Table of Contents

Chapter 15

Modules

A Python module is a file containing Python definitions and statements. A module can define functions, classes, and variables. A module can also include runnable code. Grouping related code into a module makes the code easier to understand and use. It also makes the code logically organized.
- Modules in Python can be of two types:
1. User-defined Modules
2. Built-in Modules.

1. User-defined module

Create a Module

To create a module just save the code you want in a file with the file extension .py:
Save this code in a file named mymodule.py

```
def greeting(name):
  print("Hello, " + name)
```

Use a Module

Now we can use the module we just created, by using the import statement:
Import the module named mymodule, and call the greeting function:

```
import mymodule
mymodule.greeting("Madi")
```

Or

Name a module:

You can create an alias when you import a module, by using the as keyword:

```
import mymodule as m
m.greeting("Madi")

# Hello, Madi
```

2. Built-in Modules

There are several built-in modules in Python, which you can import whenever you like.

Ex.

```
import platform
x = platform.system()
print(x) # windows
```

The dir() Function

There is a built-in function to list all the function names (or variable names) in a module. The dir() function:

Ex.

```
import math
x = dir(math)
print(x)
```

```
# ['__doc__', '__loader__', '__name__', '__package__', '__spec__', 'acos', 'acosh',
'asin', 'asinh', 'atan', 'atan2', 'atanh', 'ceil', 'comb', 'copysign', 'cos', 'cosh',
'degrees', 'dist', 'e', 'erf', 'erfc', 'exp', 'expm1', 'fabs', 'factorial', 'floor', 'fmod',
'frexp', 'fsum', 'gamma', 'gcd', 'hypot', 'inf', 'isclose', 'isfinite', 'isinf', 'isnan', 'isqrt',
'lcm', 'ldexp', 'lgamma', 'log', 'log10', 'log1p', 'log2', 'modf', 'nan', 'nextafter',
'perm', 'pi', 'pow', 'prod', 'radians', 'remainder', 'sin', 'sinh', 'sqrt', 'tan','tanh', 'tau',
'trunc', 'ulp']
```

Note: The dir() function can be used on all modules, also the ones you create yourself.

Import From Module:

You can choose to import only parts from a module, by using the from keyword.:

Ex.

```
# import only the person1 dictionary from the module:
from mymodule import person1
print (person1["age"])
```