



## Wetenskaplike Berekening 272 / Scientific Computing 272

### Tutoriaal 3: Modules in Python / Tutorial 3: Modules in Python

2020-04-16 Opgestel deur Willem Bester Gemodereer deur Brink van der Merwe

#### Agtergrond

Dié tutoriaal is 'n praktiese oorsig van modules in Python. Ter voorbereiding, bestudeer die derde reeks Python-skyfies. Ek stel voor dat u die gids `~/wb272/tut03` skep vir u werk van dié tutoriaal.

#### Uitkomst

Wanneer u die tutoriaal voltooi het, behoort u in staat te wees om die volgende te doen: (1) bestaande Python-modules in te voer en te gebruik, (2) u eie Python-modules te skep, en (3) stringmetodes te gebruik.

#### Oefeninge

- Voer die `math`-module in en gebruik sy funksies om die volgende oefeninge te voltooi:
  - Skryf 'n uitdrukking wat die waarde  $-4.3$  afrond en dan die absolute waarde van die afrondingsresultaat bereken.
  - Skryf 'n uitdrukking van die plafon van `sin(34.5)` bereken.
- In die volgende oefeninge gaan u werk met die `calendar`-module van Python.
  - Lees die dokumentasie by <https://docs.python.org/3/library/calendar.html> om te sien watter funksies beskikbaar is.
  - Vind en gebruik 'n funksie in die `calendar`-module om die aantal skrikkeljare tussen die jare 2000 en 2050, inklusief, te bepaal.
  - Vind en gebruik 'n funksie in die `calendar`-module om die dag van die week waarop 13 September 2019 val, te bepaal.
- Gebruik stringmetodes en skryf 'n uitdrukking vir elkeen van die volgende:
  - Kapitaliseer `'boolean'`.
  - Vind die eerste voorkoms van `'2'` in `'CO2 H2O'`.

#### Background

This tutorial is a practical overview of modules in Python. To prepare, study the third set of Python slides. I suggest you create the directory `~/wb272/tut03` for your work for this tutorial.

#### Outcomes

When you have completed the tutorial, you should be able to do the following: (1) import and use existing Python modules, (2) create your own Python modules, and (3) use string methods.

#### Exercises

- Import `math` module, and use its functions to complete the following exercises:
- Write a single expression that rounds the value  $-4.3$  and then calculates the absolute value of the rounding result.
- Write an expression that calculates the ceiling of `sin(34.5)`.
- In the following exercises, you will work with the `calendar` module of Python.
- Read the documentation at <https://docs.python.org/3/library/calendar.html> to see what functions are available.
- Find and use a function in the `calendar` module to determine how many leap years there will be between the years 2000 and 2050, inclusive.
- Find and use a function in the `calendar` module to determine on which day of the week 13 September 2019 will fall.
- Use string methods, and write an expression for each of the following:
- Capitalise `'boolean'`.
- Find the first occurrence of `'2'` in `'CO2 H2O'`.

- (c) Vind die laaste voorkoms van '2' in 'CO2 H2O'. Find the last occurrence of '2' in 'CO2 H2O'.
- (d) Vind die tweede voorkoms van '2' in 'CO2 NO2 H2O'. Find the second occurrence of '2' in 'CO2 NO2 H2O'.
- (e) Stel vas of 'Boolean' met 'n kleinletter begin al dan nie. Determine whether or not 'Boolean' begins with a lowercase letter.
- (f) Skakel 'MoNDaY' om na kleinletters en kapitaliseer die resultaat. Convert 'MoNDaY' to lowercase and the capitalise the result.
- (g) Verwyder die leidende spasies uit ' Monday'. Remove the leading whitespaces from ' Monday'.
4. Een manier om 'n ewekansige getal uit die Gaussiese (normaal) verdeling op te wek, is om die **Box-Muller vergelyking**, One way to generate a random number from the Gaussian (normal) distribution is to use the **Box-Muller formula**,

$$w = \sin(2\pi v)(-2 \ln u)^{1/2}, \quad (1)$$

te gebruik, waar  $u$  en  $v$  reële getalle tussen 0 en 1 is, opgewek deur die funksie `random.random()`. Skryf 'n parameterlose funksie `gaussrand` wat 'n standaard Gaussiese ewekansige waarde (dit wil sê, met gemiddeld 0 en variansie 1) terugstuur. Hoe sou u te werk gaan om dié funksie te toets? LET WEL: Aangesien ons hier met ewekansige getalle werk, moet u breër dink as die gewone manier van toets vir verwagte antwoorde.

where  $u$  and  $v$  are real numbers between 0 and 1, generated by the `random.random()` function. Write a parameterless function `gaussrand` that returns a standard Gaussian random value (that is, with mean 0 and variance 1). How would you go about testing this function? NOTE: Since we are working random numbers here, you must think wider than the normal way of testing for expected answers.

5. Die *Mercator-projeksie* is 'n konformele (dit is, hoekbehoudende) projeksie wat breedtegraad  $\phi$  en lengtegraad  $\lambda$  op reghoekige koördinate  $(x, y)$  afbeeld. Dit word wyd gebruik—byvoorbeeld, vir seekaarte en ook vir die kaarttoepassings wat 'n mens op die web kan bykom; vir meer detail, verwys na [https://en.wikipedia.org/wiki/Mercator\\_projection](https://en.wikipedia.org/wiki/Mercator_projection). Die projeksie word gedefinieer deur die vergelykings
- The *Mercator projection* is a conformal (that is, angle-preserving) projection that maps latitude  $\phi$  and longitude  $\lambda$  to rectangular coordinates  $(x, y)$ . It is widely used—for example, for nautical charts and for the maps you can access on the web; for more detail, refer to [https://en.wikipedia.org/wiki/Mercator\\_projection](https://en.wikipedia.org/wiki/Mercator_projection). The projection is defined by the equations

$$x = \lambda - \lambda_0, \text{ and} \quad (2)$$

$$y = \frac{1}{2} \ln \left( \frac{1 + \sin \phi}{1 - \sin \phi} \right), \quad (3)$$

waar  $\lambda_0$  die lengtegraad van die punt in die middel van die kaart is. Skryf 'n funksie `mercator` wat  $\lambda_0$  en die breedte- en lengtegraad van 'n punt as parameters neem (in dié volgorde, *almal in grade*) en die projeksie  $(x, y)$  terugstuur. WENK: Onthou om na radiale om te skakel waar nodig.

where  $\lambda_0$  is the longitude of the point in the centre of the map. Write a function `mercator` that takes  $\lambda_0$  and the latitude and longitude of a point as parameters (in this order, *all in degrees*) and returns its projection  $(x, y)$  as a tuple. HINT: Remember to convert to radians where necessary.

6. Gaan deur Tutoriale 1, 2 en 3, en skryf toetsgevalle vir al die vrae wat behels dat u 'n funksie moet skryf. (Maak terselfdertyd seker dat u dié vrae en hul oplossings goed verstaan—dit is goeie voorbereiding vir die semester-toets.)
- Go through Tutorials 1, 2, and 3, and write test cases for all questions that involve your writing a function. (At the same time, make sure that you understand these questions and their solutions well—it is good preparation for the semester test.)

Gebruik die `nose`-module vir toetsing. Basiese gebruiksinligting in die skyfies beskikbaar; meer gevorderde inligting is by <https://nose.readthedocs.org/en/latest/usage.html> beskikbaar.

Use the `nose` module for testing. Basic usage information is available in the slides; more advanced information is available at <https://nose.readthedocs.org/en/latest/usage.html>.