
Tip for the start

1. Ask your students what they remember from yesterday (What is a code editor, a commit? etc).
 2. Do every step together with the students.
 3. Your students will need more individual help in the afternoon than in the morning. Have 1 coach for 4-6 students in the morning, and 1 coach for 2-3 students in the afternoon.
-

Programming in Python for absolute Beginners

Slides by Marie Dedikova & Ines Guett 2017

Plan for today

1. Intro
2. Type together: First Steps in Python

Lunch!

3. Write your own game
 4. Type together: Functions and Lists
 5. Improve your game
 6. What's next
-

Do not copy - paste!!!

Writing CODE is understanding!

It is ok not to understand!

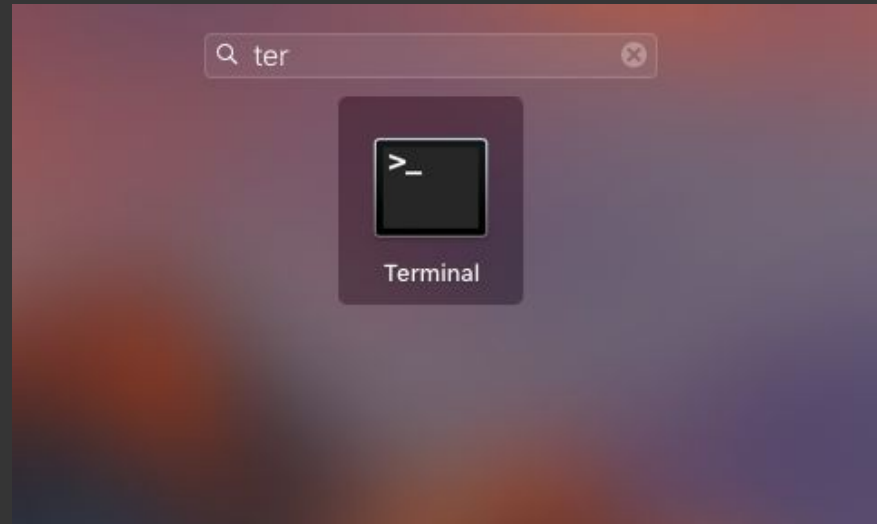
1. You will understand later
2. Most programmers do “Trial & Error”

Let's write some code!

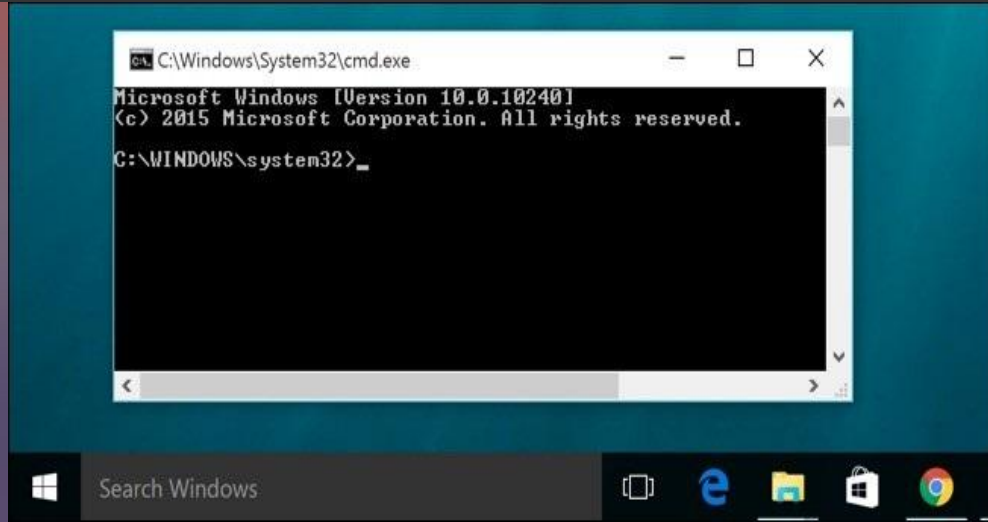
Type together

Go to your Terminal

Mac



Win



Type **python3** on Mac or
python on Windows

```
Last login: Mon Oct  2 13:13:18 on ttys000
Admins-MacBook-Pro:~ Admin$ python3
Python 3.4.4 (v3.4.4:737efcadf5a6, Dec 19 2015, 20:38:52)
[GCC 4.2.1 (Apple Inc. build 5666) (dot 3)] on darwin
Type "help", "copyright", "credits" or "license" for more
>>>
```

