Berkeley
MSSE
MOLECULAR SCIENCE &
SOFTWARE ENGINEERING

# Lecture 01:

## Introduction to Python and Command Line Basics

Markus Hohle

University California, Berkeley

**Python for Molecular Sciences**

MSSE 272, 3 Units

Outline

- What is an “environment”?

- installing **WSL** instructions here

- installing ***miniconda***

- guide through **Jupyter**

- installing ***Spyder***

- guide through ***Spyder***

Outline

- **What is an “environment”?**
- installing **WSL**
- installing ***miniconda***
- guide through **Jupyter**
- installing ***Spyder***
- guide through ***Spyder***

# Introduction

before we can start coding…we need an **environment**

## your computer

### Operational System (OS)

interface between the computer and all programs,
usually, Windows or Linux

# Introduction

before we can start coding…we need an **environment**

## your computer

### Operational System (OS)

interface between the computer and all programs,
usually Windows or **Linux**

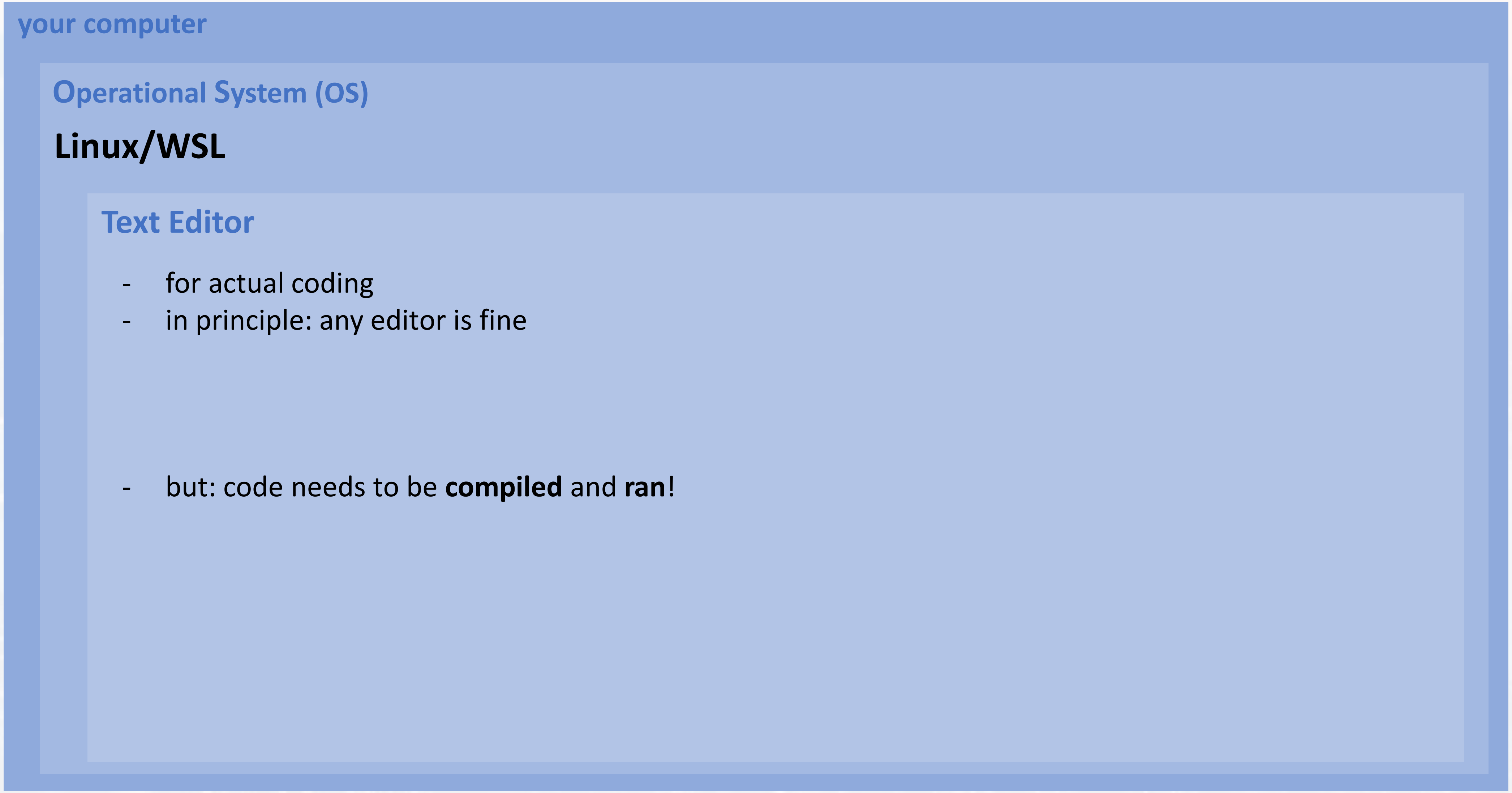
- in Computer Science, Software Engineering, Physics etc: some version of Linux/Unix

- for Windows:

**W**indows **S**ubsystem for **L**inux (**WSL**), see later

# Introduction

before we can start coding…we need an **environment**



your computer

Operational System (OS)

**Linux/WSL**

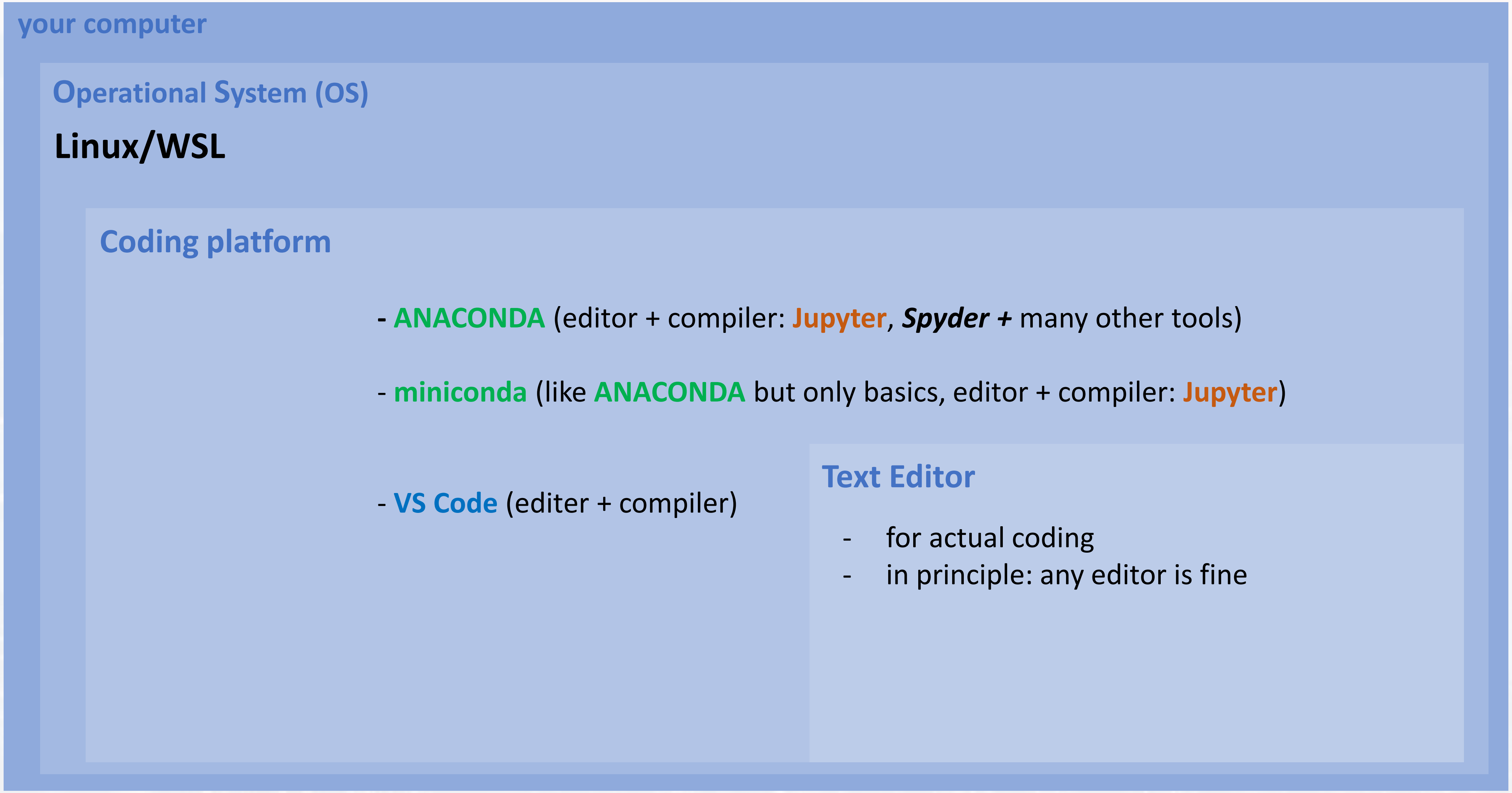
Text Editor

- for actual coding
- in principle: any editor is fine

- but: code needs to be **compiled** and **ran**!

# Introduction

before we can start coding…we need an **environment**



your computer

Operational System (OS)

**Linux/WSL**

Coding platform

- **ANACONDA** (editor + compiler: **Jupyter**, ***Spyder +*** many other tools)

- **miniconda** (like **ANACONDA** but only basics, editor + compiler: **Jupyter**)

- **VS Code** (editer + compiler)

Text Editor

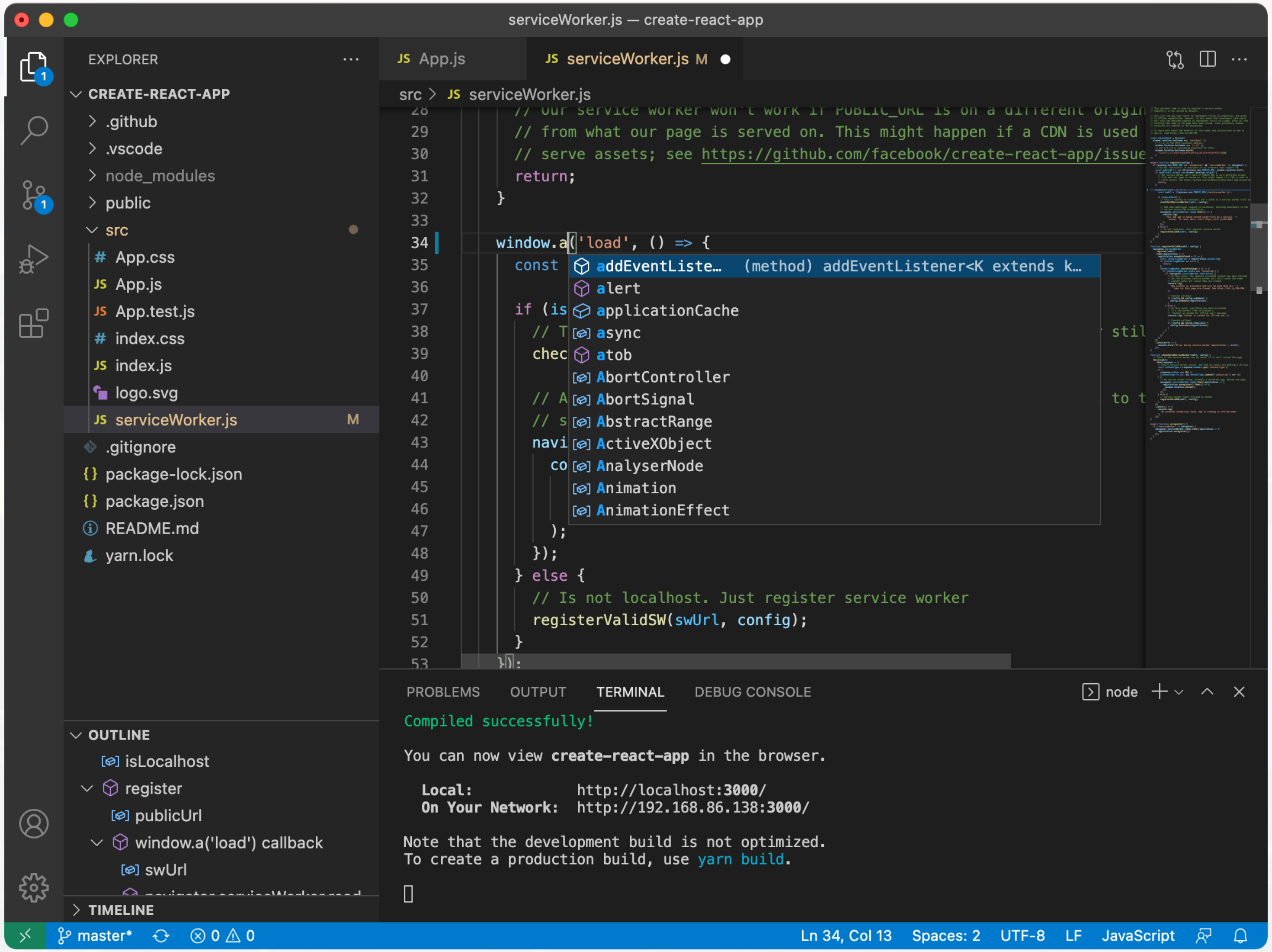
- for actual coding
- in principle: any editor is fine

