

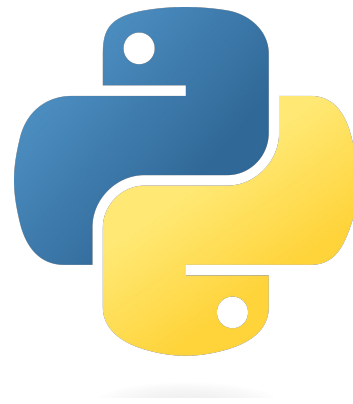
Basic usage of Python

Sergio Paniego Blanco
@sergiopaniego
sergiopaniegoblanco@gmail.com
<https://sergiopaniego.github.io/>



History

- Created in 1989
- First version in 1991
- Guido Van Rossum
- Oriented to scripting
- Open source language
- High-level language



Why Python?

- 50% of respondents in Stack Overflow Survey (most popular language). <https://survey.stackoverflow.co/2024/>
- Used in different settings: web development, AI, scripting...

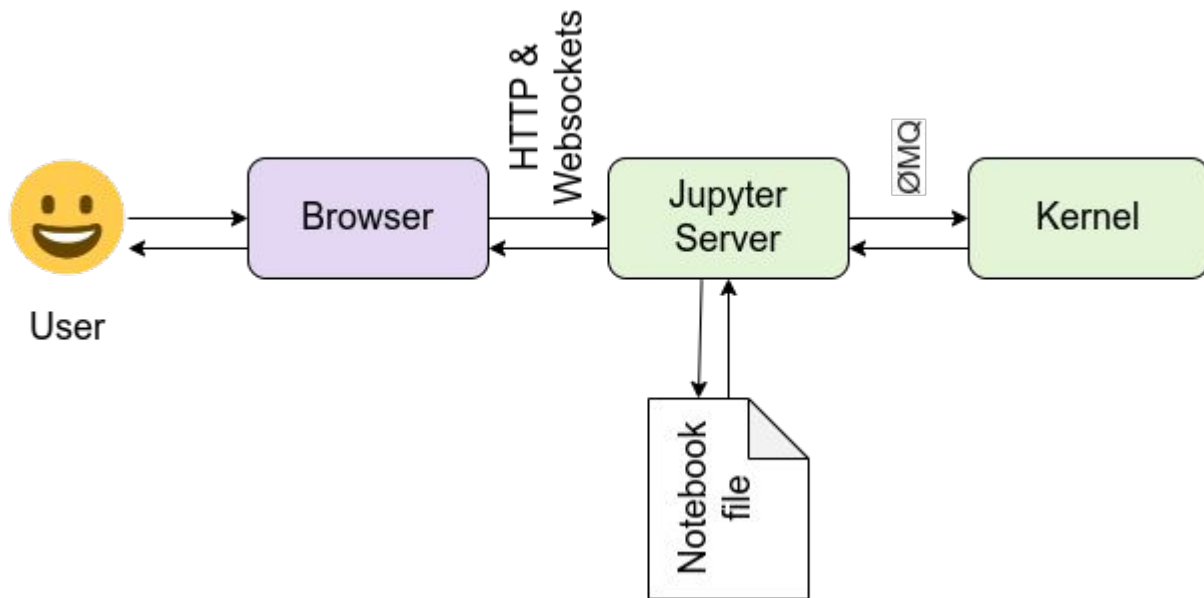
Tools for programming Python

- **Notebooks:** Jupyter Notebook, Google Colab.
- **Local editors:** PyCharm, Visual Studio Code.
- **Remote editors:** Replit.



How does a notebook work?

- Cells.
- Combining code (Python) + text (Markdown).



