# Python programming for beginners

Yao Zhang - Aalto University

# 

- Learn how to install python, useful python IDEs (PyCharm and Jupyter notebook)
- Learn basic grammar of python, how to write a simple code like 'Hello world'
- Learn how to install useful package, load data from file system
- A simple example for how to collecting data from your sensor

## What is Python?

• Python is the current most popular programming language for web development, software development, mathematics, system scripting...

# • Why Python?

- Python has a simple syntax similar to the English language.
- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has lots of easy-to-use and easy-to-install packages that make it possible to quickly implement complex functionality.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

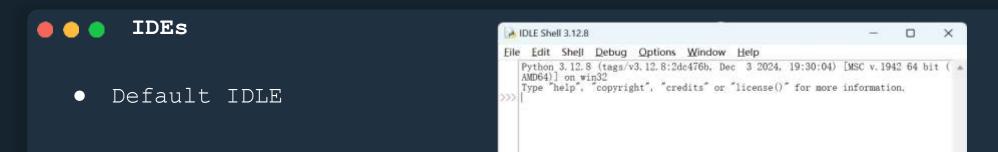
# ● ● ● How to install Python

• How to check whether you have python on your computer

C:\Users\yao zhang>python -V Python 3.12.8

• Different platforms have different installation methods

• https://realpython.com/installing-python/#windows-how-to-check-or-get-python



PyCharm

slidesmania.com

# Project Pro

# Jupyter Notebook https://jupyter.cs.aalto.fi/hub/spawn

```
[2]: # Loading Libraries
     # Numpy is utilized to manipulate data containers
     import numpy as np
[3]: # Loading the data recorded for the class "walking"
     # This Loads the data into a NumPy array for further processing or analysis in Python
     # The data is Loaded into a "matrix" structure with two dimensions
     data = np.loadtxt("/notebooks/smartwearable2025/walking.csv", delimiter=",")
[4]: # Here we access the number of components in the first dimension (number of samples)
     num_samples = data.shape[0]
     print(f"Number of samples: {num_samples}")
     # Here we access the number of components in the first dimension (number of sensor channels)
     num channels = data.shape[1]
     print(f"Number of sensor channels: {num_channels} corresponding to ACC X, Y, Z and GYRO X, Y, Z")
     Number of samples: 61266
     Number of sensor channels: 6 corresponding to ACC X, Y, Z and GYRO X, Y, Z
[5]: # Let's have a Look at the data
     print("
                 ACC X,
                              ACC Y,
                                         ACC Z,
                                                   GYRO X,
                                                                GYRO Y.
                                                                            GYRO Z")
     print(data)
          ACC X,
                       ACC Y,
                                  ACC Z,
                                             GYRO X,
                                                         GYRO Y.
                                                                     GYRO Z
     [[ 6.3906e-02 -6.5013e-02 -1.1267e-01 4.1905e-03 2.7495e-02 -8.9308e-03]
      [ 1.5697e-02 7.7307e-04 -1.1857e-01 -3.7507e-03 3.0604e-03 -8.9308e-03]
      [-1.5182e-03 5.3167e-05 -8.9513e-02 -2.4520e-02 -5.4917e-03 7.5625e-03]
      [-2.9038e+00 1.7022e+00 -5.0675e-01 -7.1477e-01 -3.5953e-03 -3.5306e-01]
      [ 7.2340e-01 -1.1946e+00 2.9736e-01 -6.5512e-01 -7.7911e-01 -4.1485e-01]
      [ 3.6843e+00 -2.3661e+00 1.7814e-01 -2.0558e-01 -2.2197e-01 -2.1458e+00]]
```

# onda(optional)

- Execute python code is based on python interpreter (python environment)
- Different environment means different packages, different versions...
- A good habit: each project corresponds to a python virtual environment
- Tool: Anaconda or miniconda
- Instruction viedo: https://www.youtube.com/watch?v=4DQGBQMvwZo



- https://www.w3schools.com/python/default.asp
- How to install useful package
- How to write simple python code
- How to load files from system
- How to add breakpoint and debug

# Receiving data by Python

## Important things:

- 1. connect Arduino(sender) and your computer(server) to same WiFi(like: your phone hotspot).
- 2. changing your Arduino code with your own WiFi id and password.
- 3. get server IP after you connect your computer to the same WiFi.
- 4. keep Port number as same for both sender and server part.



https://www.avast.com/c-how-to-findip-address

For Windows: ipconfig

For Mac: ifconfig | grep "inet"

## Collecting and labelling data

```
wifi arduino.ino
       #include <SPI.h>
       #include <WiFiNINA.h>
       // WiFi credentials
       const char* ssid = "zyyyyy"; // Replace with your WiFi SSID
       const char* password = "zy561526"; // Replace with your WiFi password
       // Server details (your computer's IP and port)
       const char* serverIP = "172.20.10.3"; // Replace with your computer's local IP
       const int serverPort = 5001; // Must match the Python script
  11
       int sensorPin1 = A2;
       int sensorValue1 = 0;
  14
       WiFiClient client;
  15
  16
  17
       void setup() {
  18
           Serial.begin(115200);
           while (!Serial);
  19
  20
  21
           // Connect to WiFi
  22
           Serial.print("Connecting to WiFi...");
  23
           while (WiFi.begin(ssid, password) != WL_CONNECTED) {
               Serial.print(".");
 25
               delay(1000);
  26
           Serial.println("Connected!");
 27
  28
```

```
HOST = "0.0.0.0" # Listens on all available interfaces
PORT = 5001 # Must match Arduino's serverPort
# Create a socket (IPv4, TCP)
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
server_socket.bind((HOST, PORT))
server_socket.listen(1)
print(f"Listening for connections on port {PORT}...")
client_socket, client_address = server_socket.accept()
print(f"Connected to {client_address}")
csv_filename = "./saving_data/training_sensor_data_1.csv"
                                                                            Listening for connections on po
with open(csv_filename, mode="w", newline="") as file:
                                                                            Connected to ('172.20.10.2', 50
   writer = csv.writer(file)
                                                                             Received: 15
 writer.writerow(["Timestamp", "Sensor Value", "label"]) # Write header
                                                                             Received: 16
                                                                            Received: 15
                                                                             Received: 15
           data = client_socket.recv(1024).decode("utf-8").strip()
                                                                            Received: 16
           if data:
               timestamp = datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S"
                                                                            Received: 16
               print(f"Received: {data}")
                                                                            Received: 15
                                                                             Received: 15
                                                                             Received: 14
               writer.writerow([timestamp, data, "pressed"])
               file.flush() # Ensure data is written immediately
                                                                            Received: 16
   except KeyboardInterrupt:
                                                                             Received: 16
                                                                             Received: 14
                                                                             Received: 16
```