# Introduction

## Environment

before we can start coding…we need an **environment**

**VS Code** (see lecture No 9)



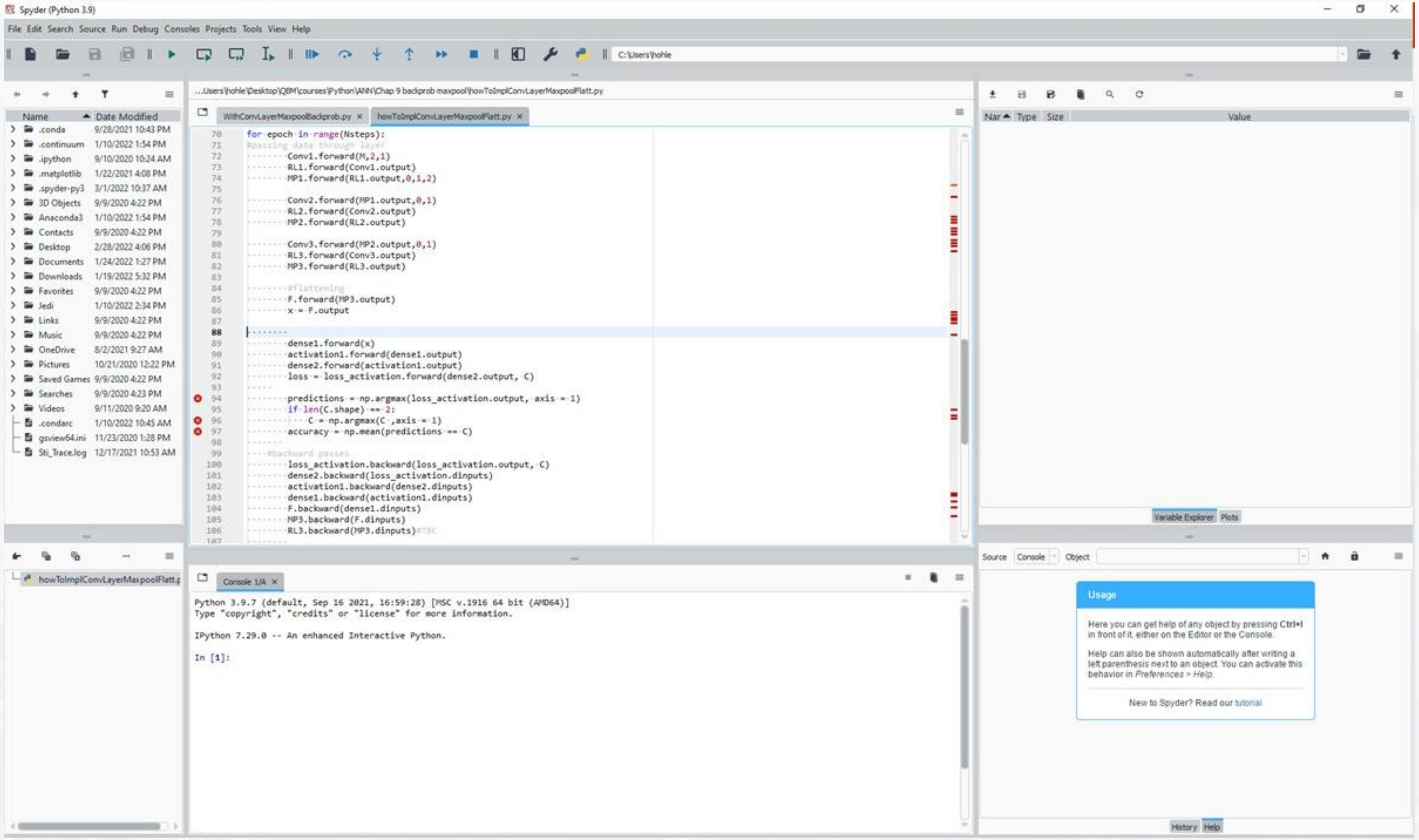
```
      // our service worker won't work if PUBLIC_URL is on a different origin
      // from what our page is served on. This might happen if a CDN is used
      // serve assets; see https://github.com/facebook/create-react-app/issue
      return;
    }

    window.a('load', () => {
      const

      if (is
        // T                                                          stil
        chec

        // A                                                          to t
        // s
        navi
          co


          );
        });
      } else {
        // Is not localhost. Just register service worker
        registerValidSW(swUrl, config);
      }
    });
```

```
Compiled successfully!

You can now view create-react-app in the browser.

  Local:            http://localhost:3000/
  On Your Network:  http://192.168.86.138:3000/

Note that the development build is not optimized.
To create a production build, use yarn build.
```

# Introduction

before we can start coding…we need an **environment**

***Spyder*** (see later)


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\hohle
...Users\hohle\Desktop\QBM\courses\Python\ANN\Chap 9 backprob maxpool\howToImplConvLayerMaxpoolFlatt.py
WithConvLayerMaxpoolBackprob.py
howToImplConvLayerMaxpoolFlatt.py
Nar Type Size Value
Variable Explorer
Plots
howToImplConvLayerMaxpoolFlatt.p
Console 1/A
Source
Console
Object
History
Help

| Name | Date Modified |
|---|---|
| .conda | 9/28/2021 10:43 PM |
| .continuum | 1/10/2022 1:54 PM |
| .ipython | 9/10/2020 10:24 AM |
| .matplotlib | 1/22/2021 4:08 PM |
| .spyder-py3 | 3/1/2022 10:37 AM |
| 3D Objects | 9/9/2020 4:22 PM |
| Anaconda3 | 1/10/2022 1:54 PM |
| Contacts | 9/9/2020 4:22 PM |
| Desktop | 2/28/2022 4:06 PM |
| Documents | 1/24/2022 1:27 PM |
| Downloads | 1/19/2022 5:32 PM |
| Favorites | 9/9/2020 4:22 PM |
| Jedi | 1/10/2022 2:34 PM |
| Links | 9/9/2020 4:22 PM |
| Music | 9/9/2020 4:22 PM |
| OneDrive | 8/2/2021 9:27 AM |
| Pictures | 10/21/2020 12:22 PM |
| Saved Games | 9/9/2020 4:22 PM |
| Searches | 9/9/2020 4:23 PM |
| Videos | 9/11/2020 9:20 AM |
| .condarc | 1/10/2022 10:45 AM |
| gsview64.ini | 11/23/2020 1:28 PM |
| Sti_Trace.log | 12/17/2021 10:53 AM |

```python
for epoch in range(Nsteps):
#passing data through layer
        Conv1.forward(M,2,1)
        RL1.forward(Conv1.output)
        MP1.forward(RL1.output,0,1,2)

        Conv2.forward(MP1.output,0,1)
        RL2.forward(Conv2.output)
        MP2.forward(RL2.output)

        Conv3.forward(MP2.output,0,1)
        RL3.forward(Conv3.output)
        MP3.forward(RL3.output)

        #flattening
        F.forward(MP3.output)
        x = F.output

        
        dense1.forward(x)
        activation1.forward(dense1.output)
        dense2.forward(activation1.output)
        loss = loss_activation.forward(dense2.output, C)
    
        predictions = np.argmax(loss_activation.output, axis = 1)
        if len(C.shape) == 2:
            C = np.argmax(C ,axis = 1)
        accuracy = np.mean(predictions == C)
        
    #backward passes
        loss_activation.backward(loss_activation.output, C)
        dense2.backward(loss_activation.dinputs)
        activation1.backward(dense2.dinputs)
        dense1.backward(activation1.dinputs)
        F.backward(dense1.dinputs)
        MP3.backward(F.dinputs)
        RL3.backward(MP3.dinputs)#TBC
        
```

```
Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.29.0 -- An enhanced Interactive Python.

In [1]:
```

Usage

Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

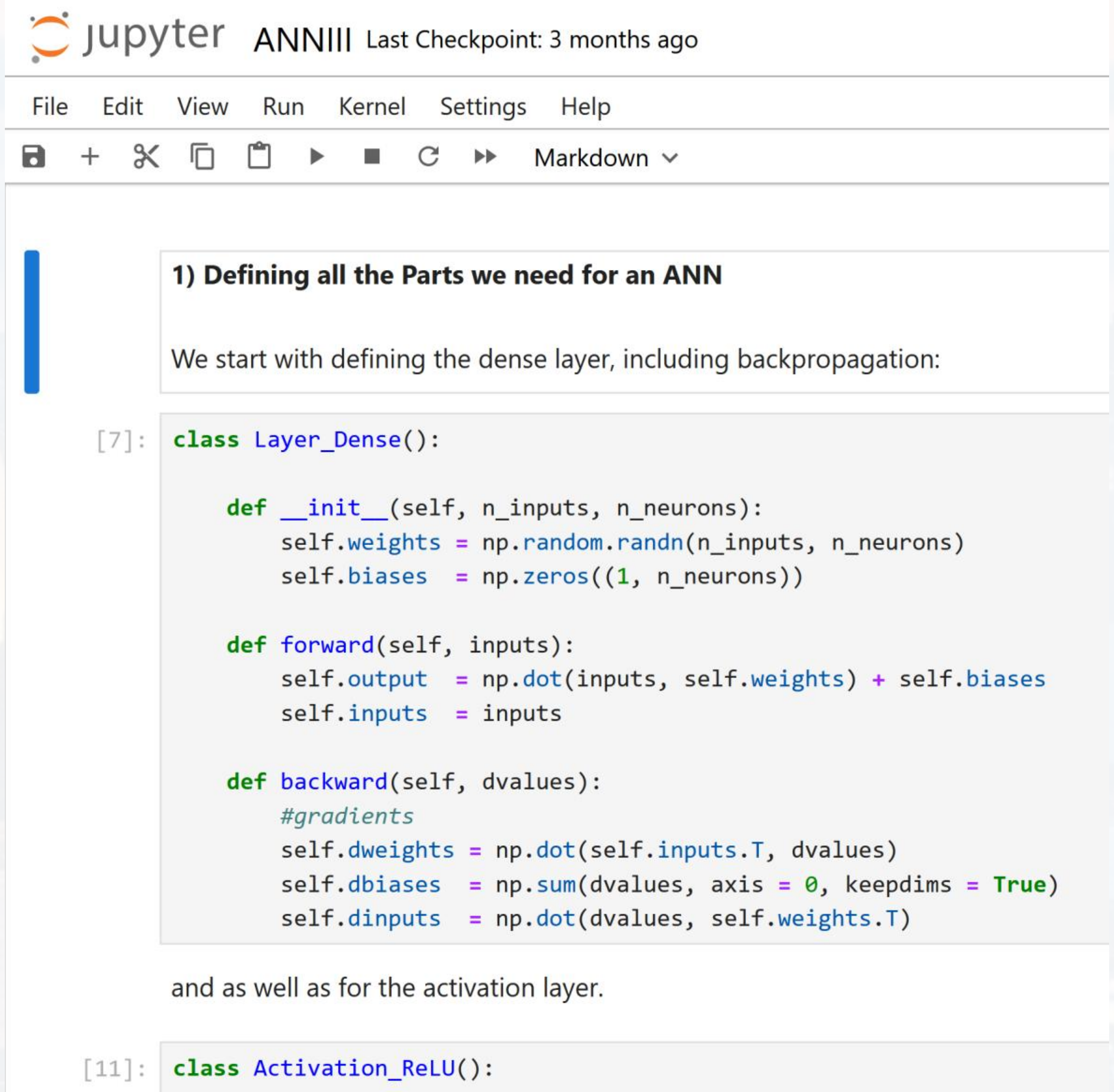
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