Printing & Strings

```
print("Hello World")
```

```
name = "Marie"
```

```
print("Hello %s" % (name))
```

Variables

```
my_variable = "coffee"  
print(my_variable)
```

```
my_variable * 4  
my_variable.upper()
```

Simple Math

```
dogs = 4
```

```
cats = 3
```

```
snakes = 2
```

```
total = dogs + cats + snakes
```

```
print(total)
```

```
# But the weather was weird and 3x more snakes were born.
```

```
snakes = 3 * snakes
```

```
dogs = dogs - 3
```

Compare Things

4 > 2

5 < 1

1 == 1

5 != 2

6 >= 12 / 2

5 <= 6

6 > 2 and 2 < 3

6 > 2 and 2 > 3

6 > 2 or 2 > 3

Boolean - True or False

a = True

a

b = 2 > 5

b

True and True

False and True

True or 1 != 1

Errors are friends! :)

```
len(23493)
```

```
city = "Tokyo"  
print(ctyi)
```

```
print(flower)  
3 > "Python"
```

IF Statements

```
people = 20  
cats = 5
```

```
# Try this:
```

```
if people > cats:
```

```
    print("The world is boring!")
```

```
# Do you get an error? Why?
```

ELSE Condition

```
cats += 20
```

```
if people > cats:  
    print("The world is boring!")  
else:  
    print("Ohh, way too many cats!")
```


ELIF Condition

```
volume = 38
```

```
if volume < 20:  
    print("It's kinda quiet.")  
elif 20 <= volume < 40:  
    print("It's nice for background music")  
else:  
    print("My ears are hurting! :(")
```

**Let's use a
code editor!**

Ways of Writing </>

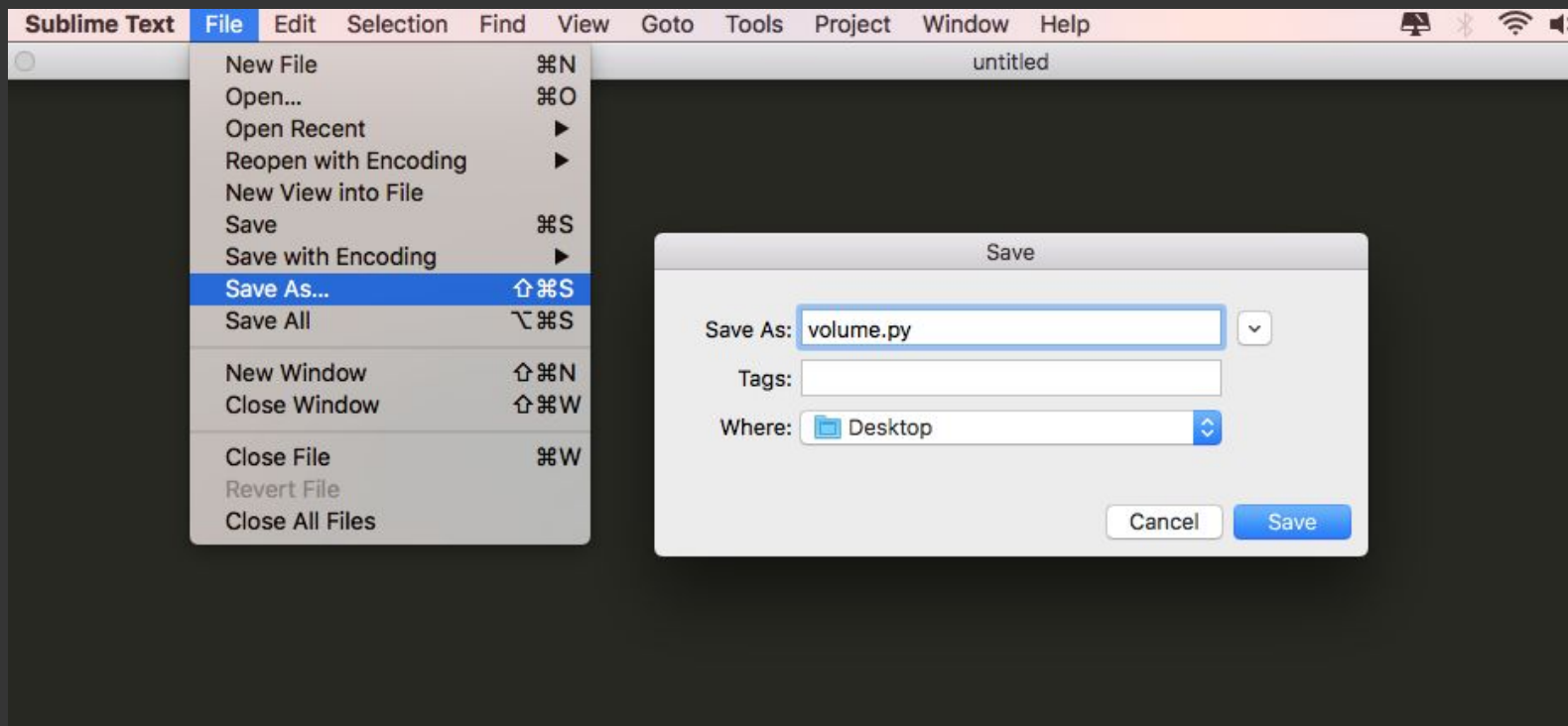
IN TERMINAL:

```
((test-env) MacBook-Air:test mariededikova$ python
Python 3.6.1 (default, Jul 4 2017, 14:32:26)
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.42.1)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
[>>> print("Hello World!")
Hello World!
>>> █
```

