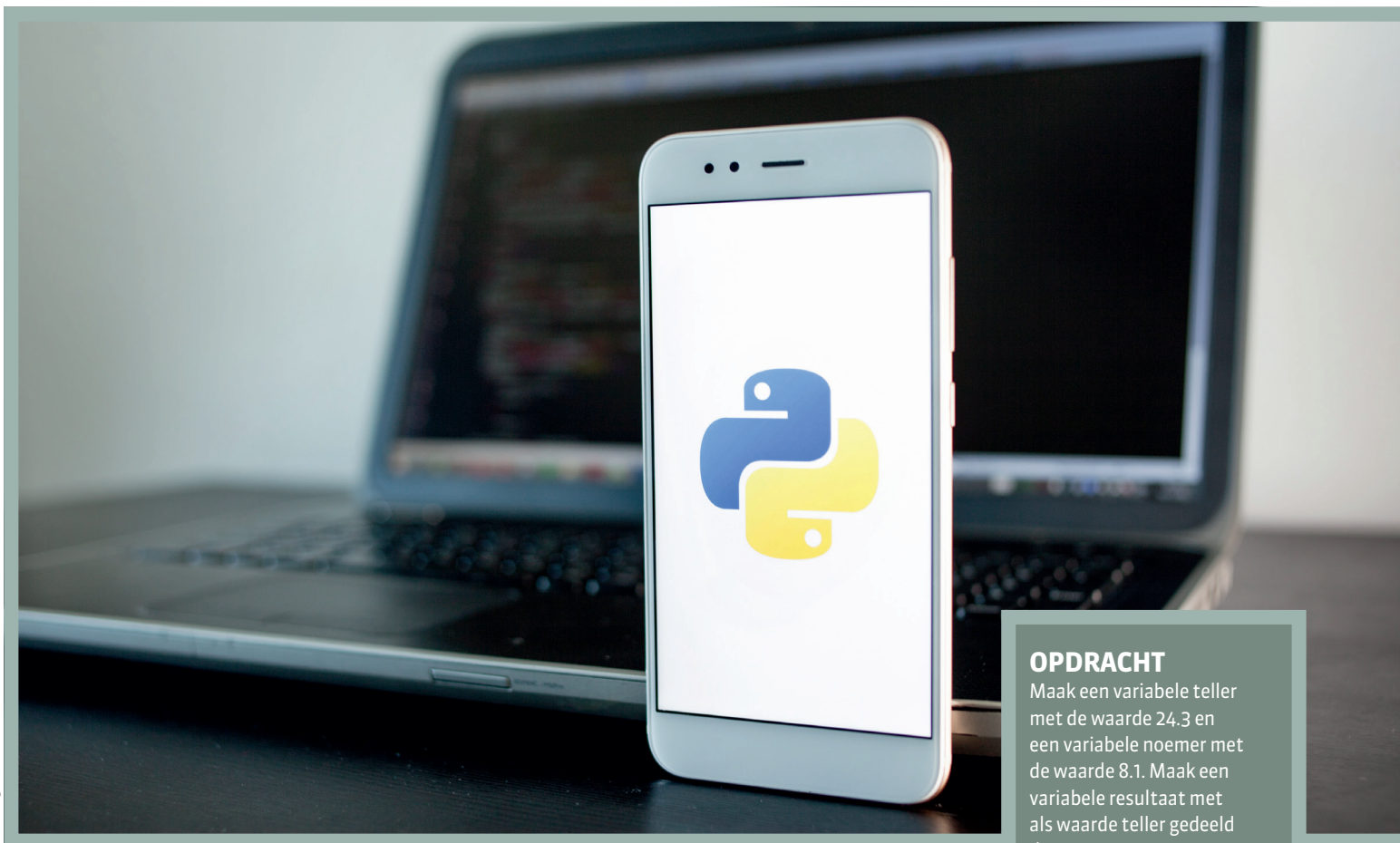


LISTING UITLEG

- > Start nieuwe regel
- regel met spatie na afbreking
- regel zonder spatie na afbreking



