Basic Python Programming

[Session 4] OpenCV

ITinerary X University of Ghana

Contents

Intro & Preparation

OpenCV

- **An open-source Computer Vision library**
- **It supports..**
  - 2D / 3D image processing
  - Facial / gesture / object recognition/detection
  - ML
  - AR
  - So on…
- **We can use it with many languages**
  - C/C++/Python/Java/Objective-C/…
  - Of course, we will use Python

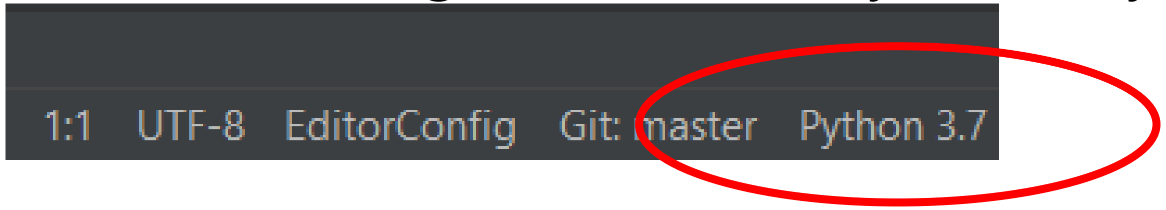
About Today’s Class…

- **We will learn by writing code.. but some backgrounds are needed**

- **OpenCV must be available in your environment!!**
  - Before the lab session, please make sure this

Required Environment (IMPORTANT)

- **As mentioned in session 1,2:**
  - Python **3.7.8**: over 3.8 may not supports OpenCV
  - In the lower-right corner of PyCharm, you can see that

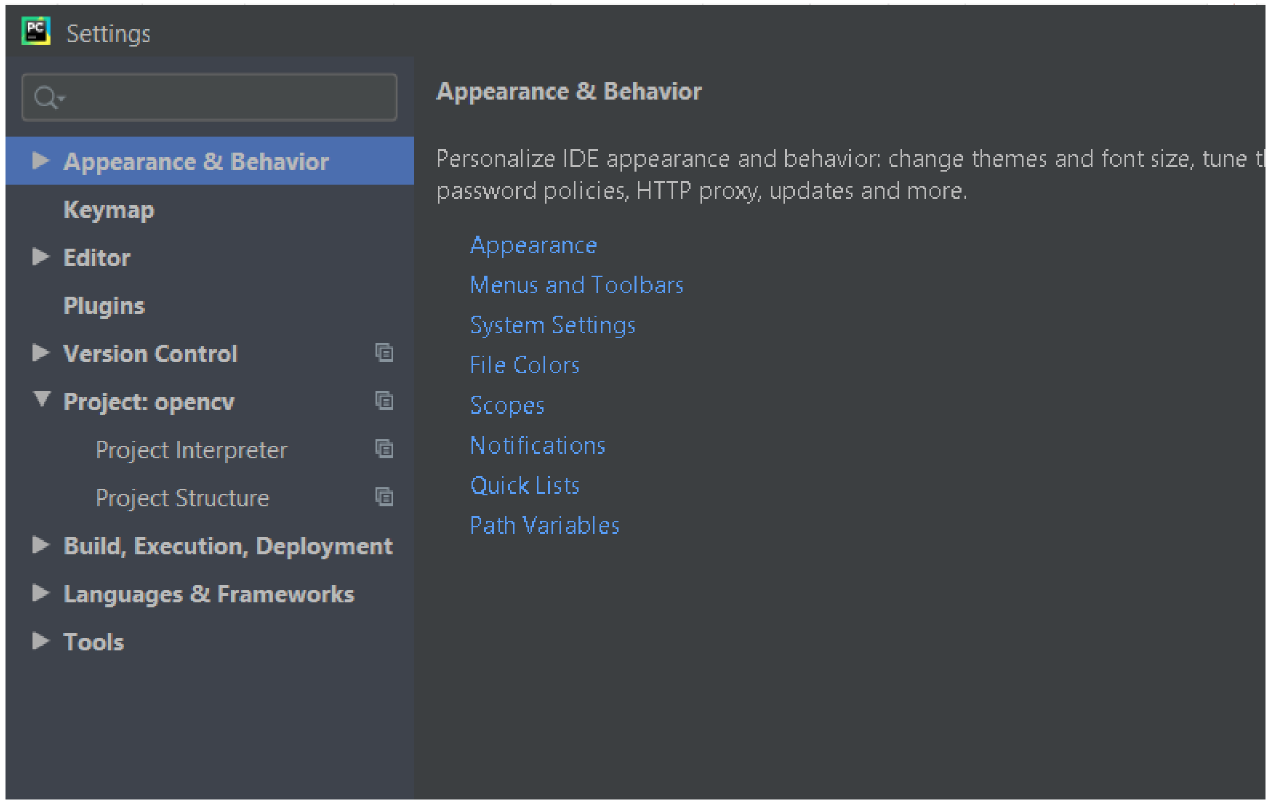


- **OpenCV is cross-platform library, so OS doesn't matter**
- **Perhaps, you don't need to worry about system requirements**
  - I think your RAM may be greater than 1GB…

- **If you cannot sure about your environment, please ask us**
  - With your device / system specifications

Installing OpenCV [1]

- In PyCharm, go to [file]>[settings]