IN CODE EDITOR:

```
1 from django.db import models
2
3
4 class Role(models.Model):
5     name = models.CharField(max_length=30)
6
7     def __str__(self):
8         return self.name
9
10
11 class Address(models.Model):
12     street = models.CharField(max_length=40)
13     house_number = models.CharField(max_length=40)
14     post_code = models.IntegerField()
15     city = models.CharField(max_length=30)
16     country = models.CharField(max_length=20)
17
18
19 class Person(models.Model):
20     first_name = models.CharField(max_length=30)
21     last_name = models.CharField(max_length=30)
22     roles = models.ManyToManyField(Role, related_name="people")
23     address = models.ForeignKey(Address, null=True)
24
25     def __str__(self):
26         return '{} {}'.format(self.first_name, self.last_name)
27
```

Open Sublime



Write This + Save:

```
print("The world is beautiful! Enjoy the sound!")
```

Switch Like a Pro!

Mac



Win



Change Directory

Mac

Change directory: OS X

On a Mac, the command will look something like this:

command-line

```
$ cd ~/Desktop
```

Win

Change directory: Windows Powershell

And on Windows Powershell, it will be like this:

command-line

```
> cd $Home\Desktop
```

Execute the File

Mac

```
$ python3 volume.py
```

Win

```
> python volume.py
```


Write This + Save:

```
print("The world is beautiful! Enjoy the sound!")
```

```
input("What is your favorite song? ")
```

```
song = input("What is your favorite song? ")  
name = input("What is your name?")
```

```
print("Your name is %s and your favorite song is %s." % (name, song))
```

Build your own text adventure.

With Python and the Shell

Instruction for task 1 and task 2

1. Get the python file task_1 from the repository you found this slides in and run it in terminal.
 2. Open the file in a code editor and ask your students if they can explain what happens in line 1, line 2, etc.
 3. Give your students ca. 30 min to create their own task_1 and another 15 min for task_2.
-

—

Task 1

Type the text adventure
and add the door “2”.

Lessons learned:
Conditional Statements
& Comparator 'equal to'

—

Task 2

Ask the player
for her name and
use it in the game.

