Introduction to Bioinformatics using Python

Lecture 2: Introduction to Python

Dr. Alexey Larionov

28 October 2024

www.cranfield.ac.uk

Lecture plan

- Python features
- Installing Python(s) on your PC
- Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- Comments & Indentation
- Python packages and environments
  - PIP, Conda, Venv

What is Python?

Python is an interpreted, high-level, general-purpose programming language

- Developed in the Netherlands, early 1990s by Guido van Rossum
- Official website: http://www.python.org

"Python is an experiment in how much freedom programmers need… … ..."
- Guido van Rossum

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

What is Python?

Python is an interpreted, high-level, general-purpose programming language

- Developed in the Netherlands, early 1990s by Guido van Rossum
- Official website: http://www.python.org

Named after a British comedy team

Monty Python !

“Python is an experiment in how much freedom programmers need… … …”

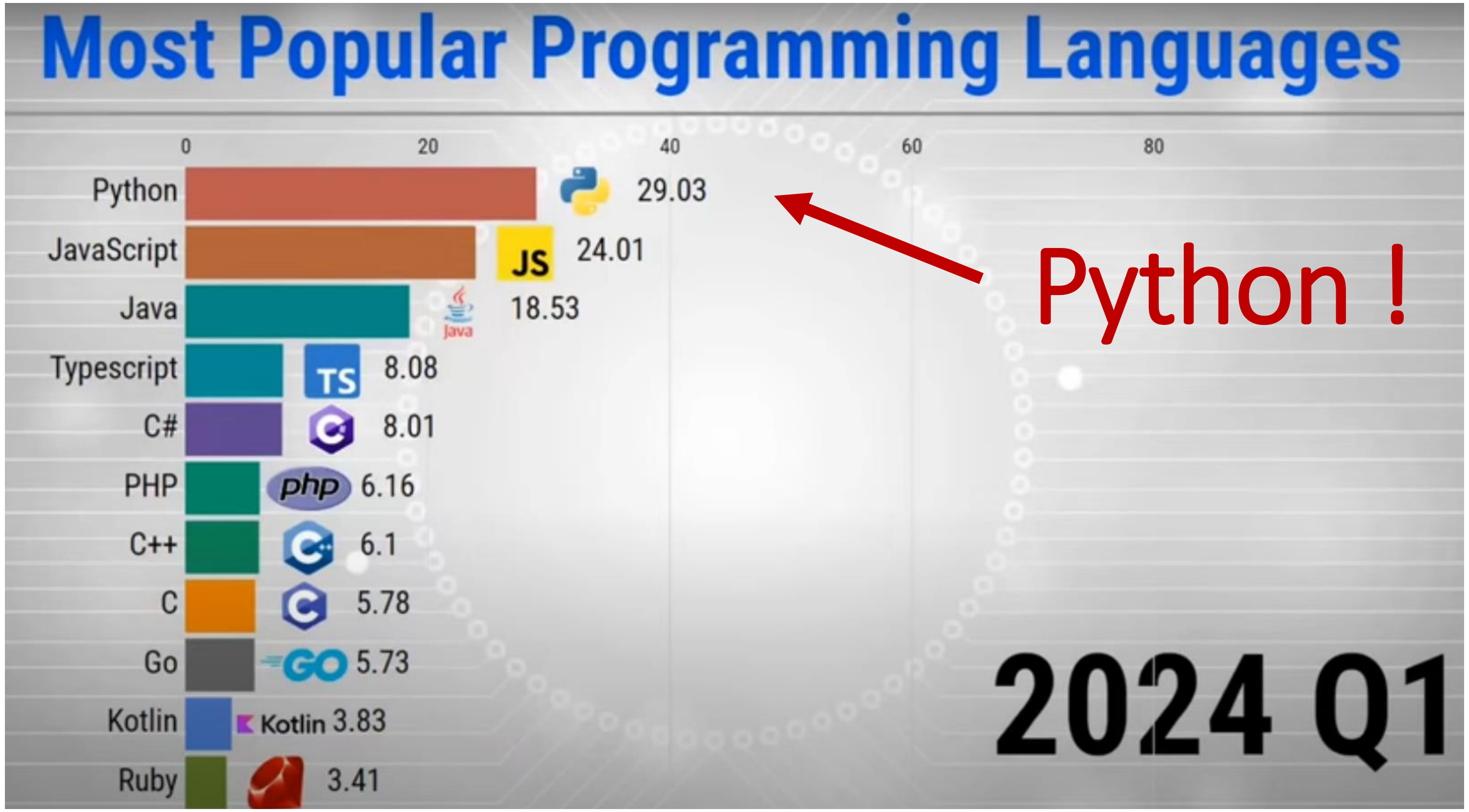
- Guido van Rossum

https://spotify.link/ADz0wBup6Db

3 hrs podcast with GVR in 2022 about Python and future of programming

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

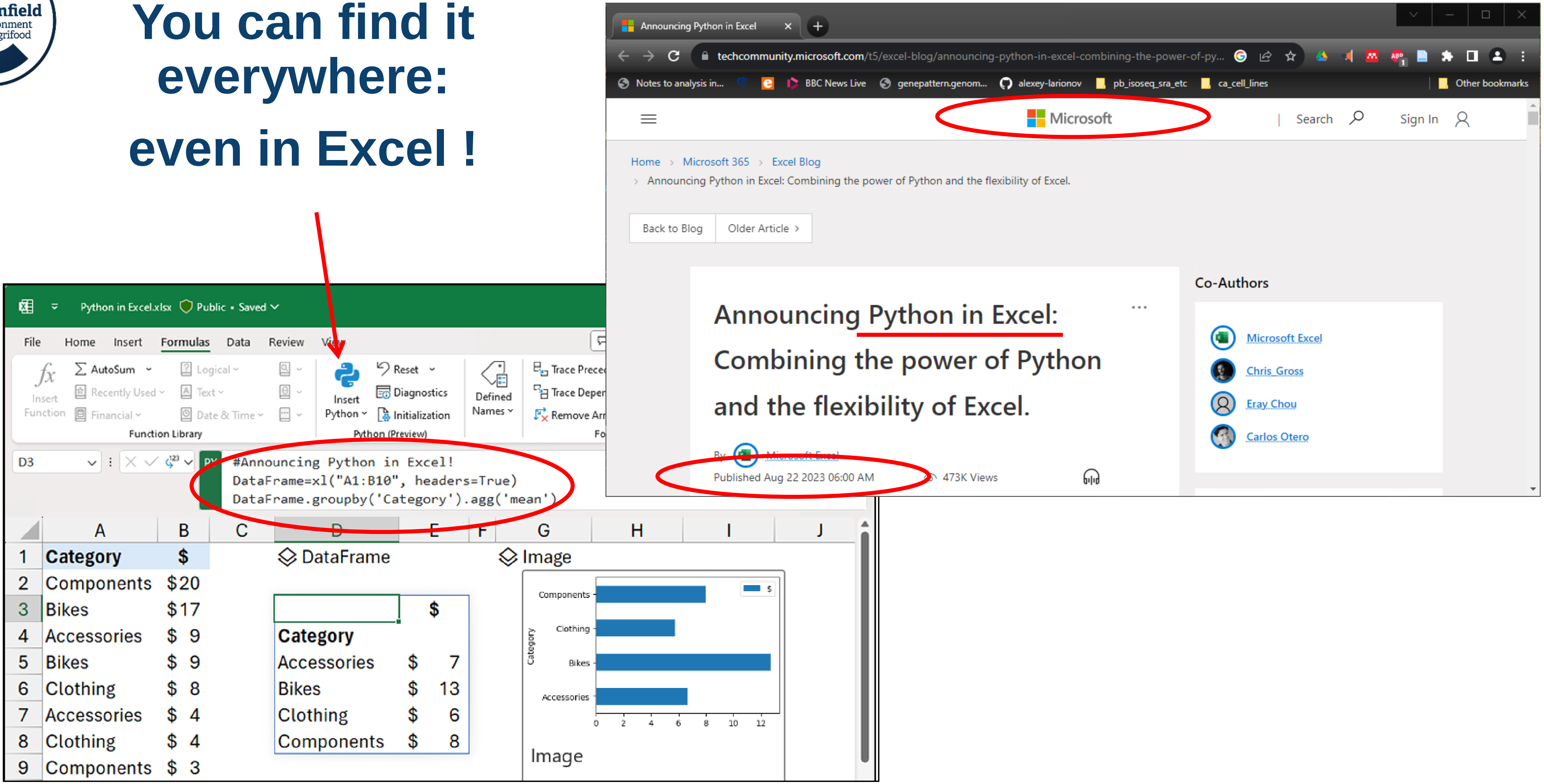
The Most Popular Programming Language in 2024



Python !

See ranks from 1965 till 2024 : https://www.youtube.com/watch?v=xOW3Cehg_qg

You can find it everywhere: even in Excel !



Home › Microsoft 365 › Excel Blog › Announcing Python in Excel: Combining the power of Python and the flexibility of Excel.

Announcing Python in Excel: Combining the power of Python and the flexibility of Excel.

By Microsoft Excel

Published Aug 22 2023 06:00 AM

Co-Authors: Microsoft Excel, Chris_Gross, Eray Chou, Carlos Otero

```
#Announcing Python in Excel!
DataFrame=xl("A1:B10", headers=True)
DataFrame.groupby('Category').agg('mean')
```

	A	B
1	Category	$
2	Components	$20
3	Bikes	$17
4	Accessories	$ 9
5	Bikes	$ 9
6	Clothing	$ 8
7	Accessories	$ 4
8	Clothing	$ 4
9	Components	$ 3

DataFrame

	$
Category	
Accessories	$ 7
Bikes	$ 13
Clothing	$ 6
Components	$ 8

Image

https://techcommunity.microsoft.com/t5/excel-blog/announcing-python-in-excel-combining-the-power-of-python-and-the/ba-p/3893439

Why Python?

- **Widely used in *bioinformatics*** : many libraries available, and their number is growing …
- Object-oriented and Functional from the beginning
- Interpreted : you can run the program straight from the source code
- Portable : Linux, Windows, Mac (anywhere where a Python interpreter is installed)
- Extensible : easily import other code
- Embeddable : easily place your code in non-python programs

Why Python?

- Widely used in *bioinformatics* : many libraries available, and their number is growing …
- **Object-oriented** and **Functional** from the beginning
- Interpreted : you can run the program straight from the source code
- Portable : Linux, Windows, Mac (anywhere where a Python interpreter is installed)
- Extensible : easily import other code
- Embeddable : easily place your code in non-python programs

Why Python?

- Widely used in *bioinformatics* : many libraries available, and their number is growing …
- Object-oriented and Functional from the beginning
- **Interpreted** : you can run the program straight from the source code
- Portable : Linux, Windows, Mac (anywhere where a Python interpreter is installed)
- Extensible : easily import other code
- Embeddable : easily place your code in non-python programs

Why Python?

- Widely used in *bioinformatics* : many libraries available, and their number is growing …
- Object-oriented and Functional from the beginning
- Interpreted : you can run the program straight from the source code
- **Portable** : Linux, Windows, Mac (anywhere where a Python interpreter is installed)
- Extensible : easily import other code
- Embeddable : easily place your code in non-python programs

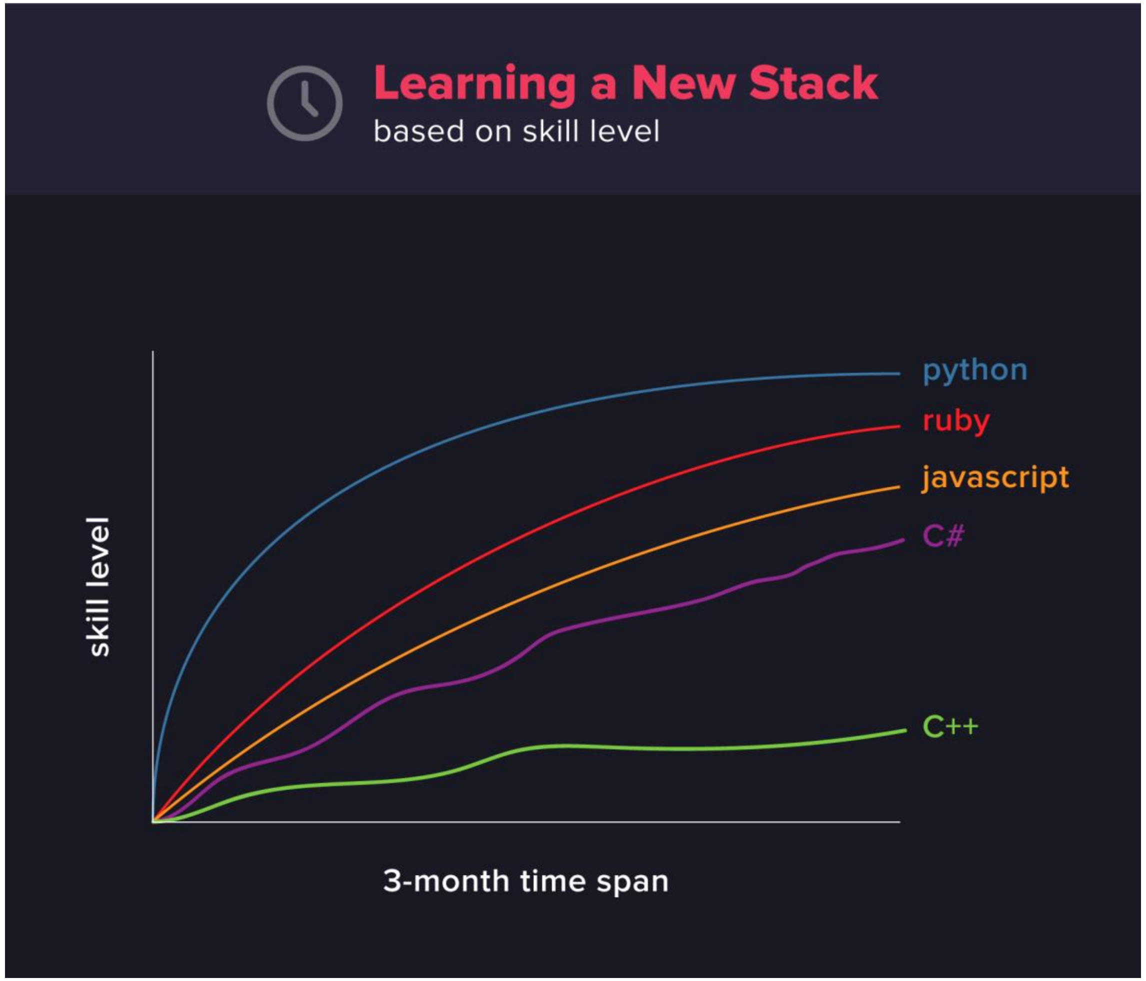
Why Python?

- Widely used in *bioinformatics* : many libraries available, and their number is growing …
- Object-oriented and Functional from the beginning
- Interpreted : you can run the program straight from the source code
- Portable : Linux, Windows, Mac (anywhere where a Python interpreter is installed)
- **Extensible** : easily import other code
- **Embeddable** : easily place your code in non-python programs

Why Python?

Easy to Learn

- Python is simple and minimalistic in nature
- Less syntax memorisation



https://dev.to/programmerthings/language-review-python-episode-one-5hh6

“Stack” in programming means a set of sequential tasks
(e.g. writing a front-end or a back-end of a web site)

Lecture plan

- ✓ Python features
- Installing Python(s) on your PC
- Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- Comments & Indentation
- Python packages and environments
  - PIP, Conda, Venv

Cranfield Environment and Agrifood

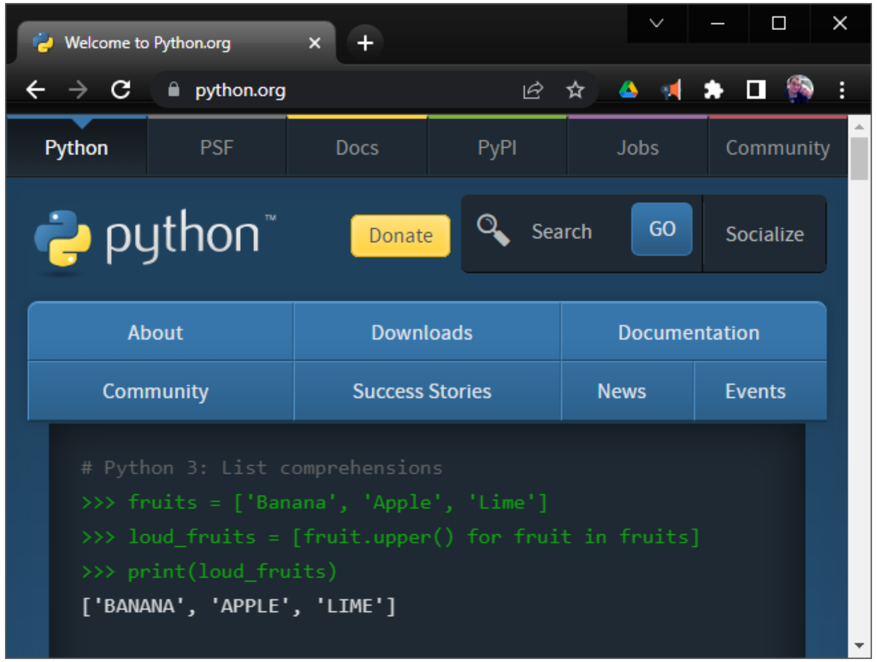
How to install Python to your PC ?