# Introduction

before we can start coding…we need an **environment**

**Jupyter** (see later)



**1) Defining all the Parts we need for an ANN**

We start with defining the dense layer, including backpropagation:

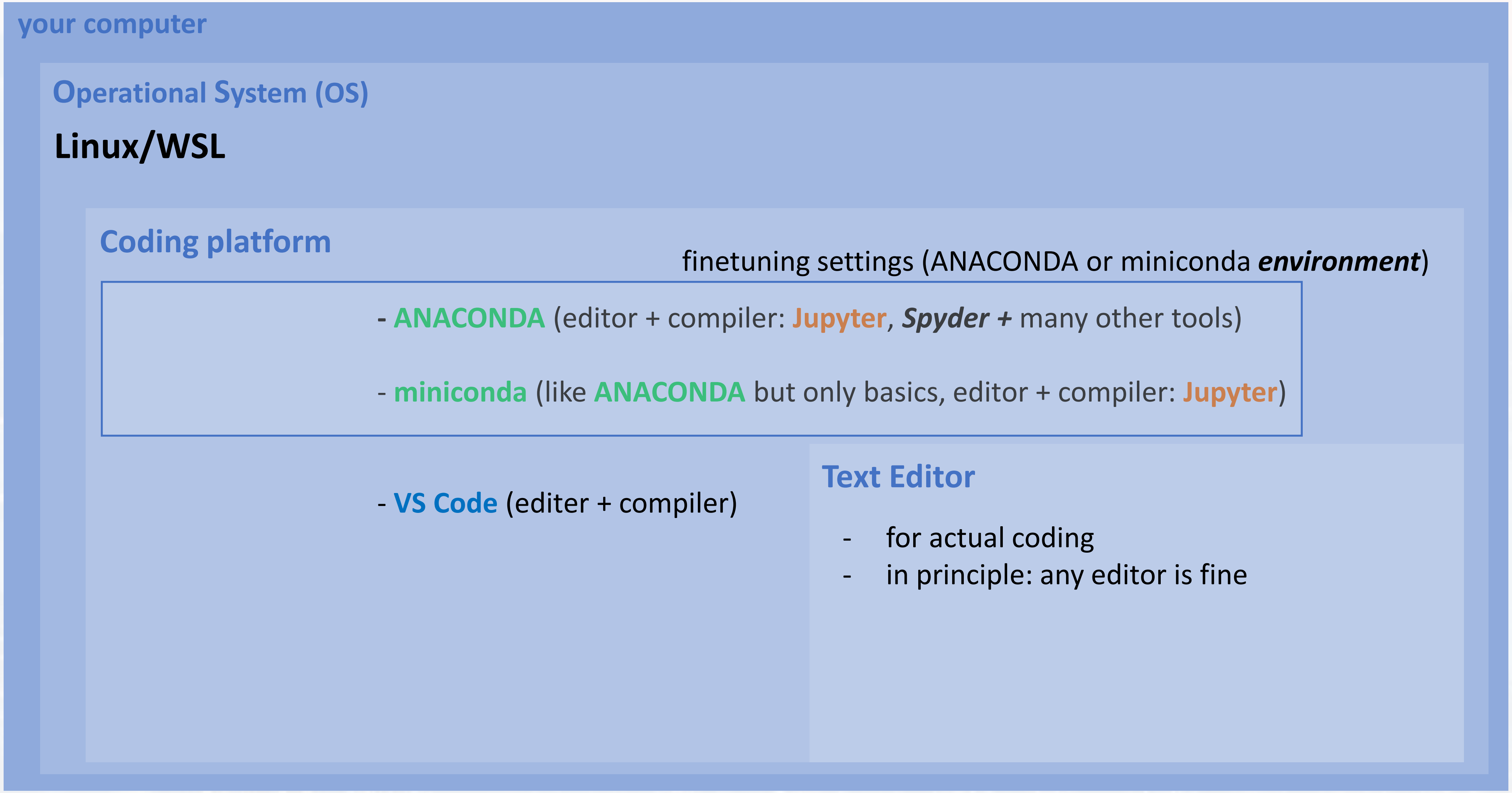
```
[7]: class Layer_Dense():

         def __init__(self, n_inputs, n_neurons):
             self.weights = np.random.randn(n_inputs, n_neurons)
             self.biases  = np.zeros((1, n_neurons))

         def forward(self, inputs):
             self.output  = np.dot(inputs, self.weights) + self.biases
             self.inputs  = inputs

         def backward(self, dvalues):
             #gradients
             self.dweights = np.dot(self.inputs.T, dvalues)
             self.dbiases  = np.sum(dvalues, axis = 0, keepdims = True)
             self.dinputs  = np.dot(dvalues, self.weights.T)
```

and as well as for the activation layer.

```
[11]: class Activation_ReLU():
```

# Introduction

before we can start coding…we need an **environment**



your computer

Operational System (OS)

**Linux/WSL**

Coding platform

finetuning settings (ANACONDA or miniconda ***environment***)

- **ANACONDA** (editor + compiler: **Jupyter**, ***Spyder +*** many other tools)

- **miniconda** (like **ANACONDA** but only basics, editor + compiler: **Jupyter**)

- **VS Code** (editer + compiler)

Text Editor

- for actual coding
- in principle: any editor is fine

Outline

- What is an "environment"?

- **installing WSL**

- installing ***miniconda***

- guide through **Jupyter**

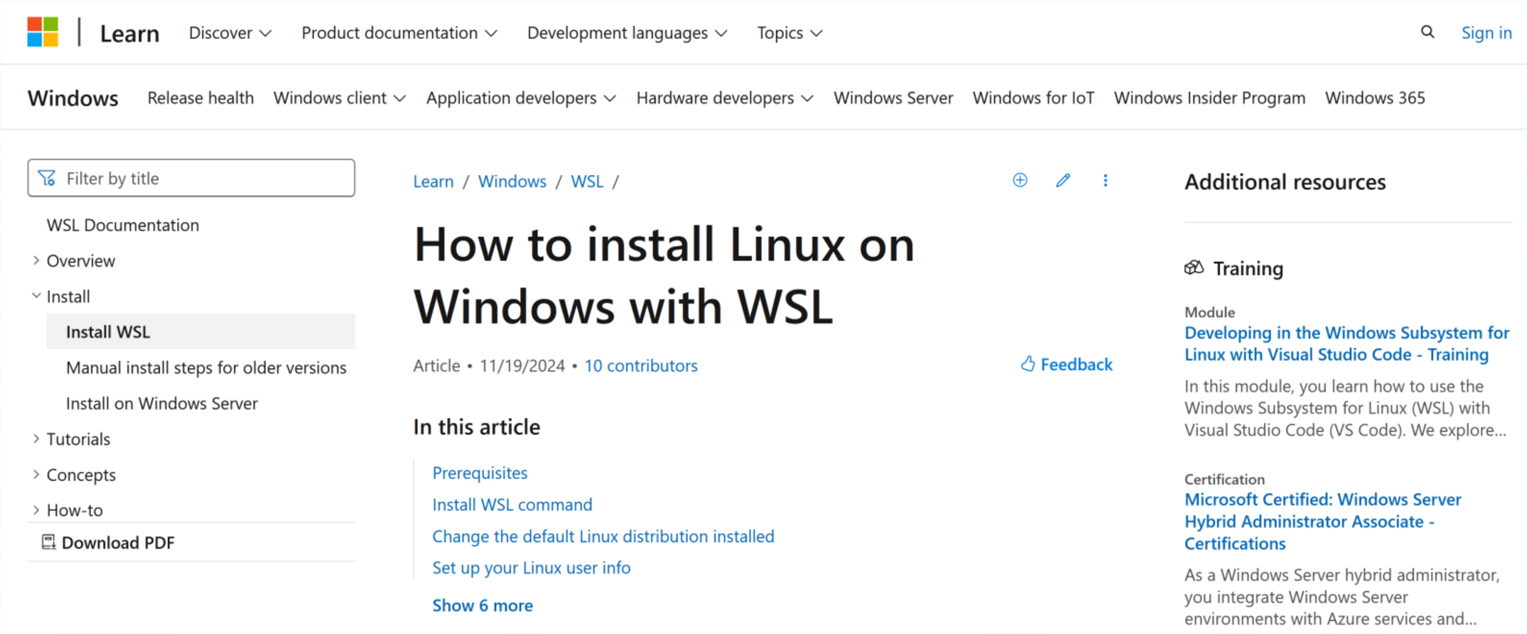
- installing ***Spyder***

- guide through ***Spyder***

# Introduction

**W**indows **S**ubsystem for **L**inux (**WSL**)

follow the instructions here



## How to install Linux on Windows with WSL

Article • 11/19/2024 • 10 contributors

In this article

- Prerequisites
- Install WSL command
- Change the default Linux distribution installed
- Set up your Linux user info

Show 6 more

Additional resources

Training

Module

Developing in the Windows Subsystem for Linux with Visual Studio Code - Training

In this module, you learn how to use the Windows Subsystem for Linux (WSL) with Visual Studio Code (VS Code). We explore...

