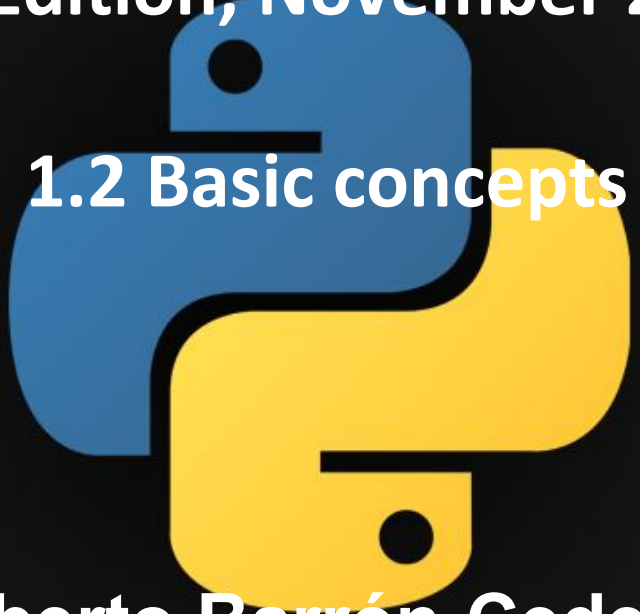
DIT gentle introduction to Python

1st Edition, November 2020



1.2 Basic concepts

Alberto Barrón-Cedeño

a.barron@unibo.it

@_albarron_

Overview

- Basics
- Algorithms
- Programming languages
- Baby steps into coding

What is a programming language?

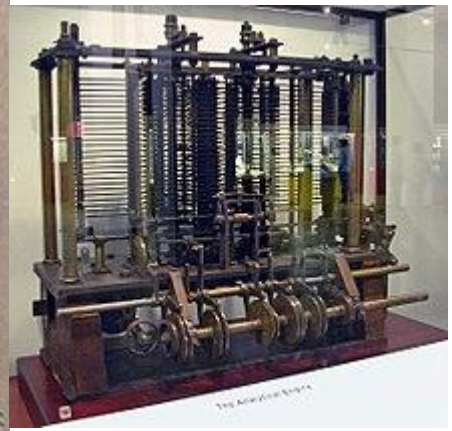
A programming language is just another language

A formal language comprising a set of instructions that produce various kinds of output [given an input]

Programming languages are used in computer programming to implement algorithm

https://en.wikipedia.org/wiki/Programming_language

The *first* programmer



Ada Lovelace* (Mathematician)

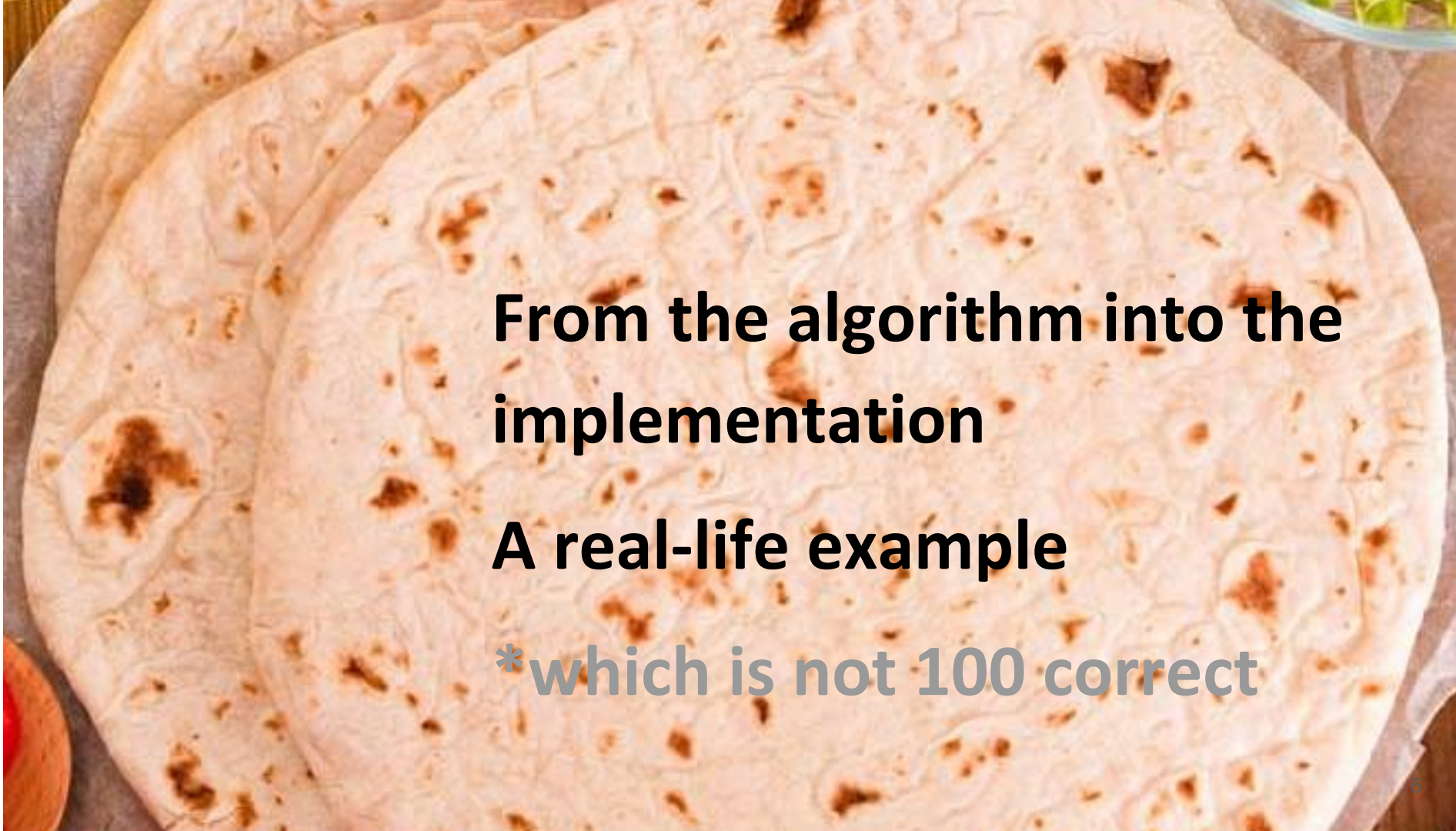
She published the first algorithm for Charles Babbage's analytical engine