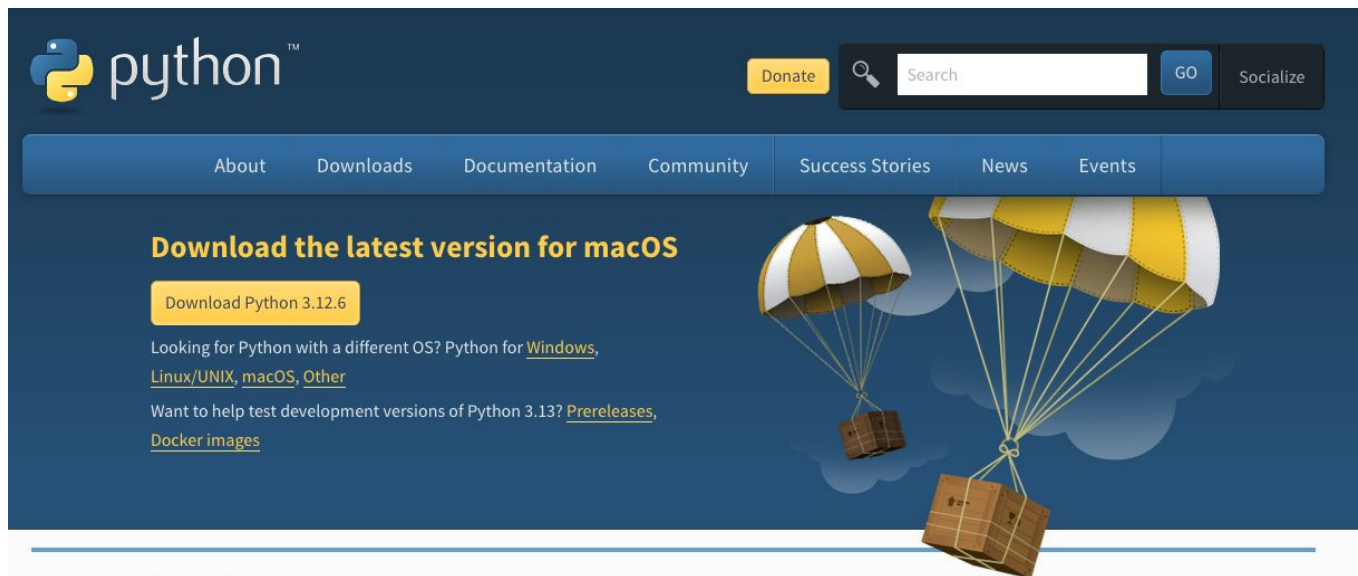
Setup

- Python version (python 3.x)
- Package management (virtualenv or Anaconda)
- IDE (VS Code, Jupyter Notebook...)

Installing Python

- python.org/downloads/

```
sudo apt update  
sudo apt install python3 python3-pip  
python3 -V  
pip -V
```