Linux has Python installed

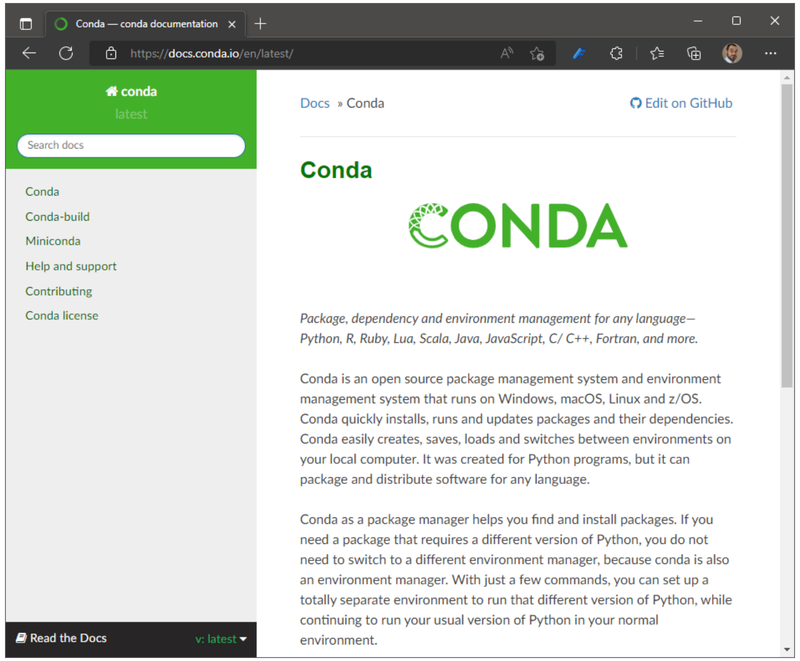
Python Foundation

Conda

Other providers
Windows: MS Store
MacOS: Brew



```
# Python 3: List comprehensions
>>> fruits = ['Banana', 'Apple', 'Lime']
>>> loud_fruits = [fruit.upper() for fruit in fruits]
>>> print(loud_fruits)
['BANANA', 'APPLE', 'LIME']
```

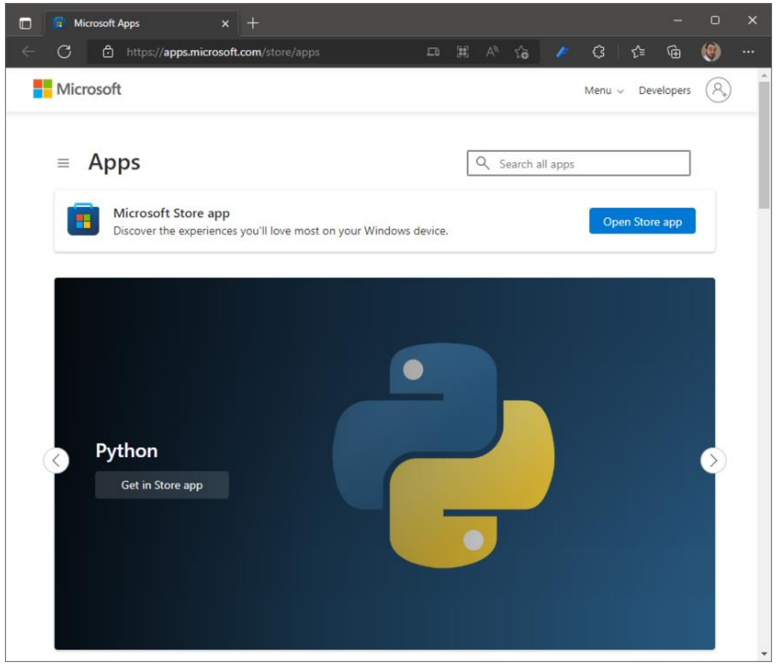


Conda

Package, dependency and environment management for any language—Python, R, Ruby, Lua, Scala, Java, JavaScript, C/ C++, Fortran, and more.

Conda is an open source package management system and environment management system that runs on Windows, macOS, Linux and z/OS. Conda quickly installs, runs and updates packages and their dependencies. Conda easily creates, saves, loads and switches between environments on your local computer. It was created for Python programs, but it can package and distribute software for any language.

Conda as a package manager helps you find and install packages. If you need a package that requires a different version of Python, you do not need to switch to a different environment manager, because conda is also an environment manager. With just a few commands, you can set up a totally separate environment to run that different version of Python, while continuing to run your usual version of Python in your normal environment.



https://www.python.org

https://docs.conda.io

https://apps.microsoft.com/store

Miniconda, Anaconda…

Cranfield Environment and Agrifood

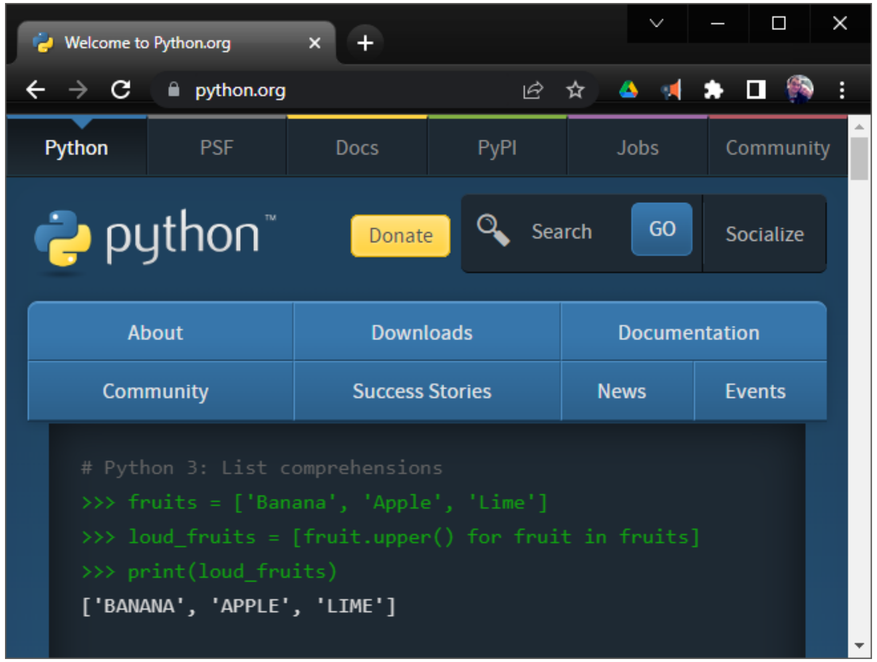
How to install Python to your PC ?

Linux has Python installed

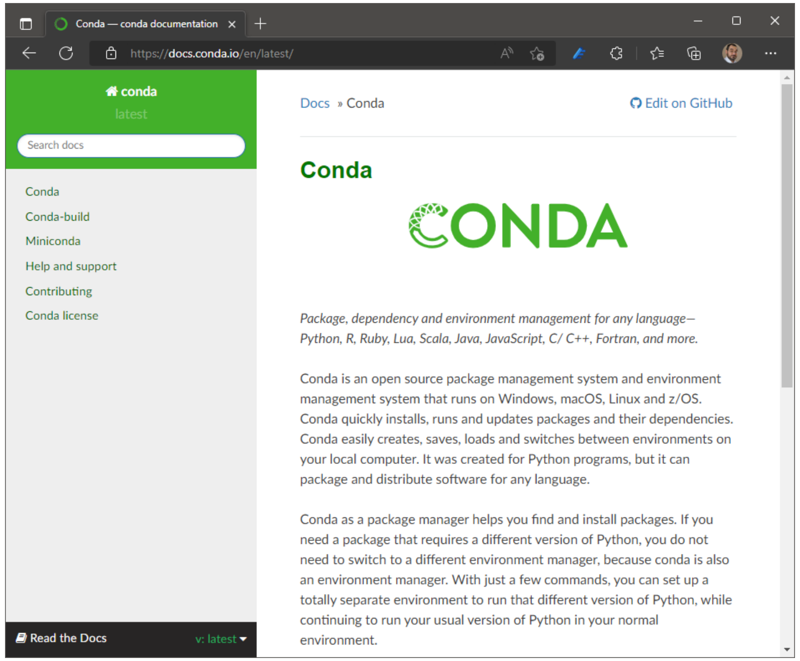
Other providers
Windows: MS Store
MacOS: Brew

Python Foundation

Conda



```
# Python 3: List comprehensions
>>> fruits = ['Banana', 'Apple', 'Lime']
>>> loud_fruits = [fruit.upper() for fruit in fruits]
>>> print(loud_fruits)
['BANANA', 'APPLE', 'LIME']
```

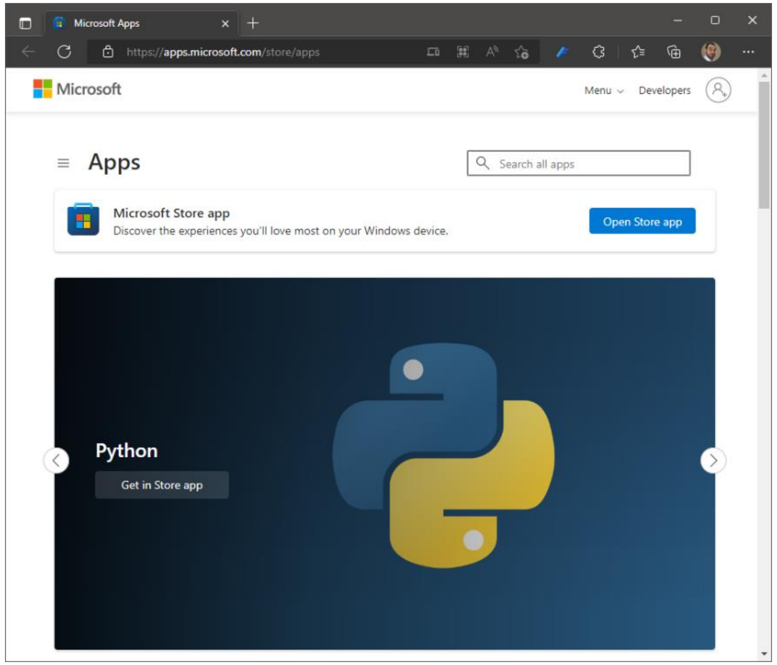


Conda

Package, dependency and environment management for any language—Python, R, Ruby, Lua, Scala, Java, JavaScript, C/ C++, Fortran, and more.

Conda is an open source package management system and environment management system that runs on Windows, macOS, Linux and z/OS. Conda quickly installs, runs and updates packages and their dependencies. Conda easily creates, saves, loads and switches between environments on your local computer. It was created for Python programs, but it can package and distribute software for any language.

Conda as a package manager helps you find and install packages. If you need a package that requires a different version of Python, you do not need to switch to a different environment manager, because conda is also an environment manager. With just a few commands, you can set up a totally separate environment to run that different version of Python, while continuing to run your usual version of Python in your normal environment.



https://www.python.org

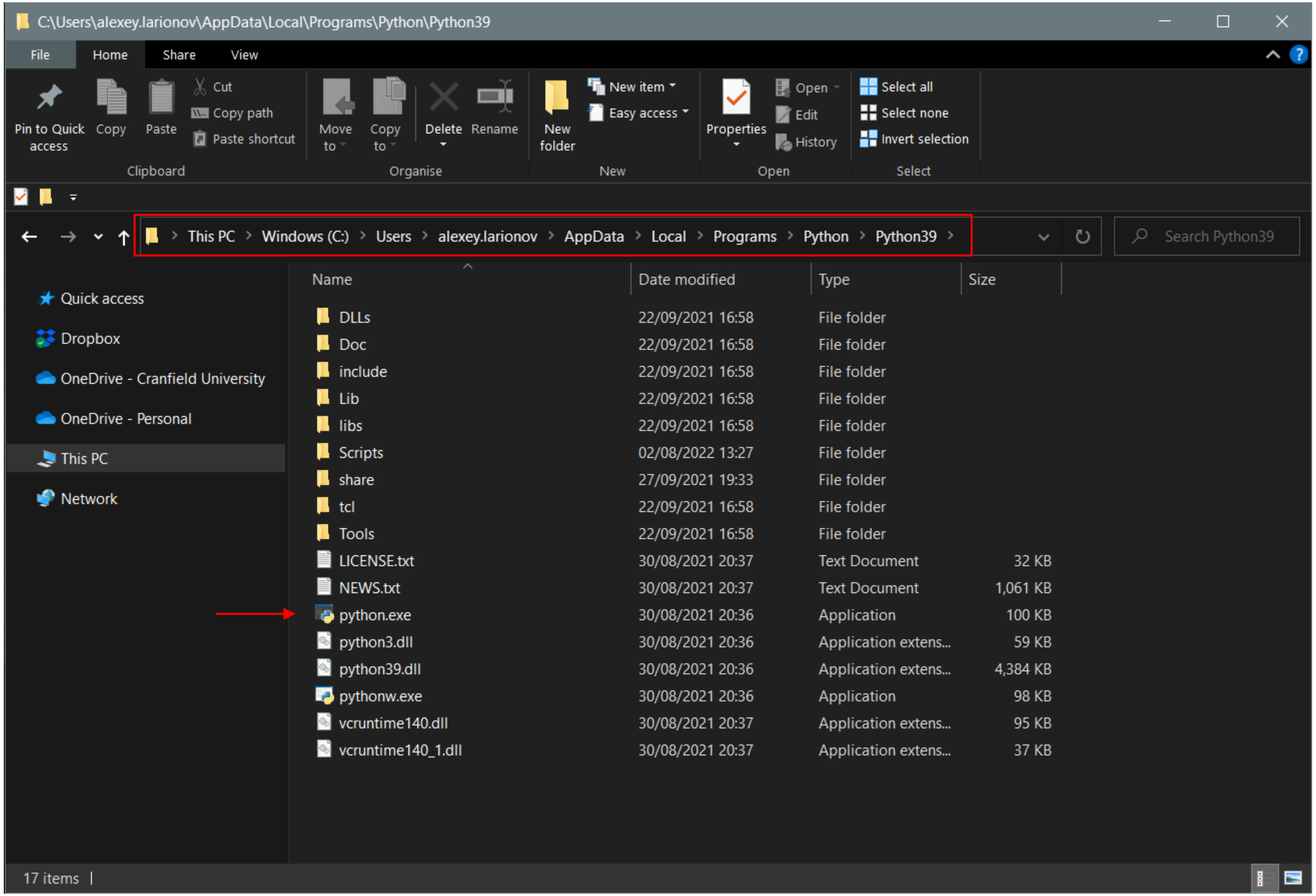
https://docs.conda.io

https://apps.microsoft.com/store

Miniconda, Anaconda…

There could (should!) be many Pythons on your PC

C:/Users/alexey.Larionov/AppData/Local/Programs/Python/Python39/**python.exe**



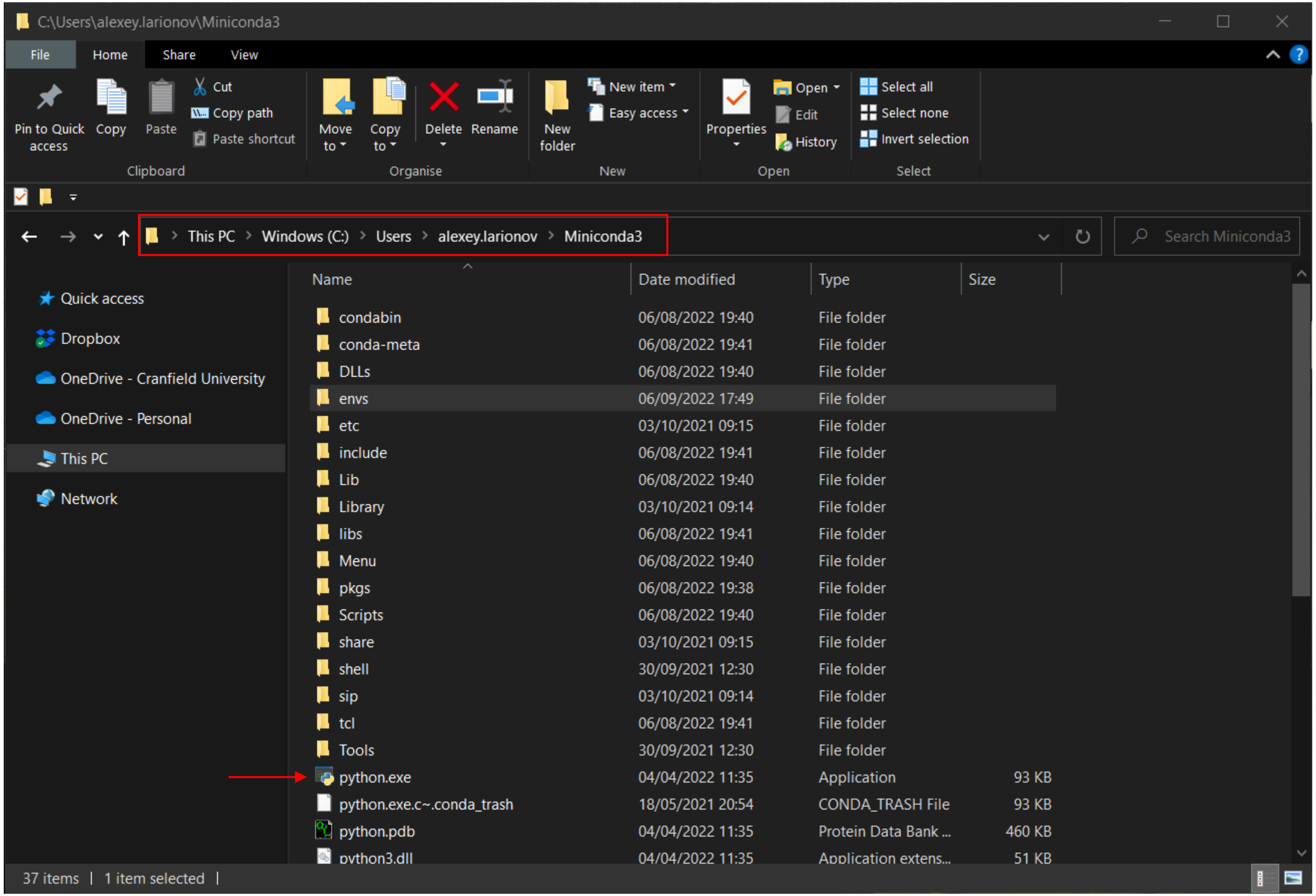
C:\Users\alexey.larionov\AppData\Local\Programs\Python\Python39

Name	Date modified	Type	Size
DLLs	22/09/2021 16:58	File folder	
Doc	22/09/2021 16:58	File folder	
include	22/09/2021 16:58	File folder	
Lib	22/09/2021 16:58	File folder	
libs	22/09/2021 16:58	File folder	
Scripts	02/08/2022 13:27	File folder	
share	27/09/2021 19:33	File folder	
tcl	22/09/2021 16:58	File folder	
Tools	22/09/2021 16:58	File folder	
LICENSE.txt	30/08/2021 20:37	Text Document	32 KB
NEWS.txt	30/08/2021 20:37	Text Document	1,061 KB
python.exe	30/08/2021 20:36	Application	100 KB
python3.dll	30/08/2021 20:36	Application extens...	59 KB
python39.dll	30/08/2021 20:36	Application extens...	4,384 KB
pythonw.exe	30/08/2021 20:36	Application	98 KB
vcruntime140.dll	30/08/2021 20:37	Application extens...	95 KB
vcruntime140_1.dll	30/08/2021 20:37	Application extens...	37 KB

