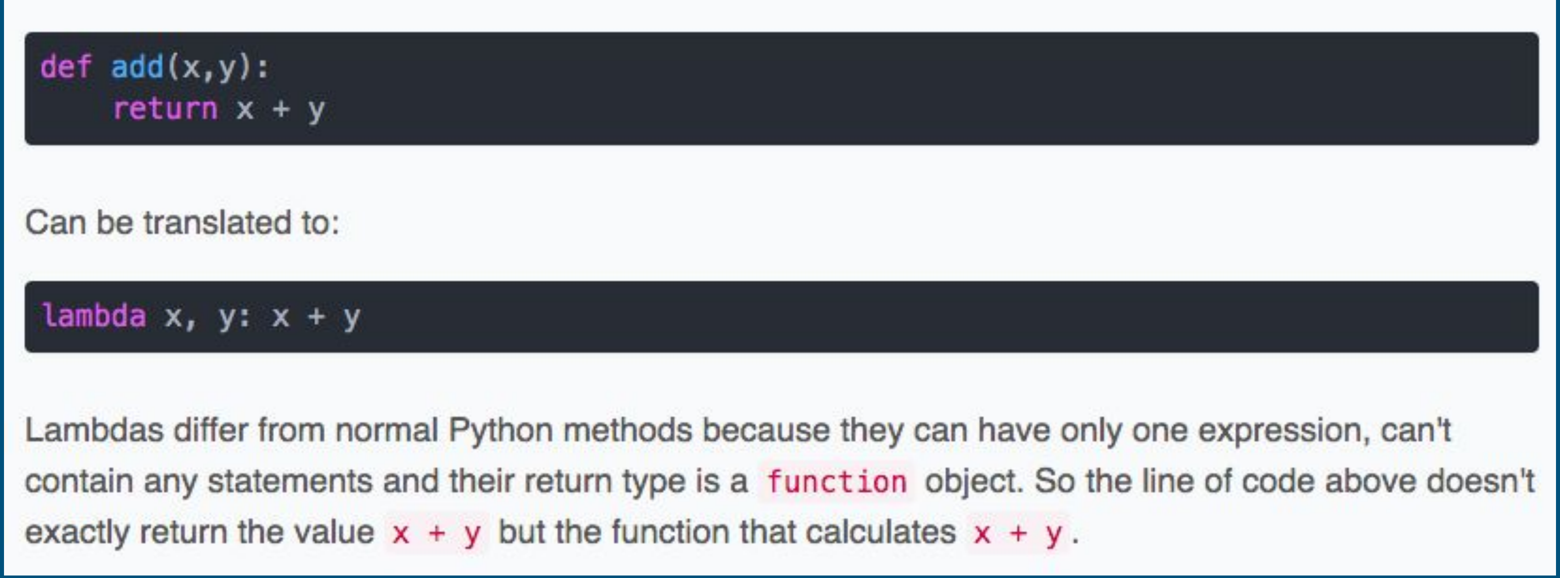
# SI 507 Lab #2

September 6

# General Information

- Recordings will be available on Canvas.
- Answers to lab exercises will be available on Canvas at the end of the week.
- Homework #1 is due on Monday, 9/12 by **7:59 PM**.

# Lambda Functions



```python
def add(x,y):
    return x + y
```

Can be translated to:

```python
lambda x, y: x + y
```

Lambdas differ from normal Python methods because they can have only one expression, can't contain any statements and their return type is a `function` object. So the line of code above doesn't exactly return the value `x + y` but the function that calculates `x + y`.

# Lambda Functions

AKA “anonymous functions”