*she is Lord Byron's daughter

Algorithm

A finite sequence of well-defined, computer-implementable instructions, typically to solve a class of problems or to perform a computation

https://en.wikipedia.org/wiki/Algorithm



From the algorithm into the implementation

A real-life example

*which is not 100 correct

The piadina maker: the ingredients

- 500g di farina 00
- 100g di latte
- 100g acqua
- 75g di strutto
- 2g di bicarbonato
- 15g di lievito istantaneo Pizzaiolo
- 10g di sale

Recipe from http://www.algoritmicucina.it/piadina-romagnola/

The piadina maker: the recipe

1. Disporre la farina con sale, bicarbonato e lievito a fontana
2. Al centro aggiungere lo strutto
3. Mescolare il latte con l'acqua e versarlo sulla farina
4. Impastare energicamente fino ad ottenere una pasta compatta
5. Lasciar riposare per almeno 30 minuti
6. Dividere l'impasto in quattro palline, e stenderle formando dischi di circa 25 cm di diametro
7. Cuocere da entrambi i lati per 1m bucherellando con una forchetta
8. Farcire a piacere

The piadina maker: into computing

- 500g di farina 00
- 100g di latte
- 100g acqua
- 75g di strutto
- 2g di bicarbonato
- 15g di lievito istantaneo Pizzaiolo
- 10g di sale

1. Disporre la farina con sale, bicarbonato e lievito a fontana
2. Al centro aggiungere lo strutto
3. Mescolare il latte con l'acqua e versarlo sulla farina

...

Input/Output

→ ingredients (e.g., data)

← piadina (e.g., more data)

Process

A series of instructions and routines

The piadina maker: into code

1. Disporre la farina con sale, bicarbonato e lievito a fontana

```
def grabber(farina,  sale, bicarbonato, lievito):
    fontana = Fontana()
    fontana.add(farina)
    fontana.add(sale)
    fontana.add(bicarbonato)
    fontana.add(lievito)
    return fontana
```

The piadina maker: into code

2. Al centro aggiungere lo strutto

```
def strutter(fontana, strutto):
    fontana.make_hole(position=”center”)
    fontana.add(strutto)
    return fontana
```

The piadina maker: into code

3. Mescolare il latte con l'acqua e versarlo sulla farina

```
def liquid_mixer(latte, acqua):
    contenitore = Contenitore(capacity = 300)
    contenitore.add(latte)
    contenitore.add(acqua)
    contenitore.mix()
    return contenitore
```

The piadina maker: into code

4. Impastare energicamente fino ad ottenere una pasta compatta

```
import time
fontana = grabber(farina,  sale, bicarbonato, lievito)
fontana = strutter(fontana, strutto)
contenitore =  liquid_mixer(latte, acqua):
pasta = kneader(fontana, contenitore)
```