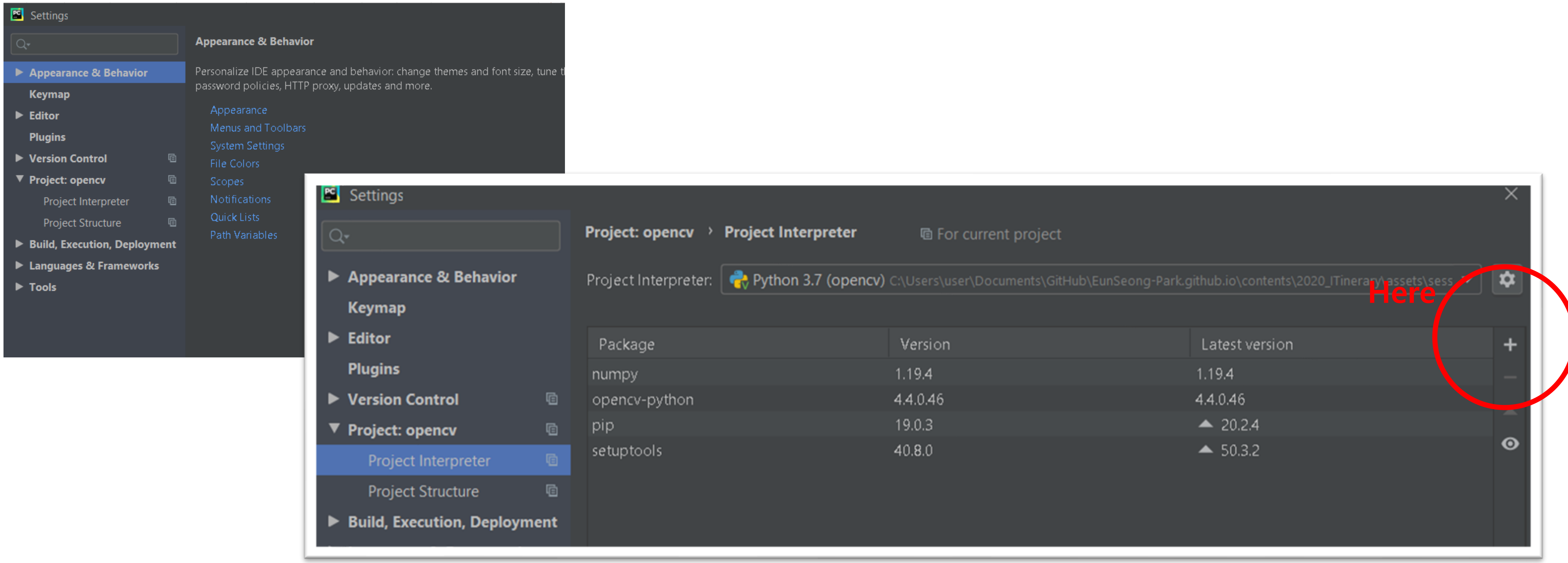

Settings

Appearance & Behavior

Personalize IDE appearance and behavior: change themes and font size, tune t
password policies, HTTP proxy, updates and more.

Installing OpenCV [2]

- Go to [Project Interpreter] and click [Install]



Settings

Appearance & Behavior

Personalize IDE appearance and behavior: change themes and font size, tune t
password policies, HTTP proxy, updates and more.

Settings

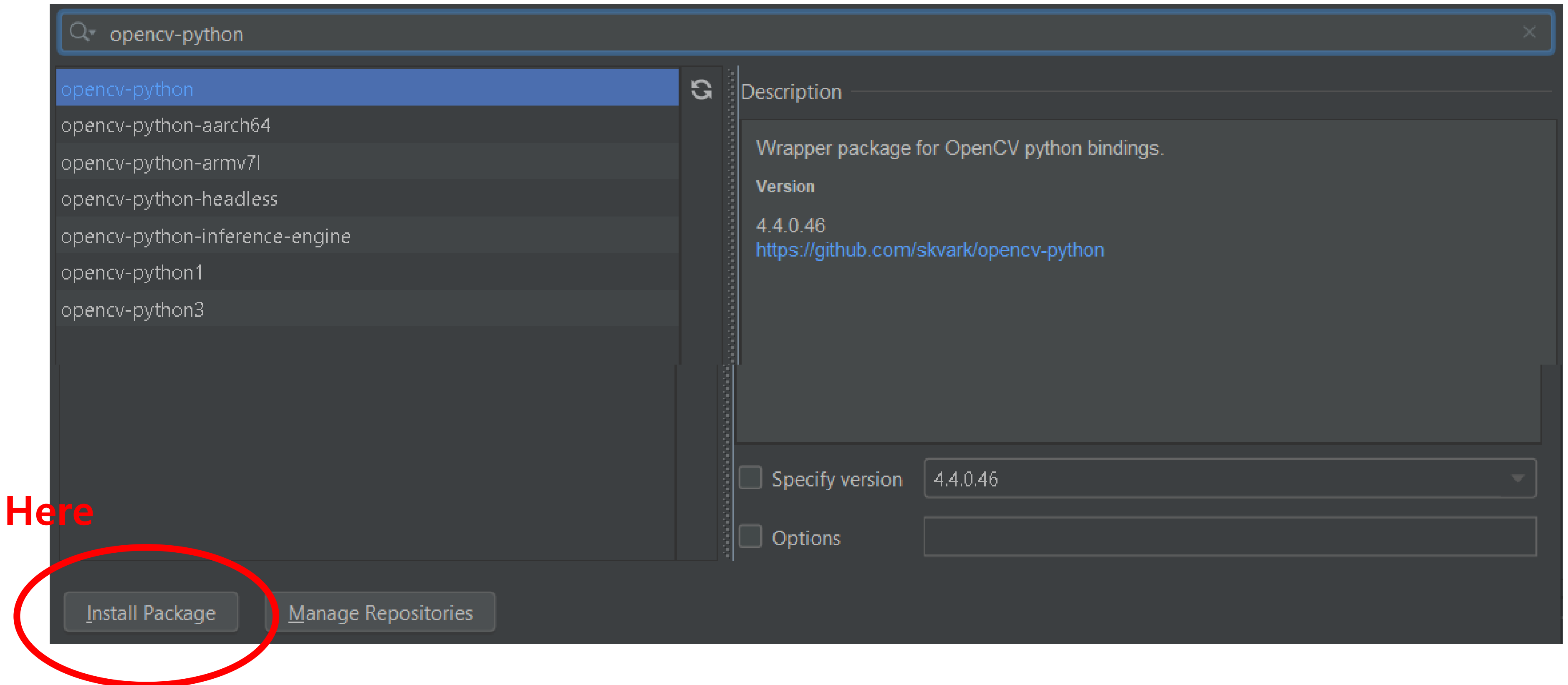
Project: opencv › Project Interpreter   For current project