- Great for one-time use
- Doesn’t require blocks of code or naming

# Lambda Functions

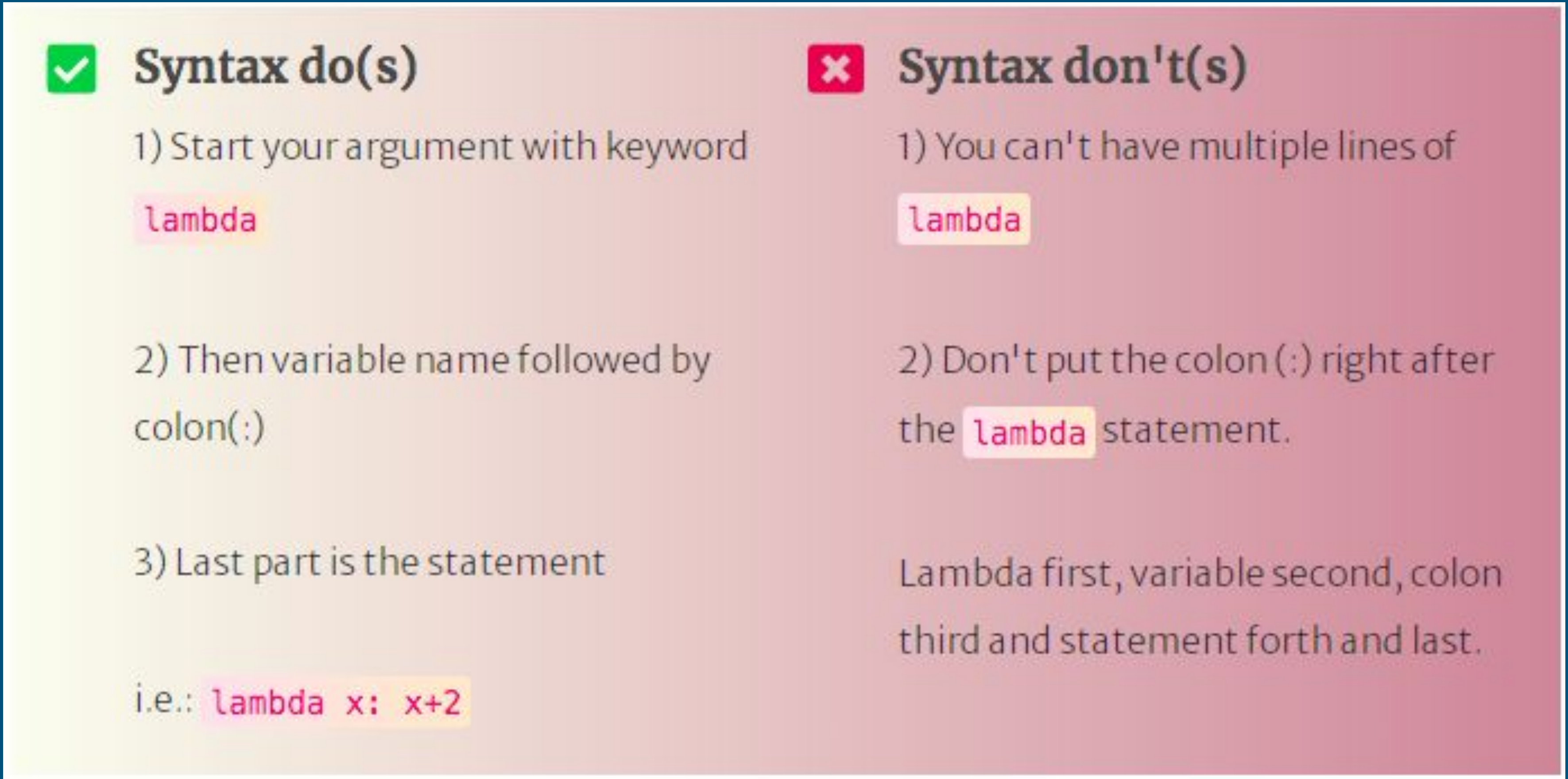
- Quick alternative to a regular user-defined function
- As an argument for some other functions:
  - .sort()
  - sorted()
  - filter()
  - map()
  - reduce()

  ...etc.

# Lambda Functions

- Quick alternative to a regular user-defined function
- As an argument for some other functions:
    - **.sort()**
    - **sorted()**
    - filter()
    - map()
    - reduce()

    ...etc.



## ✅ Syntax do(s)

1) Start your argument with keyword `lambda`

2) Then variable name followed by colon(:)