Trismegist.san/Shutterstock.com



naam, zolang het niet dezelfde naam is als een geregistreerd woord uit de Python-taal. Vervolgens kun je die variabelen gewoon in de plaats van de getallen die ze voorstellen gebruiken:

```
>>> teller=7
>>> noemer=3
>>> resultaat=teller//noemer
>>> rest=teller%noemer
>>> teller
7
>>> noemer
3
>>> resultaat
2
>>> rest
1
>>> type(rest)
<class 'int'>
```

Verwijs je naar een variabele waar je nog geen waarde aan hebt toegekend, bijvoorbeeld door een tyffout, dan krijg je een foutmelding:

```
>>> type(delta)
Traceback (most recent call
last):
> File "<stdin>", line 1, in
<module>
NameError: name 'delta' is not
defined
```

CHEATSHEET

int (integer) = geheel getal
float (floating-point arithmetic) = breuk of kommagetal
str (string) = tekenreeks
operator = bewerking

TEKST

Python kent naast int en float nog heel wat andere ingebouwde types. Een veelgebruikt datatype is **str**, dat we vaak voluit string noemen. Een string stelt tekst voor, bijvoorbeeld **'Dit is een tekst'**. Een string mag je zowel tussen dubbele als enkele aanhalingstekens zetten. Goed gebruik is wel om voor jezelf altijd één keuze te maken.

Omdat de aanhalingstekens worden gebruikt om het begin en einde van een string aan te geven, moet je iets speciaals doen als je ze in je tekst zelf wilt gebruiken: je dient ze te 'escapen'. Dat doe je door er een backslash voor te zetten. Bijvoorbeeld: **'Dit is een \'tekst\''**.

Net zoals we bewerkingen op getallen kunnen uitvoeren, kan dat

ook op tekst. Enkele voorbeelden maken dit duidelijk:

```
>>> 'Py' 'thon'
'Python'
>>> 'Py'+ 'thon'
'Python'
>>> 3*'Py'+2*'thon'
'PyPyPythonthon'
'PyPyPythonthon'
```

Daarnaast kun je allerlei functies op een string toepassen:

```
>>> len('zandzeepsodamineraal
watersteenstralen')
37
>>> 'dit is een TEKST'.
capitalize()
'Dit is een tekst'
>>> 'dit is een TEKST'.lower()
'dit is een tekst'
>>> 'dit is een TEKST'.upper()
'DIT IS EEN TEKST'
>>> 'dit is een TEKST'.swapcase()
'DIT IS EEN tekst'
```

In het tweede deel, in Linux Magazine #6 2020, leer je hoe je met individuele letters in een tekst om kunt gaan.

SAMENVATTING

In deze eerste les heb je kennisgemaakt met de programmeertaal

OPDRACHT

Maak een variabele teller met de waarde 24.3 en een variabele noemer met de waarde 8.1. Maak een variabele resultaat met als waarde teller gedeeld door noemer en zorg ervoor dat dit als geheel getal (dus niet als float) wordt weergegeven.

UITWERKING

```
>>> teller=24.3
>>> noemer=8.1
>>> resultaat=int(teller/
noemer)
>>> resultaat
3
```

De eerste twee regels spreken hopelijk voor zich. In de derde regel delen we de teller door de noemer en wijzen de uitkomst toe aan **resultaat**. Zouden we het hierbij laten, dan zou de uitkomst weergegeven worden als 3.0. Door het type **int** toe te wijzen aan **resultaat**, wordt de uitkomst als geheel getal weergegeven, in dit geval dus 3.

Python. Je kunt met de Python-interpreter werken en je hebt kennisgemaakt met basisbewerkingen op getallen en strings. Je weet ook wat datatypes en variabelen zijn. In deel 2 gaan we met complexere datatypes aan de slag. <