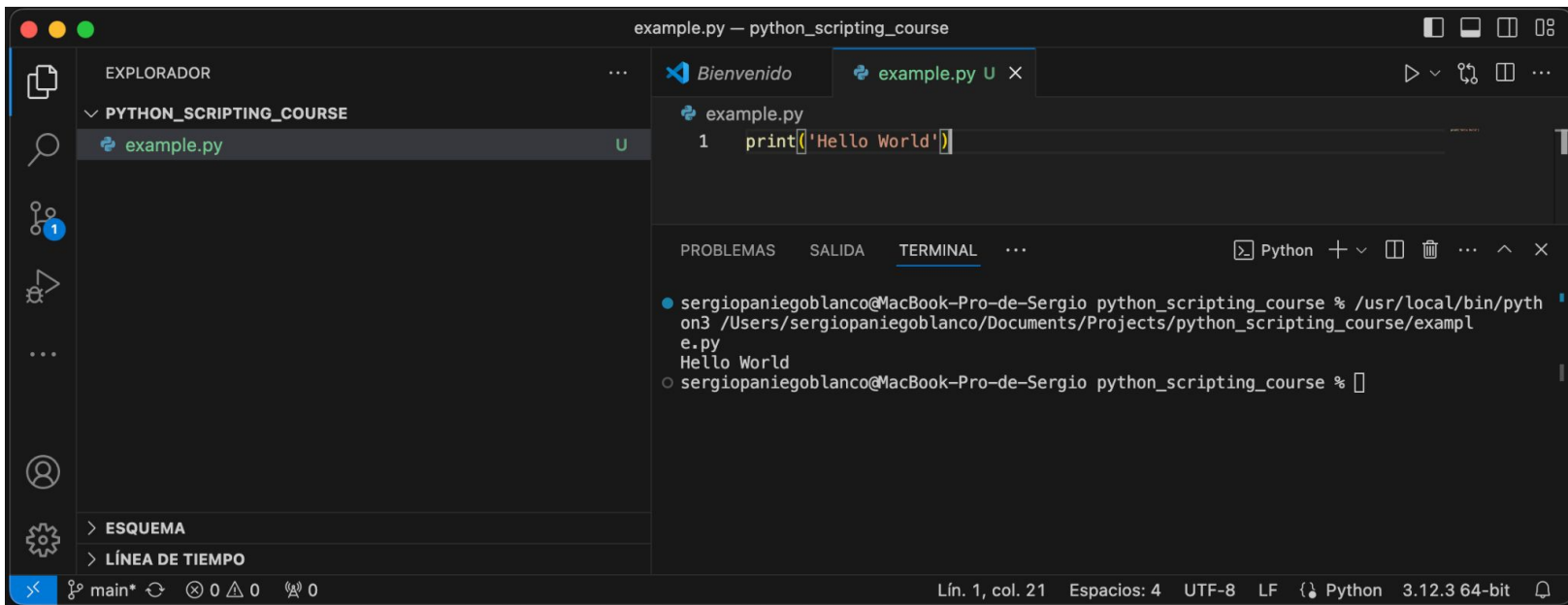


Package management

```
cd /path/to/your/project  
virtualenv venv  
source venv/bin/activate  
deactivate
```

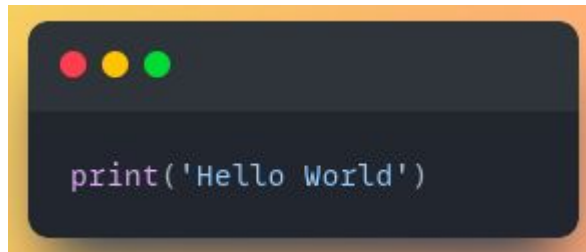
```
python -m venv venv
```


IDE



- <https://code.visualstudio.com/docs/python/environments>

Hello World in Python

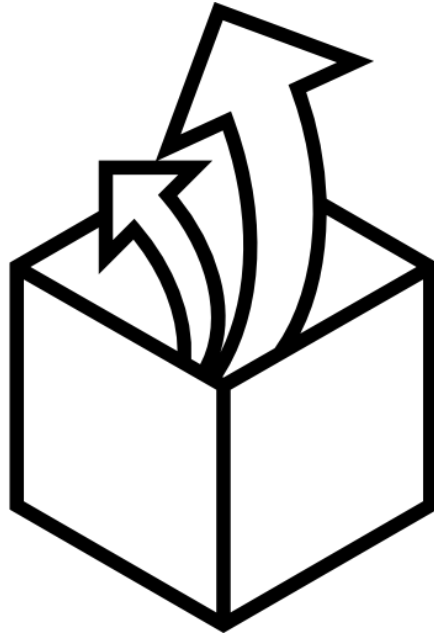


Hello World in Python



Variables in Python

- Memory space reserved to store data.
- It can store different data types: text, numbers...



Variables in Python

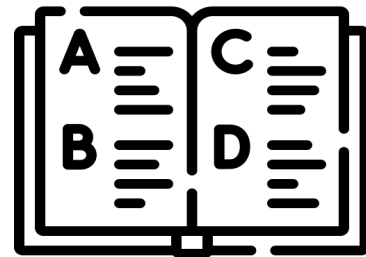
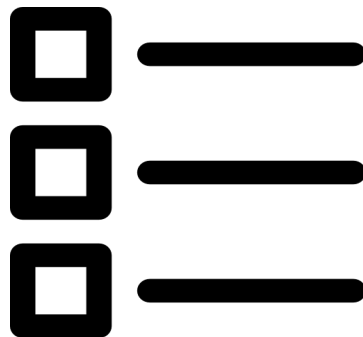


Basic data types

- Category that defines what a variable can contain.