3) Last part is the statement

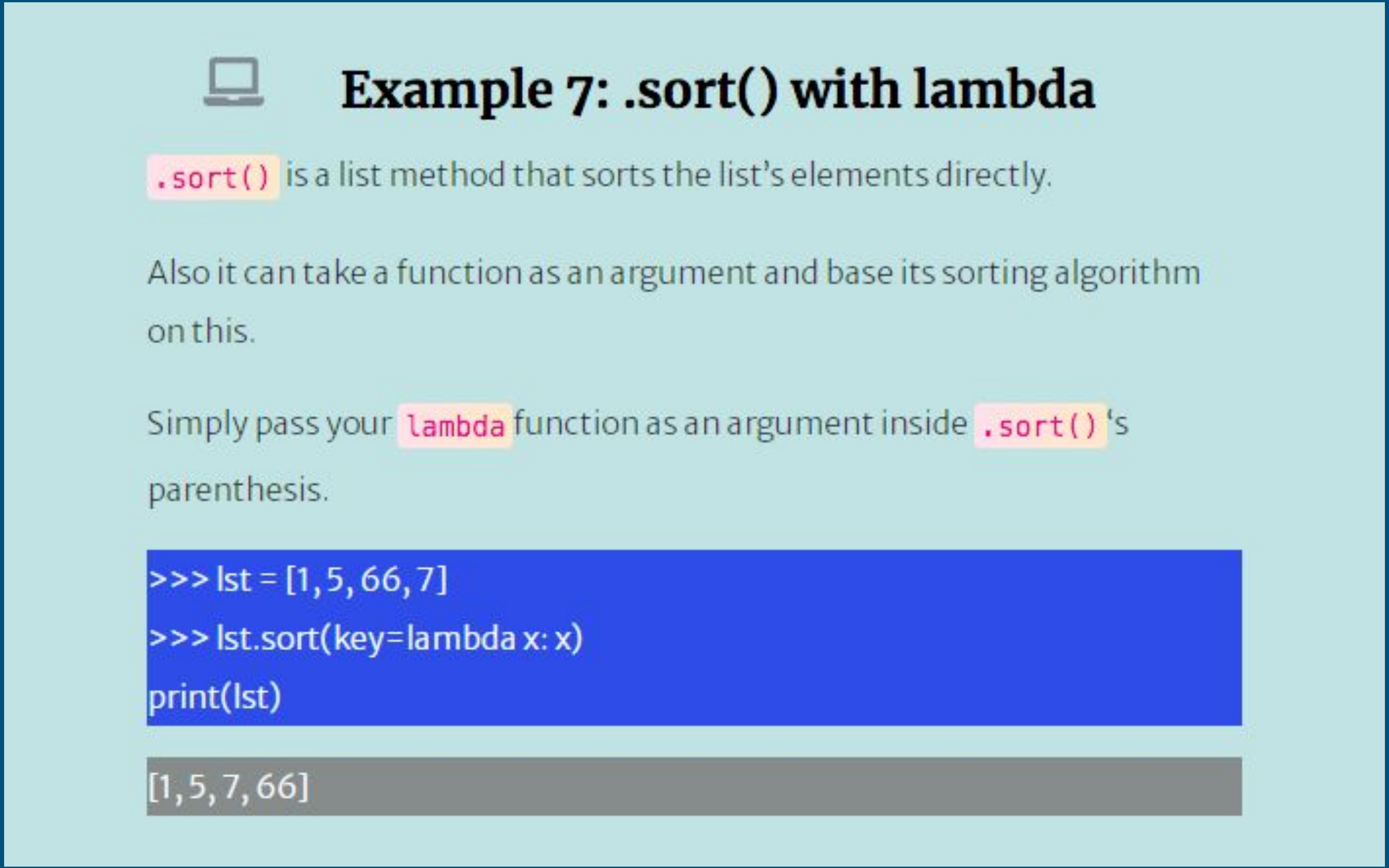
i.e.: `lambda x: x+2`

## ❌ Syntax don't(s)

1) You can't have multiple lines of `lambda`

2) Don't put the colon (:) right after the `lambda` statement.

Lambda first, variable second, colon third and statement forth and last.



# Example 7: .sort() with lambda

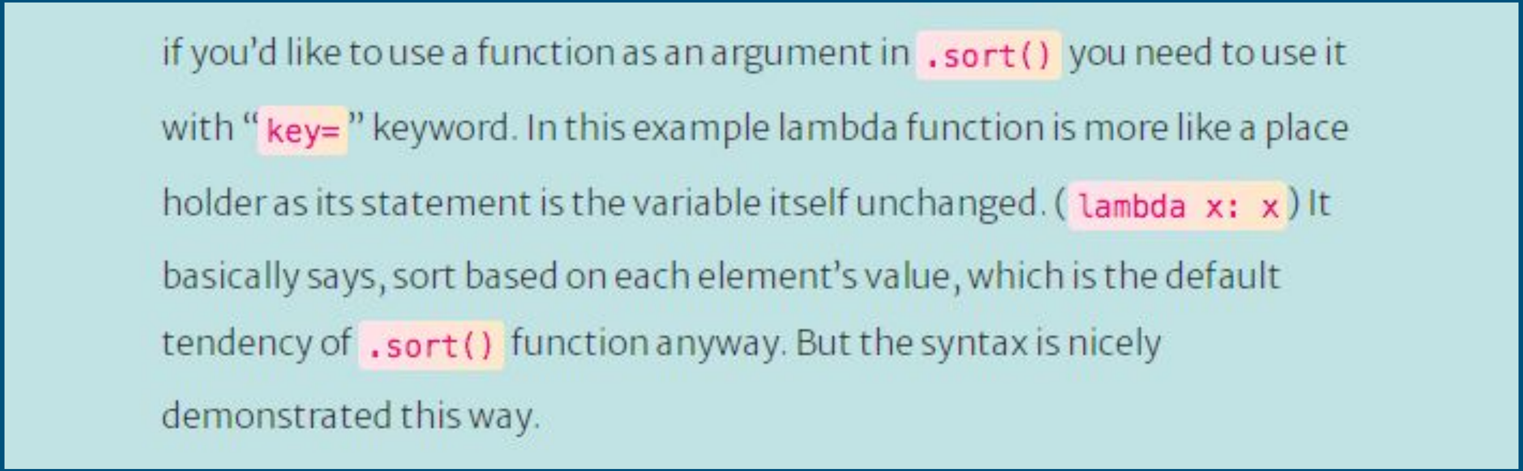
`.sort()` is a list method that sorts the list's elements directly.

