

# Programming for AI (Python)

Guoliang Ma

The Chow Institute, 2025

# 0.1 Syllabus

- What to expect
  - An introduction of basic Python data storage and functions
  - A basic understanding of data science with numpy and pandas
  - Classic AI/ML algorithms (theory and implementation with scikit-learn)
  - Python classes and objects if time permits
- What are required
  - *Take notes and ask questions!*
  - Class attendance (occasional roll calls to be taken seriously)
  - Effective communication (especially for questions and clarifications)
    - I constantly make mistakes and write bugs...
  - High school math and some straightforward extensions
  - Please sit with someone with whom you can discuss questions.

## 0.1 Syllabus

- Group projects
  - Kaggle competition
  - Huggingface model exploration and evaluation
  - Self-selected topics
- Question bank
  - We'll try to provide an optional question bank for you to practice what you've learned in class.
  - For question-bank related issues, please consult TA.

## 0.1 Syllabus

- An **Ambitious** tentative schedule (32 meetings/64 hours)
  - Basic (24 meetings)
    - 6 meetings on Python basics: variable and its construction
    - 6 meetings on functions: basic, higher-order, algorithms, generator and coroutine
    - 8 meetings on numpy and pandas
    - 2 meetings on drawing figures with matplotlib
    - 2 meetings on basic OOP
  - Advanced (6 meetings)
    - 2 meetings on classic machine learning algorithms
    - 2 meetings on large language models and huggingface
    - 2 meetings for presentation (8 groups for two hours)
  - 2 meetings standby
- Don't panic! You'll be fine (see score decomposition).
- If you really want to learn something, put efforts.

## 0.2 About you

- Course taken?
  - data analysis/statistics, math, economics
- Programming experience?
  - C/C++, R, SAS, Python, etc.
- English
  - CET4/CET6, TOEFL/IELTS, courses
  - Need to get used to my accent
- Data analysis contest?
- More?

## 0.3 Classroom behavior

- Respect the instructor and fellow students
- Use appropriate words
- Mute your devices (QQ/WeChat etc.)
- Obey the laws and rules.

## 0.4 Whys before setting off

- Why Python?
  - The most widely used language in the AI/ML community
  - One of the easiest coding language for beginners
  - Versatility [noun for versatile, capable of doing various things]
- Why in this class?
  - A concentrated course targeting on data/ML/AI (hopefully)
    - Not focused on data analysis
    - Most are too vacuous [empty] to foster your *understanding*
  - A informational course providing you with background knowledge (hopefully)
    - Every design/feature comes form a need
    - You will know the key features of Python (pythonic as said by many)

## 0.5.1 Some general concepts (of a language)

- **△code**
  - Characters, letters, etc., a form carrying information
- Editors (**software**)
  - Where to write the **code**
- Compiler/Interpreter (**software**)
  - Translating our **code** to something computers understand
- Environment
  - Interpreter's home
- Environment manager (**software**)



## 0.5.1 Some general concepts (of a language)



### In-class exercise 0.5.1.1

1. List some editors that you have used.
2. Write your first line of code (in this class) in it

```
print("Hello Python!")
```

3. Do you know how to install a software?

## 0.5.2 Setting up a Python environment

- We need VS Code and anaconda
  - VS Code is a light-weight editor
  - Anaconda is a light-weight environment manager
- Download these two software from their official websites
  - <https://code.visualstudio.com/>
  - <https://www.anaconda.com/>

## 0.5.2 Setting up a Python environment

1. Let's first work with VS Code.

### In-class exercise 0.5.2.1

Launch VS Code and check out the layout [arrangement of elements].

1. What do you observe?
2. Find this 

2. Install two **extensions**: Python and Jupyter

## 0.5.2 Setting up a Python environment

### 3. Next, let's work with anaconda

#### In-class exercise 0.5.2.2

How do you open it?

What do you observe?

Anaconda is very different from many software that you have used. It has a black box for you to enter **code**. Such pure text/code interface [a place where you communicate with your computer] is called a **command line interface (cli)**.

You enter **code** into cli to use it.

## 0.5.2 Setting up a Python environment

- We introduce environment management commands for anaconda cli:
  - Create an environment

```
(base) C:\Users\glma>conda create -n MLPython
```

- Go in that environment

```
(base) C:\Users\glma>conda activate MLPython
```

## 0.5.2 Setting up a Python environment

- (continued from previous)
  - Run a python scripts

```
(MLPython) C:\Users\glma>python "F:\2025S Python\00-introduction\test.txt"
```

- Common mistakes

```
(MLPython) C:\Users\glma>python F:\2025S Python\00-introduction\test.txt
```

## 0.5.2 Setting up a Python environment

- (continued from previous)
  - Install a package

```
(MLPython) C:\Users\glma>conda install numpy
```

- Wait! conda? Or pip?

## 0.5.2 Setting up a Python environment

### 4. Turning back to VS Code...

1. Write some code here

```
sentence = "Hello Python!"  
print(sentence)
```

2. Run [execute] the code



## 0.5.2 Setting up a Python environment

### In-class exercise 0.5.2.3

1. How did you print “Hello Python!”?
2. What are the differences?

## 0.6 What we can do with Python

Example 0.6.1 speech recognition

```
import whisper # conda install -c conda-forge ffmpeg openai-whisper

model = whisper.load_model("base")
result = model.transcribe(audio="audio.m4a", fp16=False)
print(result)
```

## 0.6 What we can do with Python

Example 0.6.2 picture recognition

```
import easyocr # pip install easyocr

reader = easyocr.Reader(['en'], gpu=False)
result = reader.readtext('F:/2025S Python/00-introduction/good.png', detail=0)
result
```

## 0.6 What we can do with Python

Example 0.6.3 picture generation

```
from openai import OpenAI # pip install openai

client = OpenAI(api_key=OPENAI_API_KEY)

response = client.images.generate(
    model="dall-e-2",
    prompt="driving in Ames in autumn",
    size="1024x1024",
    quality="standard",
    n=1,
)
print(response.data[0].url)
```

# 0.6 What we can do with Python

## In-class exercise 0.6.1

1. Based on the previous three examples, can you summarize the procedure of working on a task in Python?
2. Where does the model come from?