5. Lasciar riposare per almeno 30 minuti

```
time.sleep(1800)
```

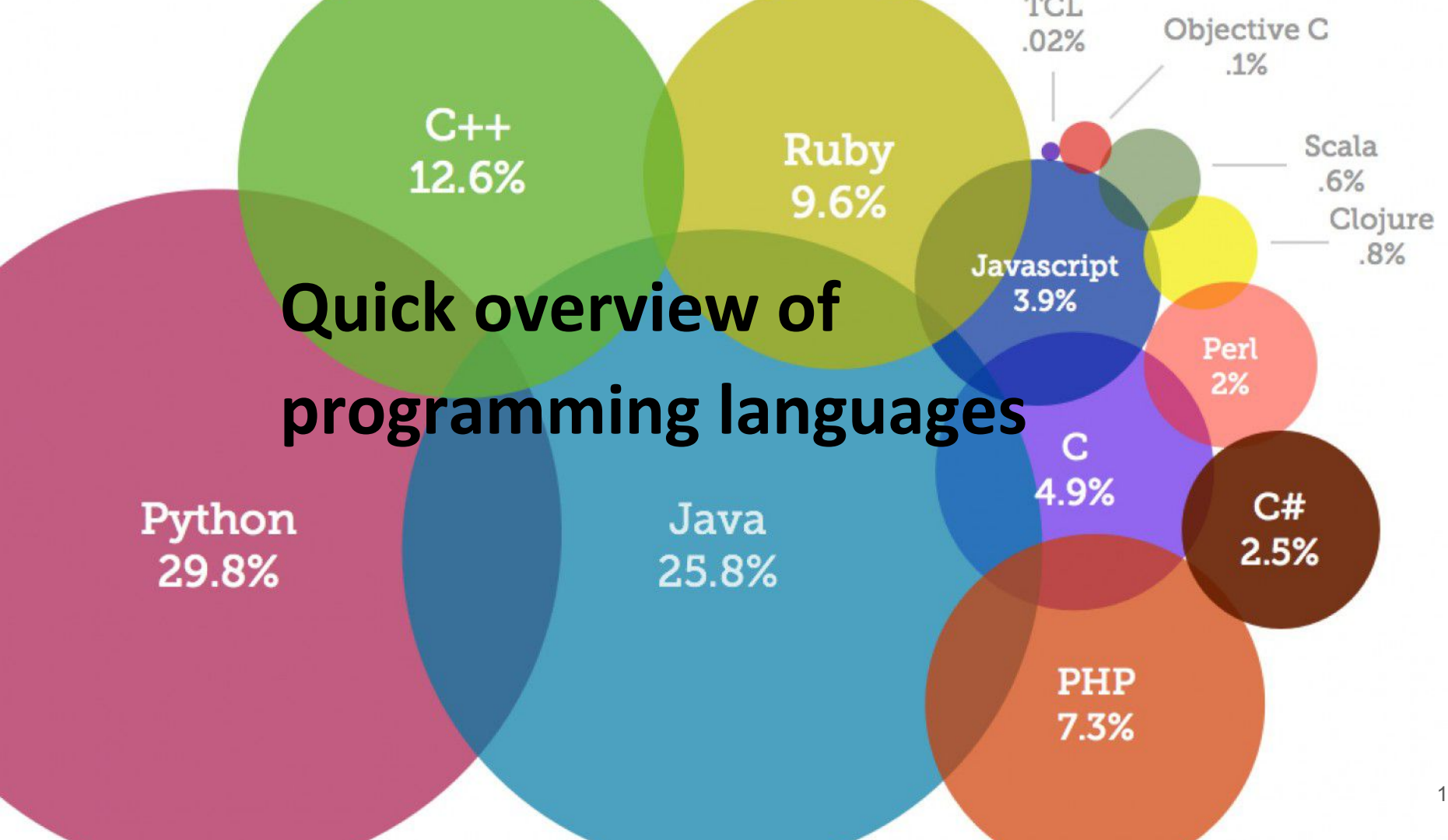
The piadina maker: into code

6. Dividere l'impasto in quattro palline, e stenderle formando dischi di circa 25 cm di diametro

7. Cuocere da entrambi i lati per 1m bucherellando con una forchetta

8. Farcire a piacere

Quick overview of programming languages



History of (some) flagship languages

year	language	highlights
1957	Fortran	compiled, imperative
1959	Lisp	Object-oriented, popular in AI, recursive functions
1964	Basic	Procedural, object-oriented (“goto”)
1970	Pascal	Imperative, procedural, lists, trees
1972	C	Procedural, recursion, static type system
1983	C++	Object-oriented, compiled, functional

History of (some) flagship languages

year	language	highlights
1989	Python	Interpreted, object-oriented, code readability
1995	Java	compiled , object-oriented
1995	Javascript	Just-in-time-compiled, object-oriented, WWW
1995	PHP	Scripting, Web-oriented
2001	Visual Basic .NET	Object-oriented, .NET framework
2009	Go	Compiled, C-like (safer)

Python is (among other things)...

