-1)+(D**(1/2)))/(2*a)      (g)
7 |     print(nulwaarde_1)                           (g)
8 |     nulwaarde_2 = ((b*-1)-(D**(1/2)))/(2*a)      (h)
9 |     print(nulwaarde_2)                           (h)
10 | if D == 0:                  (i)
11 |    nulwaarde = (b*-1)/(2*a) (j)
12 |    print(nulwaarde)         (j)
13 | if D < 0:                   (k)
14 |    print("Geen nulwaarden)  (l)
```

Campus Impuls (2025-2026)

Soms wordt hiervoor ook het **Nassi-Shneidermandiagram** hiervoor gebruikt. Hierbij maak je van de bovenstaande stappen een tabel/diagram, waarbij je bij een keuze een splitsing van jouw code maakt.

<table>
<tr><td colspan="2">Vraag getal a aan de gebruiker</td></tr>
<tr><td colspan="2">Vraag getal b aan de gebruiker</td></tr>
<tr><td colspan="2">Vraag getal c aan de gebruiker</td></tr>
<tr><td colspan="2">Bereken discriminant</td></tr>
<tr><td colspan="2">Controleer of D > 0</td></tr>
<tr><td>Ja</td><td>Nee</td></tr>
<tr><td>Bereken nulwaarde 1</td><td rowspan="4"></td></tr>
<tr><td>Toon nulwaarde 1</td></tr>
<tr><td>Bereken nulwaarde 2</td></tr>
<tr><td>Toon nulwaarde 2</td></tr>
<tr><td colspan="2">Controleer of D = 0</td></tr>
<tr><td>Ja</td><td>Nee</td></tr>
<tr><td>Bereken nulwaarde</td><td rowspan="2">Toon dat er geen oplossingen zijn.</td></tr>
<tr><td>Toon nulwaarde</td></tr>
</table>

Je ziet dat je dan verschillende kleuren kan gebruiken om bepaalde acties aan te duiden:

- Geel: invoer/uitvoer
- Blauw: berekening
- Grijs: if/else

Een paar tips bij het opstellen van een Nassi-Schneidermandiagram:

- Bedenk welke variabelen/geheugenvelden je nodig hebt? Moet ik bepaalde informatie ingeven, een waarde berekenen of iets uitvoeren, ga dan na of je hier een variabele voor nodig hebt.
- Schrijf per variabele/waarde een kolom in een tabel, waarbij je de ingegeven waarden hiervan kan opschrijven. Voor bovenstaande algoritme, zou dit er als volgt uit kunnen zien:

a:	1
b:	-5
c:	6
D:	1
nulwaarde_1	2
nulwaarde_2	3

- Ga na welke manipulatie (bewerkingen, toekenningen aan variabelen, ..) nodig zijn om het juiste resultaat op de juiste plaats te krijgen.

Campus Impuls (2025-2026)