Certification

Microsoft Certified: Windows Server Hybrid Administrator Associate - Certifications

As a Windows Server hybrid administrator, you integrate Windows Server environments with Azure services and...

check out the video ***01a_Installing_WSL*** in bcourses

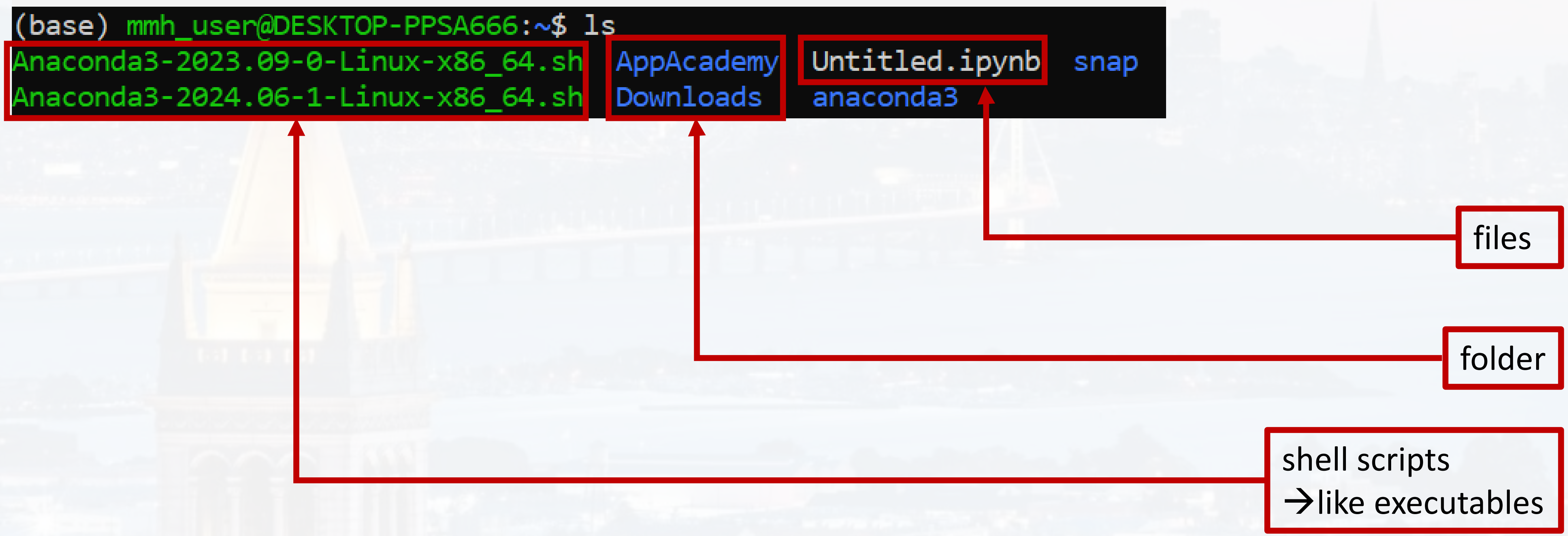
# Introduction

**Linux commands**

list your files and folders

`ls`

check out the video ***01b_Quick_Guide_Unix_Commands*** on bcourses



```
(base) mmh_user@DESKTOP-PPSA666:~$ ls
Anaconda3-2023.09-0-Linux-x86_64.sh  AppAcademy  Untitled.ipynb  snap
Anaconda3-2024.06-1-Linux-x86_64.sh  Downloads   anaconda3
```

files

folder

shell scripts
→like executables

# Introduction

**Linux commands**

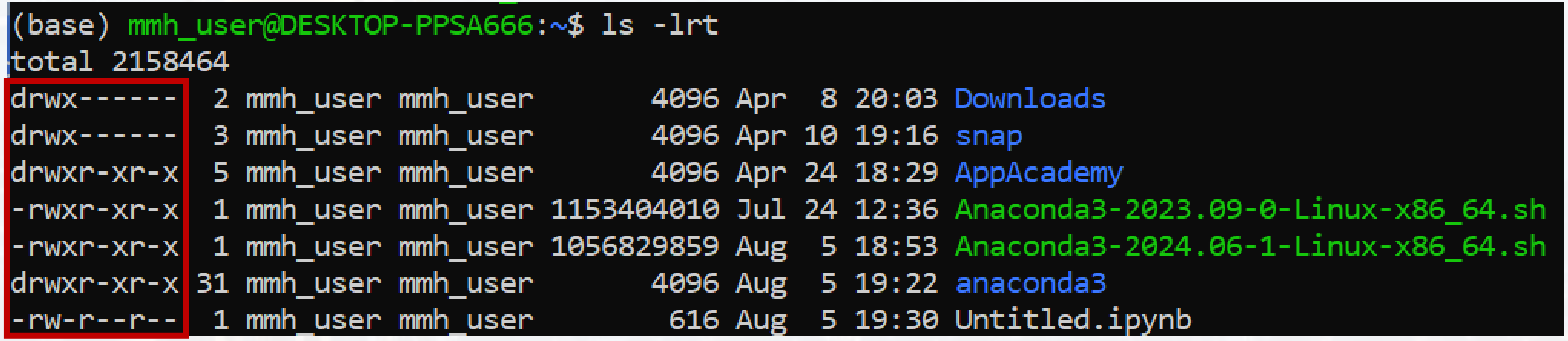
check out the video ***01b_Quick_Guide_Unix_Commands*** on bcourses

list your files and folders

`ls      -lrt`

called *flags*

- **`l`** stands for *long*: shows a file with all permissions and properties, each per line
- **`r`** stands for *reverse* (oldest first)
- **`t`** stands for the *time* flag (as criterium for r)



```
(base) mmh_user@DESKTOP-PPSA666:~$ ls -lrt
total 2158464
drwx------  2 mmh_user mmh_user       4096 Apr  8 20:03 Downloads
drwx------  3 mmh_user mmh_user       4096 Apr 10 19:16 snap
drwxr-xr-x  5 mmh_user mmh_user       4096 Apr 24 18:29 AppAcademy
-rwxr-xr-x  1 mmh_user mmh_user 1153404010 Jul 24 12:36 Anaconda3-2023.09-0-Linux-x86_64.sh
-rwxr-xr-x  1 mmh_user mmh_user 1056829859 Aug  5 18:53 Anaconda3-2024.06-1-Linux-x86_64.sh
drwxr-xr-x 31 mmh_user mmh_user       4096 Aug  5 19:22 anaconda3
-rw-r--r--  1 mmh_user mmh_user        616 Aug  5 19:30 Untitled.ipynb
```

# Introduction

**Linux commands**

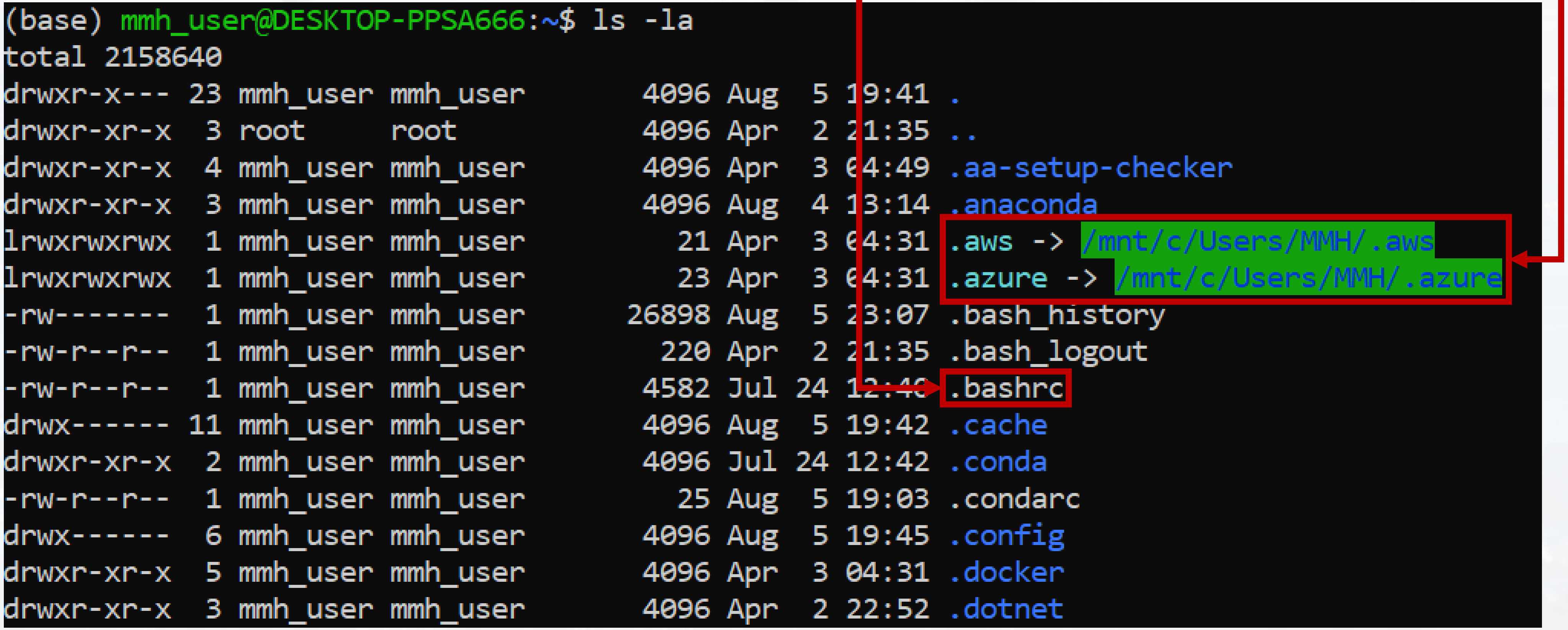
check out the video ***01b_Quick_Guide_Unix_Commands*** on bcourses

but these are only those files you *see.*
try:

```
ls      -la
```

will be important (see later)

pointer/ links/aliases



```
(base) mmh_user@DESKTOP-PPSA666:~$ ls -la
total 2158640
drwxr-x--- 23 mmh_user mmh_user      4096 Aug  5 19:41 .
drwxr-xr-x  3 root     root          4096 Apr  2 21:35 ..
drwxr-xr-x  4 mmh_user mmh_user      4096 Apr  3 04:49 .aa-setup-checker
drwxr-xr-x  3 mmh_user mmh_user      4096 Aug  4 13:14 .anaconda
lrwxrwxrwx  1 mmh_user mmh_user        21 Apr  3 04:31 .aws -> /mnt/c/Users/MMH/.aws
lrwxrwxrwx  1 mmh_user mmh_user        23 Apr  3 04:31 .azure -> /mnt/c/Users/MMH/.azure
-rw-------  1 mmh_user mmh_user     26898 Aug  5 23:07 .bash_history
-rw-r--r--  1 mmh_user mmh_user       220 Apr  2 21:35 .bash_logout
-rw-r--r--  1 mmh_user mmh_user      4582 Jul 24 12:40 .bashrc
drwx------ 11 mmh_user mmh_user      4096 Aug  5 19:42 .cache
drwxr-xr-x  2 mmh_user mmh_user      4096 Jul 24 12:42 .conda
-rw-r--r--  1 mmh_user mmh_user        25 Aug  5 19:03 .condarc
drwx------  6 mmh_user mmh_user      4096 Aug  5 19:45 .config
drwxr-xr-x  5 mmh_user mmh_user      4096 Apr  3 04:31 .docker
drwxr-xr-x  3 mmh_user mmh_user      4096 Apr  2 22:52 .dotnet
```

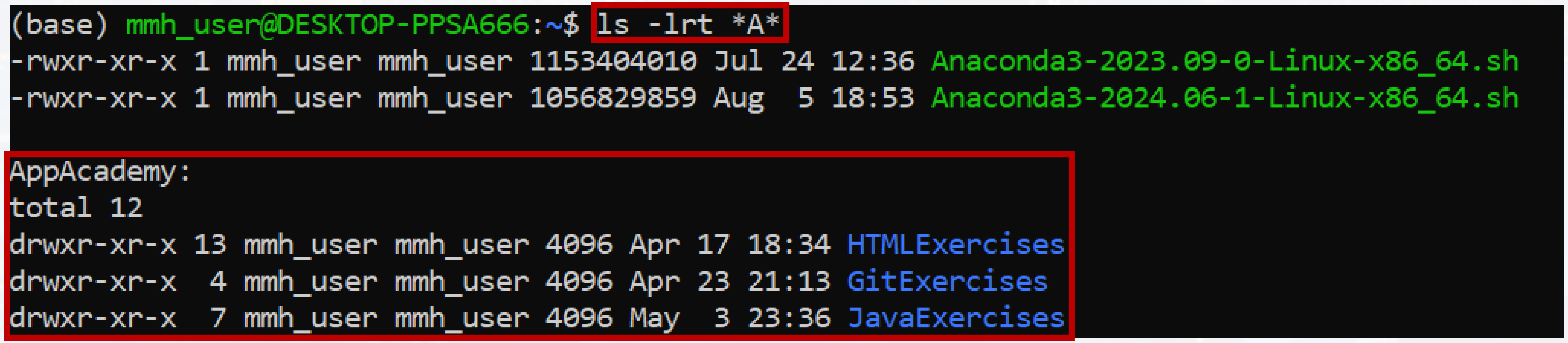
# Introduction

**Linux commands**

check out the video ***01b_Quick_Guide_Unix_Commands*** on bcourses

You can search for files/folders with particular substrings using a “wildcard”

```
ls      *.py
ls      *py*
```



```
(base) mmh_user@DESKTOP-PPSA666:~$ ls -lrt *A*
-rwxr-xr-x 1 mmh_user mmh_user 1153404010 Jul 24 12:36 Anaconda3-2023.09-0-Linux-x86_64.sh
-rwxr-xr-x 1 mmh_user mmh_user 1056829859 Aug  5 18:53 Anaconda3-2024.06-1-Linux-x86_64.sh

AppAcademy:
total 12
drwxr-xr-x 13 mmh_user mmh_user 4096 Apr 17 18:34 HTMLExercises
drwxr-xr-x  4 mmh_user mmh_user 4096 Apr 23 21:13 GitExercises
drwxr-xr-x  7 mmh_user mmh_user 4096 May  3 23:36 JavaExercises
```

