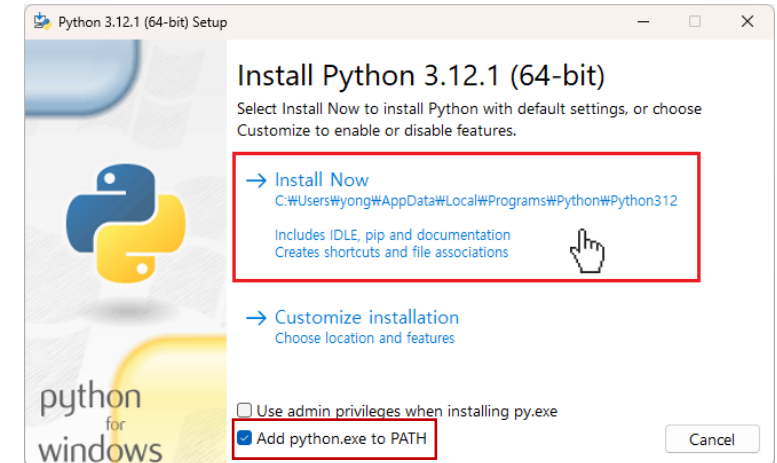


Python Programming

Jaewoong Han

Install Necessary Programs

- Python
 - <https://www.python.org/downloads/>
 - Choose latest version or appropriate version.
- Visual Studio Code (VSCode)
 - <https://code.visualstudio.com/>
 - It's most popular open-source code editor.
- Anaconda (not now...)
 - <https://www.anaconda.com/>
 - It is useful when you use virtual environment.
- CUDA, cuDNN for using GPU
 - <https://pytorch.org/get-started/locally/>
 - Find appropriate CUDA, cuDNN version for your computer
 - <https://en.wikipedia.org/wiki/CUDA>



Python coding you should study on your own

- Data Structures
 - List, Tuples, Dictionaries, ...
- Control Structures
 - if, elif, else
 - for, while
- Functions
 - Define functions, parameters, return, ...
- Object-Oriented Programming (OOP)
 - `class class_name(father_class_name):`
 - `super(class_name, self). __init__()`
- Error Handling
 - try, except, finally
 - raise `ValueError`, ...
- Modules, Packages, Library
 - `pip install ~`
 - `from ~ import ~`

How to run a Python file using Command Prompt (cmd)

```

PS C:\Users\> cd "C:\Users\Documents\MLCS\인턴십\MLCS-Internship-Program\001 Python Programming\002 numpy"
PS C:\Users\Documents\MLCS\인턴십\MLCS-Internship-Program\001 Python Programming\002 numpy> python -m test
70 90 100
[array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
        34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
        51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67,
        68, 69]), array([70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86,
        87, 88, 89]), array([90, 91, 92, 93, 94, 95, 96, 97, 98, 99]), array([100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112,
        113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125,
        126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138,
        139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151,
        152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164,
        165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177,
        178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190,
        191, 192, 193, 194, 195, 196, 197, 198, 199])]
PS C:\Users\Documents\MLCS\인턴십\MLCS-Internship-Program\001 Python Programming\002 numpy> python test.py
70 90 100
[array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
        34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
        51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67,
        68, 69]), array([70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86,
        87, 88, 89]), array([90, 91, 92, 93, 94, 95, 96, 97, 98, 99]), array([100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112,
        113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125,
        126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138,
        139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151,
        152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164,
        165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177,
        178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190,
        191, 192, 193, 194, 195, 196, 197, 198, 199])]
  