1

S

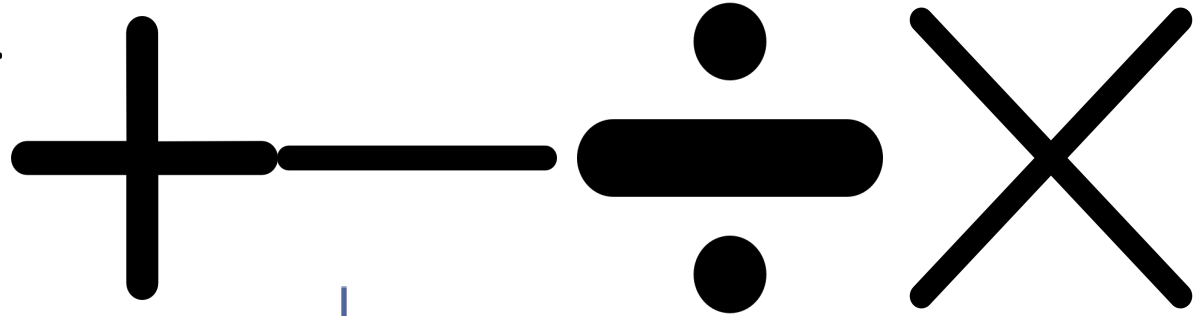


Basic data types



Operators

- Category that defines what a variable can contain.
- Different operands types
 - Arithmetic.
 - Comparison.
 - ...



Operators

↓ ↓ ↓

1 + 4 - 2 = 3

Operators

↓ ↓ ↓

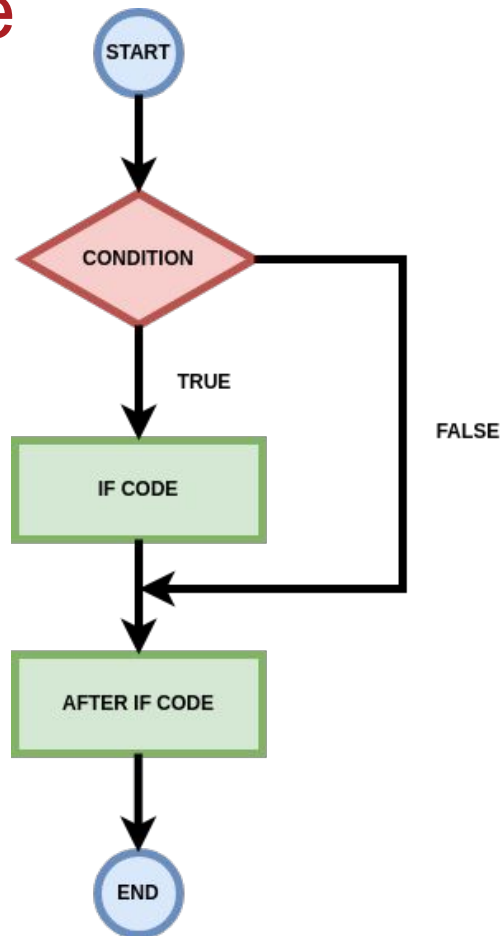
1 * 4 / 2 > 1

Operators



Decision structure

- Construct that allows decision-making based on conditions.
- Flow control.
- `if/else`.