17 items

There could (should!) be many Pythons on your PC

C:/Users/alexey.larionov/Miniconda3/**python.exe**

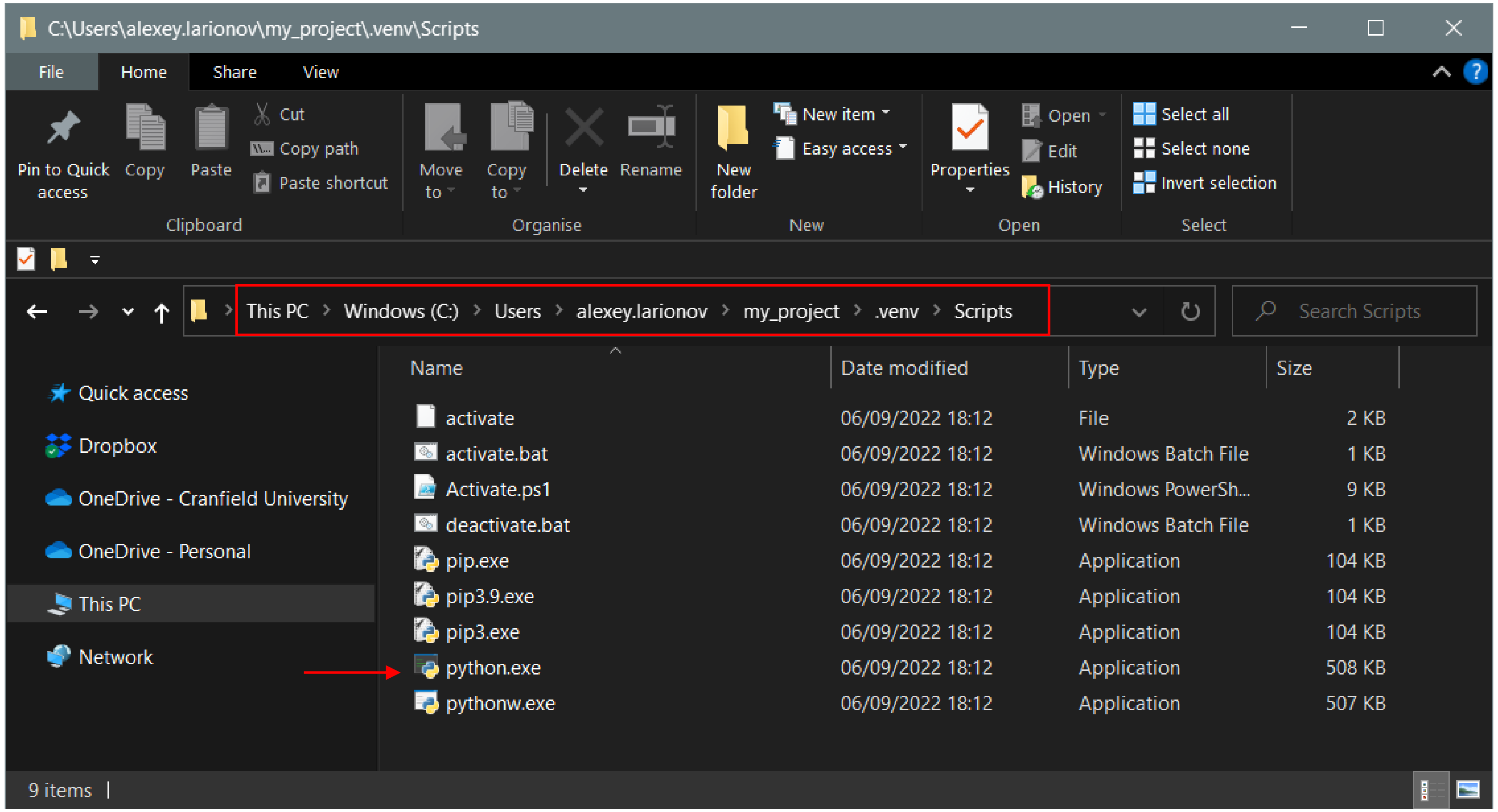


Name	Date modified	Type	Size
condabin	06/08/2022 19:40	File folder	
conda-meta	06/08/2022 19:41	File folder	
DLLs	06/08/2022 19:40	File folder	
envs	06/09/2022 17:49	File folder	
etc	03/10/2021 09:15	File folder	
include	06/08/2022 19:41	File folder	
Lib	06/08/2022 19:40	File folder	
Library	03/10/2021 09:14	File folder	
libs	06/08/2022 19:41	File folder	
Menu	06/08/2022 19:40	File folder	
pkgs	06/08/2022 19:38	File folder	
Scripts	06/08/2022 19:40	File folder	
share	03/10/2021 09:15	File folder	
shell	30/09/2021 12:30	File folder	
sip	03/10/2021 09:14	File folder	
tcl	06/08/2022 19:41	File folder	
Tools	30/09/2021 12:30	File folder	
python.exe	04/04/2022 11:35	Application	93 KB
python.exe.c~.conda_trash	18/05/2021 20:54	CONDA_TRASH File	93 KB
python.pdb	04/04/2022 11:35	Protein Data Bank ...	460 KB
python3.dll	04/04/2022 11:35	Application extens...	51 KB

37 items | 1 item selected |

There could (should!) be many Pythons on your PC

C:/Users/alexey.larionov/my_project/venv/Scripts/**python.exe**



C:\Users\alexey.larionov\my_project\.venv\Scripts

This PC › Windows (C:) › Users › alexey.larionov › my_project › .venv › Scripts

Name	Date modified	Type	Size
activate	06/09/2022 18:12	File	2 KB
activate.bat	06/09/2022 18:12	Windows Batch File	1 KB
Activate.ps1	06/09/2022 18:12	Windows PowerSh...	9 KB
deactivate.bat	06/09/2022 18:12	Windows Batch File	1 KB
pip.exe	06/09/2022 18:12	Application	104 KB
pip3.9.exe	06/09/2022 18:12	Application	104 KB
pip3.exe	06/09/2022 18:12	Application	104 KB
python.exe	06/09/2022 18:12	Application	508 KB
pythonw.exe	06/09/2022 18:12	Application	507 KB

9 items

There could (should!) be many ~~Pythons~~ on your PC

C:/Users/alexey.larionov/AppData/Local/Programs/Python/Python39/**python.exe**

C:/Users/alexey.larionov/my_project/venv/Scripts/**python.exe**

C:/Users/alexey.larionov/Miniconda3/**python.exe**

In fact, I have many more Python environments on my PC …

Called

Interpreters (VS Code)
or
Kernels (Jupyter)
or
Virtual Environments (venv)

It's a good programming practice to create a separate virtual environment for each Python project !

(using conda or venv – as we will discuss later)

The "system / global" Python interpreter is the one that is called by default when you type "python" on the system terminal

The "system" Python interpreter is defined by what of the available Pythons is in PATH variable of the terminal (shell)

Lecture plan

- ✓ Python features
- ✓ Installing Python(s) on your PC
- Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- Comments & Indentation
- Python packages and environments
  - PIP, Conda, Venv

Interactive use in terminal

Open command prompt

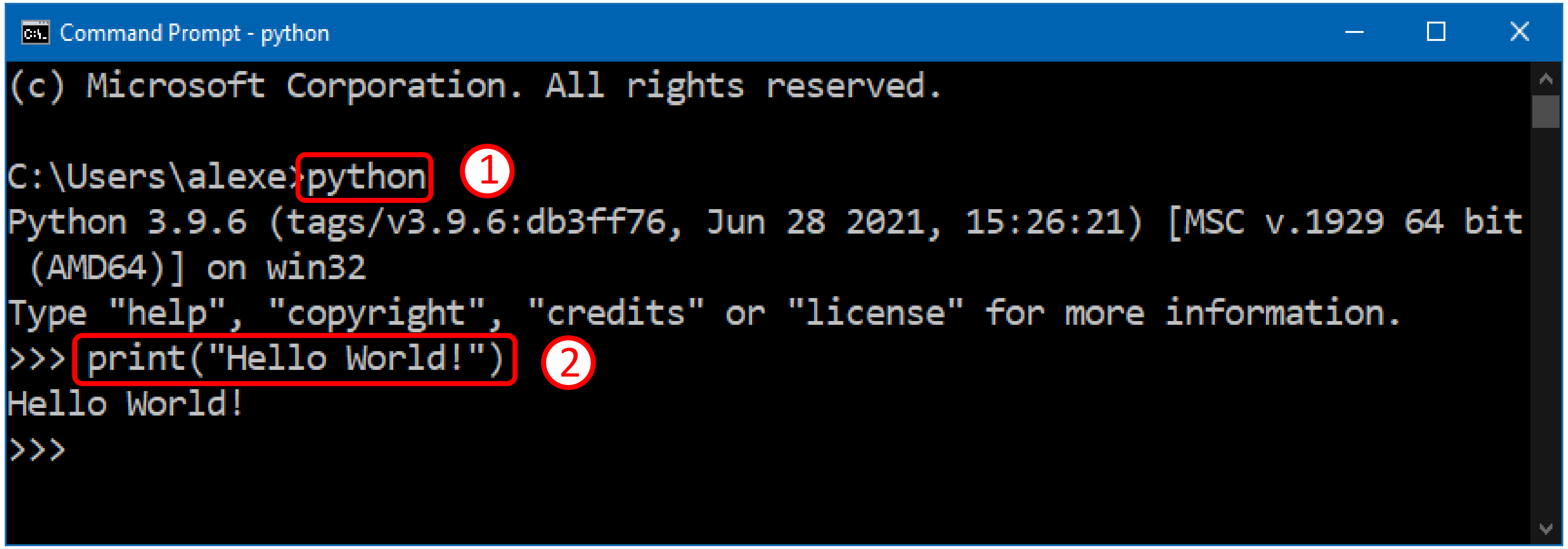
1 Type `python`
to start Python interpreter

2 Type `print(”Hello, world!”)`
within the Python interpreter

In some Linux systems you may need to use **`python3`** instead of `python`

In Windows you may use **`py`**
instead of `python`
https://docs.python.org/3/using/windows.html#launcher

Use `exit()` command to exit

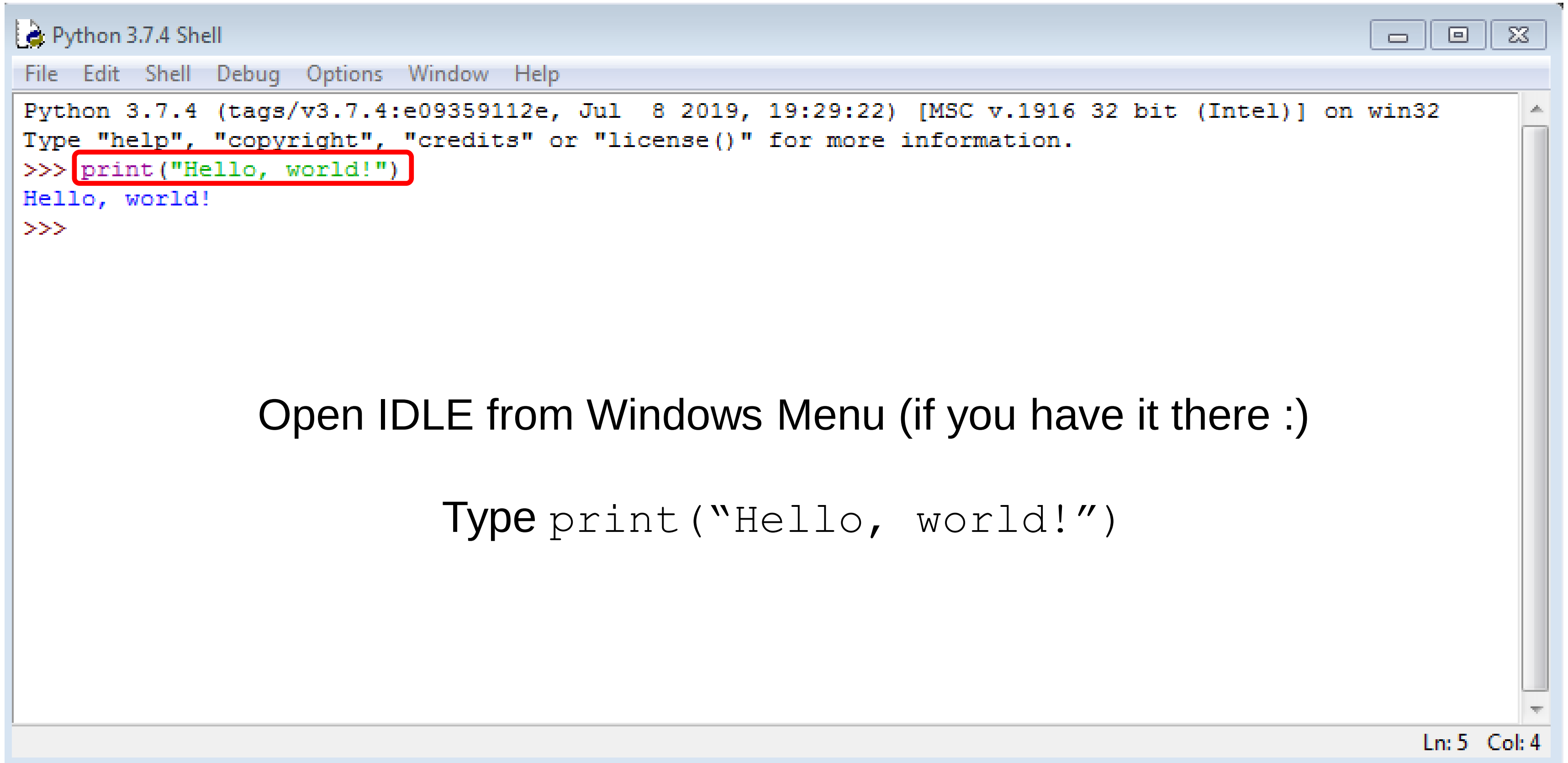


```
(c) Microsoft Corporation. All rights reserved.

C:\Users\alexe>python ①
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit
 (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello World!") ②
Hello World!
>>>
```

Sometime this interactive way of using Python is called Read-Evaluate-Print-Loop: REPL

Python IDLE

- Integrated DeveLopment Environment: a tool traditionally supplied with many Python distributions
- It is not actually a proper IDE (like RStudio, NetBeans, VisualStudio etc)
- It’s an interactive Python terminal with syntax highlighting, auto-completion, smart indentation etc



```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print("Hello, world!")
Hello, world!
>>>
```

Open IDLE from Windows Menu (if you have it there :)

Type `print(“Hello, world!”)`

I personally don’t know a single person who uses Python IDLE …

Lecture plan

- ✓ Python features
- ✓ Installing Python(s) on your PC
- Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- Comments & Indentation
- Python packages and environments
  - PIP, Conda, Venv

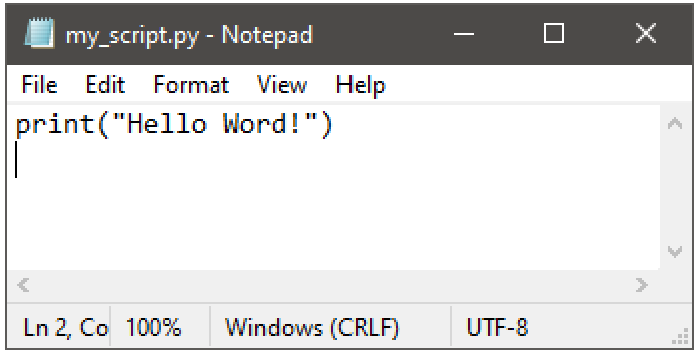
Writing a Python script

Create a text file called `my_script.py`

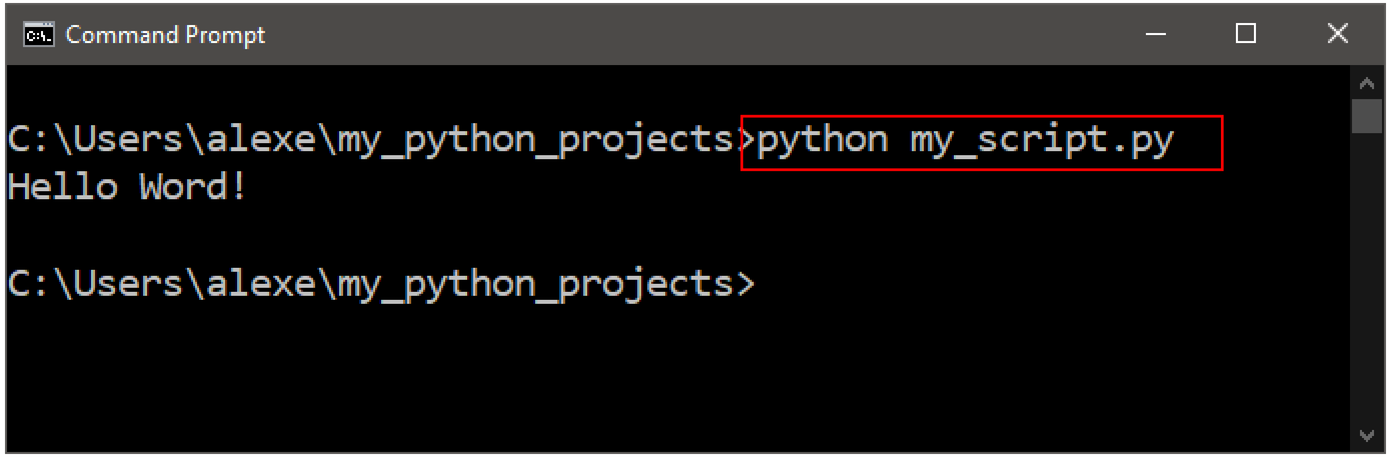
Write in the file `print("Hello, world!")`

Open terminal in the folder containing your script

Type in the terminal `python my_script.py`


my_script.py - Notepad
File Edit Format View Help
Ln 2, Co 100% Windows (CRLF) UTF-8

```
print("Hello Word!")
```



```
C:\Users\alexe\my_python_projects>python my_script.py
Hello Word!

C:\Users\alexe\my_python_projects>
```

