Introduction

The given log file in the resource section has to be evaluated with the aim of determining the proportions of HTTP status codes. The log file is in common log format (CLF).

The status code of the server is 503 in the example below.

```
53.149.82.223 - - [07/Apr/2022:16:16:40 +0000] "POST /playbooks HTTP/1.1" 503
4103
```

Before you start, verify the integrity of the downloaded log file in the resource section by calculating and comparing the MD5 checksum.

```
MD5 (c5-access.log) = f835044521c653a1fca956fbde72dd02
```

Goal and tasks

Develop a Python program which meets the following requirements:

- The program reads the given log file c5-access.log as an argument. If no argument is specified on the command line, a corresponding message must be printed and execution must be aborted.
- The program reads and counts the occurrence of every status code in the log file. Hint: We recommend the use of the class collections.Counter.
- Errors in file handling: The program must print a specific error message to the console, distinguishing errors while reading files.
- The program creates an ordered list with pairs of keys (status code) and values (counter), ordered by status code, ascending.
- The program writes the ordered key value pairs into a YAML file named results.yml, representing every pair on a new line.
- The program generates a pie chart with labels (status code) and its percentage share, rounded to two decimals. Save that pie chart as statuscode.png. Hint: We recommend the use of matplotlib.
- In terms of code quality, the program must be commented accordingly.

Submission

Submit a ZIP archive containing your Python program (regardless of whether it is finished or not) and the generated outputs results.yml and statuscode.png (if available). Ensure that used external modules are properly declared with import statements.