Lessons learned:
String replacement
with %s and %

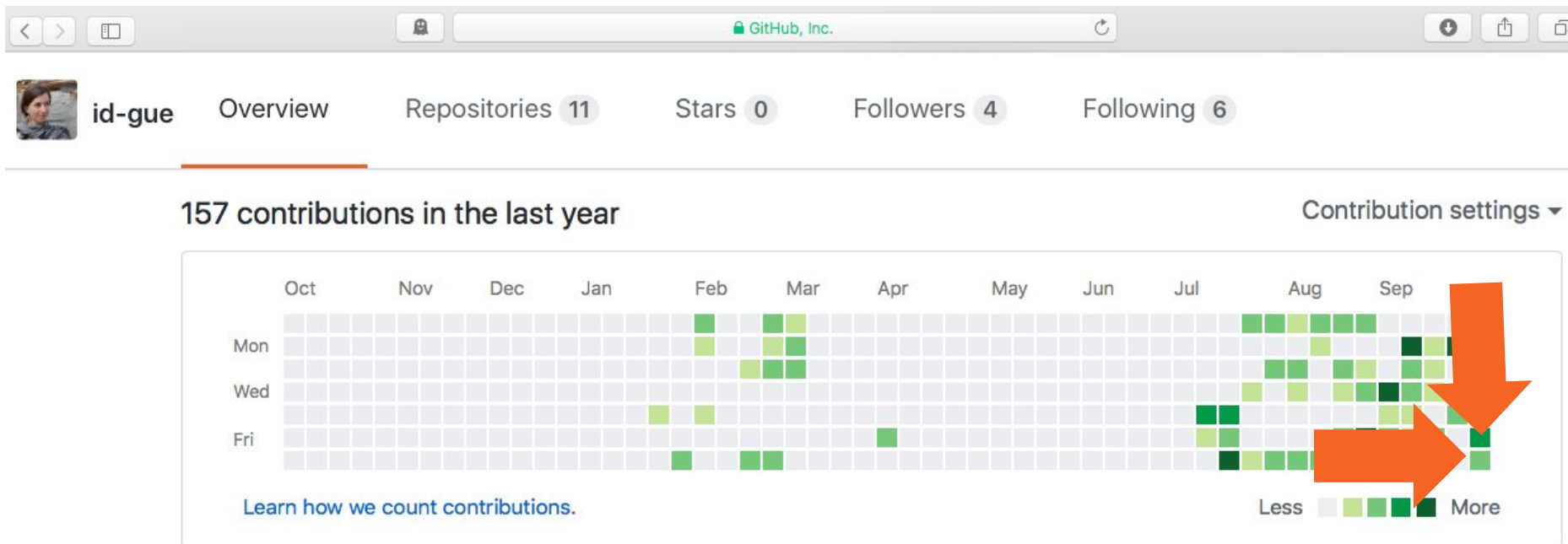
Push your files to Github

In Github Desktop:

1. Commit
2. Push

Then check if you find your files on github.com

Green fields on Github Profile!!! Yeay!!



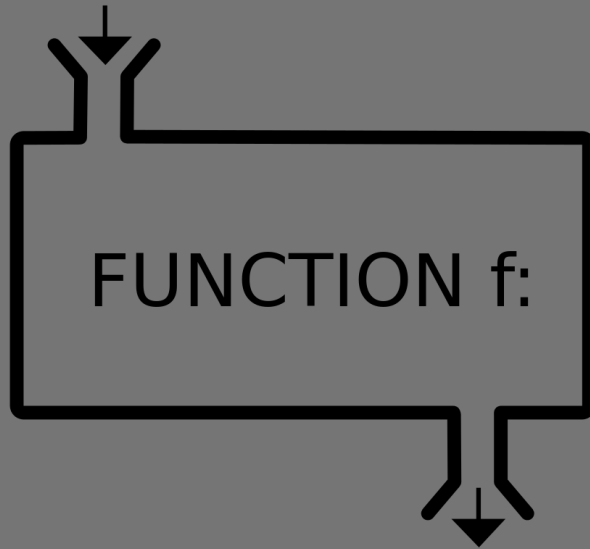
Functions

```
def sound(volume):  
    if volume < 20:  
        print("It's kinda quiet.")  
    elif volume < 40:  
        print("It's nice for background music")  
    else:  
        print("My ears are hurting! :(")
```

```
sound(42)
```

Create whole world!