Also it can take a function as an argument and base its sorting algorithm on this.

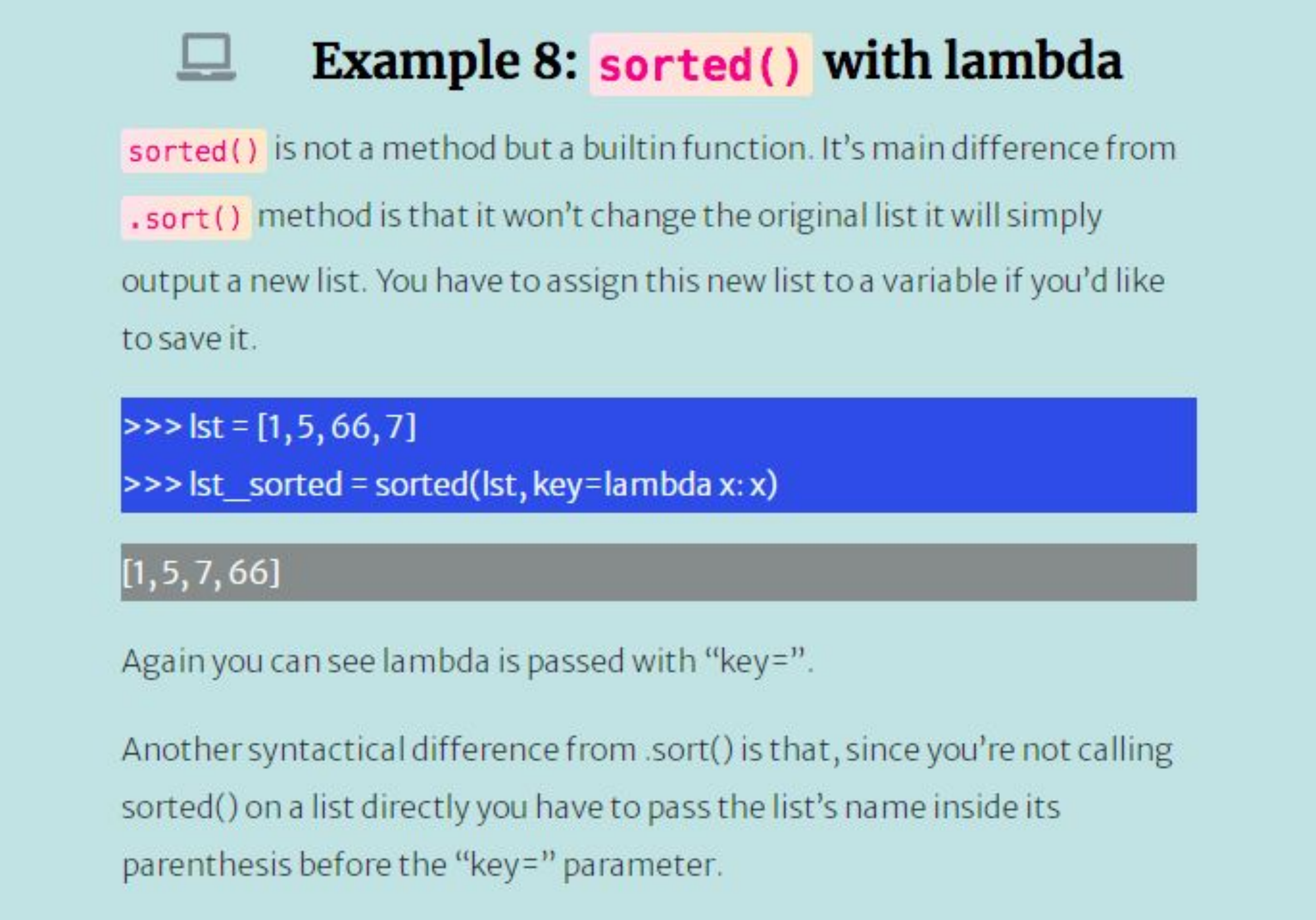
Simply pass your `lambda` function as an argument inside `.sort()` 's parenthesis.

```
>>> lst = [1, 5, 66, 7]
>>> lst.sort(key=lambda x: x)
print(lst)
```

```
[1, 5, 7, 66]
```