A good Python IDE shipped with Anaconda …

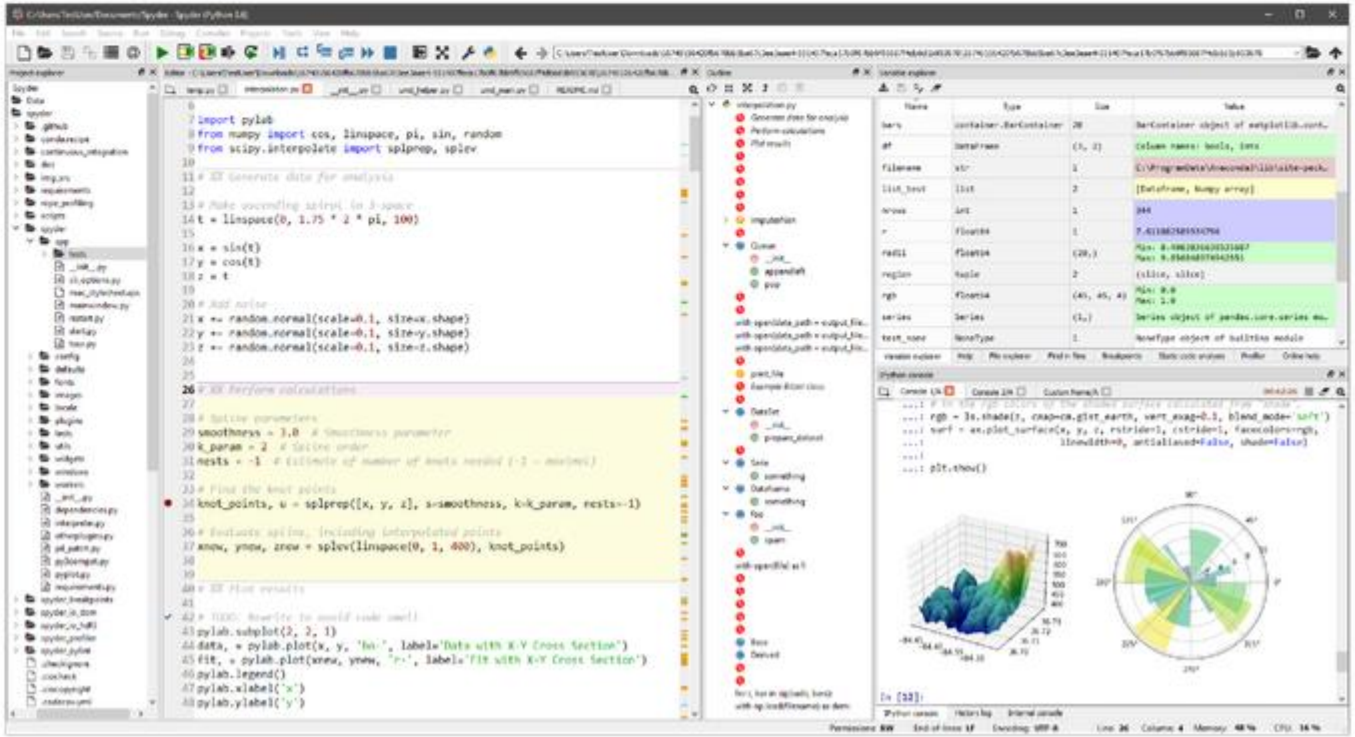
Spyder or Spider ?

https://wikidiff.com/spyder/spider

Spyder is an open-source cross-platform integrated development environment (IDE) for scientific programming in Python

- Spyder installer includes a number of important Python packages, including NumPy, SciPy, Matplotlib, pandas, IPython, SymPy and Cython, as well as other open-source software

Feels like R-Studio for Python …



```
import pylab
from numpy import cos, linspace, pi, sin, random
from scipy.interpolate import splprep, splev

# %% Generate data for analysis

# Make ascending spiral in 3-space
t = linspace(0, 1.75 * 2 * pi, 100)

x = sin(t)
y = cos(t)
z = t

# Add noise
x += random.normal(scale=0.1, size=x.shape)
y += random.normal(scale=0.1, size=y.shape)
z += random.normal(scale=0.1, size=z.shape)


# %% Perform calculations

# Spline parameters
smoothness = 3.0  # Smoothness parameter
k_param = 2  # Spline order
nests = -1  # Estimate of number of knots needed (-1 = maximal)

# Find the knot points
knot_points, u = splprep([x, y, z], s=smoothness, k=k_param, nests=-1)

# Evaluate spline, including interpolated points
xnew, ynew, znew = splev(linspace(0, 1, 400), knot_points)


# %% Plot results

# TODO: Rewrite to avoid code smell
pylab.subplot(2, 2, 1)
data, = pylab.plot(x, y, 'bo-', label='Data with X-Y Cross Section')
fit, = pylab.plot(xnew, ynew, 'r-', label='Fit with X-Y Cross Section')
pylab.legend()
pylab.xlabel('x')
pylab.ylabel('y')
```

The proper ways to write Python code

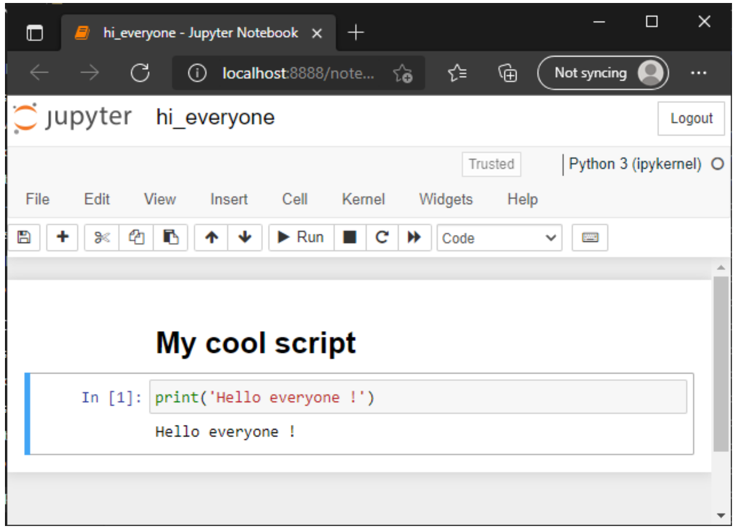
[Jupyter or Jupiter?](https://en.wikipedia.org/wiki/Project_Jupyter)

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

Now, forget all the ways of using Python shown in the previous slides !

- **Jupyter Notebook for interactive analysis**
- De-facto standard in data science & bioinformatics
- Keeps well-formatted comments, plots and code together: Reproducible Research !

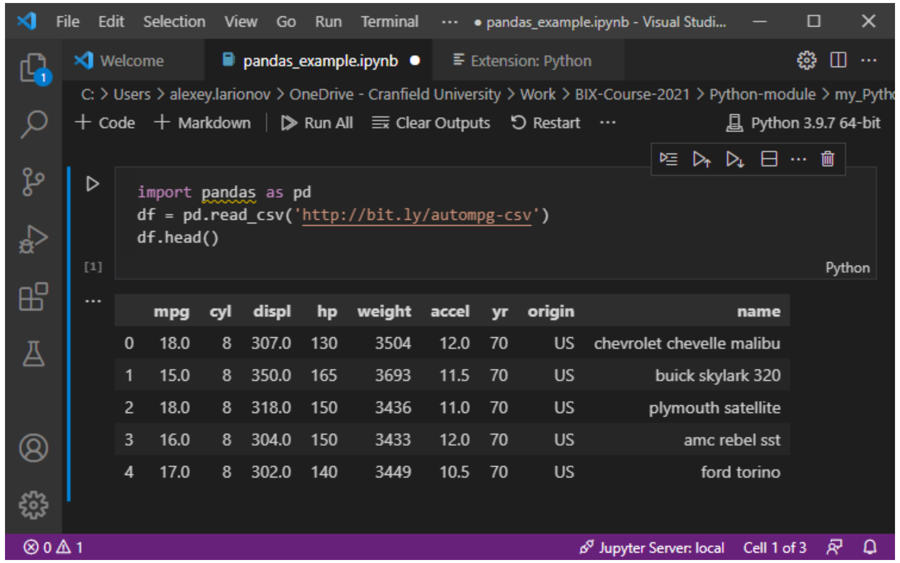
- **VS Code or PyCharm for coding**
- Good for software developers
- Could be confusing for beginners
- Supports Jupyter Notebooks too !



My cool script

```
In [1]: print('Hello everyone !')

        Hello everyone !
```

https://jupyter.org



```
import pandas as pd
df = pd.read_csv('http://bit.ly/autompg-csv')
df.head()
```

```
[1]
    mpg  cyl  displ   hp  weight  accel  yr  origin                        name
0  18.0    8  307.0  130    3504   12.0  70      US   chevrolet chevelle malibu
1  15.0    8  350.0  165    3693   11.5  70      US           buick skylark 320
2  18.0    8  318.0  150    3436   11.0  70      US          plymouth satellite
3  16.0    8  304.0  150    3433   12.0  70      US               amc rebel sst
4  17.0    8  302.0  140    3449   10.5  70      US                 ford torino
```

https://code.visualstudio.com

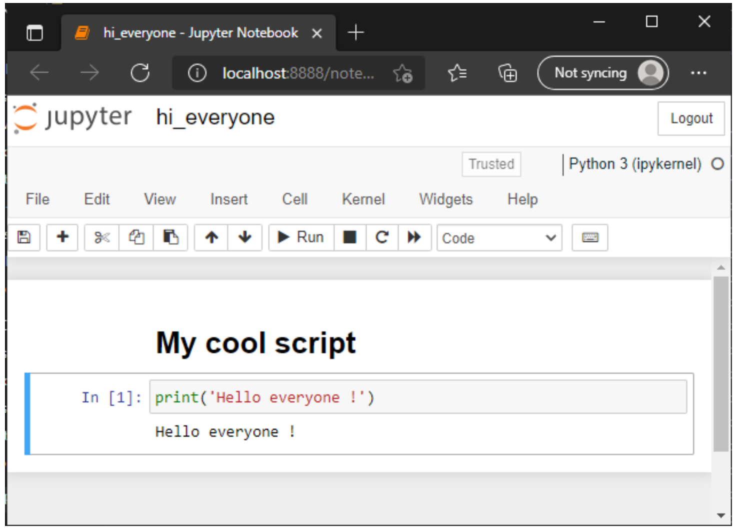
Cranfield
Environment
and Agrifood

The proper ways to write Python code

https://blog.jetbrains.com/pycharm/2017/09/pycharm-community-edition-and-professional-edition-explained-licenses-and-more

Mac users ?

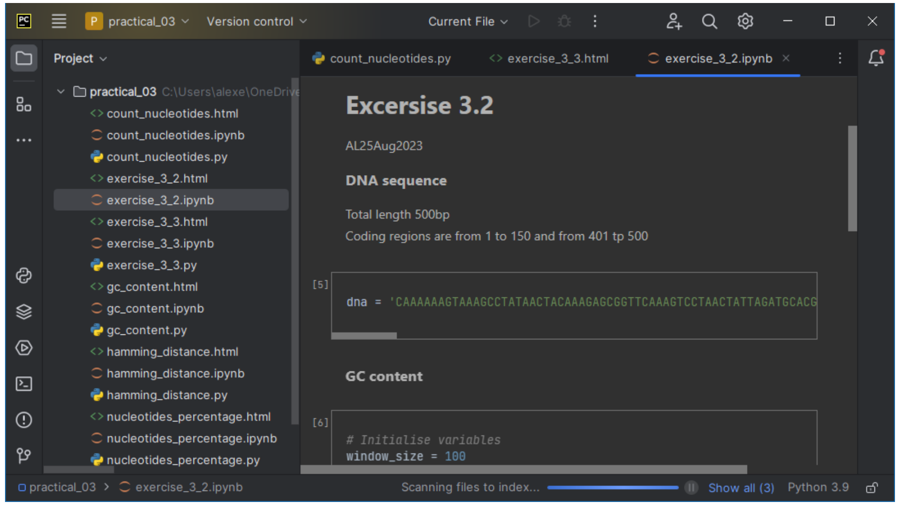
- **Jupyter Notebook for interactive analysis**
- De-facto standard in data science & bioinformatics
- Keeps well-formatted comments, plots and code together: Reproducible Research !



My cool script

```
In [1]: print('Hello everyone !')

        Hello everyone !
```

- **VS Code or PyCharm for coding**
- Good for software developers
- Could be confusing for beginners
- Supports Jupyter Notebooks too !



Excersise 3.2

AL25Aug2023

DNA sequence

Total length 500bp

Coding regions are from 1 to 150 and from 401 tp 500

```
[5]
dna = 'CAAAAAAGTAAAGCCTATAACTACAAAGAGCGGTTCAAAGTCCTAACTATTAGATGCACG
```

GC content

```
[6]
# Initialise variables
window_size = 100
```

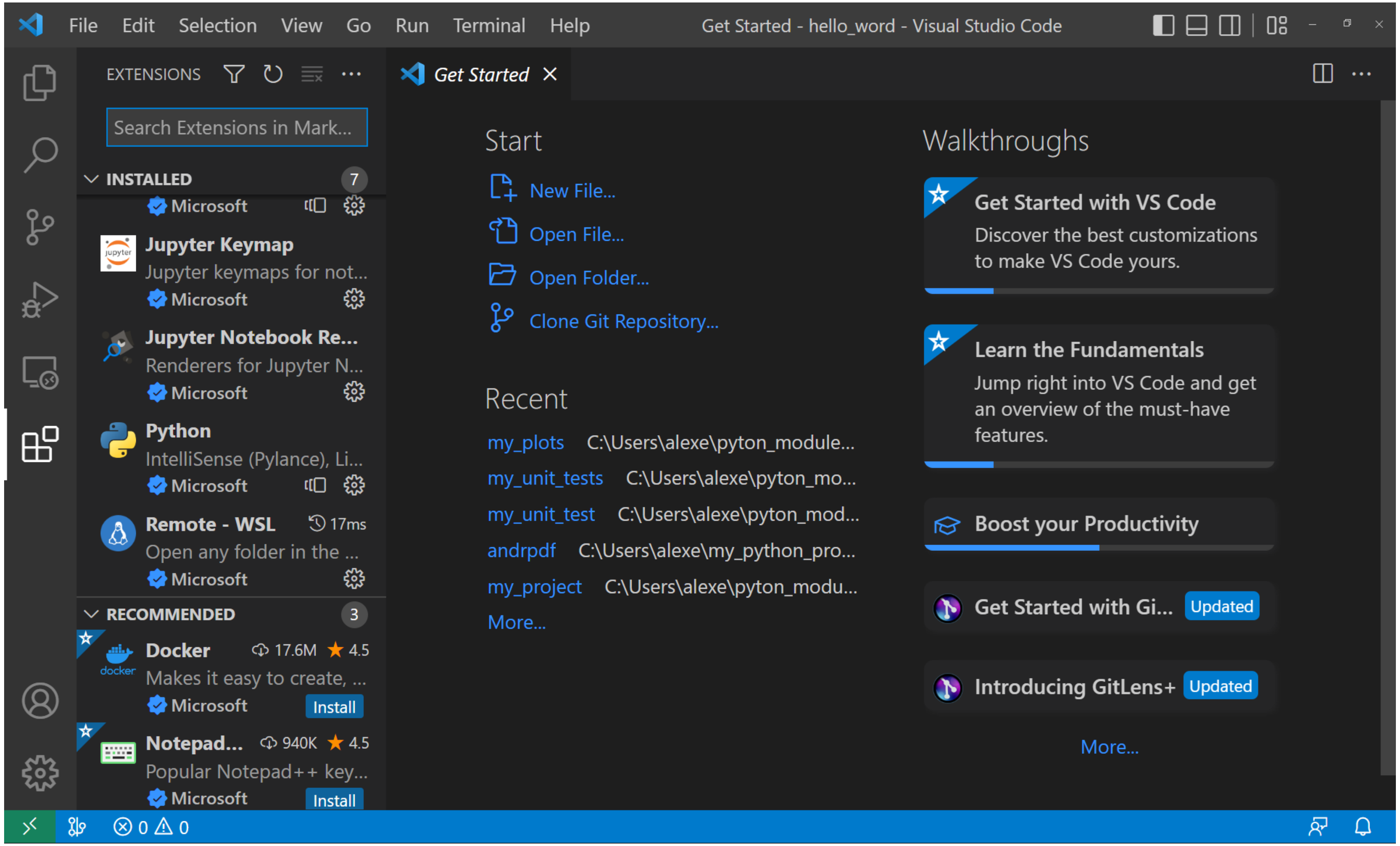
https://www.jetbrains.com/pycharm/download/other.html

Lecture plan

- ✓ Python features
- ✓ Installing Python(s) on your PC
- Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- Comments & Indentation
- Python packages and environments
  - PIP, Conda, Venv

Using VS-Code for “Hello World !” script in Python

You will have time for this during the practical session !