Project Interpreter: Python 3.7 (opencv) C:\Users\user\Documents\GitHub\EunSeong-Park.github.io\contents\2020_ITinerary\assets\sess

Package	Version	Latest version
numpy	1.19.4	1.19.4
opencv-python	4.4.0.46	4.4.0.46
pip	19.0.3	▲ 20.2.4
setuptools	40.8.0	▲ 50.3.2

Installing OpenCV [3]

- **Find [opencv-python] and install it**
  - You don't need to change the version, just click [Install]



opencv-python

- opencv-python
- opencv-python-aarch64
- opencv-python-armv7l
- opencv-python-headless
- opencv-python-inference-engine
- opencv-python1
- opencv-python3

Description

Wrapper package for OpenCV python bindings.

Version

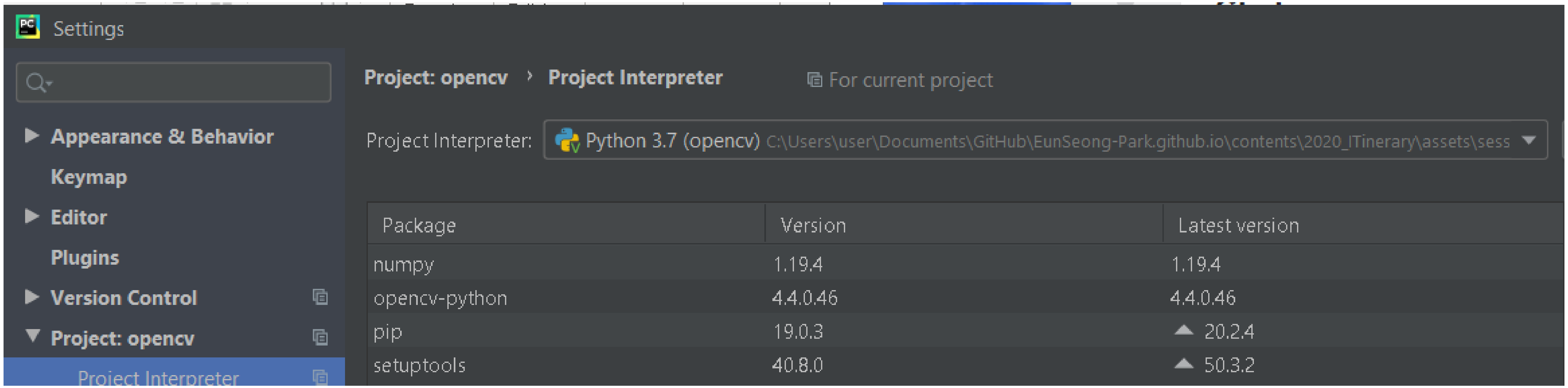
4.4.0.46
https://github.com/skvark/opencv-python

Specify version 4.4.0.46

Options

Installing OpenCV [4]

- **Installation takes time (so don't worry)**
- **After installation, you can find opencv-python and numpy**



Settings

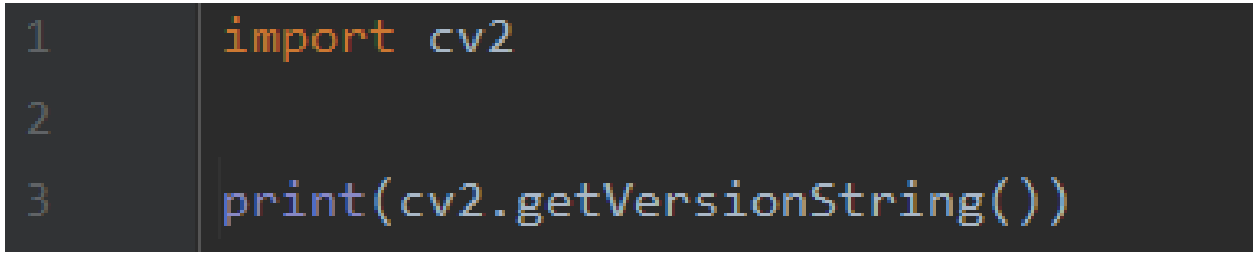
Project: opencv › Project Interpreter   For current project

Project Interpreter: Python 3.7 (opencv) C:\Users\user\Documents\GitHub\EunSeong-Park.github.io\contents\2020_ITinerary\assets\sess