if you'd like to use a function as an argument in `.sort()` you need to use it with "`key=`" keyword. In this example lambda function is more like a place holder as its statement is the variable itself unchanged. (`lambda x: x`) It basically says, sort based on each element's value, which is the default tendency of `.sort()` function anyway. But the syntax is nicely demonstrated this way.



## Example 8: `sorted()` with lambda

`sorted()` is not a method but a builtin function. It’s main difference from `.sort()` method is that it won’t change the original list it will simply output a new list. You have to assign this new list to a variable if you’d like to save it.

```
>>> lst = [1, 5, 66, 7]
>>> lst_sorted = sorted(lst, key=lambda x: x)
```

```
[1, 5, 7, 66]
```

Again you can see lambda is passed with “key=”.

Another syntactical difference from .sort() is that, since you’re not calling sorted() on a list directly you have to pass the list’s name inside its parenthesis before the “key=” parameter.

# Some more exercises...

Functions and Strings

Exercise 5.17

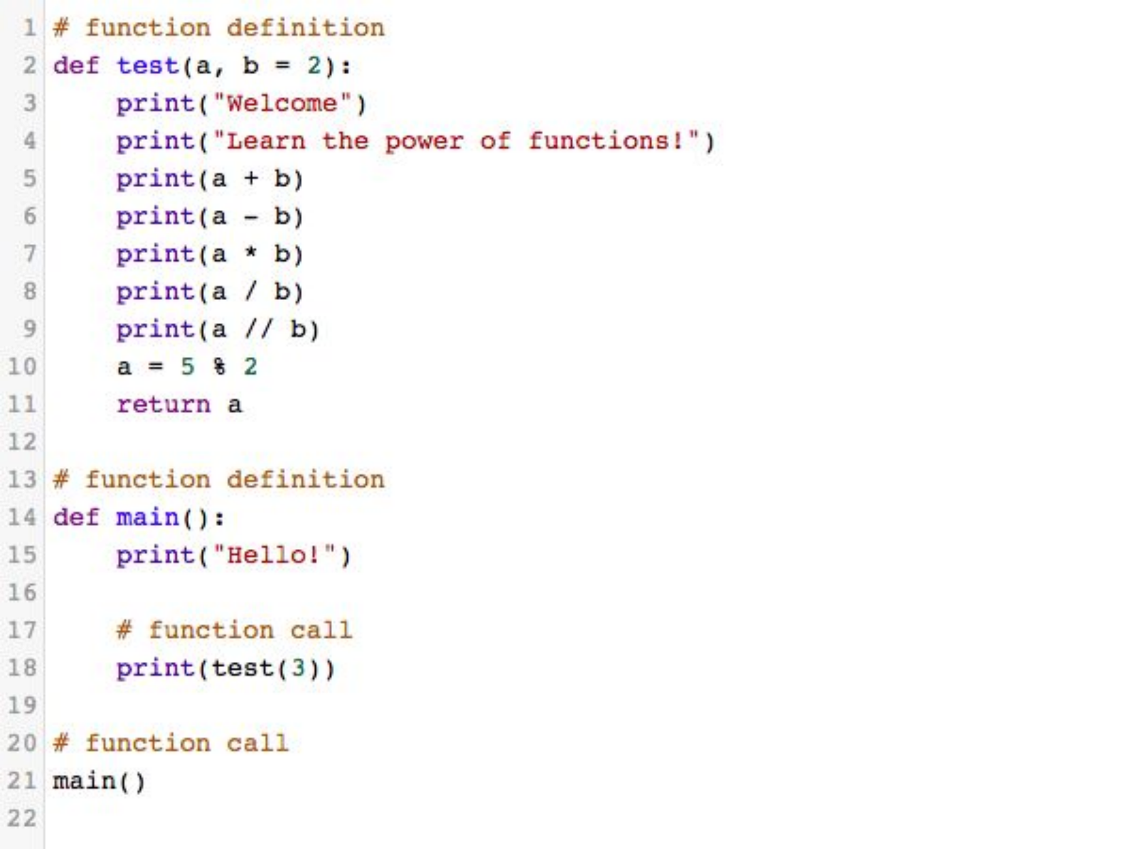
**https://bit.ly/3x1AFMp**

Exercise 5.21

**https://bit.ly/3QeJEkl**

# Exercise 5.17

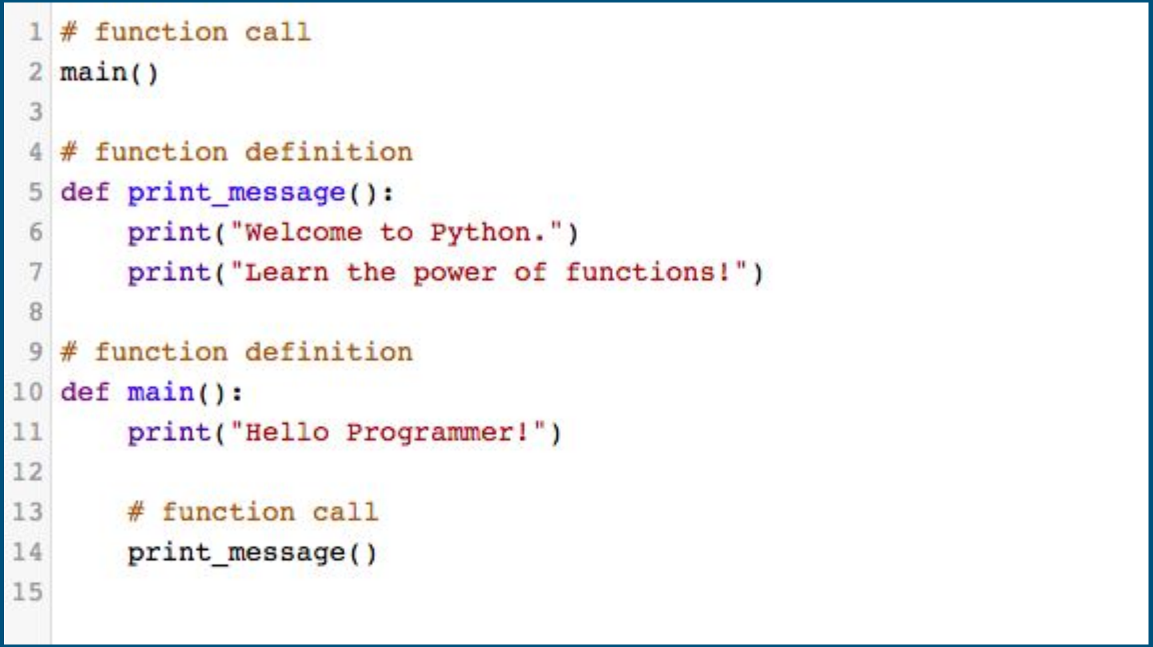
Use this code to answer 5.17.1.1 -- 5.17.1.3, 5.17.1.7



```python
# function definition
def test(a, b = 2):
    print("Welcome")
    print("Learn the power of functions!")
    print(a + b)
    print(a - b)
    print(a * b)
    print(a / b)
    print(a // b)
    a = 5 % 2
    return a

# function definition
def main():
    print("Hello!")

    # function call
    print(test(3))

# function call
main()