# Introduction

**Linux commands** check out the video ***01b_Quick_Guide_Unix_Commands*** on bcourses

changing your directory

| | |
|---|---|
| `cd` | always leads back to the home directory |
| `cd ../` | one level up |
| `cd ../my_dir` | one level up, down to my_dir |
| `cd another/dir` | one level down to dir |
| `mkdir test` | creating the new directory *test* |
| `rm -r test` | removing the directory using the flag *r* (here: recursively) |
| `rm any_file` | when removing a file, no flag is needed |

# Introduction

**Linux commands**

check out the video ***01b_Quick_Guide_Unix_Commands*** on bcourses

copying files and folders

```
cp my_file ../somewhere/else

cp my_file_original my_file_copy

cp my_file ../somewhere/else/my_file_copy

cp -r entireDirectory somewhere/else/to/new_destination
```

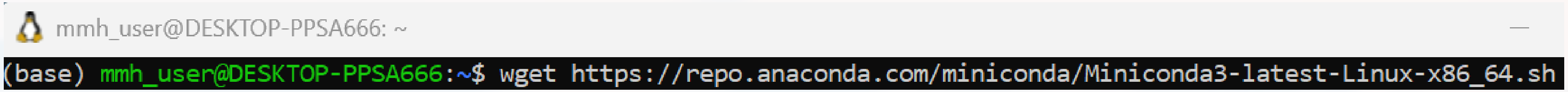
note: there are way more commands and flags we will be learning soon : )

Outline

- What is an "environment"?
- installing **WSL**
- **installing *miniconda***
- guide through **Jupyter**
- installing ***Spyder***
- guide through ***Spyder***

# Introduction

open your **WSL** shell and run the following commands:



```
mmh_user@DESKTOP-PPSA666: ~
(base) mmh_user@DESKTOP-PPSA666:~$ wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
```

**1) download installer:** `wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh`

**2) run installer:** `bash  Miniconda3-latest-Linux-x86_64.sh`

confirm
- license agreement,
- folder location and
- settings

**3) close and reopen WSL**

**4) activate conda:** `conda activate`

**for windows**

see the video ***01c_Installing_Miniconda*** on bcourses

**for mac**

see the video ***01cd_Installing_Miniconda_and_Jupyter_Mac*** on bcourses

# Introduction

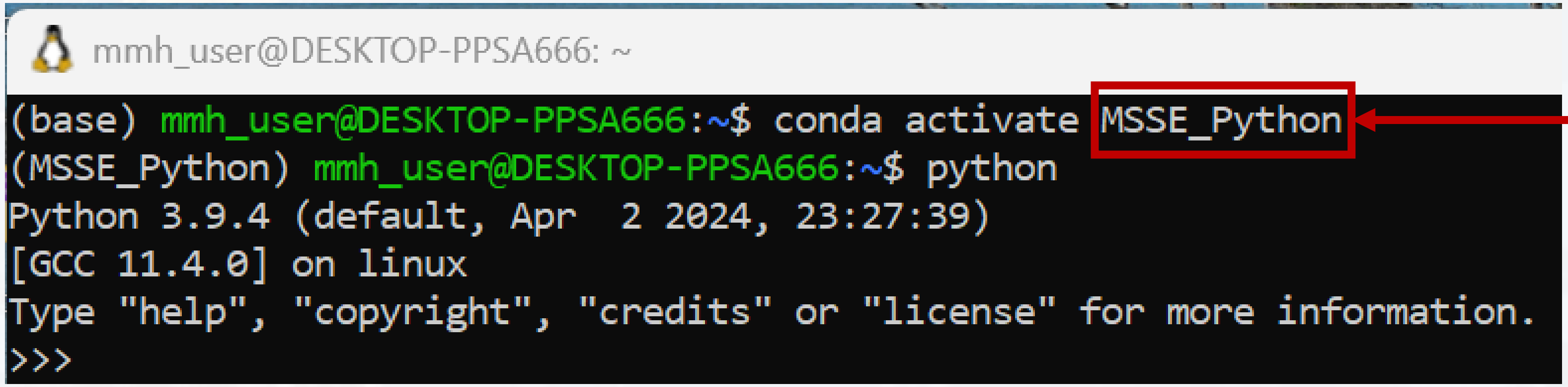
**5) create environment:** `conda create --name <My_Environment>`

**6) check environment:** `conda env list`

**7) activate environment:** `conda activate <My_Environment>`

<My_Environment>

**8) check python:**



mmh_user@DESKTOP-PPSA666: ~

```
(base) mmh_user@DESKTOP-PPSA666:~$ conda activate MSSE_Python
(MSSE_Python) mmh_user@DESKTOP-PPSA666:~$ python
Python 3.9.4 (default, Apr  2 2024, 23:27:39)
[GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

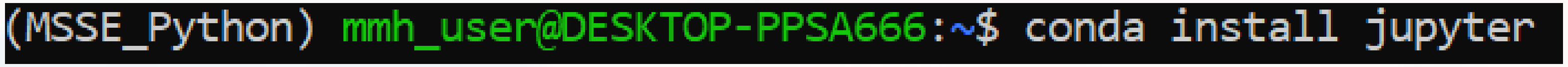
Outline

- What is an “environment”?
- installing **WSL**
- installing *miniconda*
- **guide through Jupyter**
- installing ***Spyder***
- guide through ***Spyder***

# Introduction

**1) install Jupyter:**

```
conda install jupyter
```



```
(MSSE_Python) mmh_user@DESKTOP-PPSA666:~$ conda install jupyter
```

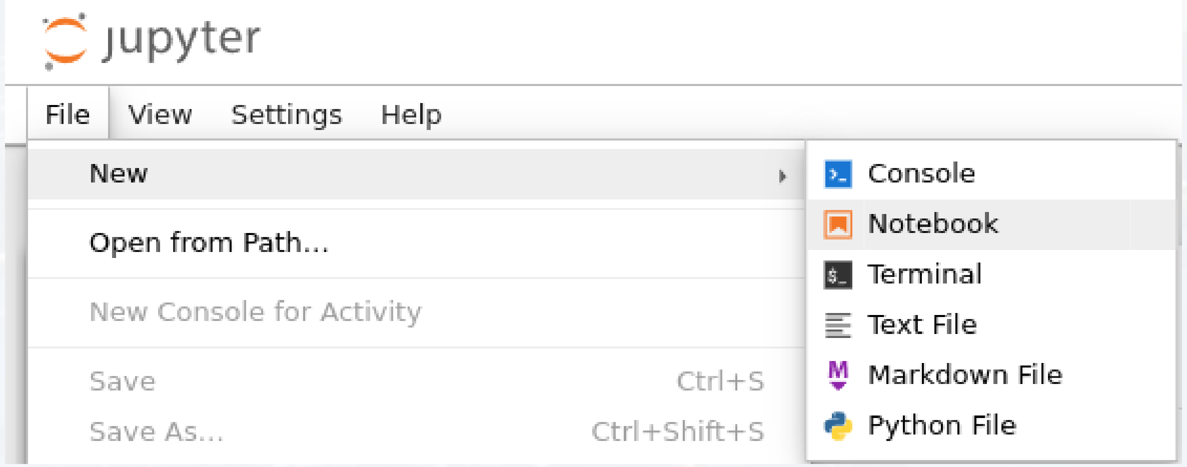
confirm settings

**2) open Jupyter:**