INPUT x



OUTPUT $f(x)$

Lists & Loops

```
corb = ["apple", "strawberry", "kiwi", "grape"]
```

```
for fruit in corb:  
    print(fruit)
```

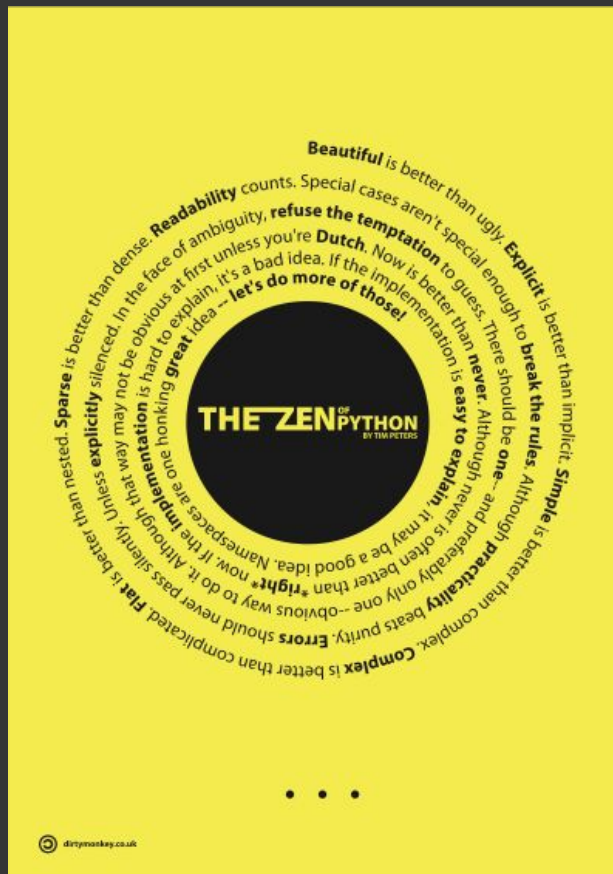
**Algorithmic
thinking!!!**

Well done!



Libraries

import this



Build your own text adventure.

Part 2

—

Task 3:

Create a function
to avoid to repeating
yourself

Lessons learned:
Create functions
and use them

Push your files to Github

In Github Desktop:

1. Commit
2. Push

Then check if you find your files on github.com

Yeay, you did it!!!

What's next?

Learning materials

Learn

PYTHON SYNTAX

Variables

Creating web apps, games, and search engines all involve storing and working with different types of data. They do so using **variables**. A **variable** stores a piece of data, and gives it a specific name.

For example:

```
spam = 5
```

The variable `spam` now stores the number `5`.

☒ Instructions

1. Set the variable `my_variable` equal to the value `10`.

Click the Run button to run your code.

? Stuck? Get a hint

? Report a Bug

script.py

```
1 # Write your code below!  
2 my_variable = 10
```

Run





repl.it is a cloud coding environment for PHP

join a community of

engineers, teachers, and students



Search for a language, e.g. c++

Rust

QBasic

Python

CoffeeScript

[browse all languages](#)

Tutorial

django girls

[Introduction](#)

Installation

Installation (chromebook)

How the Internet works

Introduction to command line

Python installation

Code editor

Introduction to Python

What is Django?

Django installation

Be a Part of Community!

That makes learning much more fun!

OpenTechSchool Berlin



django
girls

Rails Girls 
Summer of Code

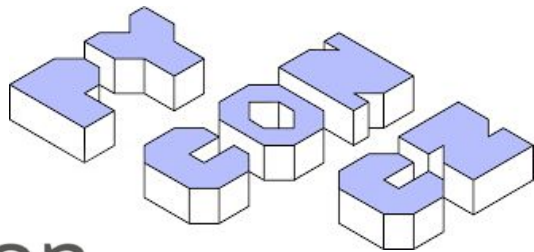
WOMEN WHO
CODE

Financial Aid as Woman

You get paid for travelling to conferences: LGBT, women ...

pyladies

{codemotion}



**DJANGO
CON
EUROPE
2017**



europython
9-16 JULY 2017 Rimini

Happy coding!!!

Thanks to: PyLadies, SoundCloud
and all of our great coaches!!!