General-purpose

Applicable across application domains

High-level

Strong abstraction from the computer (hardware)

Interpreted

No previous compilation into machine-level instructions necessary

(Not-necessarily) object-oriented paradigm

An object contains data (attributes) and procedures (methods)

Some notable features (1/2)

- Elegant syntax (indentation-based) → easy to read
- Simple and ideal for prototyping
- It has a large standard library for diverse tasks (e.g., web servers, text search, file reading/modifying)
- Interactive mode → continuous snippet testing

https://wiki.python.org/moin/BeginnersGuide/Overview

Some notable features (2/2)

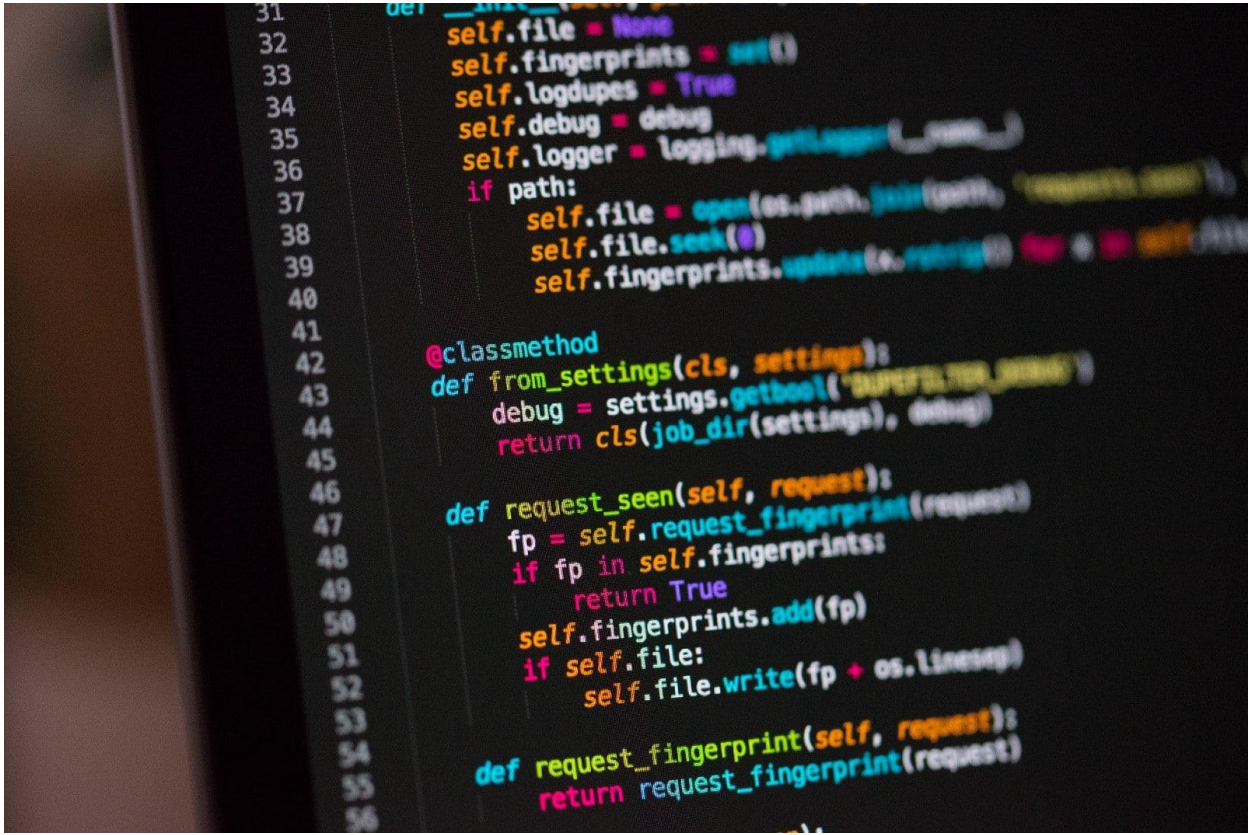
- Extendable with modules in compiled languages (e.g., C++)
- Multi-platform (e.g., Mac OS X, MS Windows, Linux, Unix)
- Free: zero-cost to download/use; open-source license
- It has a large and friendly community

https://wiki.python.org/moin/BeginnersGuide/Overview

Some programming-language features

- A variety of basic data types are available:
  - numbers (floating point, complex, integers) ← later today
  - strings (both ASCII and Unicode)
  - Lists
  - Dictionaries
- It supports object-oriented programming ← 3rd session
- Code can be grouped into modules and packages