```

# Exercise 5.17

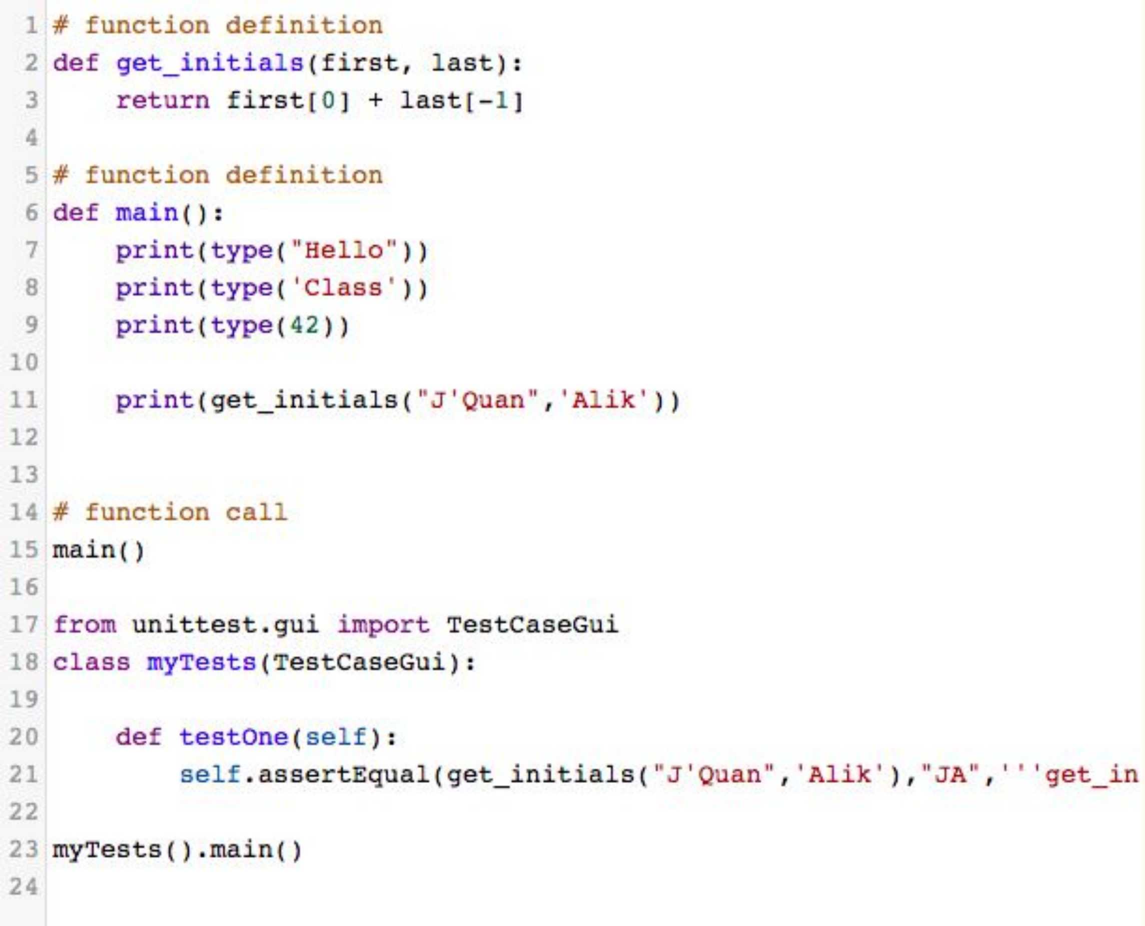
Use this code to answer 5.17.3.1



```python
# function call
main()

# function definition
def print_message():
    print("Welcome to Python.")
    print("Learn the power of functions!")

# function definition
def main():
    print("Hello Programmer!")

    # function call
    print_message()

```

# Exercise 5.21

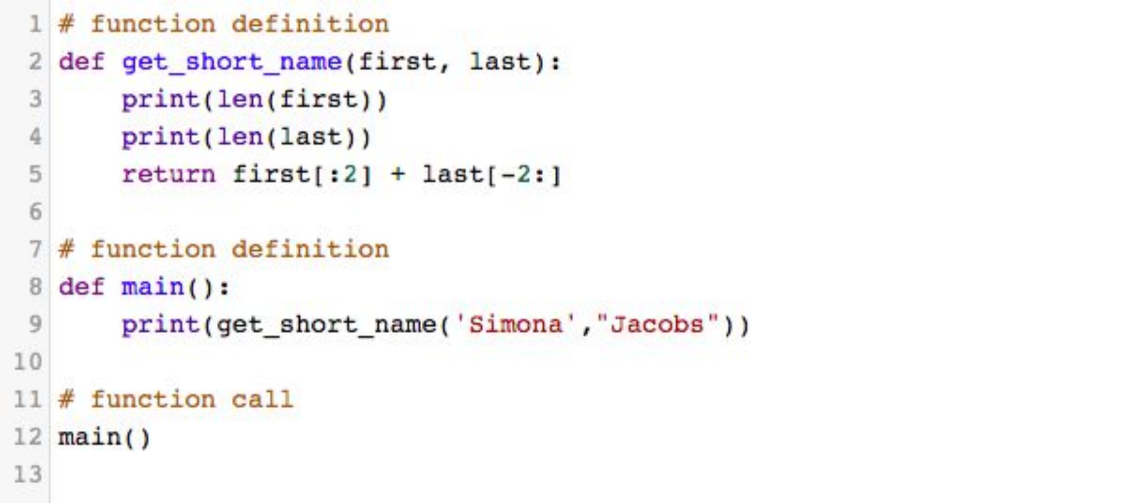
Use this code to answer 5.21.1.1



```python
# function definition
def get_initials(first, last):
    return first[0] + last[-1]

# function definition
def main():
    print(type("Hello"))
    print(type('Class'))
    print(type(42))

    print(get_initials("J'Quan",'Alik'))


# function call
main()

from unittest.gui import TestCaseGui
class myTests(TestCaseGui):

    def testOne(self):
        self.assertEqual(get_initials("J'Quan",'Alik'),"JA",'''get_in

myTests().main()

