Lecture 2: Functions (part I)

Morten Rieger Hannemose, Vedrana Andersen Dahl

Fall 2023

Today's lecture

1. Functions (ca. 20 min)
2. Functions live demo (ca. 20 min)
3. Course material setup live demo (ca. 20 min)

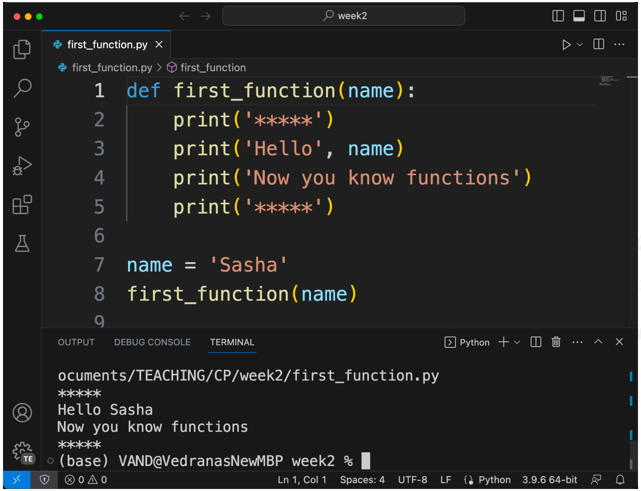
Definition

A function (in the context of programming) is a named sequence of statements that performs a computation.

Today we learn. . .

- Why write functions? Grouping, re-use, breaking down the problem. . .
- Coming up next: How to write functions? How to use functions?

First function



```
def first_function(name):
    print('*****')
    print('Hello', name)
    print('Now you know functions')
    print('*****')

name = 'Sasha'
first_function(name)
```

```
ocuments/TEACHING/CP/week2/first_function.py
*****
Hello Sasha
Now you know functions
*****
(base) VAND@VedranasNewMBP week2 %
```

- Writing functions:
  - function header
  - function body (Careful about the indentation!)
- Using functions:
  - function call

Functions come in different flavors

- Often, a function takes an argument (input) and returns a result, a return value (output)
- A function may have no, one, or several arguments
- A function may return a value (fruitful function) or not (void function)
- A function may have side effects

Functions can be

- Built-in (provided in Python), e.g. `print()`, `str()`
- Part of a package, e.g. `math.sin()`
- User-made

Important for functions (common pitfalls)

- Statements in the function body are executed when function is called. Not before!
- Parameters and variables defined inside the function are local

A problem solved using a function

Problem

Write a function `rectangle_area` that calculates and prints the area of the rectangle. As input, the function should take two variables `length` and `width`. For example, given as input 5 and 3, the function should print the message `The area is: 15`.

Test the function on an input `length=14.5` and `width=12`. The function should print the message `The area is: 74.0`.

Solution

```python
def rectangle_area(length, width):
    area = length * width
    print('The area is:', area)

rectangle_area(14.5, 12)
```

What is printed?

Example

```python
def my_function(a):
    print(a)

b = 72.2
my_function(b)
```

Examples

What is printed?

Example

```python
def my_function(a):
    print(a)

a = 13.6
my_function(17)
```

What is printed?

Example

```python
def full_price(price):
    rate = 0.2
    tip = rate * price
    total = price + tip
    print('Full price is', total)

cake_price = 100
full_price(cake_price)
print(cake_price)
print(tip)
print(price)
```

Problem

Write a function `area` that calculates and prints the area of the rectangle. (...)

Test the function on an input `a=8` and `b=16`. The function should print the message `Area is: 128`.

What is strange (wrong!) in the suggested solution?

Solution

```python
def area(a, b):
	a = 8
	b = 16
	print('Area is:', a*b)

area(8, 16)
```

What is printed?

Example

```python
def shout_name(name):
    print('Hey, ' + name + '!!!!')

def shout_twice(name):
    shout_name(name)
    shout_name(name)

shout_twice('Emmy')
```

Good practice

- A function should do one thing
- A function not be more than 20 lines long.
- Choose a descriptive name for your function, and its arguments
- Start by writing a program. Then, group and encapsulate (turn into functions)

Advanced

- Positional arguments and keyword arguments
- Default arguments