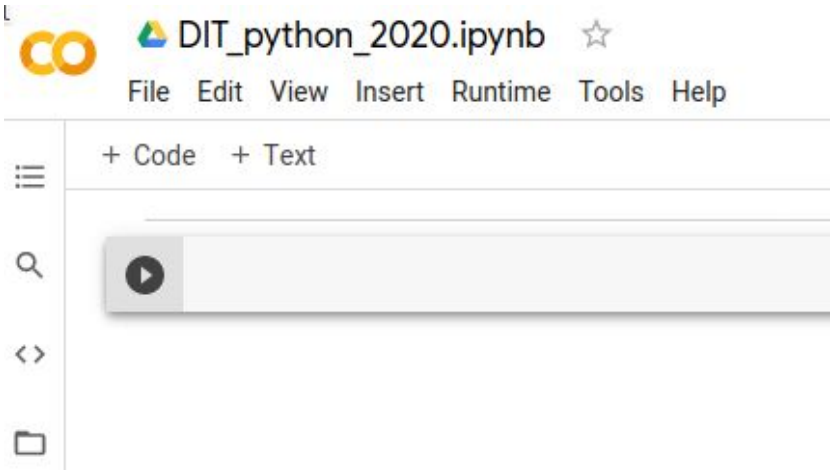
Enough! Let us look at some code!



Google’s colab

“a free Jupyter notebook environment that runs in the cloud and stores its notebooks on Google Drive”

https://colab.research.google.com



Let’s go to our first jupyter notebook

What we know so far: input/output

- print() displays stuff to the screen
- input() captures information from the user

What we know so far: variables

x = 5	- x is a variable - We assign values to a variable with '='	
	- x = 5 - x = 5.5 - x = 'ciao' - x = "ciao" - x = '5'	is an integer is a float is a string is also a string is what?
x = x * 3	- We can apply operators to variables - We can assign the output to a variable	

What we know so far: flow control - conditionals

```
if (condition):
    execute something
elif (condition):
    execute something
else:
    execute something
```

Only one of these three snippets is executed

How is this different?

```
if (condition):
    execute something
if (condition):
    execute something
else:
    execute something
```

What we know so far: flow control - loops

for(iterator):

    execute something

while(condition):

    execute something

The code snippet will be executed during a number of iterations

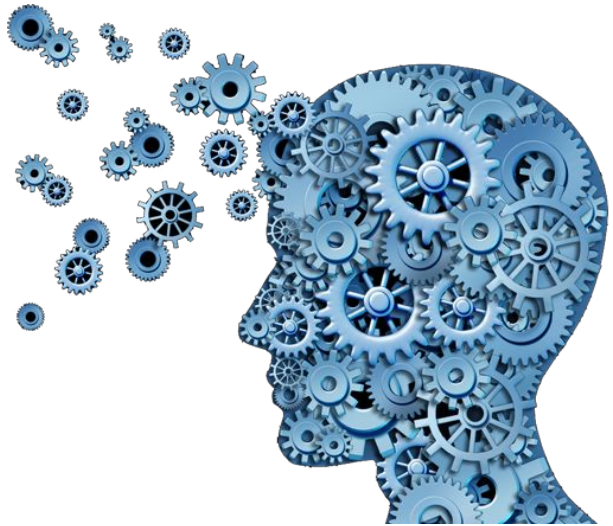
* Danger: if you make a mistake, a loop could run forever

What we know so far: basic formatting

```
# my code
x = 0
while x < 50:
 for i in range(x):
   print('x', end="")
 print()
 x += 1
```

- Comments start with #
- A line break is enough to close an instruction (in Java or C, we need ;)
- Colon opens a *special* code snippet
- Indentation is crucial

You know a lot already!



It is your turn to play with the notebook

https://colab.research.google.com/drive/1-WpX_L4wBaPfbZApkhhtwo7j1yG49JNF?usp=sharing

DIT gentle introduction to Python

1st Edition, November 2020

Alberto Barrón-Cedeño

a.barron@unibo.it

@_albarron