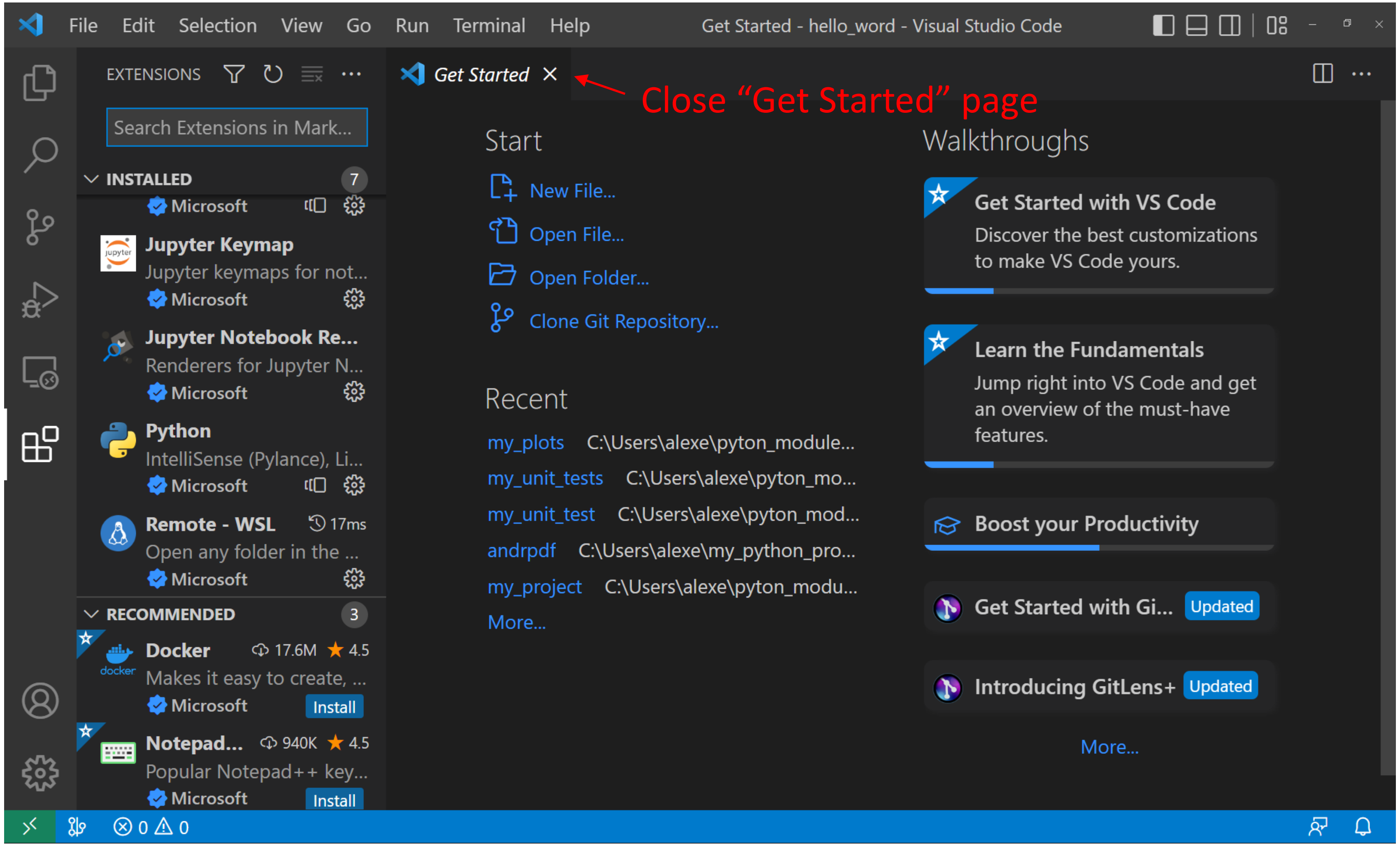

File Edit Selection View Go Run Terminal Help
Get Started - hello_word - Visual Studio Code
EXTENSIONS
Search Extensions in Mark...
INSTALLED 7
Microsoft
Jupyter Keymap
Jupyter keymaps for not...
Microsoft
Jupyter Notebook Re...
Renderers for Jupyter N...
Microsoft
Python
IntelliSense (Pylance), Li...
Microsoft
Remote - WSL 17ms
Open any folder in the ...
Microsoft
RECOMMENDED 3
Docker 17.6M ★ 4.5
Makes it easy to create, ...
Microsoft Install
Notepad... 940K ★ 4.5
Popular Notepad++ key...
Microsoft Install
Get Started
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Recent
my_plots C:\Users\alexe\pyton_module...
my_unit_tests C:\Users\alexe\pyton_mo...
my_unit_test C:\Users\alexe\pyton_mod...
andrpdf C:\Users\alexe\my_python_pro...
my_project C:\Users\alexe\pyton_modu...
More...
Walkthroughs
Get Started with VS Code
Learn the Fundamentals
Boost your Productivity
Get Started with Gi... Updated
Introducing GitLens+ Updated
More...
0 0

Discover the best customizations to make VS Code yours.

Jump right into VS Code and get an overview of the must-have features.

VS Code may look confusing at the beginning …

Using VS-Code for "Hello World !" script in Python



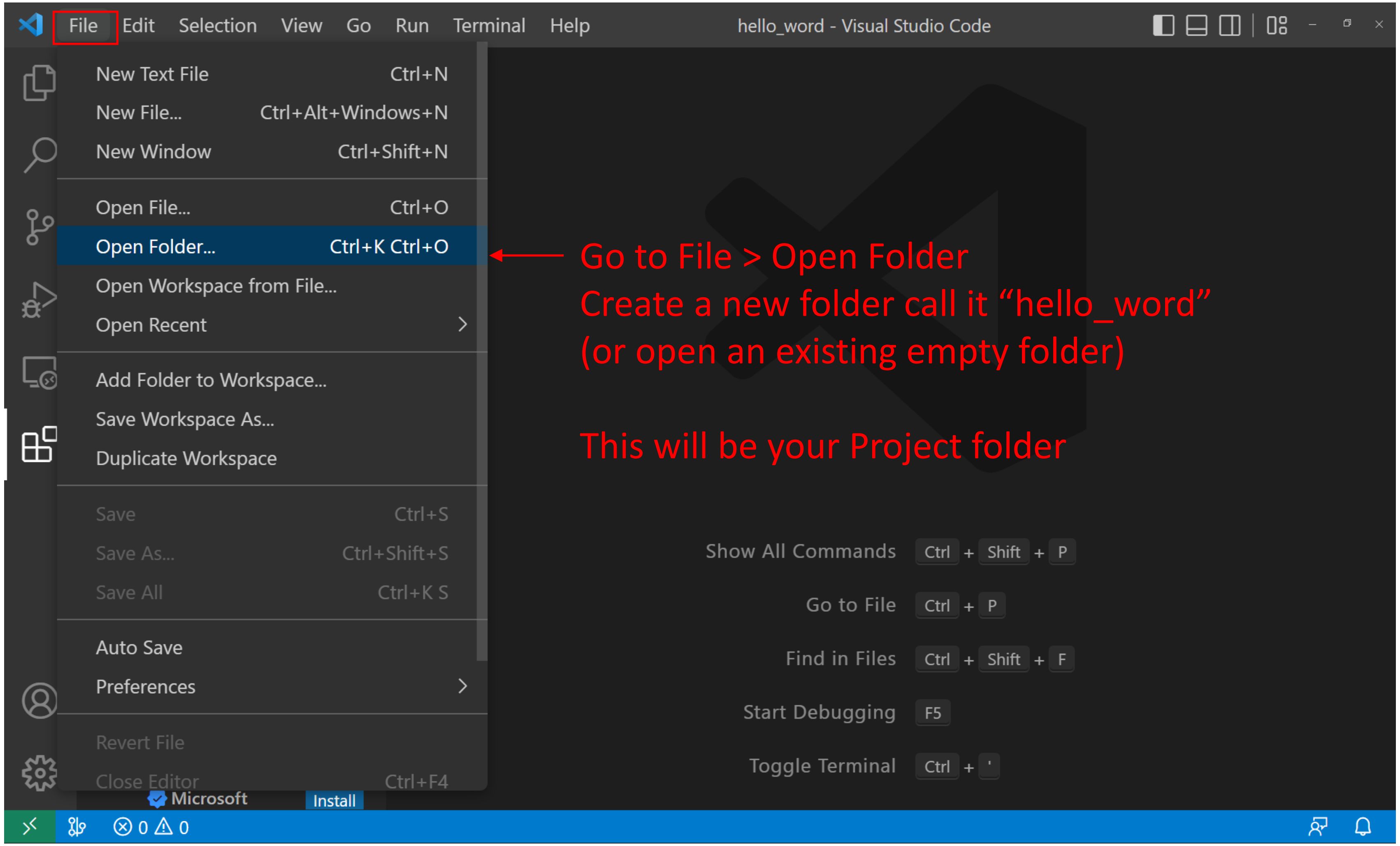
Close "Get Started" page

Get Started - hello_word - Visual Studio Code

File Edit Selection View Go Run Terminal Help
EXTENSIONS
Search Extensions in Mark...
INSTALLED 7
Microsoft
Jupyter Keymap
Jupyter keymaps for not...
Microsoft
Jupyter Notebook Re...
Renderers for Jupyter N...
Microsoft
Python
IntelliSense (Pylance), Li...
Microsoft
Remote - WSL 17ms
Open any folder in the ...
Microsoft
RECOMMENDED 3
Docker 17.6M ★ 4.5
Makes it easy to create, ...
Microsoft Install
Notepad... 940K ★ 4.5
Popular Notepad++ key...
Microsoft Install
Get Started
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Recent
my_plots C:\Users\alexe\pyton_module...
my_unit_tests C:\Users\alexe\pyton_mo...
my_unit_test C:\Users\alexe\pyton_mod...
andrpdf C:\Users\alexe\my_python_pro...
my_project C:\Users\alexe\pyton_modu...
More...
Walkthroughs
Get Started with VS Code
Discover the best customizations to make VS Code yours.
Learn the Fundamentals
Jump right into VS Code and get an overview of the must-have features.
Boost your Productivity
Get Started with Gi... Updated
Introducing GitLens+ Updated
More...
0 0

VS Code may look confusing at the beginning ...

Using VS-Code for “Hello World !” script in Python …



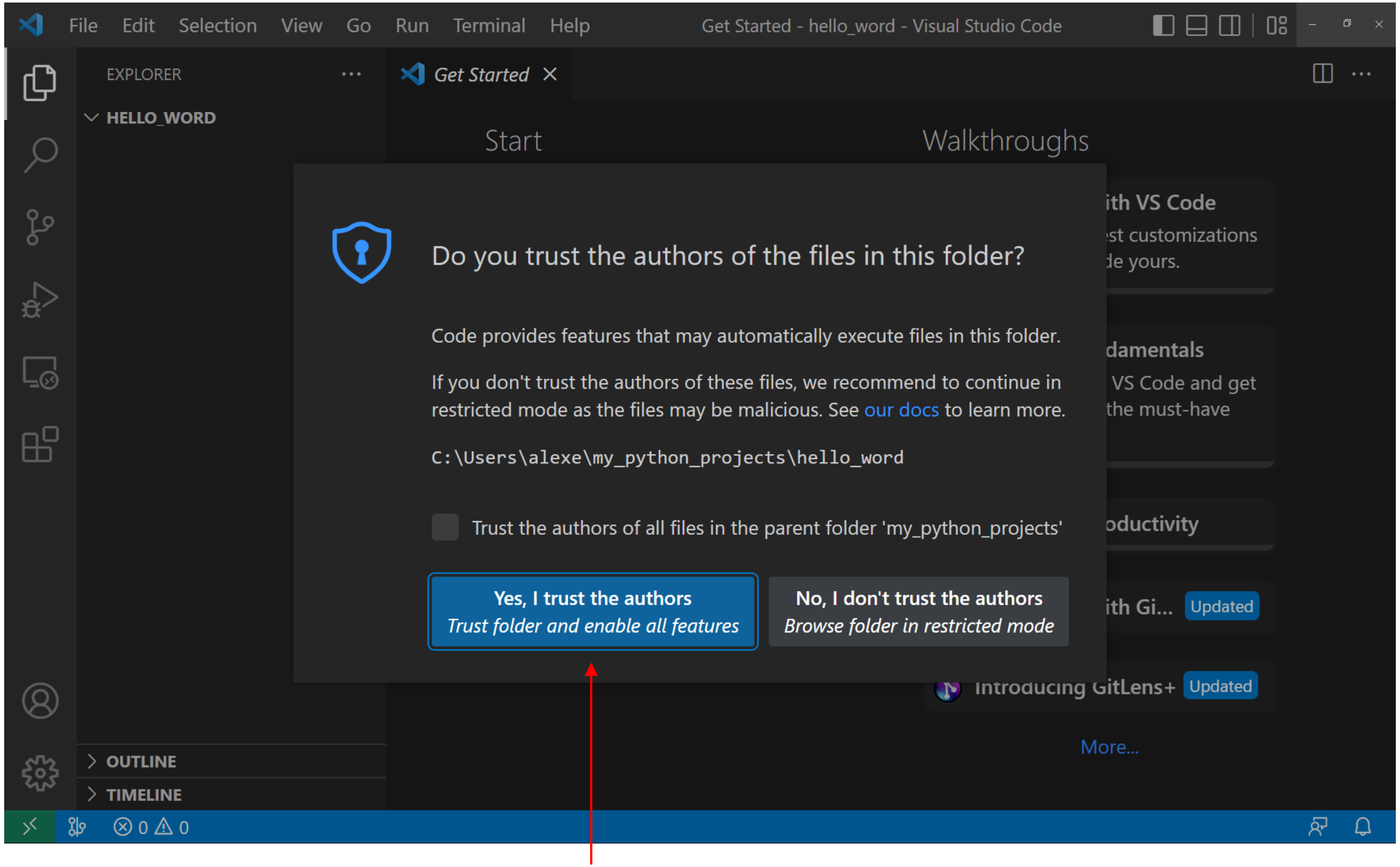
Go to File > Open Folder
Create a new folder call it “hello_word”
(or open an existing empty folder)

This will be your Project folder

It’s a good programming practice to make a Project folder for any Python project, even if it includes a single file

Using VS-Code for “Hello World !” script in Python

Sorry, VS-Code is slightly paranoid :)



Do you trust the authors of the files in this folder?

Code provides features that may automatically execute files in this folder.

If you don't trust the authors of these files, we recommend to continue in restricted mode as the files may be malicious. See our docs to learn more.

`C:\Users\alexe\my_python_projects\hello_word`

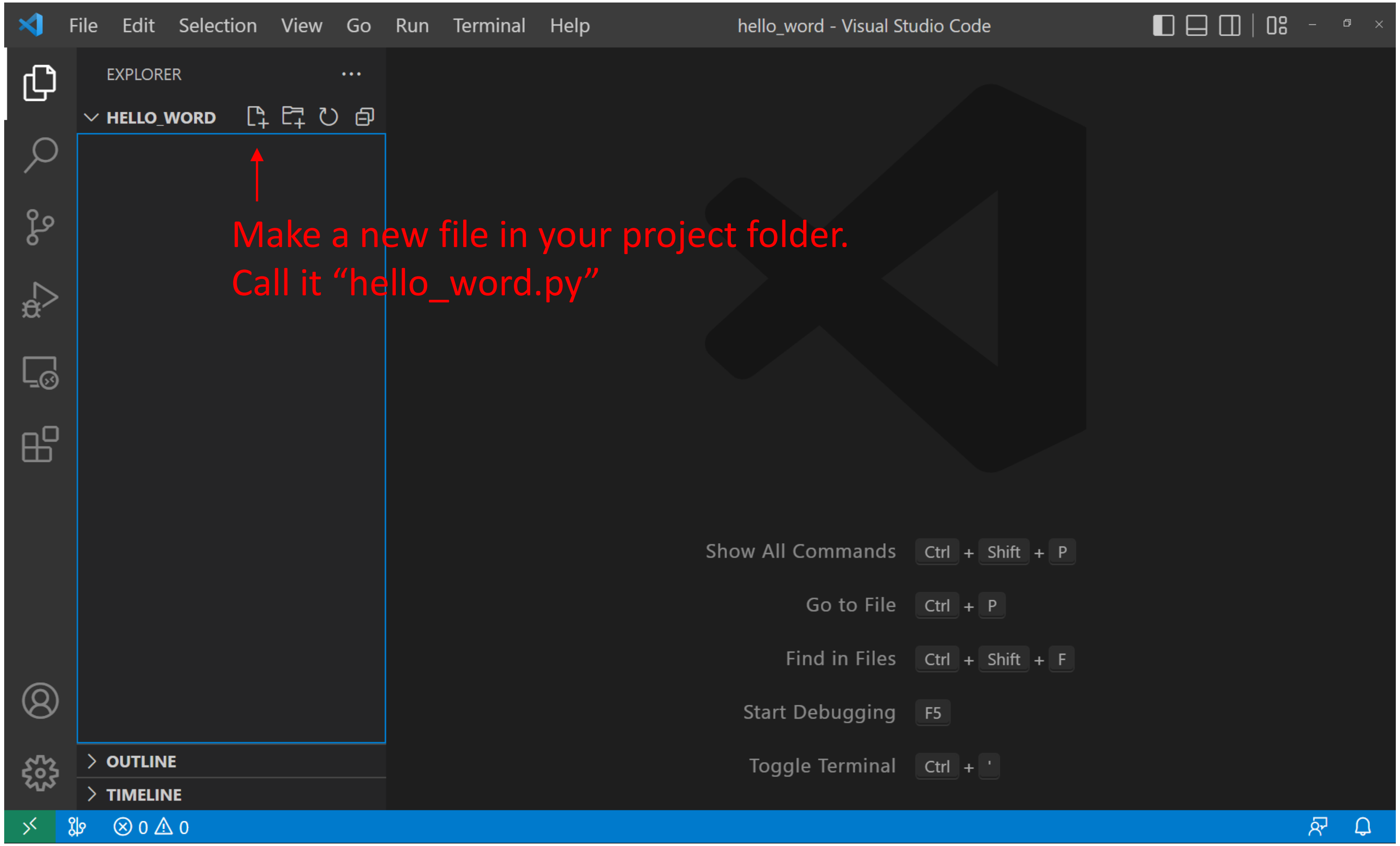
□ Trust the authors of all files in the parent folder 'my_python_projects'

Yes, I trust the authors
Trust folder and enable all features
No, I don't trust the authors
Browse folder in restricted mode

Decide whether you trust yourself …

Using VS-Code for “Hello World !” script in Python

Make sure you selected file explorer tab



Make a new file in your project folder.
Call it “hello_word.py”

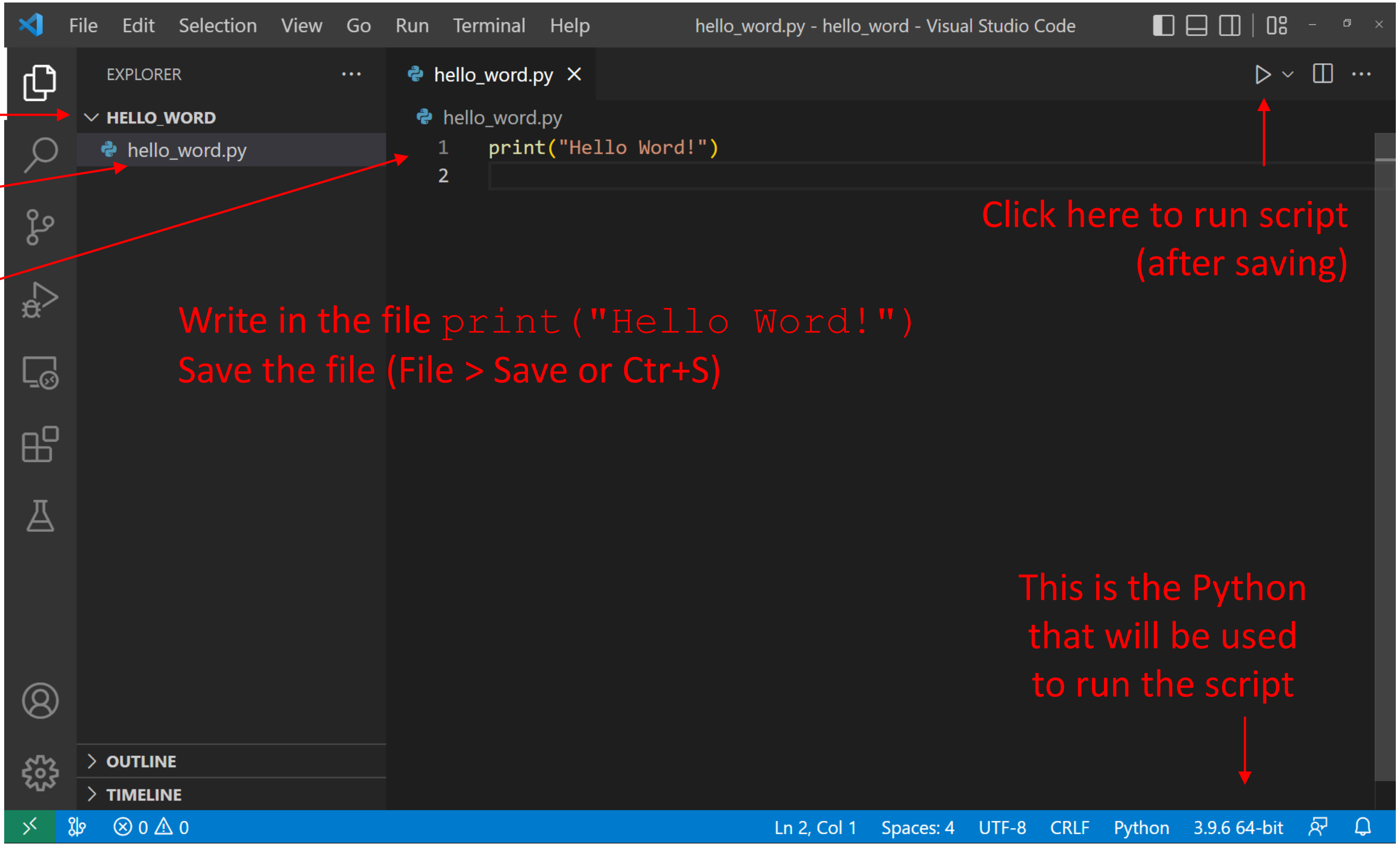
If you are prompted to install Python extension: agree to it :)