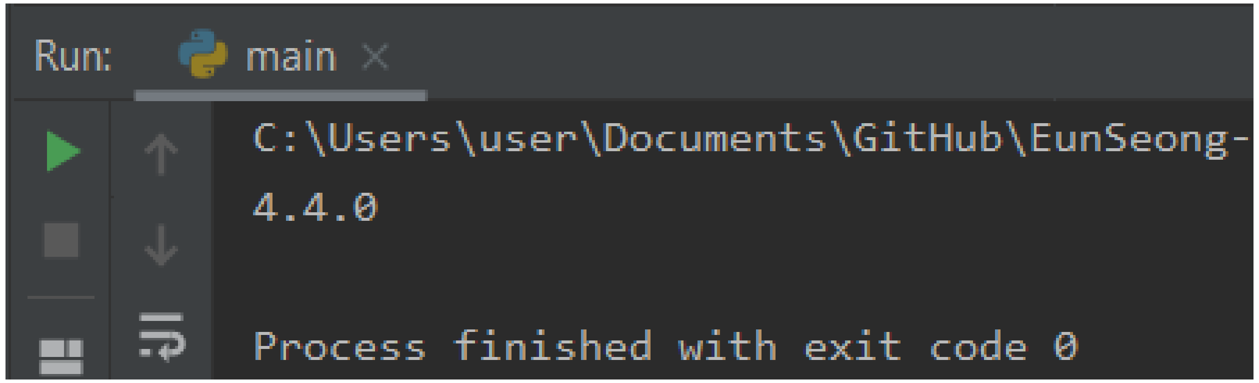
Package	Version	Latest version
numpy	1.19.4	1.19.4
opencv-python	4.4.0.46	4.4.0.46
pip	19.0.3	▲ 20.2.4
setuptools	40.8.0	▲ 50.3.2

Installing OpenCV [5]

- **Then let's check if the installation was successful.**
- **Write this and run:**
- **You got an error?**
  - Check if you've done correctly and retry
  - Or please ask us



```python
import cv2

print(cv2.getVersionString())
```



```
C:\Users\user\Documents\GitHub\EunSeong-
4.4.0

Process finished with exit code 0
```

Backgrounds

Boring time is coming…

Image is a matrix

- **Why?**

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

- **An Image contains (Width) * (Height) pixels**
  - So it is a (H)x(W) matrix

Color

- **In RGB, each component can be 0~255**
  - So we can represent 256 * 256 * 256 = 16M colors!
- **So each pixel can be dealt with as a 3-tuple(R,G,B) but…**

- **We can also represent as “an” integer!**
  - By hexadecimal representation

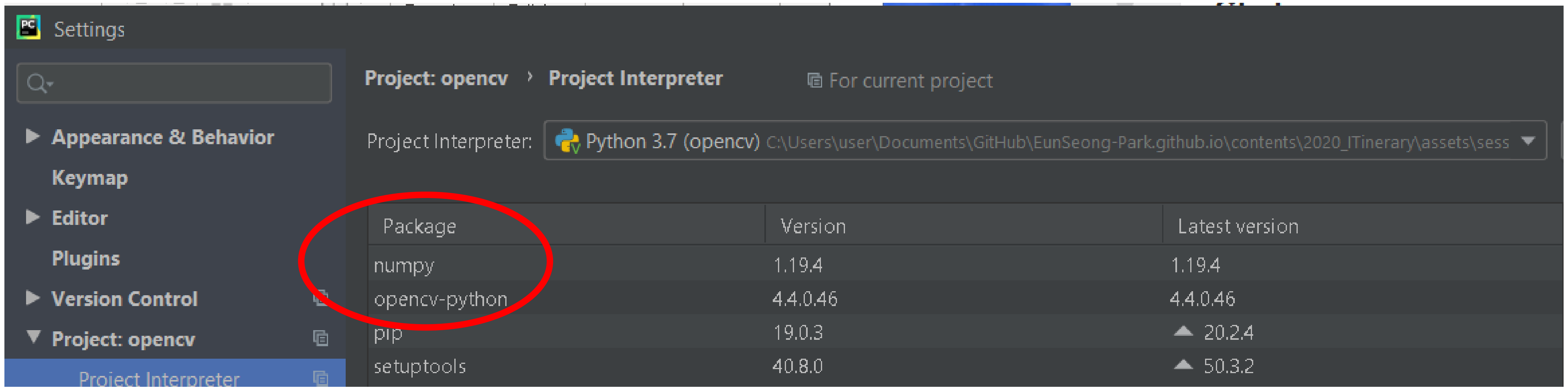
#FFFF00

Notes

- **Anyway, in OpenCV, image is regarded as 3-dimensional matrix(array)**
    - Height X Width X 3 (RGB)

NumPy [1]

- You might see numpy when we install OpenCV


Settings

Project: opencv › Project Interpreter   For current project

Project Interpreter: Python 3.7 (opencv) C:\Users\user\Documents\GitHub\EunSeong-Park.github.io\contents\2020_ITinerary\assets\sess

Package	Version	Latest version
numpy	1.19.4	1.19.4
opencv-python	4.4.0.46	4.4.0.46
pip	19.0.3	▲ 20.2.4
setuptools	40.8.0	▲ 50.3.2

- What is this?

