

Introduction to Programming with Python 1/?

DUCSS - Dublin University Computer Science Society

Volunteers are here to help

Ask Questions

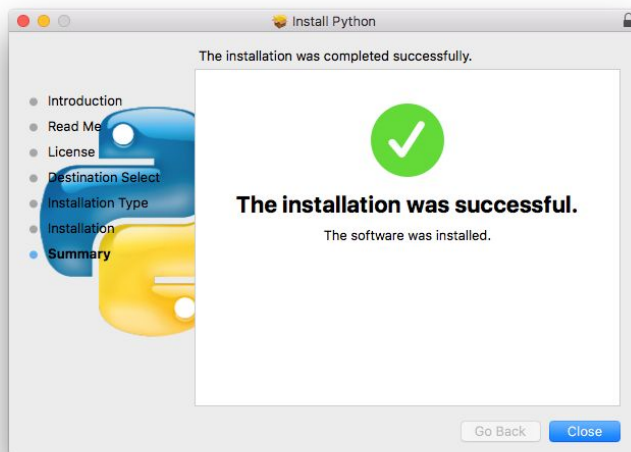
Experiment with the Code

Plan for Today

- Installing Python
- Python Tool
- “Hello World!”
- Variables
- Branching
- Loops
- Functions
- Exercises

Installing Python

- Go to python.org/downloads
- Download Version 3.x.x
- Follow the Installation Instructions



Checking your Installation

- Open Terminal (Linux/macOS) or the Command Line (Windows)
- Run “py” (Windows) or “python” (macOS, Linux)

Expected output:

```
Python 3.6.2 (v3.6.2:5fd33b5926, Jul 16 2017, 20:11:06)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "help", "copyright", "credits" or "license" for more
information.
```

```
>>>
```

- If version is not 3.x.x
 - Enter “exit()”
 - Run “python3”

Python Tool

- Entering manual commands:

```
python
```

```
>>> #your python commands here
```

```
>>> exit()
```

- Running Python scripts (.py files):

```
python path/to/your/python/script.py
```

py not python
on Windows!

Sample Code

github.com/arneph/python-workshops

“Hello World!”

```
python
```

```
>>> 'Hello World!'
```

```
'Hello World!'
```

py not python
on Windows!

“Hello World!”

```
>>> 'Hel' + 'lo!'  
'Hello!'
```

“Hello World!”

```
>>> A = 'Hello'  
>>> B = ' '  
>>> C = 'World!'  
>>> A + B + C  
'Hello World!'
```

“Hello World!”

- Open Editor/Notepad/your favorite text editor
- Create a new file
- Enter the following:

```
X = 'Hello World!'
```

```
print(X)
```

- Save the file as “hello.py” on your desktop
- In Terminal/the Command Line:

```
python /Desktop/hello.py
```

```
Hello World!
```

py not python
on Windows!

Variables

- Storing values:

```
firstName = 'Gina'  
lastName = 'Hopper'  
born = 1903
```

- Changing values:

```
firstName = 'Grace'  
born = 1906
```

Variables

- Values can be used in computation and for function calls

```
age = 2017 - born
name = firstName + ' ' + lastName
print(name + ' would now be')
print(age)
print('years old today')
```

Variables

```
firstName = 'Gina'  
lastName = 'Hopper'  
born = 1903
```

```
firstName = 'Grace'  
born = 1906
```

```
age = 2017 - born  
name = firstName + ' ' + lastName  
print(name + ' would now be')  
print(age)  
print('years old today')
```

Variables

```
Grace Hopper would be  
111  
years old today
```



Branching

- if-statements:

```
A = 9
```

```
B = 7
```

```
if A > B:
```

```
    print('A is larger')
```

```
    print('A wins!')
```

```
if B > A:
```

```
    print('B is larger')
```

```
    print('A loses! :-(')
```

```
print('Game over')
```


Branching

A is larger

A wins!

Game over

Branching

- if-statements (continued):

```
A = 42
```

```
B = 42
```

```
if A > B:
```

```
    print('A wins!')
```

```
elif A == B:
```

```
    print('Draw!')
```

```
else:
```

```
    print('B wins!')
```

Branching

Draw!

Branching

- Combining Conditions:

```
username = 'marc z'  
password = 'dadada'  
if username == 'marc z' and password == 'dadada':  
    print('Hello Marc')  
    print('Get a better password!')  
else:  
    print('You are not Marc!')
```

Branching

Hello Marc

Get a better password!



Branching

- Combining Conditions (continued):

```
age = 42
```

```
if age < 21 or age >= 30:
```

```
    print('Not a primary target')
```

```
    if age >= 18 and age < 21:
```

```
        print('Keep watching')
```

```
    else:
```

```
        print('Remove from watchlist')
```

Branching

Not a primary target

Remove from watchlist



Loops

- while-loops:

```
i = 1
while i <= 30:
    print(i ** 2)    #Print i2
    i += 1           #Add 1 to i
```


Loops

1

4

9

16

25

36

49

64

81

100

121

...

Loops

- for-loops:

```
factorial = 1
for x in range(1, 12):
    factorial *= x
    print(factorial)
```

Loops

1

2

6

24

120

720

5040

40320

362880

3628800

39916800

Loops

- continue & break:

```
for x in range(2, 40):  
    prime = True  
    for y in range(2, x):  
        if x % y == 0:  
            prime = False  
            break                #Break out of inner for-loop  
    if prime == False:  
        continue                #Continue with next item in  
    print(x)                     #outer for-loop
```

Loops

2

3

5

7

11

13

17

19

23

29

31

37

Functions

- Using Functions:

```
print('Hello!')
```

```
A = pow(19, 2)
```

```
B = input()          #Get keyboard input from the user
```

```
for x in range(42, 888):
```

```
    #...
```

Functions

- Defining Functions:

```
def isOdd(x):  
    return x % 2 == 1
```

```
print(isOdd(27))  
print(isOdd(2048))  
print(isOdd(22))
```

Functions

True

False

False

Functions

- Defining Functions (continued):

```
def fizzbuzz(x):  
    a = (x % 3 == 0)  
    b = (x % 5 == 0)  
    if a and b:  
        print('Fizz Buzz')  
    elif a:  
        print('Fizz')  
    elif b:  
        print('Buzz')  
    else:  
        print(x)
```

Functions

- Defining Functions (continued):

```
# . . .
```

```
for x in range(1, 50):  
    fizzbuzz(x)
```

Functions

```
1  
2  
Fizz  
4  
Buzz  
Fizz  
7  
8  
Fizz  
Buzz  
11  
...
```

Exercises

- Prompt the user for a name and password and check if they are correct
Hint: use the `input()` function
- Rewrite the prime numbers example with a function
Hint: define a function `isPrime(x)`
- Print all Fibonacci numbers up to 10946

Exercises

Solutions online:

github.com/arneph/python-workshops