Using VS-Code for “Hello World !” script in Python

This is the name of your folder

This is the name of your file

This is the content of your file



```
print("Hello Word!")
```

Click here to run script (after saving)

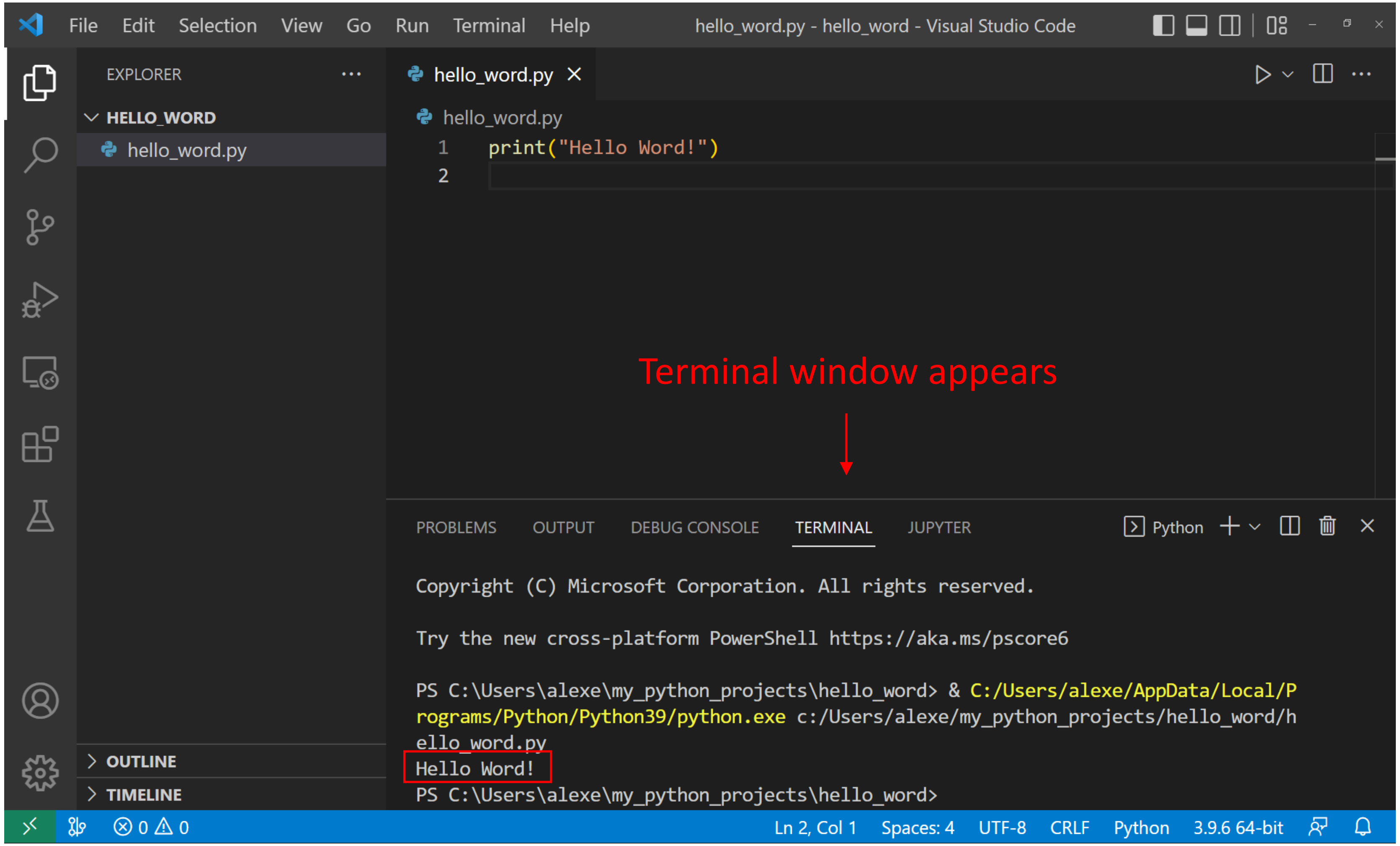
Write in the file `print("Hello Word!")`

Save the file (File > Save or Ctr+S)

This is the Python that will be used to run the script

If you work on Windows, you may need to configure terminal when you attempt running the script

Using VS-Code for “Hello World !” script in Python



File Edit Selection View Go Run Terminal Help
hello_word.py - hello_word - Visual Studio Code
EXPLORER
HELLO_WORD
hello_word.py
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python
Ln 2, Col 1 Spaces: 4 UTF-8 CRLF Python 3.9.6 64-bit

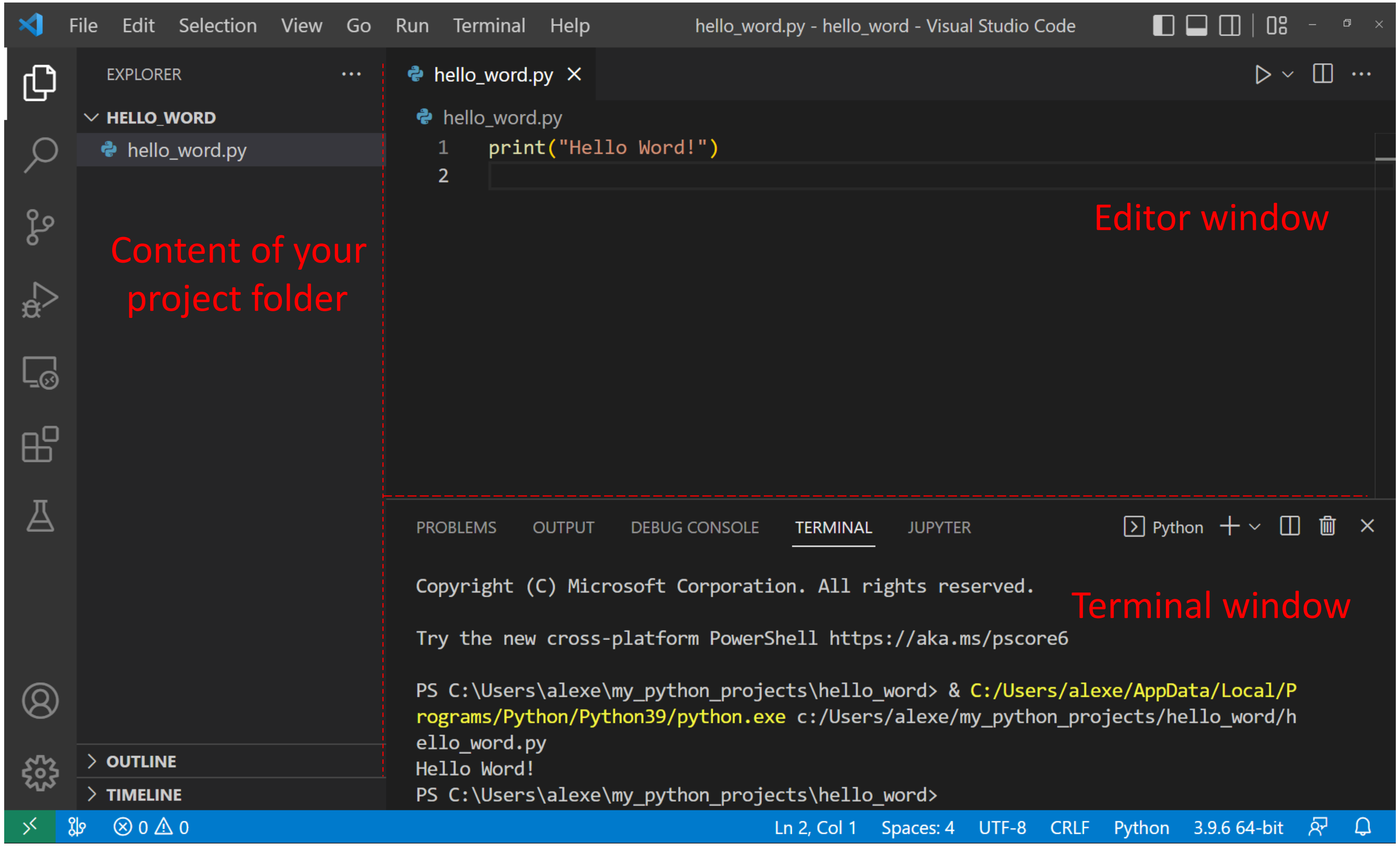
```python
print("Hello Word!")
```

Terminal window appears

```
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\alexe\my_python_projects\hello_word> & C:/Users/alexe/AppData/Local/P
rograms/Python/Python39/python.exe c:/Users/alexe/my_python_projects/hello_word/h
ello_word.py
Hello Word!
PS C:\Users\alexe\my_python_projects\hello_word>
```

Hurray !!! You have run your first Python script in VS Code !!!

Using VS-Code for “Hello World !” script in Python



File Edit Selection View Go Run Terminal Help
hello_word.py - hello_word - Visual Studio Code
EXPLORER
HELLO_WORD
hello_word.py
OUTLINE
TIMELINE
hello_word.py
Content of your project folder
Editor window
Terminal window
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python
Ln 2, Col 1 Spaces: 4 UTF-8 CRLF Python 3.9.6 64-bit

```python
print("Hello Word!")
```

```
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\alexe\my_python_projects\hello_word> & C:/Users/alexe/AppData/Local/P
rograms/Python/Python39/python.exe c:/Users/alexe/my_python_projects/hello_word/h
ello_word.py
Hello Word!
PS C:\Users\alexe\my_python_projects\hello_word>
```

VS Code is not so confusing after all ☺

Lecture plan

- ✓ Python features
- ✓ Installing Python(s) on your PC
- ✓ Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- Comments & Indentation
- Python packages and environments
  - PIP, Conda, Venv

Comments in your script

Comments are not executed.
However, it’s a good practice to put comments in your script

```
# A traditional one-line comment

"""
This is an example of multi-line comment:
You should not worry if comments make a half of your script !
"""
```

Comments in your script

Use English US Keyboard

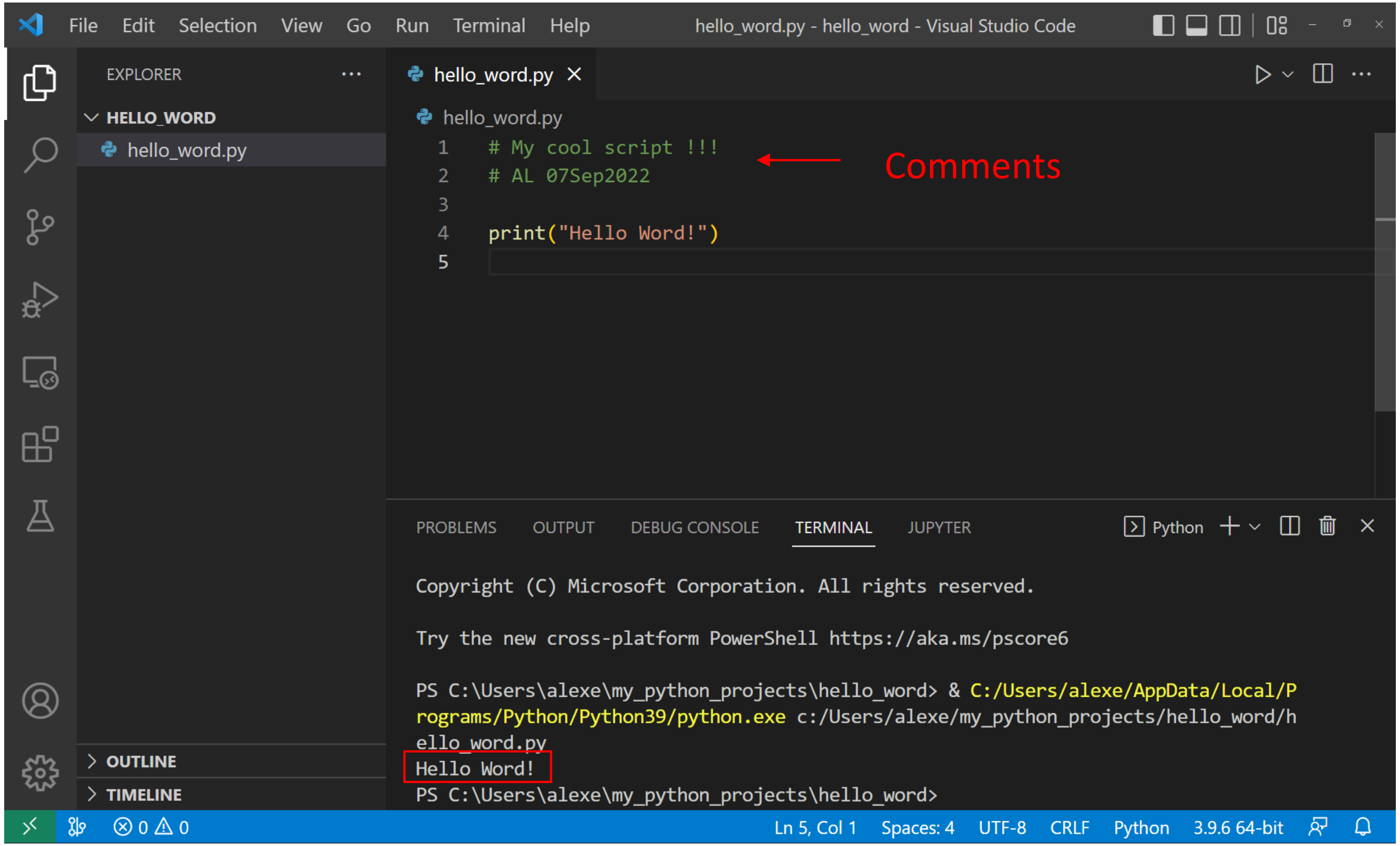
English UK Keyboard may type £

Comments are not executed.

However, it’s a good practice to put comments in your script

```
# A traditional one-line comment


"""
This is an example of multi-line comment:
You should not worry if comments make a quote of your script !
"""
```

Comments in Python



File Edit Selection View Go Run Terminal Help
hello_word.py - hello_word - Visual Studio Code
EXPLORER
HELLO_WORD
hello_word.py
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python
Ln 5, Col 1 Spaces: 4 UTF-8 CRLF Python 3.9.6 64-bit
Comments

```python
# My cool script !!!
# AL 07Sep2022

print("Hello Word!")

```

```
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\alexe\my_python_projects\hello_word> & C:/Users/alexe/AppData/Local/P
rograms/Python/Python39/python.exe c:/Users/alexe/my_python_projects/hello_word/h
ello_word.py
Hello Word!
PS C:\Users\alexe\my_python_projects\hello_word>
```

Result is the same: comments are not executed !

Indentation

Most programming languages do not care about indentations …

But most human do …

Non-indented C-style Code:

```
if(condition_1){
if(condition_2){
function_X()
}
else{
function_Y()
}
}
```

Indented C-style Code:

```
if(condition_1){
      if(condition_2){
            function_X()
      }
      else{
            function_Y()
      }
}
```

Indentation is required in Python

Indented C-style Code:

```
if(condition_1){
      if(condition_2){
            function_X()
      }
      else{
            function_Y()
      }
}
```

Indented Python Code:

```
if(condition_1):
      if(condition_2):
            function_X()

      else:
            function_Y()

...
```

Lecture plan

- ✓ Python features
- ✓ Installing Python(s) on your PC
- ✓ Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- ✓ Comments & Indentation
- Python packages and environments
  - PIP, Conda, Venv

The Python philosophy

A small (!) core language …

and lots (!) of modules

Module = Package = Library

math

sys

os

only load what you need &
do not reinvent the wheel

Built-in modules: for standard tasks

built-in modules come with Python, they don’t need to be installed

- However, to use a built-in module, you still need to import it: `import <module name>`

Examples:

```
import sys # system module
sys.executable   # the Python that you use …

import os # os module
os.getcwd() # working folder …

import math # math module
math.sin(0.5) # math functions …
```

If you want to find what modules are installed in your Python environment:

```
help("modules")
```

For specific module:

```
help("math")
```

Multiple ways of importing a module

- Simple import

```
import math
factorial = math.factorial(5)
print(factorial)
```

- Modify the names of modules and their functions by aliasing

```
import math as m
factorial = m.factorial(5)
print(factorial)
```

- A specific function can be imported from a module

```
from math import factorial
factorial = factorial(5)
print(factorial)
```

NOT import factorial from math

- `import` statement may reference everything defined within the module by using an asterisk (*)

```
from math import *
factorial = factorial(5)
print(factorial)
```

Don’t use this method !

Additional modules: your customised toolbox

There are LOTS of modules for specialized bioinformatics tasks !

- You must install additional modules (=packages) before use

- Packages (=modules) are installed outside of Python
  - You can not ***install*** a package inside a Python script
  - Inside a Python script you only ***import*** already installed packages

- Two main tools to install Python packages are PIP and Conda

Some cool Python modules:

https://www.youtube.com/watch?v=o06MyVhYte4

PIP - Package Installer for Python

- PIP is PyPA recommended tool to install packages from PyPI
  - Tool to install packages
  - The repository of Python packages
- Installs Python packages and their Python dependencies
- Installs into currently active Python environment
- PIP usually comes with Python installation
  - May be missed in Conda Python environments (although could be easily added by Conda)

pypa

Python Packaging Authority

python™ Package Index

PyPI

Python Packages Repository

```
pip install …
```

This is how you install a module with PIP from the system terminal !