```
jupyter notebook &
```

a browser will open

go to → File
→ New
→ Notebook
→ confirm kernel

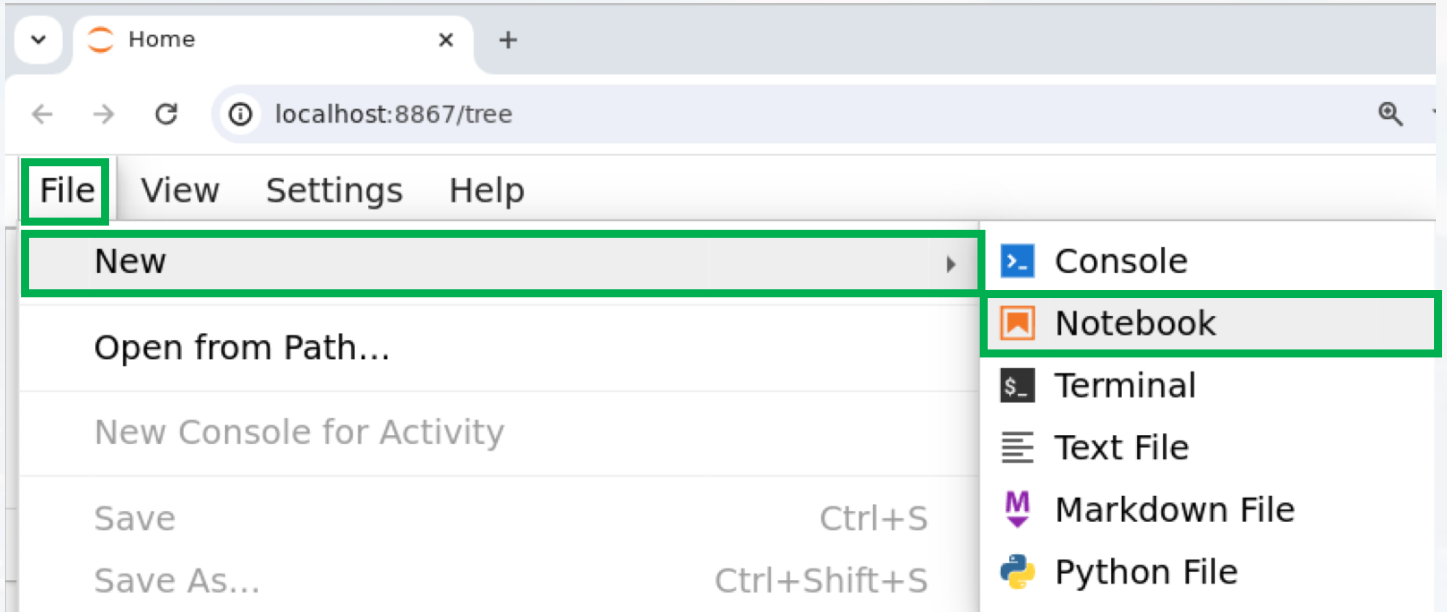


the notebook will open

check out the video ***01d_Installing_Jupyter*** on bcourses

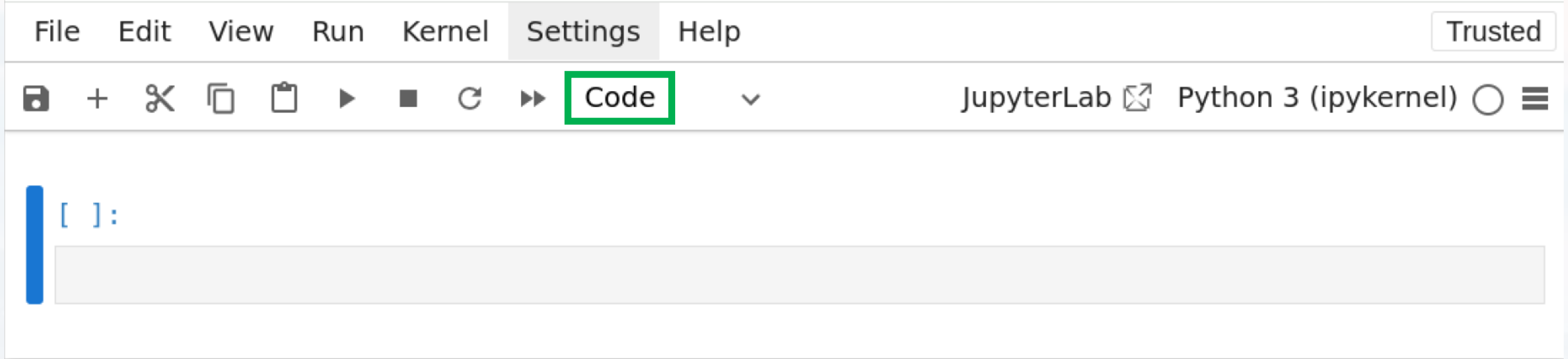
# Introduction

open new notebook



# Introduction

an empty cell will show up



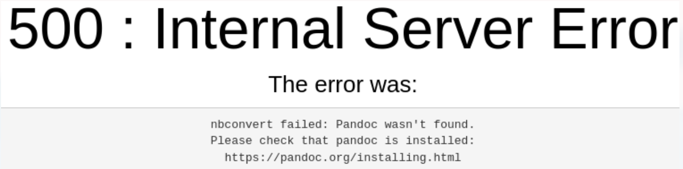
**markdown** → text
**code** → actual code

check out the video ***01e_Quick_Guide_Jupyter*** in bcourses

# Introduction

saving notebook as pdf

**note:** you might get an error message



500 : Internal Server Error

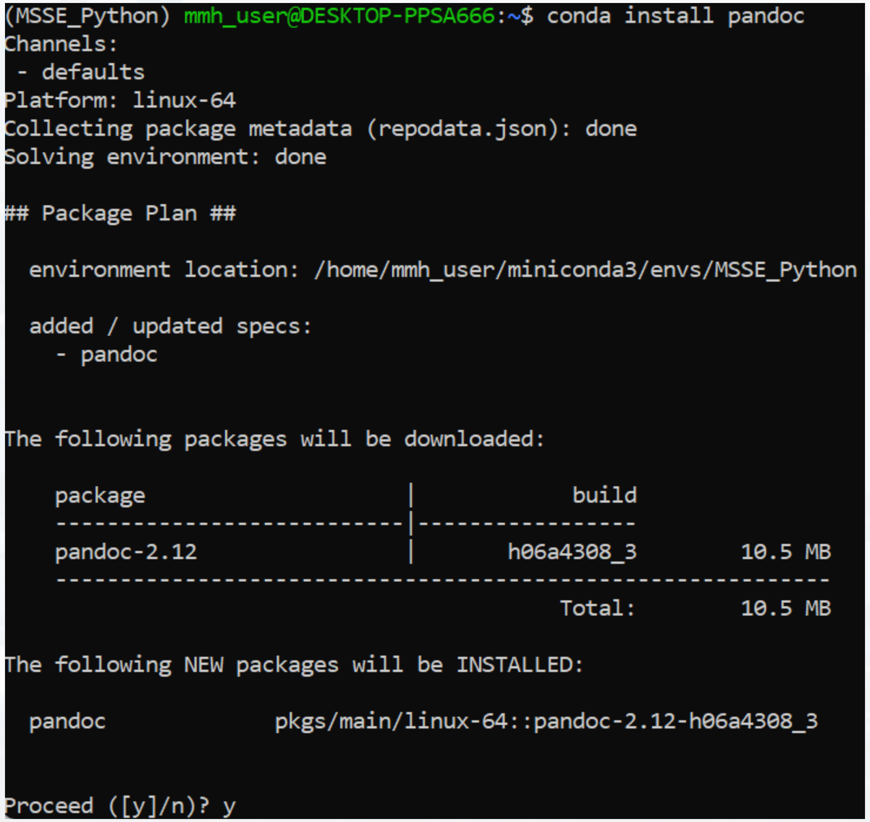
The error was:

```
nbconvert failed: Pandoc wasn't found.
Please check that pandoc is installed:
https://pandoc.org/installing.html
```

**fix:**

in your **conda** environment, run:

```
conda install pandoc
```

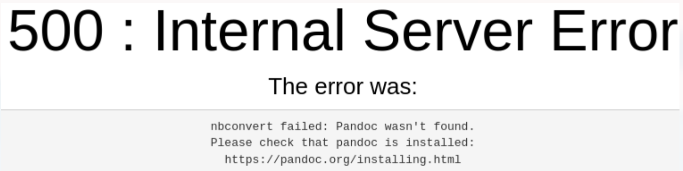


```
(MSSE_Python) mmh_user@DESKTOP-PPSA666:~$ conda install pandoc
Channels:
 - defaults
Platform: linux-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/mmh_user/miniconda3/envs/MSSE_Python

  added / updated specs:
    - pandoc


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    pandoc-2.12                |       h06a4308_3        10.5 MB
    ------------------------------------------------------------
                                           Total:        10.5 MB

The following NEW packages will be INSTALLED:

  pandoc             pkgs/main/linux-64::pandoc-2.12-h06a4308_3


Proceed ([y]/n)? y
```

# Introduction

saving notebook as pdf

**note:** you might get an error message



500 : Internal Server Error

The error was:

```
nbconvert failed: Pandoc wasn't found.
Please check that pandoc is installed:
https://pandoc.org/installing.html
```

**fix:**

in your **shell**, run:

```
sudo apt-get install texlive-xetex
```

& confirm your password

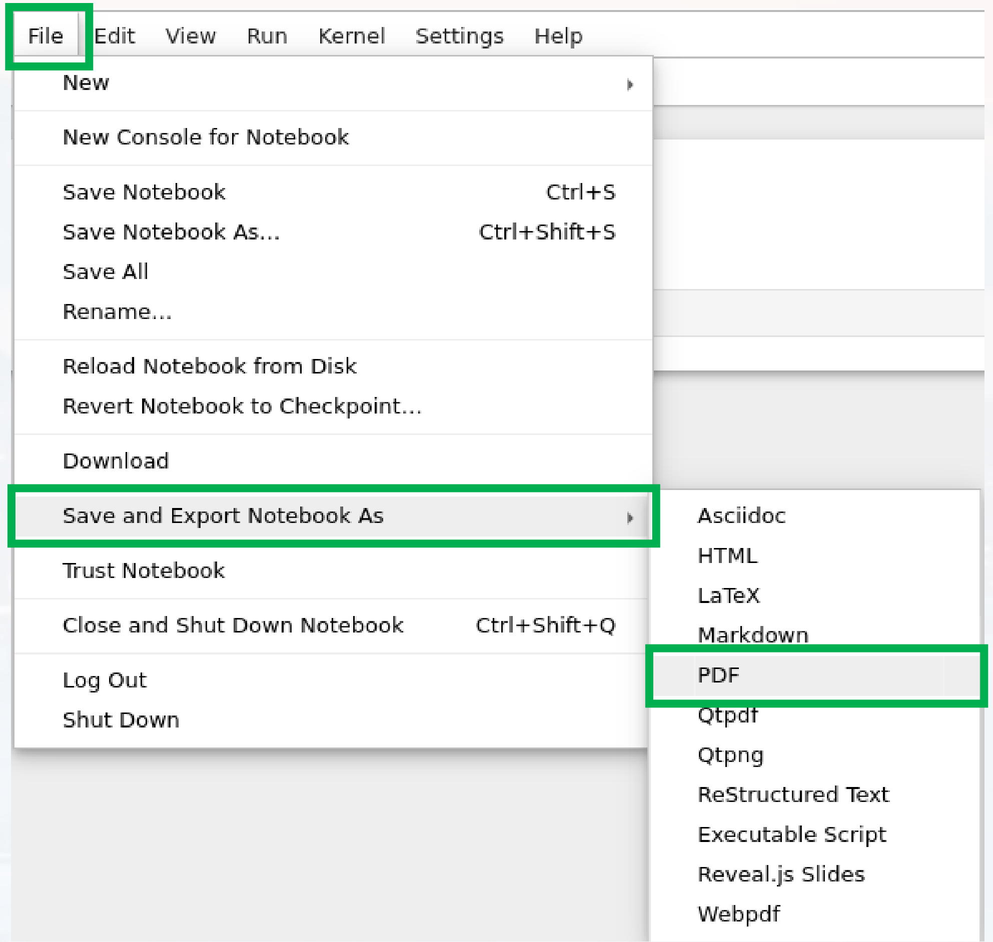
# Introduction

saving notebook as pdf

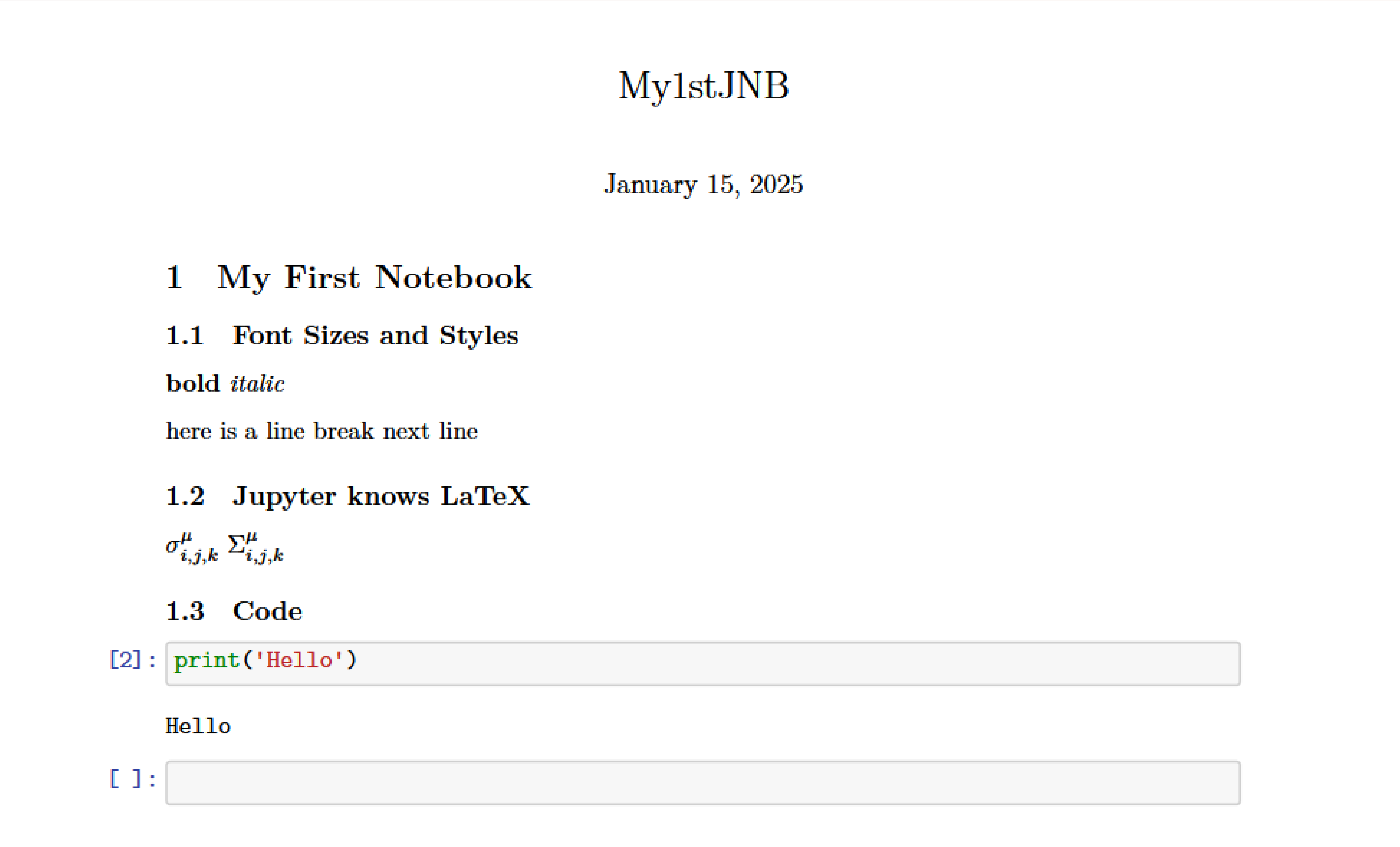
**Now it works!**

alternatively, the notebook can be converted via command line:

