

December 3, 2024

## 1 Advent of Code 2024

Welcome to my solution repository for the Advent of Code (edition 2024)! This repository contains (hopefully) a solution to all exercises. The solutions should be simple enough that anyone with a minor Python knowledge, but a solid internet connection and a curious mind should be able to comprehend all of them. I did my best to implement them in the most readable, yet elegant way and pythonic way I could imagine. Below, we will talk quickly revise each of the solutions, building upon what we learned in previous quests!

### 1.1 Day 1: *Historian Hysteria*

The first two quests are simple, but require knowledge on how to read files in Python. I implemented the solution in the files *aoc\_01.py* and *aoc\_02.py*, however, it's best practice to keep the interaction with your file system as short as possible. That is why I always declared and implemented the input handling functions in *aoc\_utils.py*.

For this first day, we're looking at the function `read_lists_01` (part of the utilities file), which reads the contents of an input file into two lists *left\_list* and *right\_list*.

```
def read_lists_01(input_file="data/24-aoc-01.in"):
    left_list = []
    right_list = []

    with open(input_file, "r") as file:
        for line in file:
            values = line.split(' ')

            left_list.append(int(values[0]))
            right_list.append(int(values[-1]))

    left_list.sort()
    right_list.sort()

    return left_list, right_list
```

This function has some features that might look unfamiliar to those new to Python.

- The function's declaration shows that it takes one argument, called `input_file`. We also see that this argument is set to `"data/24-aoc-01.in"` if it is omitted. This means that if we call the function without specifying an input file, this argument will be set to that standard value.
- We are opening a file in *read-only* mode. This is done with the `with open(filename, mode)` `as` statement. This statement opens a file in the required mode (*write*, *read*, *append*) and gives you a handle to the file. With this handle, you can use the file as an iterator which returns the lines of the file in-order. There's some other functionality for these file handles, but that would take us too far for this day.