NumPy [2]

- **NumPy is an open-source library for arrays and matrices**
  - It shows good performance even on very large / multi-dimensional case
  - Using Python list for large-scale calculation is too slow…

- **It is used in many other Python libraries**
  - SciPy
  - Matplotlib
  - Pandas
  - So on…

NumPy [3]

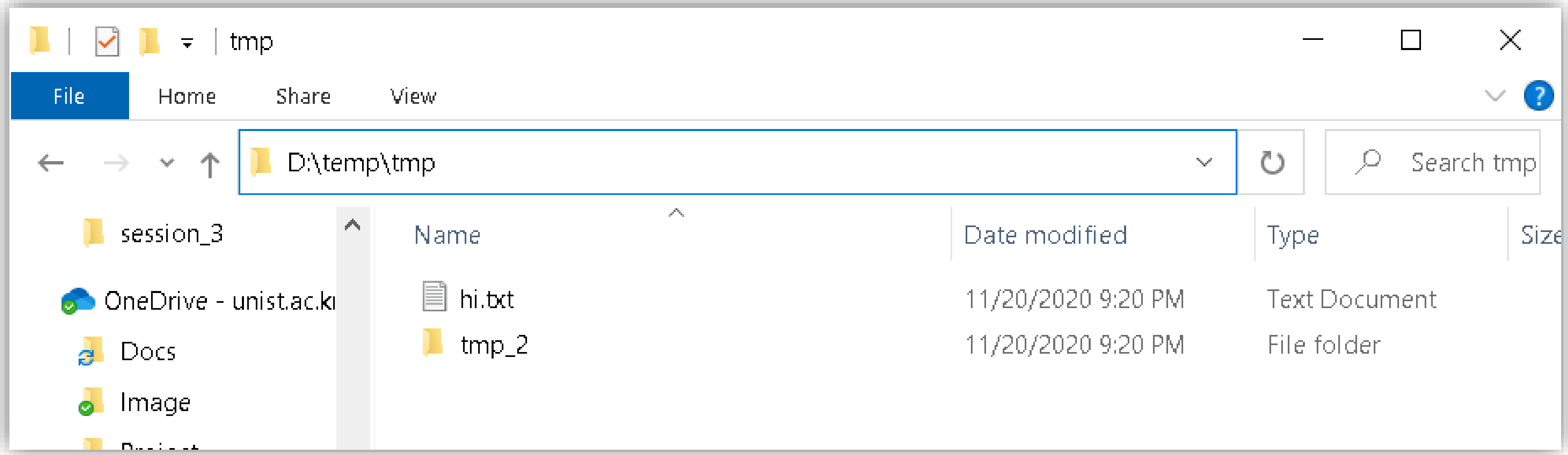
- **But we do not practice it in this class**
  - We don’t need to learn about “miscellaneous” things

- **We only use/learn what we need**
  - But some supplement will be given
  - And feel free to ask us!

  - A good reference is here: https://numpy.org/devdocs/user/whatisnumpy.html

Path [1]

- **Sometimes, we read/write some file**
  - In OpenCV, we may read/write images/videos
- **Every file has a path**
  - Path is a way to find the file (So, the answer of “where is it?”)
  - For example, the following hi.txt has a path: `D:\temp\tmp\hi.txt`



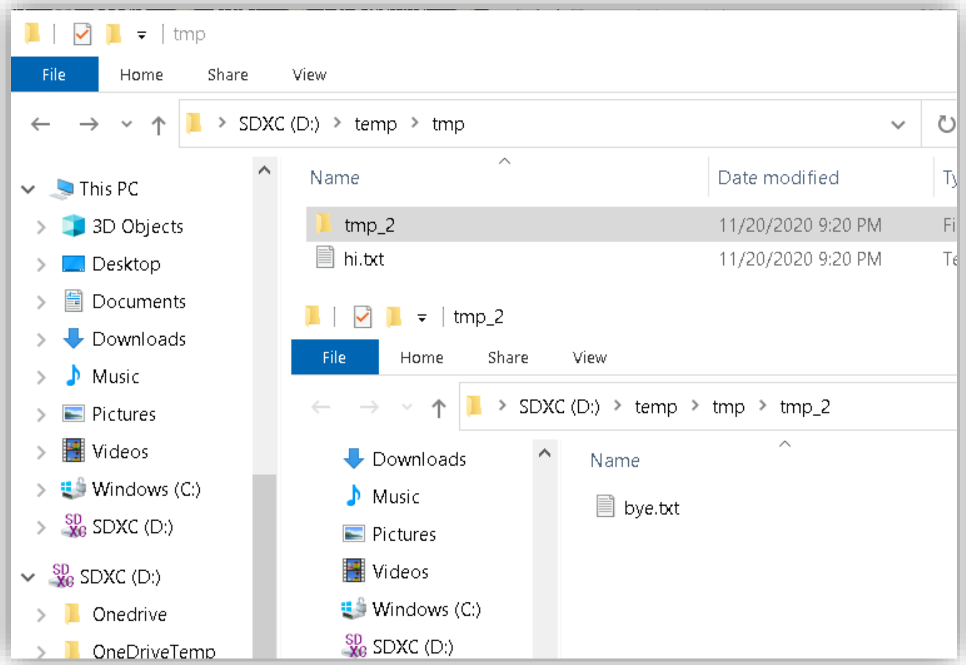
Name	Date modified	Type	Size
hi.txt	11/20/2020 9:20 PM	Text Document	
tmp_2	11/20/2020 9:20 PM	File folder	

Path [2]

- **Two types of path:**
  - Absolute path: A path from some fixed (reference) point (usually root)
    - Previous example used absolute path
    - Because the reference is fixed, absolute path of a file is unique
  - Relative path: A path from the current point
    - Let's take an example!

Path [3]

- In a view of `hi.txt`, what is the relative path of `bye.txt`?