```
jupyter nbconvert My1stNB.ipynb --to pdf
```



# Introduction



My1stJNB

January 15, 2025

## 1 My First Notebook

### 1.1 Font Sizes and Styles

**bold** *italic*

here is a line break next line

### 1.2 Jupyter knows LaTeX

$\boldsymbol{\sigma^{\mu}_{i,j,k}}$ $\boldsymbol{\Sigma^{\mu}_{i,j,k}}$

### 1.3 Code

```
[2]: print('Hello')
```

```
Hello
```

```
[ ]:
```

Outline

- What is an "environment"?
- installing **WSL**
- installing *miniconda*
- guide through Jupyter
- **installing *Spyder***
- guide through ***Spyder***

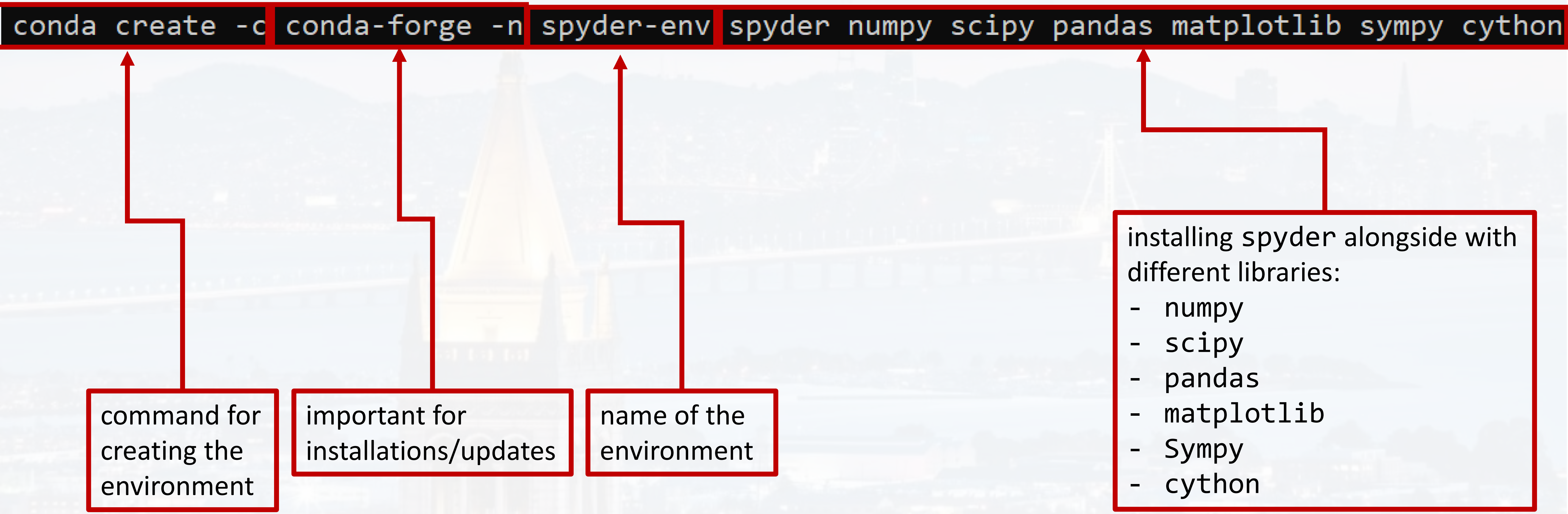
# Introduction

***Installing spyder***

check out the video ***01f_Installing_Spyder*** on bcourses

or follow instructions [here](here):

1) creating spyder environment



```
conda create -c conda-forge -n spyder-env spyder numpy scipy pandas matplotlib sympy cython
```

command for creating the environment

important for installations/updates

name of the environment

installing `spyder` alongside with different libraries:

- `numpy`
- `scipy`
- `pandas`
- `matplotlib`
- `Sympy`
- `cython`

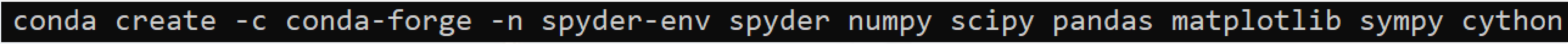
# Introduction

***Installing spyder***

check out the video ***01f_Installing_Spyder*** on bcourses

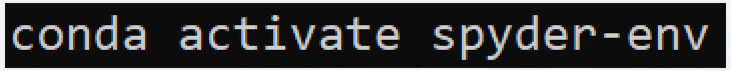
or follow instructions here:

1) creating spyder environment

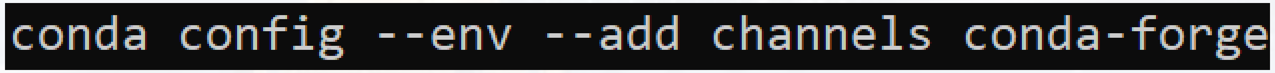


```
conda create -c conda-forge -n spyder-env spyder numpy scipy pandas matplotlib sympy cython
```

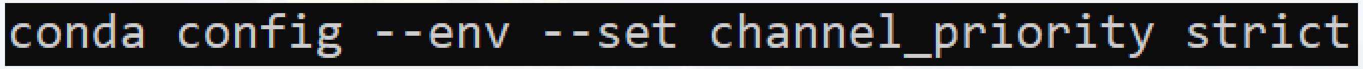
2) activating and configure environment



```
conda activate spyder-env
```

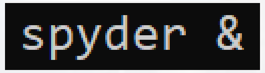


```
conda config --env --add channels conda-forge
```



```
conda config --env --set channel_priority strict
```

3) open spyder



```
spyder &
```

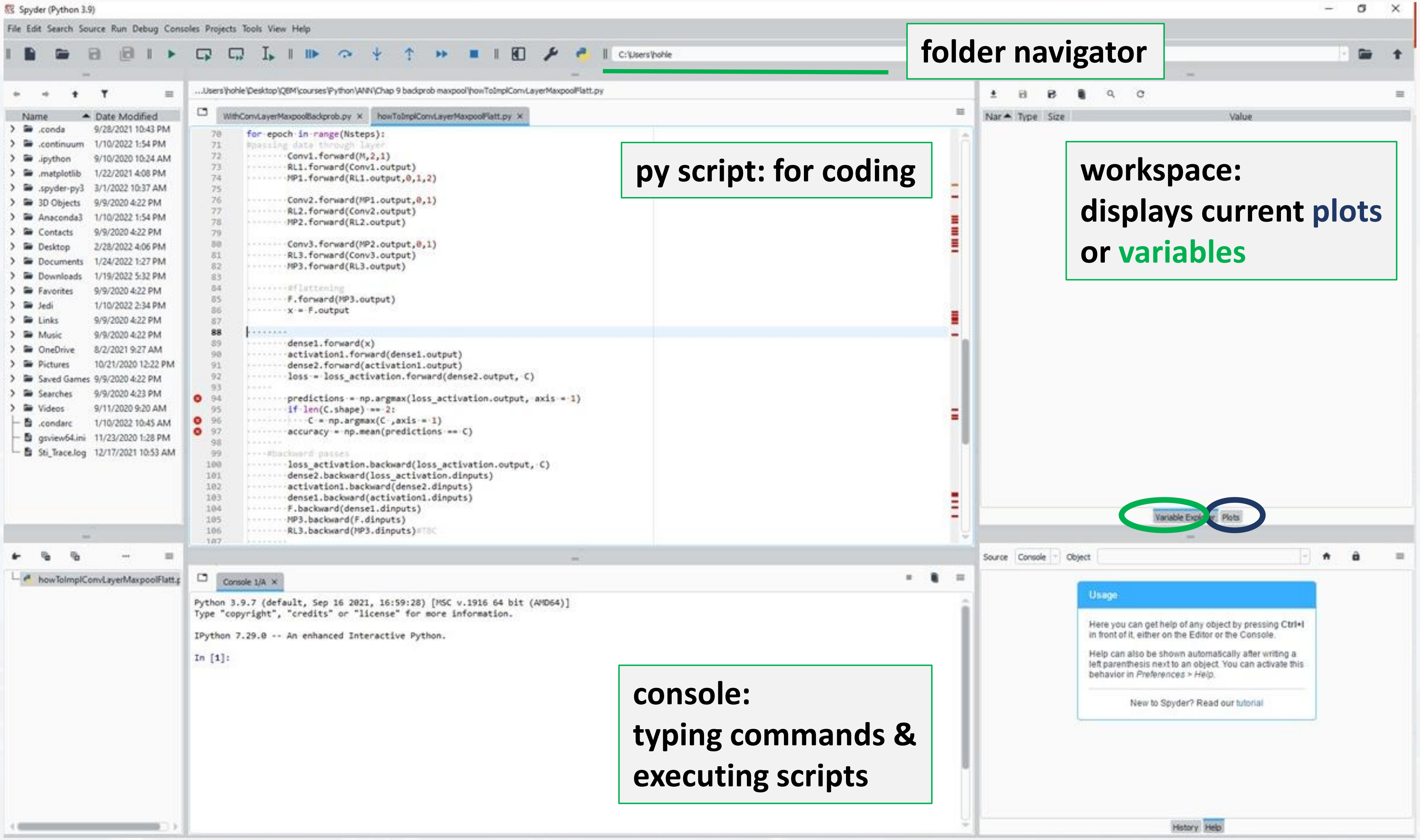
Outline

- What is an “environment”?
- installing **WSL**
- installing *miniconda*
- guide through Jupyter
- installing *Spyder*
- **guide through *Spyder***

# Introduction

check out the video ***01g_Quick_Guide_Spyder*** on bcourses

**content of current folder**



Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\hohle
WithConvLayerMaxpoolBackprob.py
howToImplConvLayerMaxpoolFlatt.py
Nar Type Size Value
Variable Explorer
Plots
Source Console Object
Console 1/A
History Help

...Users\hohle\Desktop\QBM\courses\Python\ANN\Chap 9 backprob maxpool\howToImplConvLayerMaxpoolFlatt.py

| Name | Date Modified |
|---|---|
| .conda | 9/28/2021 10:43 PM |
| .continuum | 1/10/2022 1:54 PM |
| .ipython | 9/10/2020 10:24 AM |
| .matplotlib | 1/22/2021 4:08 PM |
| .spyder-py3 | 3/1/2022 10:37 AM |
| 3D Objects | 9/9/2020 4:22 PM |
| Anaconda3 | 1/10/2022 1:54 PM |
| Contacts | 9/9/2020 4:22 PM |
| Desktop | 2/28/2022 4:06 PM |
| Documents | 1/24/2022 1:27 PM |
| Downloads | 1/19/2022 5:32 PM |
| Favorites | 9/9/2020 4:22 PM |
| Jedi | 1/10/2022 2:34 PM |
| Links | 9/9/2020 4:22 PM |
| Music | 9/9/2020 4:22 PM |
| OneDrive | 8/2/2021 9:27 AM |
| Pictures | 10/21/2020 12:22 PM |
| Saved Games | 9/9/2020 4:22 PM |
| Searches | 9/9/2020 4:23 PM |
| Videos | 9/11/2020 9:20 AM |
| .condarc | 1/10/2022 10:45 AM |
| gsview64.ini | 11/23/2020 1:28 PM |
| Sti_Trace.log | 12/17/2021 10:53 AM |

howToImplConvLayerMaxpoolFlatt.py

```
for epoch in range(Nsteps):
#passing data through layer
        Conv1.forward(M,2,1)
        RL1.forward(Conv1.output)
        MP1.forward(RL1.output,0,1,2)

        Conv2.forward(MP1.output,0,1)
        RL2.forward(Conv2.output)
        MP2.forward(RL2.output)

        Conv3.forward(MP2.output,0,1)
        RL3.forward(Conv3.output)
        MP3.forward(RL3.output)

        #flattening
        F.forward(MP3.output)
        x = F.output

        
        dense1.forward(x)
        activation1.forward(dense1.output)
        dense2.forward(activation1.output)
        loss = loss_activation.forward(dense2.output, C)
    
        predictions = np.argmax(loss_activation.output, axis = 1)
        if len(C.shape) == 2:
            C = np.argmax(C ,axis = 1)
        accuracy = np.mean(predictions == C)
        
    #backward passes
        loss_activation.backward(loss_activation.output, C)
        dense2.backward(loss_activation.dinputs)
        activation1.backward(dense2.dinputs)
        dense1.backward(activation1.dinputs)
        F.backward(dense1.dinputs)
        MP3.backward(F.dinputs)
        RL3.backward(MP3.dinputs)#TBC
```

```
Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.29.0 -- An enhanced Interactive Python.