Where PIP installs the additional libraries ?

What is the “currently active environment” ?

C:/Users/alexey.Larionov/AppData/Local/Programs/Python/Python39/**python.exe**
/**pip.exe**
/Libs/<libraries>

C:/Users/alexey.larionov/my_project/.venv/Scripts/**python.exe**
/Scripts/**pip.exe**
/Libs/<a different set of libraries>

C:/Users/alexey.larionov/Miniconda3/**python.exe**
/**pip.exe**
/Libs/<a third set of libraries>

Packages/libraries
are only visible within a
specific Python environment

Different Python environments
may have different sets of
installed packages/libraries

Lecture plan

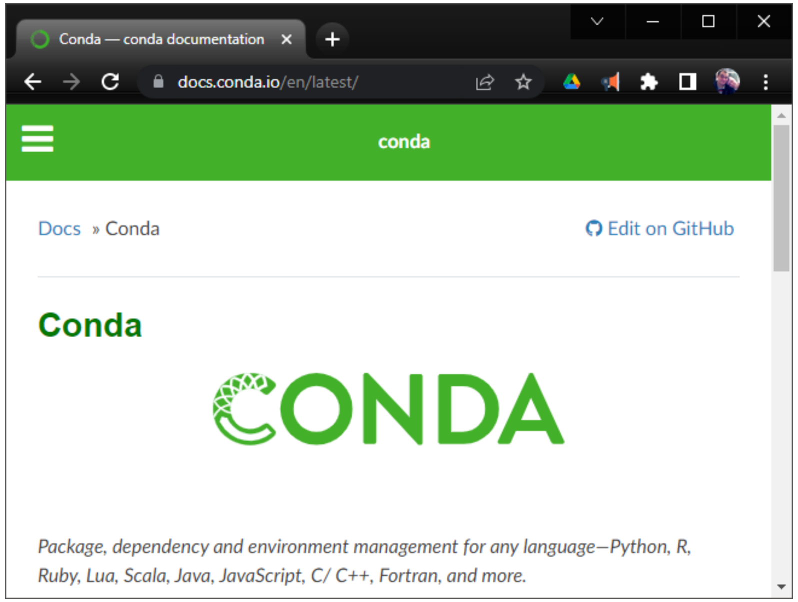
- ✓ Python features
- ✓ Installing Python(s) on your PC
- ✓ Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- ✓ Comments & Indentation
- Python packages and environments
  - PIP, Conda, Venv

Conda: a third-party package manager

Conda ≠ Python

C:/Users/alexey.larionov/Miniconda3/**python.exe**
/**conda.exe**

- Supports Python and other languages
- Installs Python and non-Python dependencies (uses pre-compiled binaries for installation)
- Many repositories (in addition to defaults)
  - conda-forge, bioconda, etc
- Organises Python environments
  - maintains CONDA environments tree
- By default, installs into currently active CONDA environment



Conda

Package, dependency and environment management for any language—Python, R, Ruby, Lua, Scala, Java, JavaScript, C/ C++, Fortran, and more.

https://docs.conda.io/en/latest

CONDA-FORGE

BIOCONDA®

CONDA environments

```
C:/Users/alexey.larionov/Miniconda3/python.exe
                                   /conda.exe
                                   /Libs/<base set of libraries>
                                   /envs/emoji_env
                                        /matplotlib
                                        /cutatapt
                                        /my_project_1
                                        /my_project_2
                                        /...
```

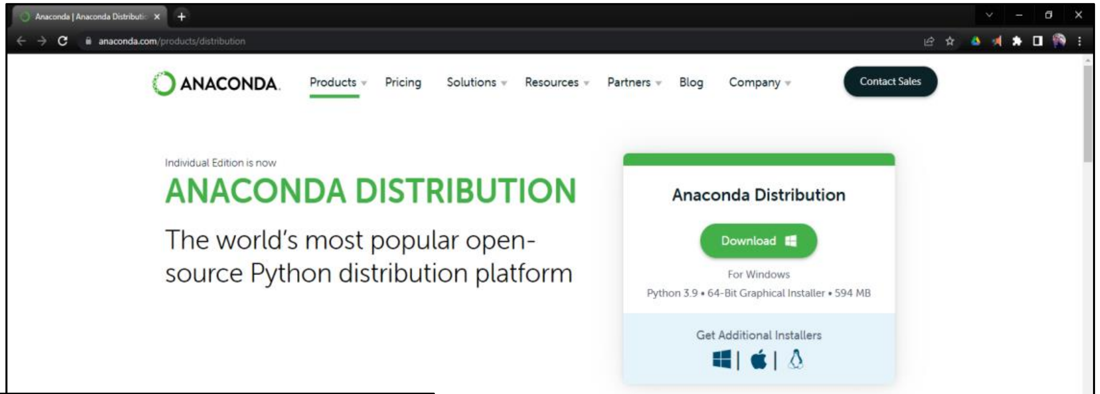
Currently active Conda environment

```
(base)> conda env list
(base)> conda create --name emoji_env
(base)> conda activate emoji_env
(emoji_env)> conda install emoji
(emoji_env)> ...
(emoji_env)> conda deactivate
(base)> ...
```

Anaconda

Anaconda = Python + Conda + Many packages + …

https://www.anaconda.com/download



Individual Edition is now

ANACONDA DISTRIBUTION

The world's most popular open-source Python distribution platform

Anaconda Distribution

For Windows

Python 3.9 • 64-Bit Graphical Installer • 594 MB

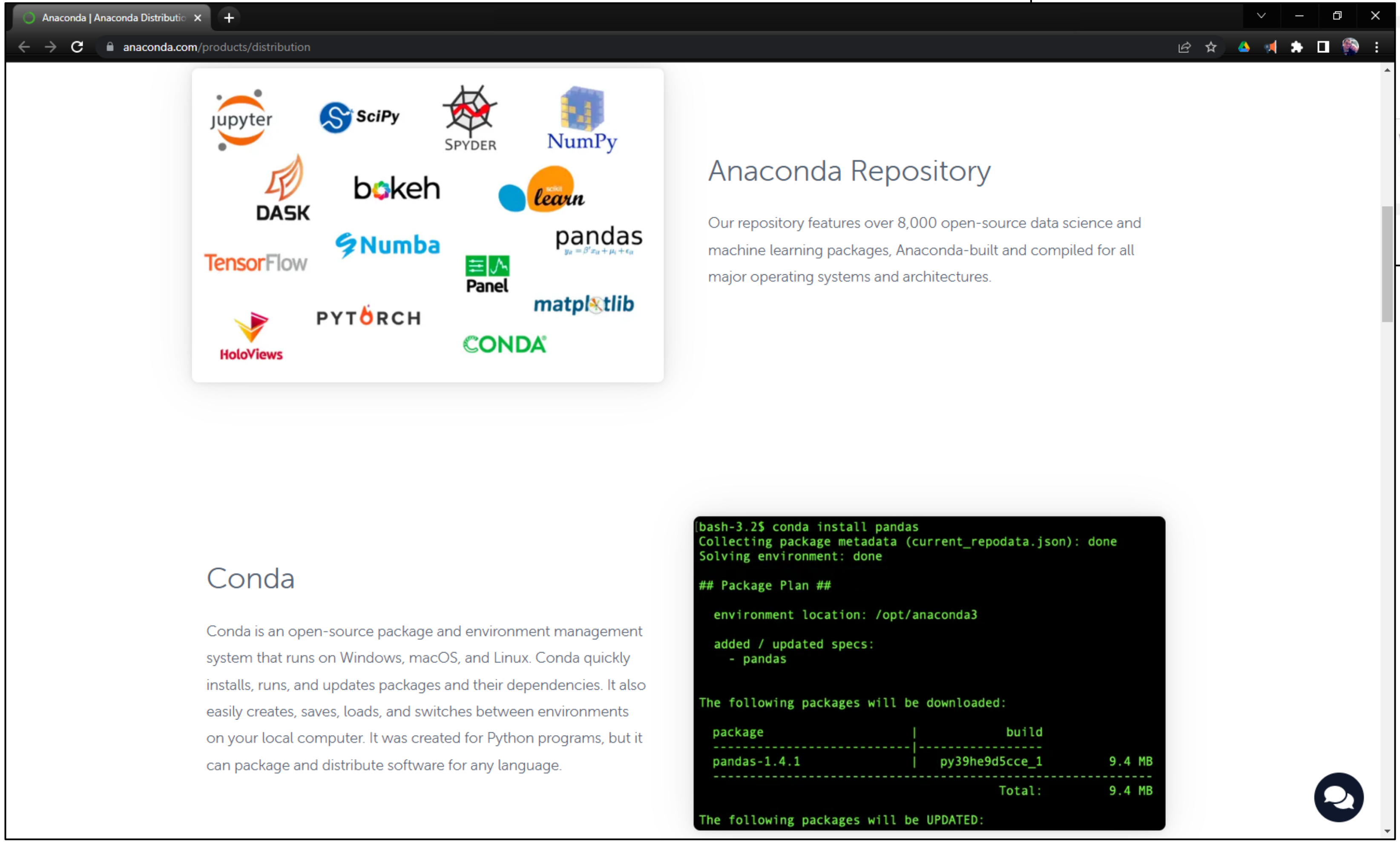
Get Additional Installers

User-friendly

With our intuitive platform, you can easily search and install packages and create, load, and switch between environments.

Trusted

Our securely hosted packages and artifacts are methodically tested and regularly updated.



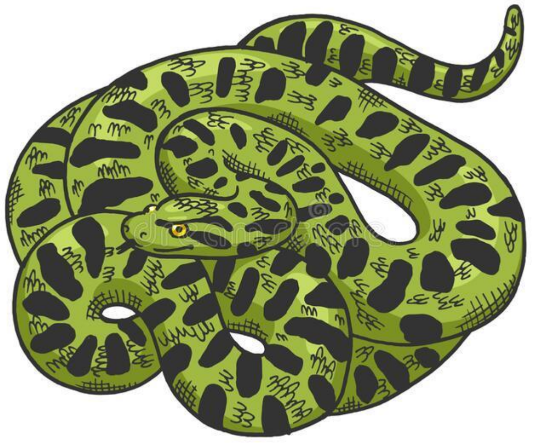
Anaconda Repository

Our repository features over 8,000 open-source data science and machine learning packages, Anaconda-built and compiled for all major operating systems and architectures.

Conda

Conda is an open-source package and environment management system that runs on Windows, macOS, and Linux. Conda quickly installs, runs, and updates packages and their dependencies. It also easily creates, saves, loads, and switches between environments on your local computer. It was created for Python programs, but it can package and distribute software for any language.

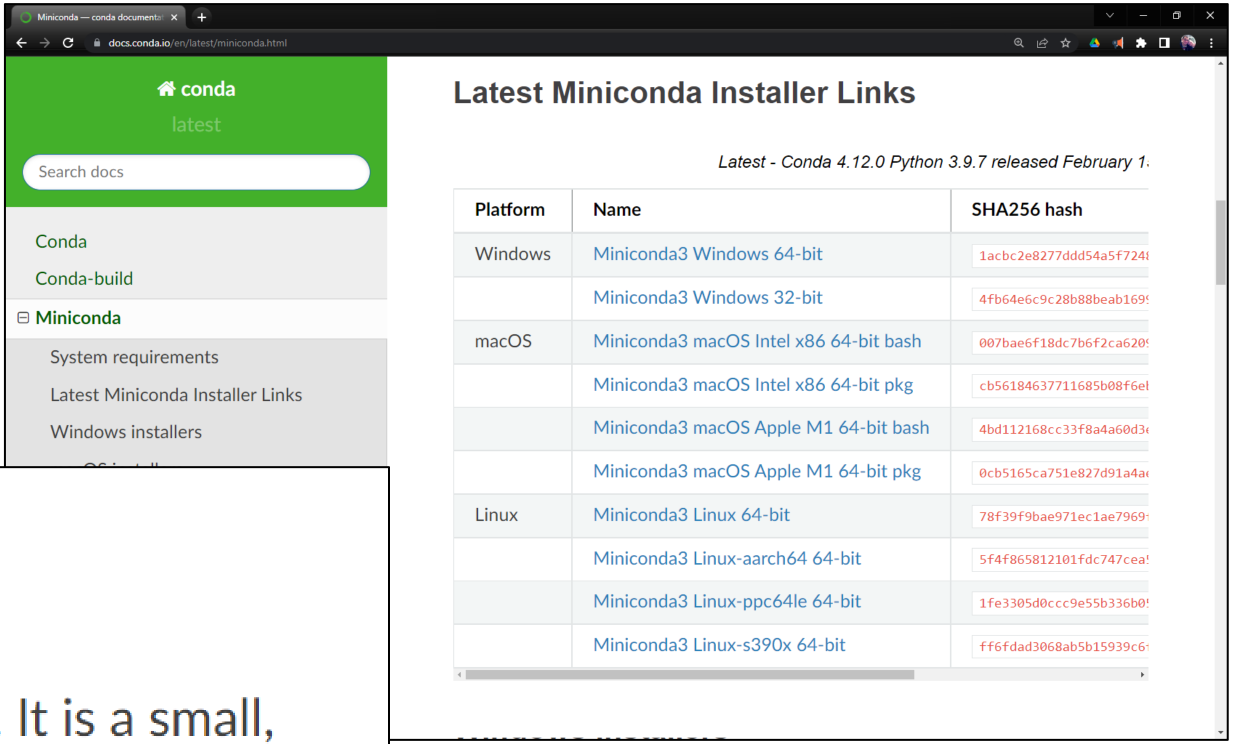
```
bash-3.2$ conda install pandas
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /opt/anaconda3

  added / updated specs:
    - pandas


The following packages will be downloaded:

  package                    |            build
  ---------------------------|-----------------
  pandas-1.4.1               |   py39he9d5cce_1        9.4 MB
  ------------------------------------------------------------
                                           Total:        9.4 MB

The following packages will be UPDATED:
```



Miniconda

Miniconda = Python + Conda

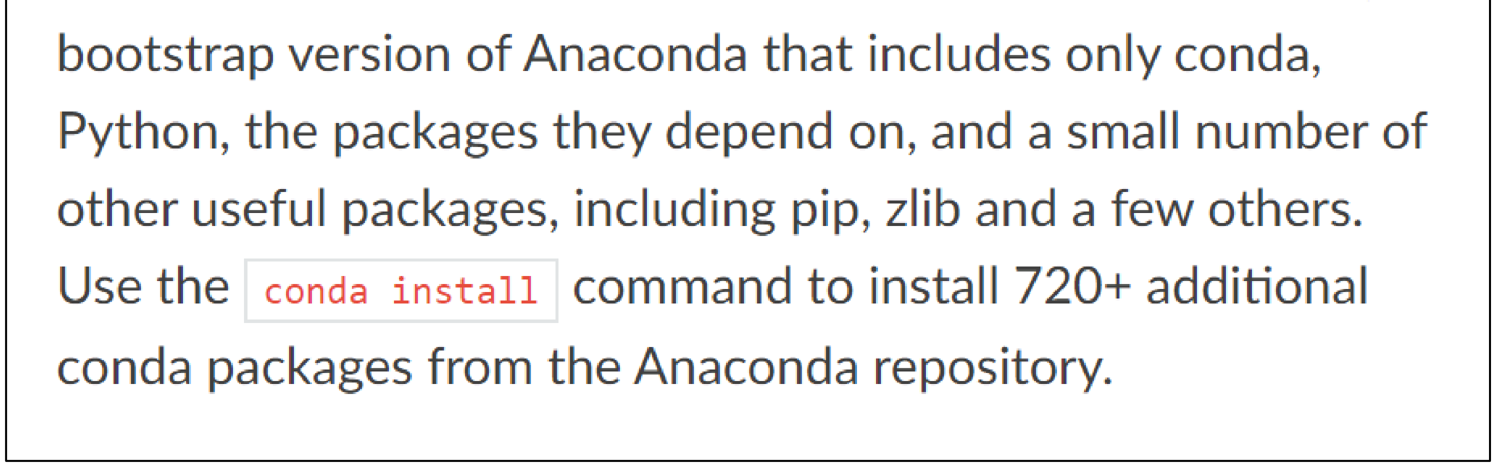
https://docs.conda.io/en/latest/miniconda.html


Miniconda — conda documenta
docs.conda.io/en/latest/miniconda.html
conda
latest
Search docs
Conda
Conda-build
Miniconda
System requirements
Latest Miniconda Installer Links
Windows installers

Latest Miniconda Installer Links

Latest - Conda 4.12.0 Python 3.9.7 released February 1

Platform	Name	SHA256 hash
Windows	Miniconda3 Windows 64-bit	1acbc2e8277ddd54a5f724
	Miniconda3 Windows 32-bit	4fb64e6c9c28b88beab169
macOS	Miniconda3 macOS Intel x86 64-bit bash	007bae6f18dc7b6f2ca620
	Miniconda3 macOS Intel x86 64-bit pkg	cb5618463771l685b08f6e
	Miniconda3 macOS Apple M1 64-bit bash	4bd112168cc33f8a4a60d3
	Miniconda3 macOS Apple M1 64-bit pkg	0cb5165ca751e827d91a4a
Linux	Miniconda3 Linux 64-bit	78f39f9bae971ec1ae7969
	Miniconda3 Linux-aarch64 64-bit	5f4f865812101fdc747cea
	Miniconda3 Linux-ppc64le 64-bit	1fe3305d0ccc9e55b336b0
	Miniconda3 Linux-s390x 64-bit	ff6fdad3068ab5b15939c6