- It's denoted by
  - "`tmp_2\bye.txt`", or
  - "`.\tmp_2\byte.txt`"

Path [4]

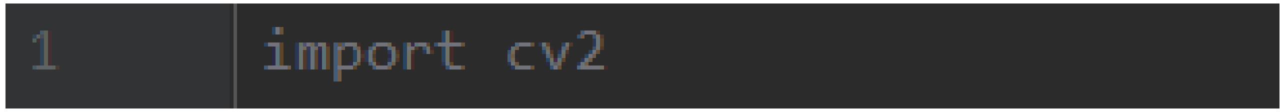
- **Some practices are in exercise!**

OpenCV Practice

Boring time is over!

Preparation

- **Just import this**



```
import cv2
```

Image Read [1]

- **Put any image in your project directory**

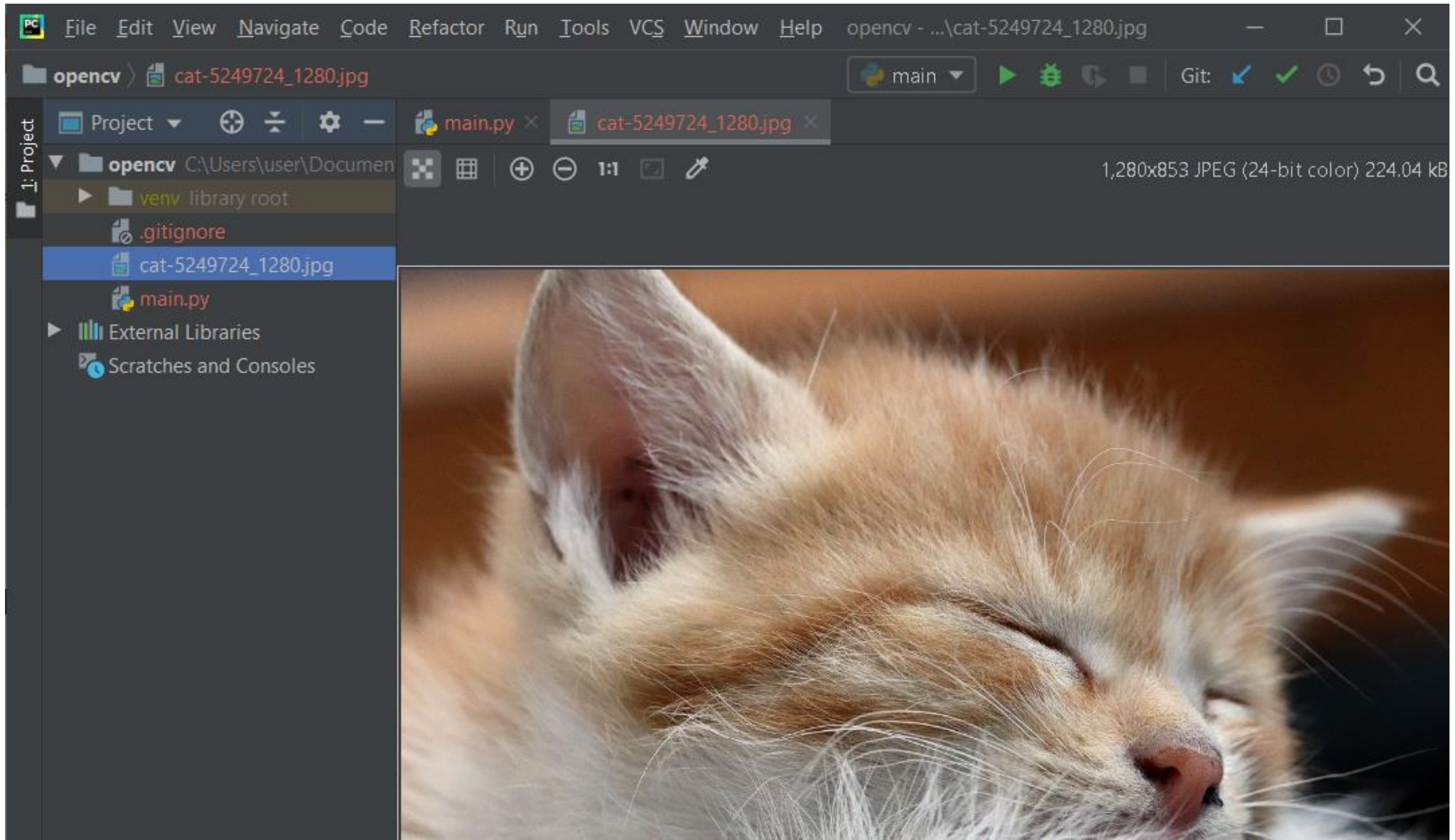
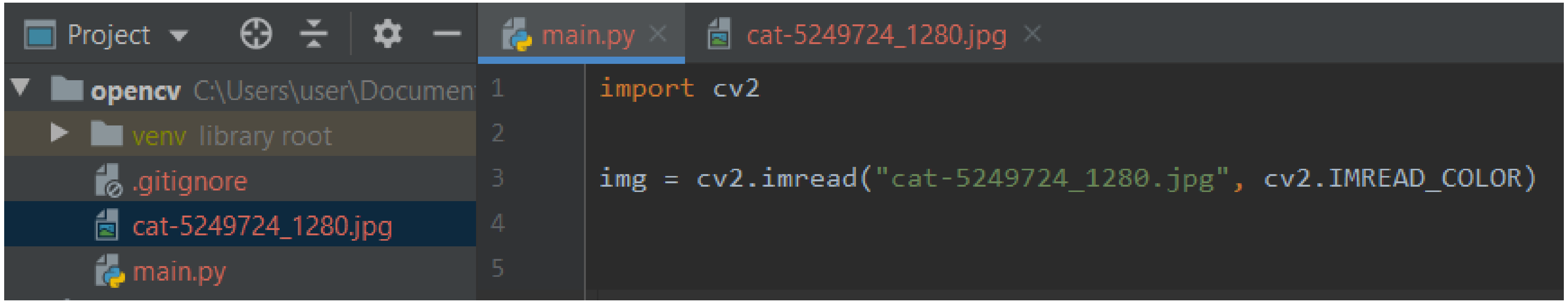