In [1]:
```

Usage

Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.

New to Spyder? Read our tutorial

**folder navigator**

**py script: for coding**

**workspace: displays current plots or variables**

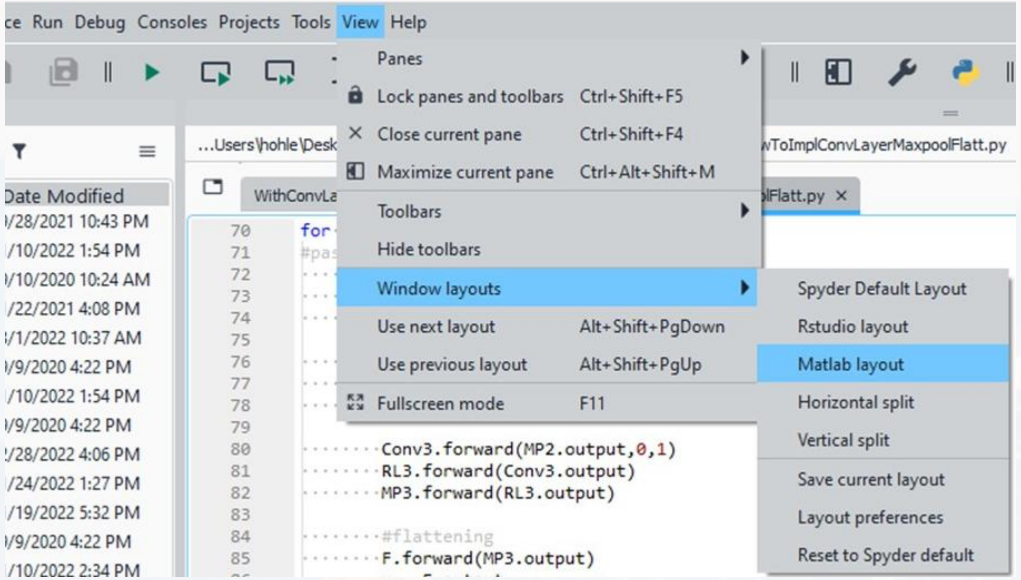
**console: typing commands & executing scripts**

# Introduction

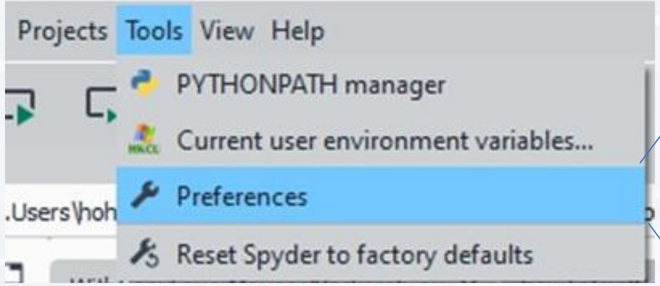
***Spyder***

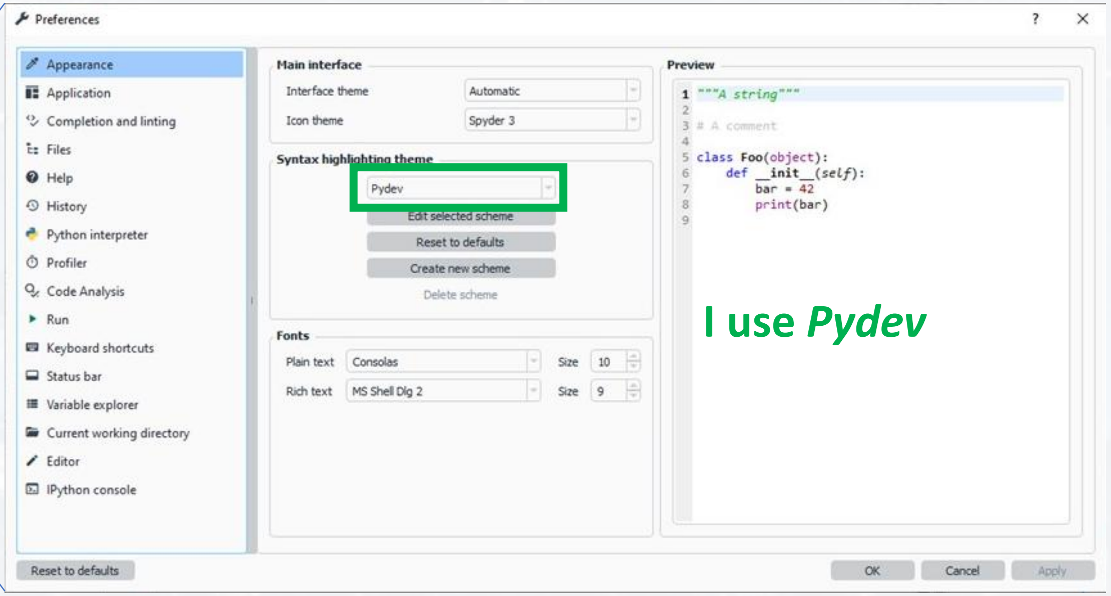
**settings:**
toolbar: *View* → *Window layouts*

e. g. Matlab



```
for
#pas
        Conv3.forward(MP2.output,0,1)
        RL3.forward(Conv3.output)
        MP3.forward(RL3.output)

        #flattening
        F.forward(MP3.output)
```


Projects Tools View Help
PYTHONPATH manager
Current user environment variables...
Preferences
Reset Spyder to factory defaults
.Users\hoh



Main interface
Interface theme: Automatic
Icon theme: Spyder 3

Syntax highlighting theme: Pydev

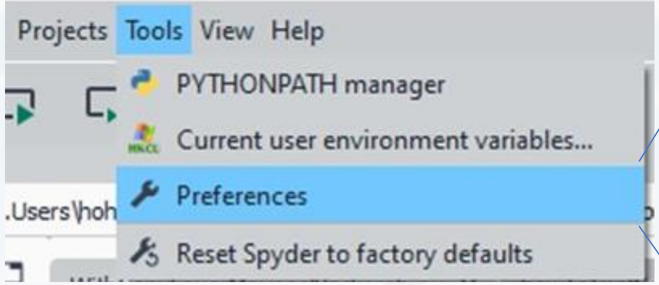
Fonts
Plain text: Consolas Size: 10
Rich text: MS Shell Dlg 2 Size: 9

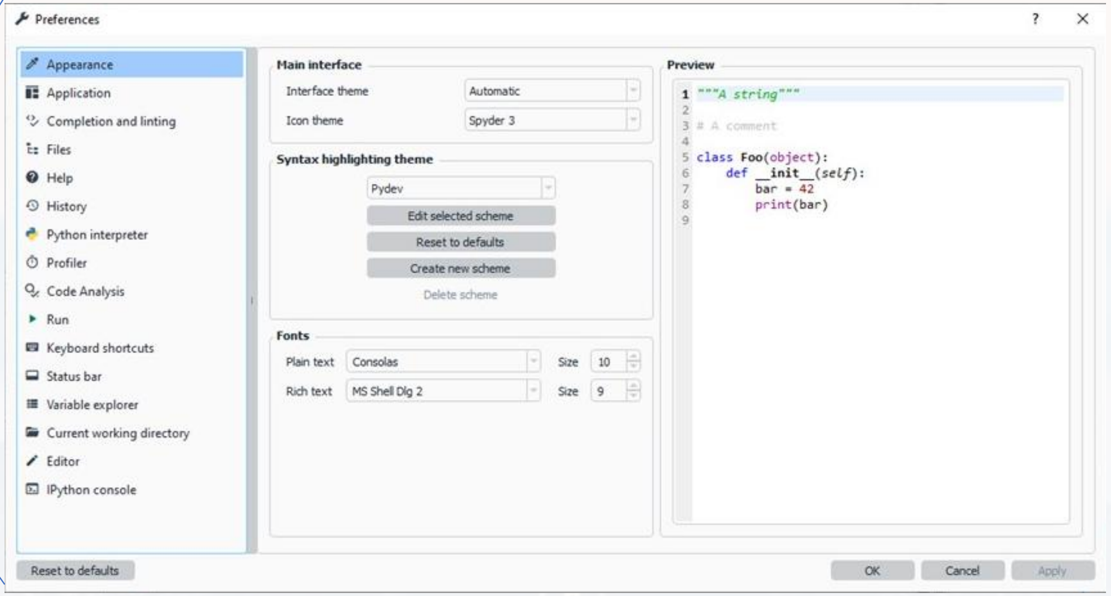
Preview

```
"""A string"""

# A comment

class Foo(object):
    def __init__(self):
        bar = 42
        print(bar)
```

**I use *Pydev***

# Introduction


Projects
Tools
View
Help
PYTHONPATH manager
Current user environment variables...
Preferences
Reset Spyder to factory defaults
.Users\hoh



Main interface

Interface theme: Automatic

Icon theme: Spyder 3

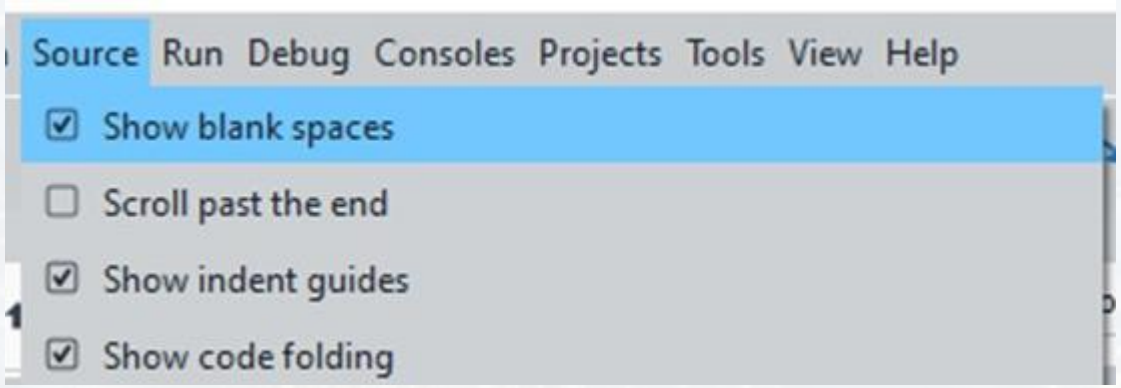
Syntax highlighting theme: Pydev

Fonts

Plain text: Consolas, Size: 10

Rich text: MS Shell Dlg 2, Size: 9

```
"""A string"""

# A comment

class Foo(object):
    def __init__(self):
        bar = 42
        print(bar)

```



☑ Show blank spaces

□ Scroll past the end

☑ Show indent guides

☑ Show code folding

spaces are relevant for syntax!

## Thank you very much for your attention