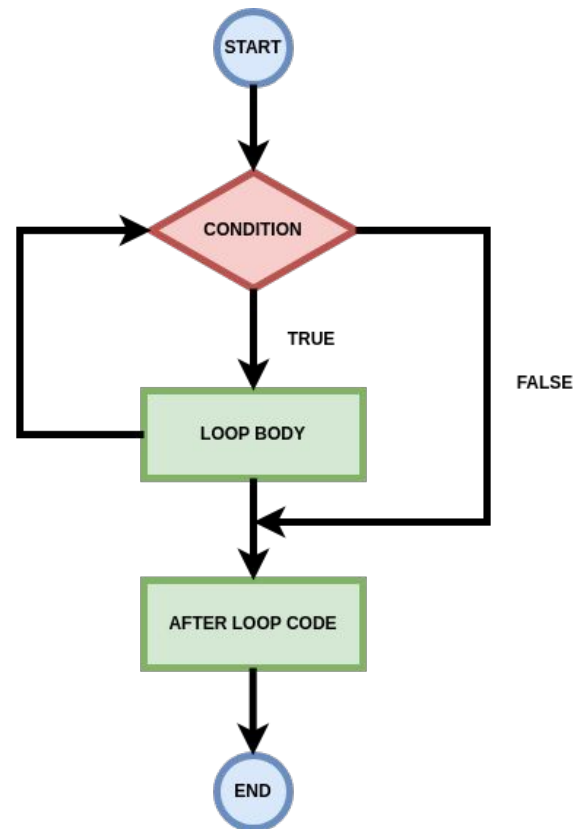
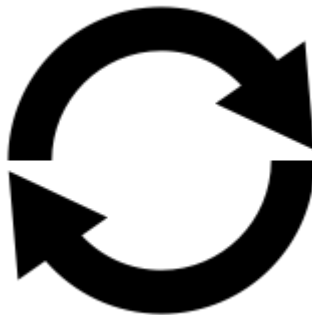


Decision structure



Repetition structure

- Construct for executing code repeatedly. Two cases:
 - While a condition lasts.
 - Specific number of times.
- `for/while`.

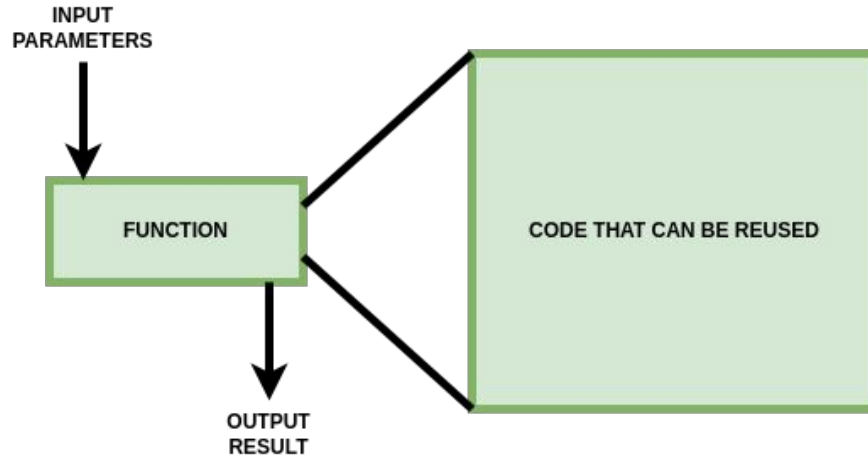


Repetition structure



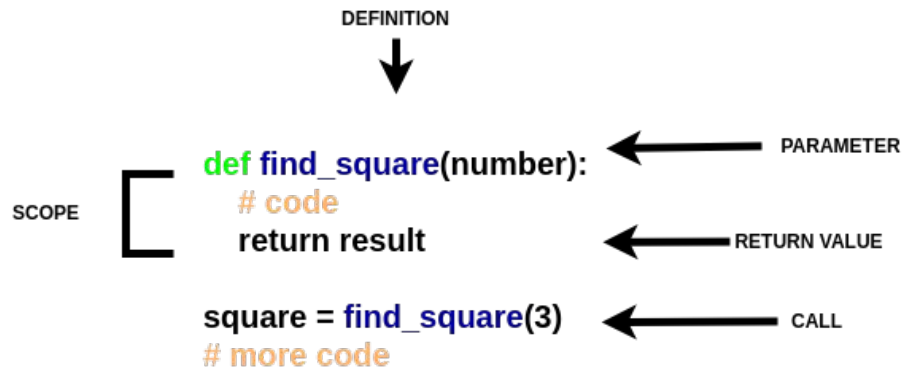
Functions

- Block of code that performs a specific task.
- Reusable along the program.
- Allow division of code into smaller parts.