File Edit View Navigate Code Refactor Run Tools VCS Window Help opencv - ...\cat-5249724_1280.jpg
opencv cat-5249724_1280.jpg
main
Git:
1: Project
Project
main.py
cat-5249724_1280.jpg
opencv C:\Users\user\Documen
venv library root
.gitignore
cat-5249724_1280.jpg
main.py
External Libraries
Scratches and Consoles
1:1
1,280x853 JPEG (24-bit color) 224.04 kB

Image Read [2]

- **Use `cv2.imread("filename", "flag")`**
  - Filename: the path
  - Flag:
    - `cv2.IMREAD_COLOR`: load image with color
    - `cv2.IMREAD_GRAYSCALE`: load image with grayscale
    - `cv2.IMREAD_UNCHANGED`: load image with color (also with alpha-channel)



```python
import cv2

img = cv2.imread("cat-5249724_1280.jpg", cv2.IMREAD_COLOR)
```

Image Show [1]

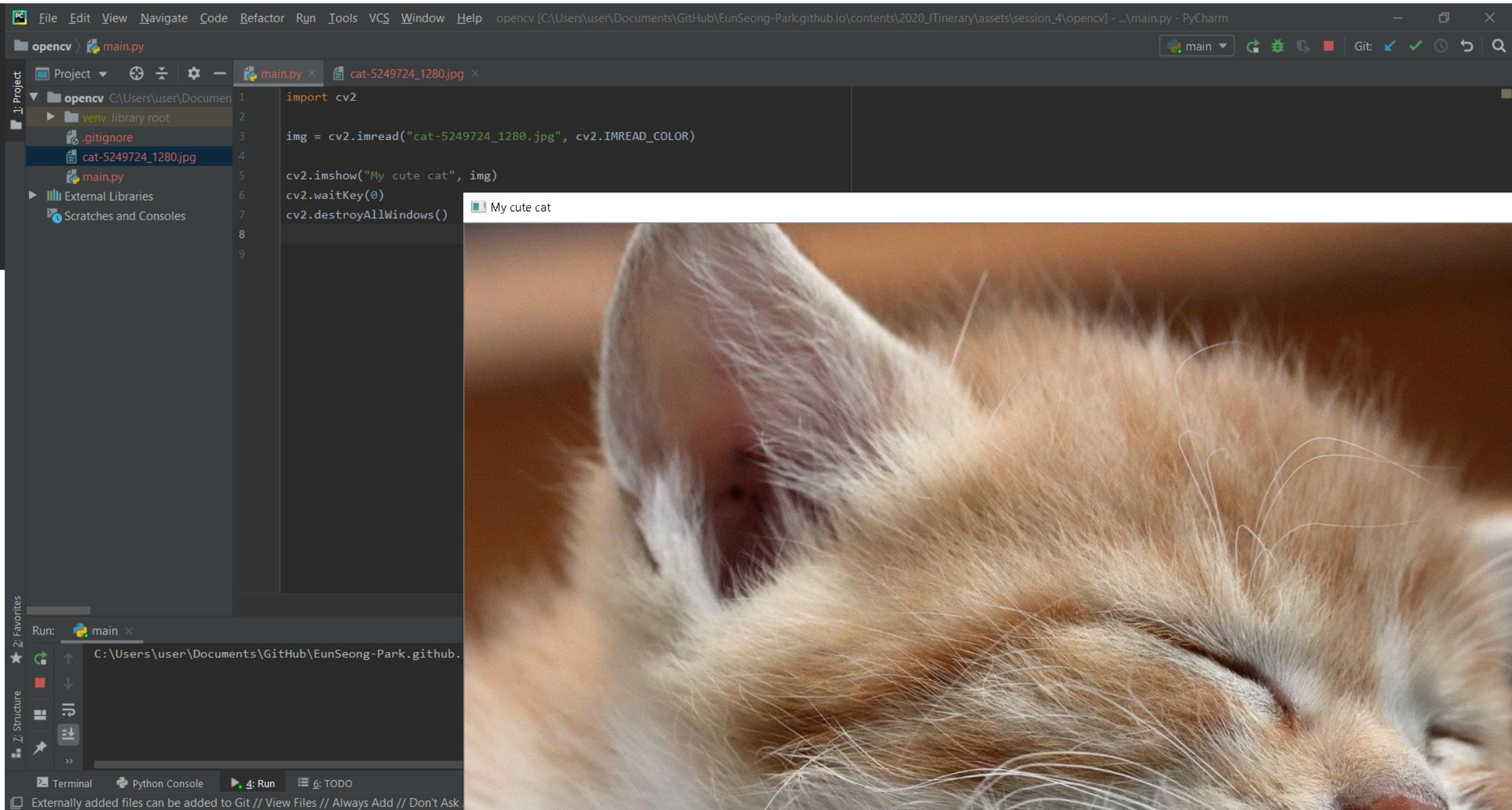
- **Usually, we use three functions to show image**
  - `cv2.imshow(“title”, “image”):` Show image(we got by `imread()`) with title
  - `cv2.waitkey(“time”):` Wait for any keyboard input or time(in ms, infinite when time=0)
  - `cv2.destroyAllWindows():` Destroy all windows