```

Module Path

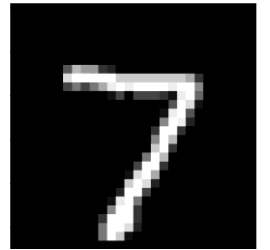
File Path

Handling csv files

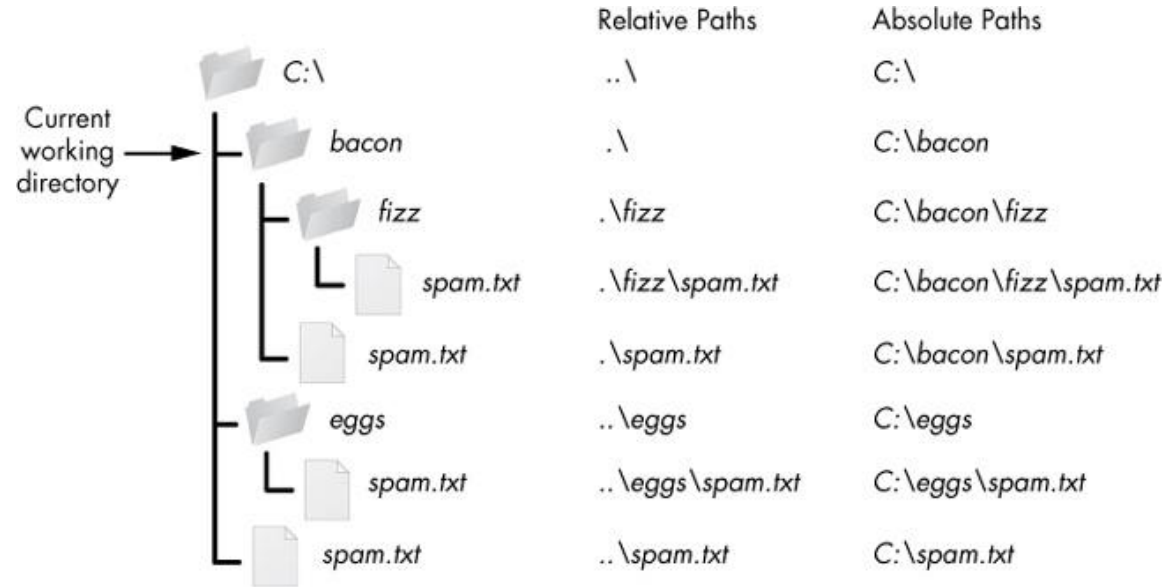
- Reading and writing csv files is one of the basic functions that we use a lot when dealing with data. In this course, we are going to read the MNIST dataset in the form of 1D vector and convert it into 2D vector form.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	84	185	159	151	60	36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	222	254	254	254	254	241	198	198	198	198	198	198	198	198	170	52	0	0	0	0	0	0
10	0	0	0	0	0	0	67	114	72	114	163	227	254	225	254	254	254	250	229	254	254	140	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	17	66	14	67	67	67	59	21	236	254	106	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	83	253	209	18	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22	233	255	83	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	129	254	238	44	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	249	254	62	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	133	254	187	5	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	205	248	58	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	126	254	182	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	75	251	240	57	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	221	254	166	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	3	203	254	219	35	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	38	254	254	77	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	31	224	254	115	1	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	133	254	254	52	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	61	242	254	254	52	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	121	254	254	219	40	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	121	254	207	18	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



Load files



- It is preferred to use relative paths in most cases
- Use 'os.path' library instead of using string path

Load files

```
PS C:\Users\ [redacted] > cd "C:\Users\ [redacted] \Documents\MLCS\인턴십\MLCS-Internship-Program_revise\002 Neural Networks"
PS C:\Users\ [redacted] \Documents\MLCS\인턴십\MLCS-Internship-Program_revise\002 Neural Networks> python
Python 3.8.10 (tags/v3.8.10:3d8993a, May 3 2021, 11:48:03) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import os, glob
>>> current_path = os.getcwd()
>>> current_path
'C:\\Users\\ [redacted] \\Documents\\MLCS\\인턴십\\MLCS-Internship-Program_revise\\002 Neural Networks'
>>> data_path = os.path.join('data', 'stl10_binary')
>>> data_path
'data\\stl10_binary'
>>> data = glob.glob(os.path.join(data_path, '*.bin'))
>>> data
['data\\stl10_binary\\test_X.bin', 'data\\stl10_binary\\test_y.bin', 'data\\stl10_binary\\train_X.bin', 'data\\stl10_bin
ary\\train_y.bin', 'data\\stl10_binary\\unlabeled_X.bin']
```

- 'os.path.join' creates path.
- 'glob.glob' returns the files in given pathB but note that the files are not sorted.

002 Numpy

Using Numpy Library

- Using numpy library to treat numerical data in matrix form will be a daily job when you become a data scientist.
- In this course, we are going to read 2D MNIST dataset (from the previous course) as numpy arrays.
- Subsequently, we will split the data into train, validation, and test dataset and save those files as .npz files.

Our current directory is

... /MLCS-Internship-Program/001 Python Programming/002 numpy

We need to load train datasets in

... /MLCS-Internship-Program/001 Python Programming/001 csv/ ~

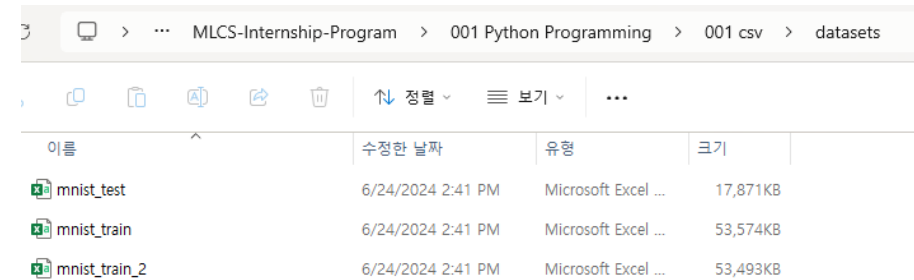
Create an empty numpy array.

```
import os, glob
import numpy as np

path_to_train_2d_datasets = os.path.join '..', '001 csv', 'outputs', 'train', '*.csv')
train_2d_files = glob.glob(path_to_train_2d_datasets)

# We already know that the width and height of the 2D MNIST dataset is 28.
train = np.empty((len(train_2d_files), 28, 28))

for data_idx, data_path in enumerate(train_2d_files):
    # Read the csv and replace the elements
    csv_data = # may need csv library to load the data
    for i in range(28):
        for j in range(28):
            train[data_idx, i, j] = csv_data[i][j]
```



이름	수정된 날짜	유형	크기
mnist_test	6/24/2024 2:41 PM	Microsoft Excel ...	17,871KB
mnist_train	6/24/2024 2:41 PM	Microsoft Excel ...	53,574KB
mnist_train_2	6/24/2024 2:41 PM	Microsoft Excel ...	53,493KB

Question