Functions

- Features:
 - Definition
 - Call
 - Parameters
 - Return
 - Scope

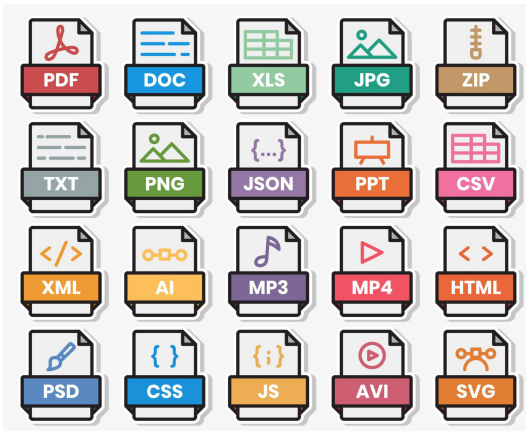


Functions



Reading and writing files

- We can use different file types inside a python script: `txt`, `csv`, `json`...
- We use `with` structure so the file is closed when the pipeline is finished.



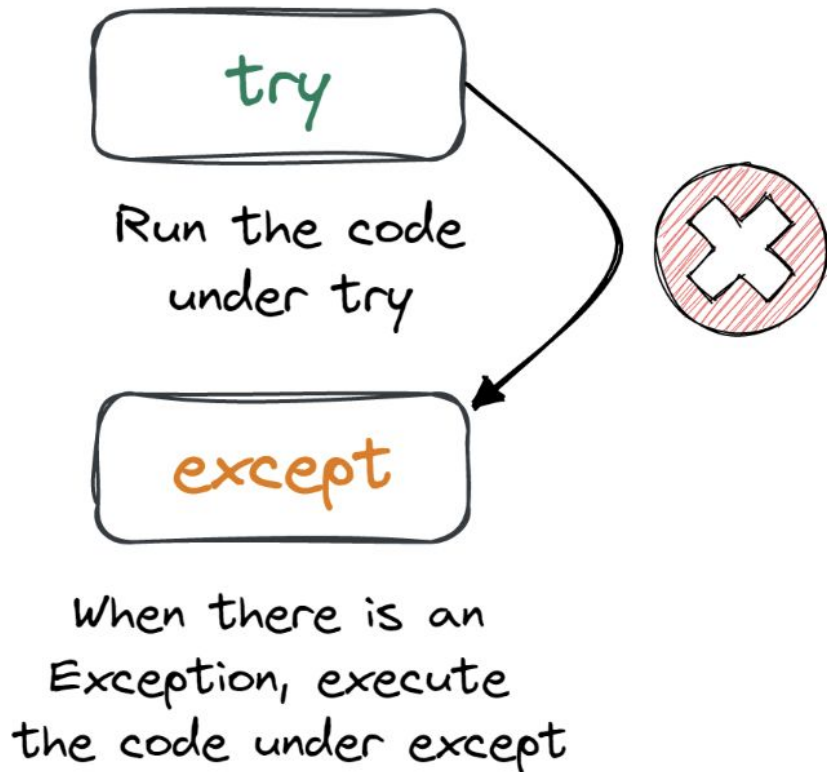
Reading and writing files



Errors and exceptions

- **Error:** problem that occurs during the execution of a program that prevents it to continue.
- **Exceptions:** way of handling and recovering from errors.
- **Examples:** `SyntaxError`, `ZeroDivisionError`, `NameError`, `TypeError`...
- **Structure for handling:** `try-except-finally`