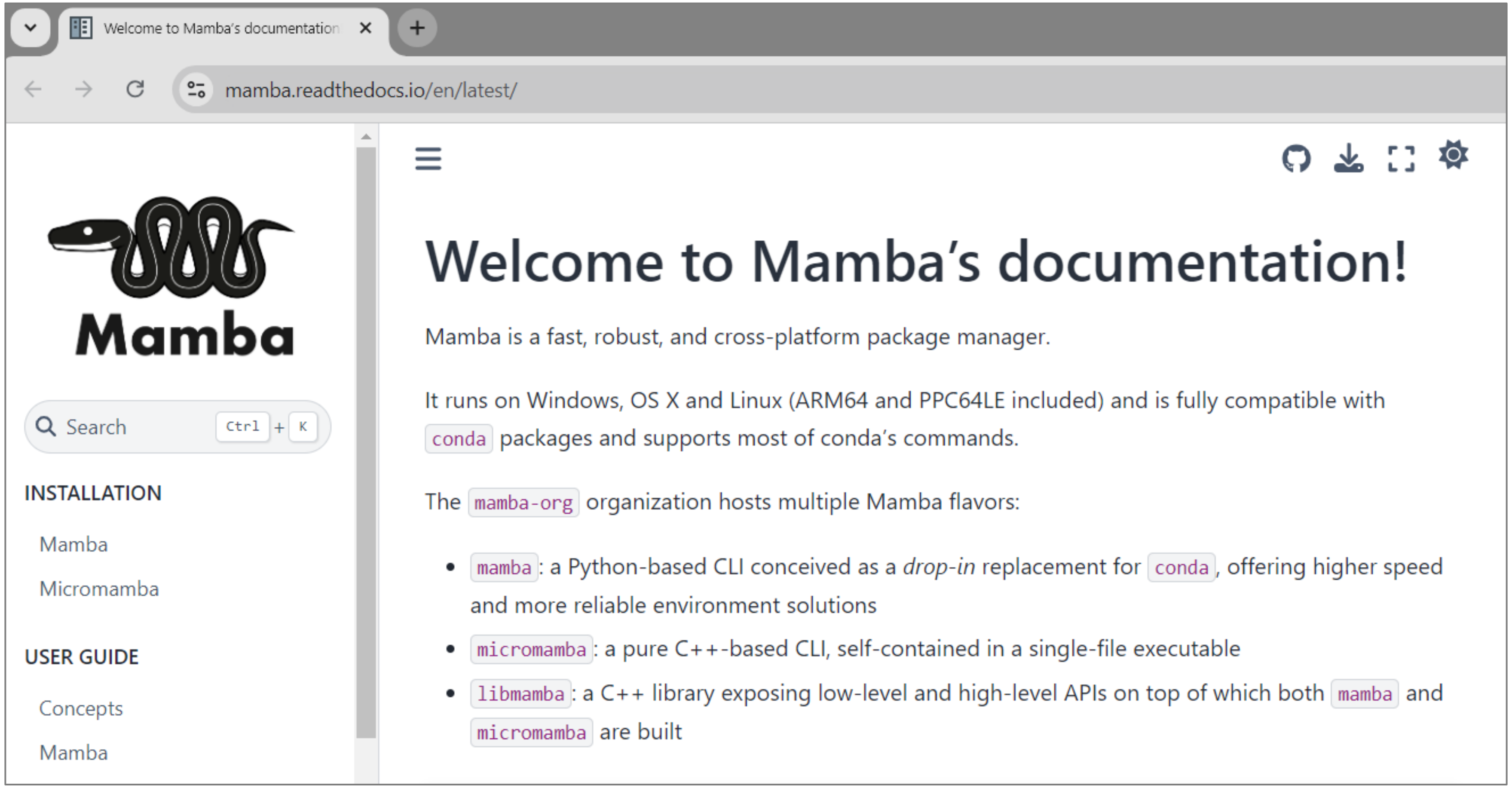
Miniconda

Miniconda is a free minimal installer for conda. It is a small, bootstrap version of Anaconda that includes only conda, Python, the packages they depend on, and a small number of other useful packages, including pip, zlib and a few others. Use the `conda install` command to install 720+ additional conda packages from the Anaconda repository.

Mamba

An efficient implementation of Conda

Not recommended in this course



Welcome to Mamba's documentation!

Mamba is a fast, robust, and cross-platform package manager.

It runs on Windows, OS X and Linux (ARM64 and PPC64LE included) and is fully compatible with `conda` packages and supports most of conda's commands.

The `mamba-org` organization hosts multiple Mamba flavors:

- `mamba`: a Python-based CLI conceived as a *drop-in* replacement for `conda`, offering higher speed and more reliable environment solutions
- `micromamba`: a pure C++-based CLI, self-contained in a single-file executable
- `libmamba`: a C++ library exposing low-level and high-level APIs on top of which both `mamba` and `micromamba` are built

https://mamba.readthedocs.io/en/latest

Lecture plan

- ✓ Python features
- ✓ Installing Python(s) on your PC
- ✓ Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- ✓ Comments & Indentation
- Python packages and environments
  - PIP, Conda, Venv

Python environments

“Global” environments

Created when you install Python (including ***base*** Conda env)

C:/Users/alexey.Larionov/AppData/Local/Programs/Python/Python39/…

Never recommended to use directly

CONDA environments created by user

C:/Users/alexey.larionov/Miniconda3/…

/envs/**emoji_env**

/**matplotlib**

/**cutatapt**

/**my_project_1**

/**my_project_2**

/**…**

```
(base)> conda env list
(base)> conda create ...
(base)> conda activate ...
(emoji_env)> conda install ...
(emoji_env)> ...
(emoji_env)> conda deactivate
(base)> ...
```

Recommended to use during this course

“Local” venv environment

Within project folder, created by venv module

C:/Users/alexey.larionov/my_project/venv/…

```
> python -m venv venv
> venv/Scripts/activate
(venv)> pip install ...
(venv)> ...
(venv)> venv/Scripts/deactivate
> ...
```

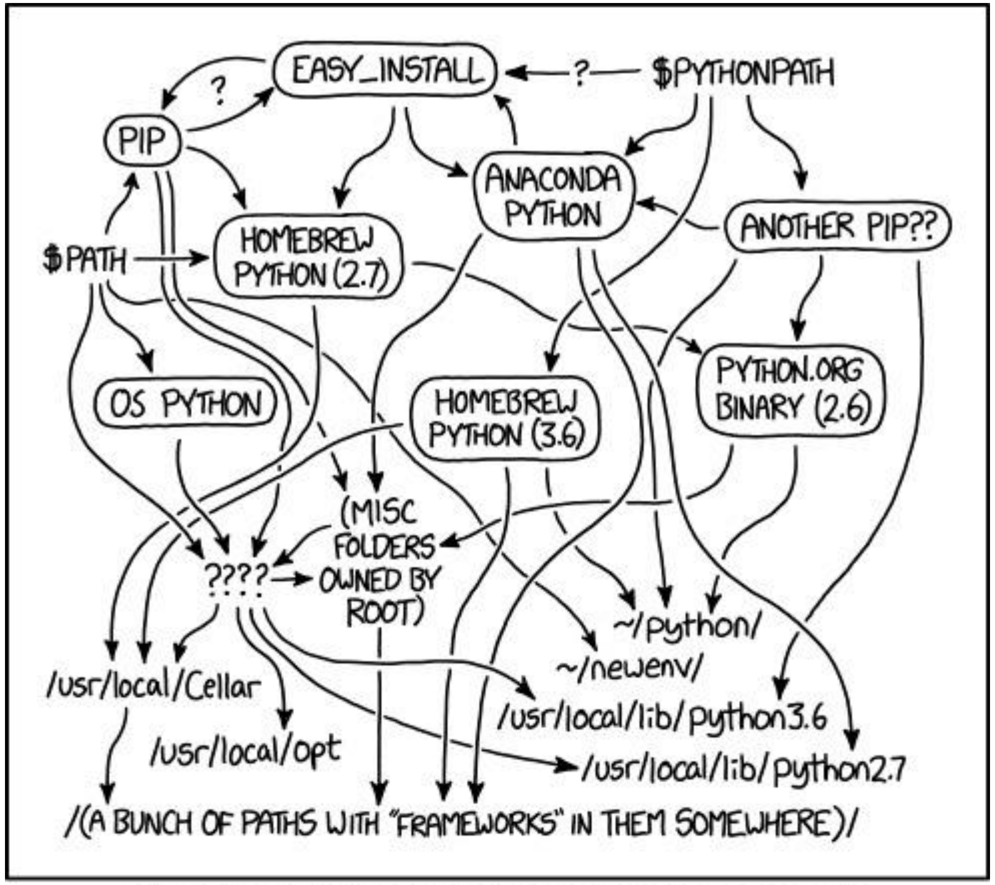
Not recommended during this course
unless you already know what you are doing

Python environments

- Using environments allows avoiding conflicts between Python packages

- It’s **NOT** recommended to install bioinformatics packages into global (system) environments !!!

- A good practice is to create a separate environment for each bioinformatics project



MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.*

* Superfund sites are “polluted locations requiring a long-term response to clean up hazardous materials” (Wikipedia)

https://www.explainxkcd.com/wiki/index.php/1987:_Python_Environment

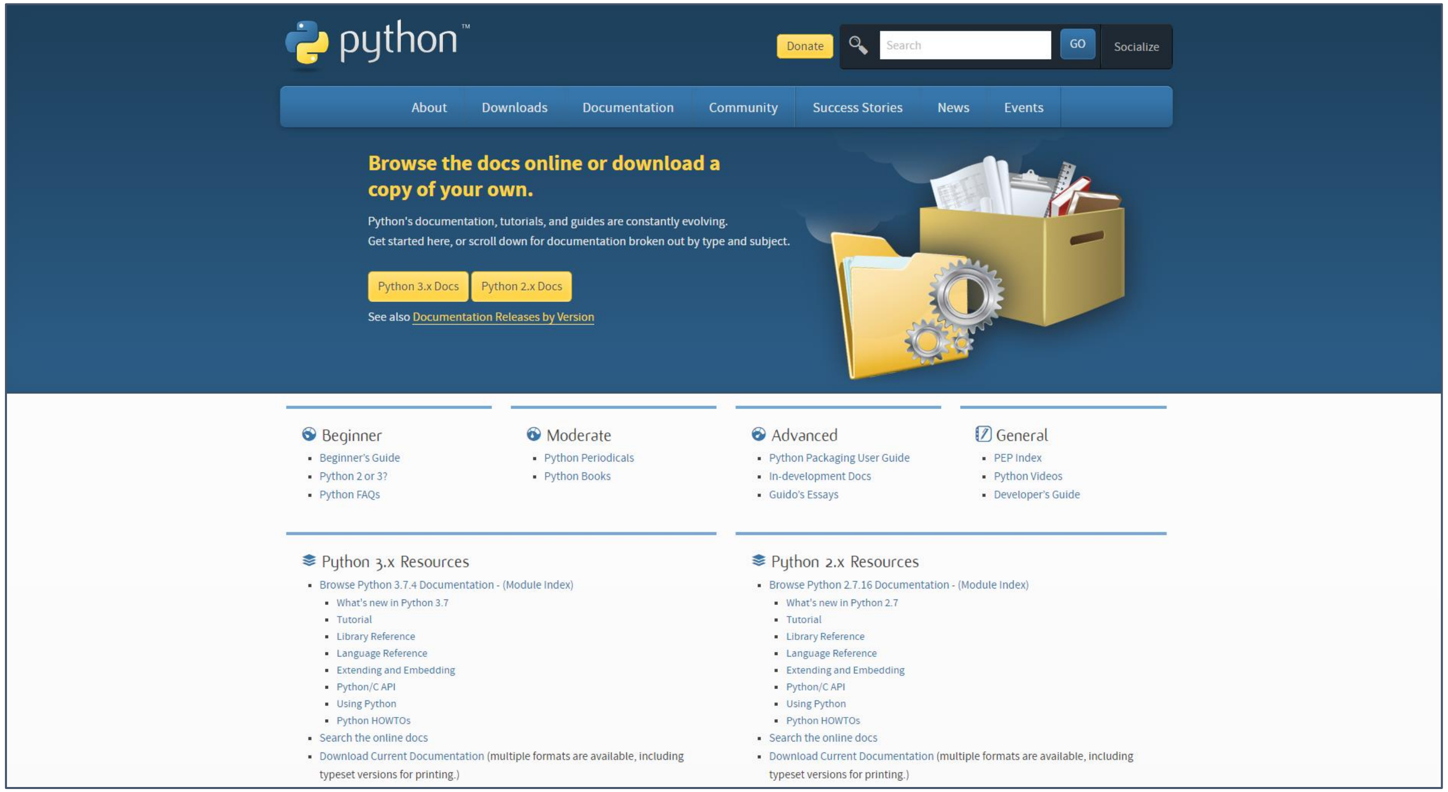
Cranfield
Environment
and Agrifood

Lecture plan

DONE

- ✓ Python features
- ✓ Installing Python(s) on your PC
- ✓ Using Python
  - Interactive use in terminal & IDLE
  - Scripts & Python IDEs
  - Hello Word in VS Code
- ✓ Comments & Indentation
- ✓ Python packages and environments
  - PIP, Conda, Venv

https://www.python.org/doc/



Browse the docs online or download a copy of your own.

Python's documentation, tutorials, and guides are constantly evolving.
Get started here, or scroll down for documentation broken out by type and subject.

See also Documentation Releases by Version

Beginner

- Beginner's Guide
- Python 2 or 3?
- Python FAQs

Moderate

- Python Periodicals
- Python Books

Advanced

- Python Packaging User Guide
- In-development Docs
- Guido's Essays

General

- PEP Index
- Python Videos
- Developer's Guide

Python 3.x Resources

- Browse Python 3.7.4 Documentation - (Module Index)
  - What's new in Python 3.7
  - Tutorial
  - Library Reference
  - Language Reference
  - Extending and Embedding
  - Python/C API
  - Using Python
  - Python HOWTOs
- Search the online docs
- Download Current Documentation (multiple formats are available, including typeset versions for printing.)

Python 2.x Resources

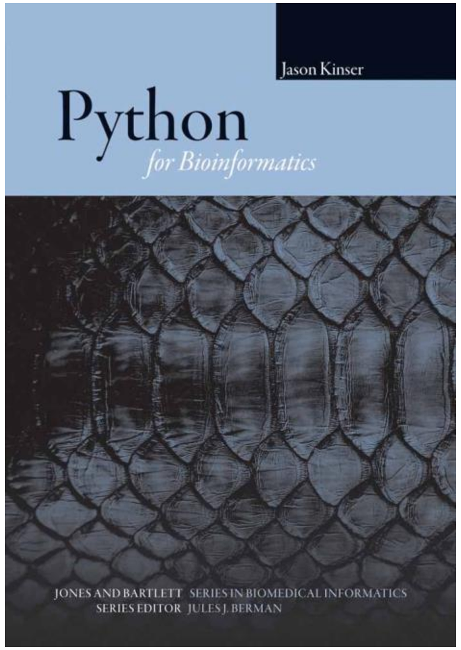
- Browse Python 2.7.16 Documentation - (Module Index)
  - What's new in Python 2.7
  - Tutorial
  - Library Reference
  - Language Reference
  - Extending and Embedding
  - Python/C API
  - Using Python
  - Python HOWTOs
- Search the online docs
- Download Current Documentation (multiple formats are available, including typeset versions for printing.)

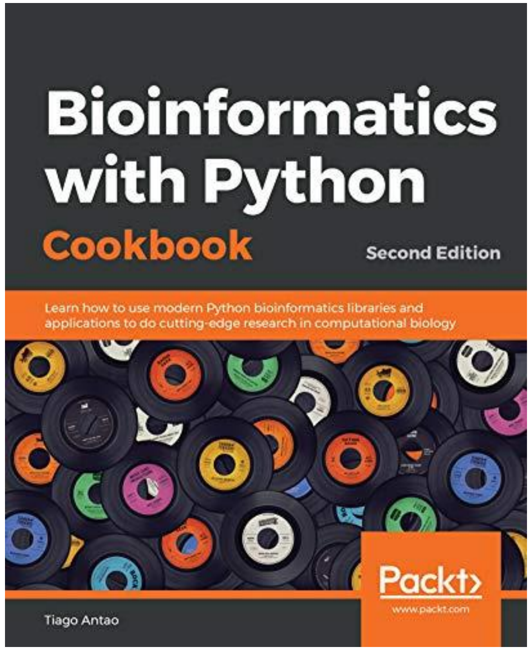
https://wiki.python.org/moin/BeginnersGuide

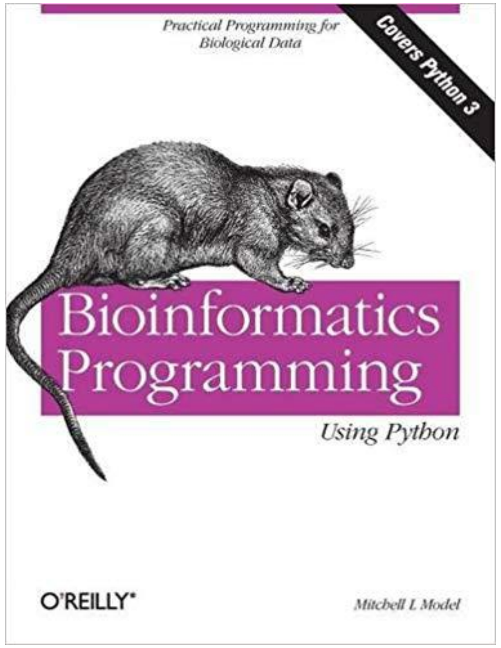
https://realpython.com

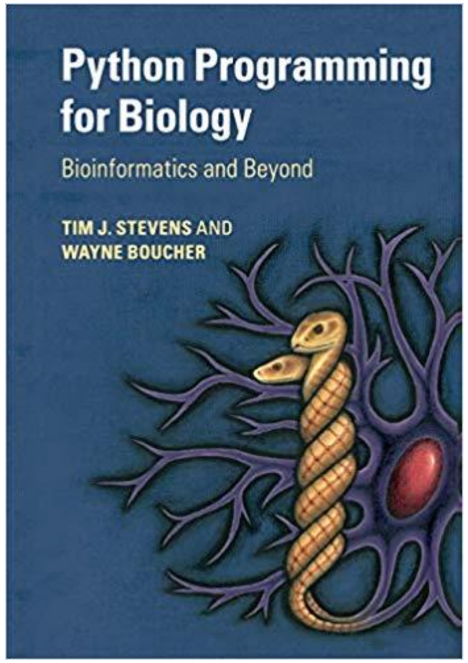
https://realpython.com/python-beginner-tips

Some books (see more in the Reading List)


Jason Kinser
Python
for Bioinformatics
JONES AND BARTLETT SERIES IN BIOMEDICAL INFORMATICS
SERIES EDITOR JULES J. BERMAN


Bioinformatics with Python
Cookbook
Second Edition
Learn how to use modern Python bioinformatics libraries and applications to do cutting-edge research in computational biology
Packt>
www.packt.com
Tiago Antao


Practical Programming for Biological Data
Covers Python 3
Bioinformatics Programming
Using Python
O'REILLY
Mitchell L Model



Many other resources on the Internet …

e.g. LinkedIn-Learning course “Python Essential Training”
https://www.linkedin.com/learning/python-essential-training-18764650
(set of video lectures and tasks, available through Cranfield LinkedIn)

Questions