- **Let’s try it!**

Image Show [2]

- Oh… slightly big but OK

```python
import cv2

img = cv2.imread("cat-5249724_1280.jpg", cv2.IMREAD_COLOR)

cv2.imshow("My cute cat", img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

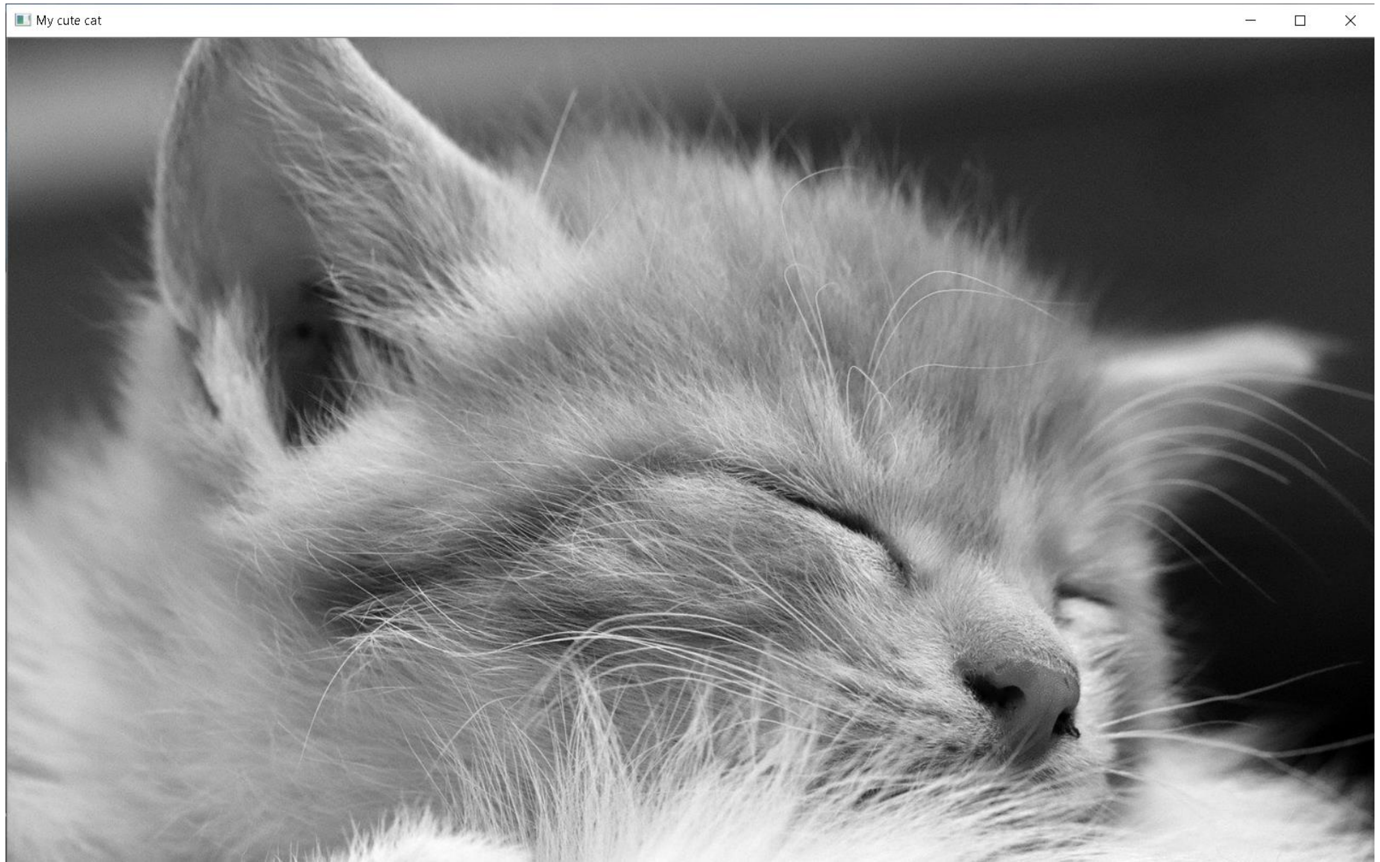

File Edit View Navigate Code Refactor Run Tools VCS Window Help
opencv [C:\Users\user\Documents\GitHub\EunSeong-Park.github.io\contents\2020_ITinerary\assets\session_4\opencv] - ...\main.py - PyCharm
opencv main.py
Project
main.py
cat-5249724_1280.jpg
opencv C:\Users\user\Documen
venv library root
.gitignore
cat-5249724_1280.jpg
main.py
External Libraries
Scratches and Consoles
My cute cat
Run: main
Terminal
Python Console
4: Run
6: TODO
Externally added files can be added to Git // View Files // Always Add // Don't Ask

```python
import cv2

img = cv2.imread("cat-5249724_1280.jpg", cv2.IMREAD_COLOR)

cv2.imshow("My cute cat", img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
C:\Users\user\Documents\GitHub\EunSeong-Park.github.
```

Image Show [3]

- **Try to use grayscale**
  - `cv2.imread("filename", cv2.IMREAD_GRAYSCALE)`



Thank you