Errors and exceptions



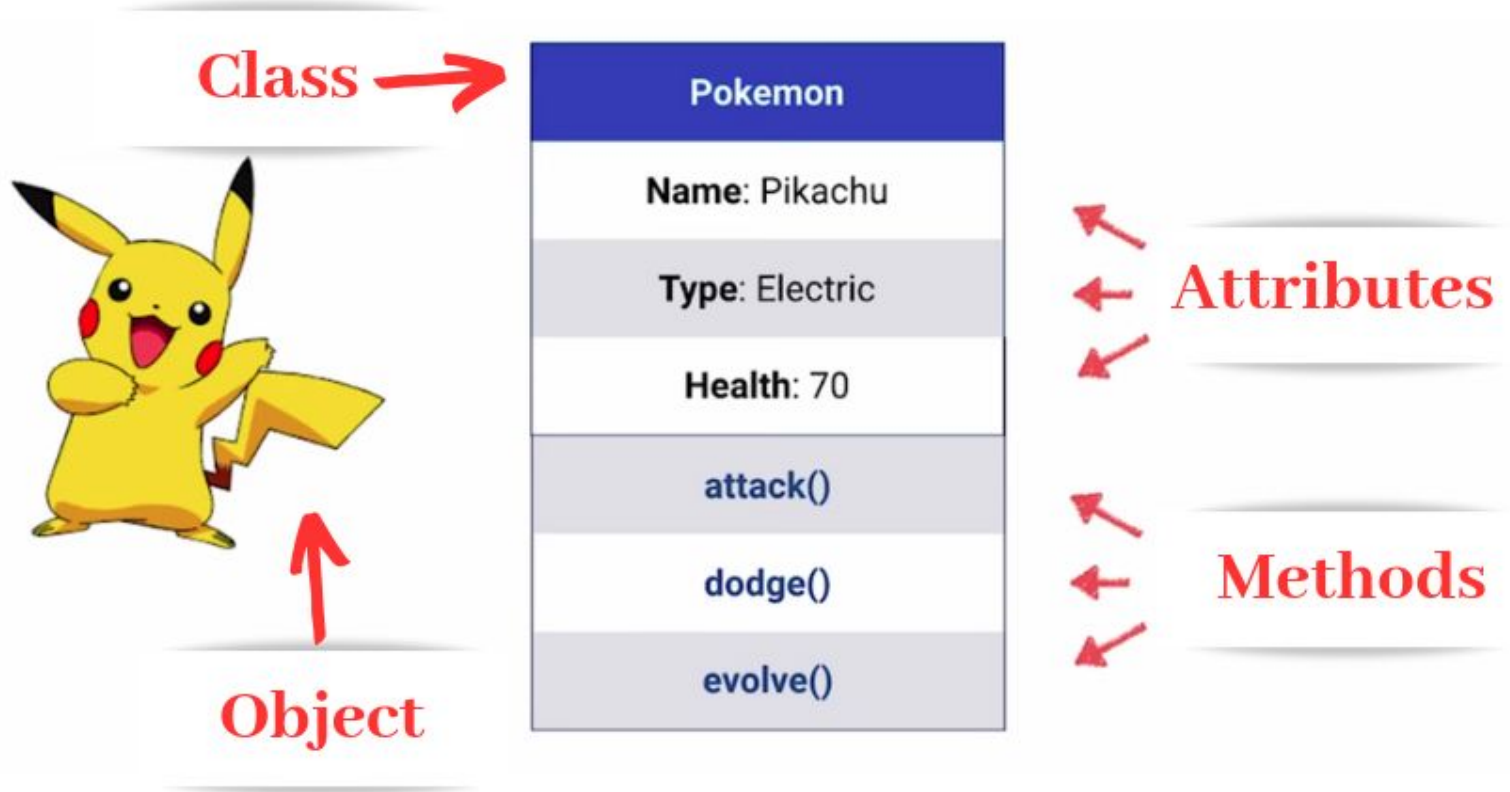
Errors and exceptions



Object oriented programming

- Programming paradigm.
 - Model the world using objects: data (attributes) + behaviors (methods).
 - Everything is an object in python
- Key concepts:
 - Class and object
 - Attribute and method
 - Abstraction: hiding complex implementation details and showing only the essential.
 - Encapsulation: bundling data and methods within a class and restricting access to them.
 - Inheritance: a class inherits attributes and methods from another class, promoting code reuse.
 - Polymorphism: allows different classes to be used interchangeably if they implement similar methods.

Object oriented programming



Object oriented programming