```

# Exercise 5.21

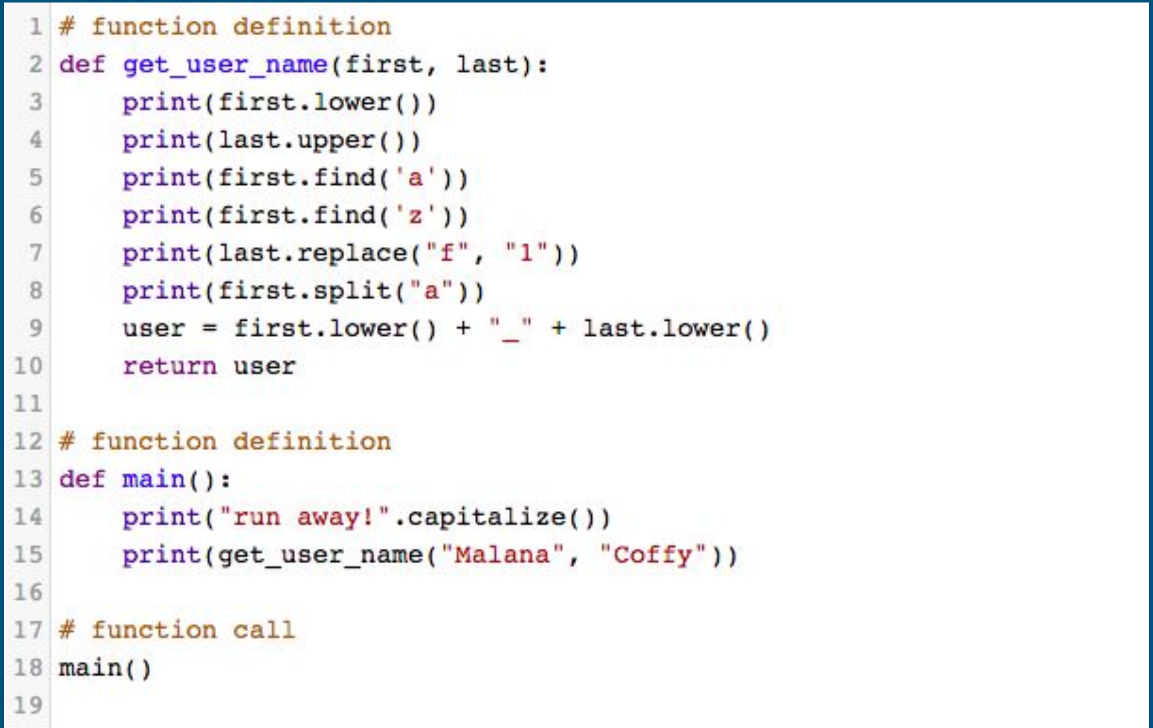
Use this code to answer 5.21.2.1



```python
# function definition
def get_short_name(first, last):
    print(len(first))
    print(len(last))
    return first[:2] + last[-2:]

# function definition
def main():
    print(get_short_name('Simona',"Jacobs"))

# function call
main()

```

# Exercise 5.21

Use this code to answer 5.21.4.1



```python
# function definition
def get_user_name(first, last):
    print(first.lower())
    print(last.upper())
    print(first.find('a'))
    print(first.find('z'))
    print(last.replace("f", "1"))
    print(first.split("a"))
    user = first.lower() + "_" + last.lower()
    return user

# function definition
def main():
    print("run away!".capitalize())
    print(get_user_name("Malana", "Coffy"))

# function call
main()

```

# Homework

Tic-Tac-Toe

Please reach out if you ever have issues with submitting an assignment on time.

# Sources

https://holypython.com/intermediate-python-lessons/lesson-11-python-lambda/