



PyQB

Monga

Files

Types,  
docstrings,  
doctests

# Programming in Python<sup>1</sup>

Mattia Monga

Dip. di Informatica  
Università degli Studi di Milano, Italia

`mattia.monga@unimi.it`

Academic year 2022/23, I semester



PyQB

Monga

Files

Types,  
docstrings,  
doctests

## Lecture VII: Files



# Files

A **file** is an abstraction the operating system uses to preserve data among the execution of programs. Data must be accessed **sequentially**. (Italian reading people might enjoy **this**)

- We need commands to ask to the OS to give access to a file (**open**).
- It is easy to read or write data **sequentially**, otherwise you need special commands (**seek**) to move the file “cursor”
- The number of open files is limited ( $\approx$  thousands), thus it is better to **close** files when they are not in use

Files contain bits (normally considered by group of bytes, 8 bits), the interpretation (“format”) is given by the programs which manipulate them. However, “lines of printable characters” (**plain text**) is a rather universal/predefined interpretation, normally the easiest to program.

PyQB

Monga

Files

Types,  
docstrings,  
doctests



# File read access

```
f = open('filename.txt', 'r') # read only

# iterating on a file reads (all) the lines
for i in f:
    print(i)

# End of file already reached, result is ''
f.readline()

f.close()

# File closed, error!
f.readline()
```

To avoid remembering to close explicitly, Python provides the **context manager** syntax.

```
with open('filename.txt', 'r') as f:
    for i in f:
        print(i)
```

PyQB

Monga

Files

Types,  
docstrings,  
doctests



# Make a program readable

You never write a program only for a machine! You, others, tools will *read* the program for different purposes. Every minute spent in making a program more understandable pays off hours saved later.

- **Type hinting** makes clear what a function needs to work properly, and what it produces
- **Documentation** helps understanding without the need to read implementation details
- **Examples of use** make easy to remember how to use a function and can be used for verification

PyQB

Monga

Files

Types,  
docstrings,  
doctests



# Example

```
from typing import Union

Num = Union[int, float]

def cube(x: Num) -> Num:
    """Return the cube of x.

    >>> cube(-3)
    -27

    >>> abs(cube(0.2) - 0.008) < 10e-5
    True
    """
    return x * x * x
```

Examples can be tested by:

```
python -m doctest filename.py.
```

PyQB

Monga

Files

Types,  
docstrings,  
doctests