Exercise Sheet 05 – Handling Errors and Debugging

Sebastian Höffner Aline Vilks

Deadline: Mon, 8 May 2017 08:00 +0200

Submission

By the end of this sheet you will have a number of different files to submit. In Stud.IP you will have a directory for your own group, please upload them there. It is easier for you if you just archive all files and upload your archive (preferably zip), but it is okay if you upload them one by one.

Exercise 1: Reversed class room

Sebastian solved a task. Or at least, he tried really hard. But unfortunately, he again failed miserably. Of course Aline could fix it, but she is busy correcting your homework submissions, so she needs your help:

a) Open the accompanying file `whatdoesitdo.py` and examine it. What does it do? Add/update documentation comments for all functions, describing what their inputs and expected outputs are, what the functions do. You are allowed to rename it.
b) Write down what is wrong, so that Sebastian will be able to correct it himself the next time.
c) Correct the code.
d) Make sure everything is properly indented. Be sure to also
   - correct any potential white space conventions Sebastian disobeyed.
   - ensure enough blank lines between functions and import statements.

Hints:

- There are about 20 errors in the script, depending on how you count them.
- Some are already found by spyder or when you try to run it with python.
- Some are very subtle and hard to spot, and can only be found when you read carefully or try out the program's functionality (for which you have to correct the other errors first).

- Try to get it to run first, then search for logical mistakes.
- It's okay if you don't find all, but we expect you to get the code to run somehow.

Exercise 2: Reading and writing csv files

At the UC Irvine Machine Learning Repository[1] you can find a lot of interesting data sets to do some Machine Learning. While we don't want to do Machine Learning today, we still want to read and process some data.

Check out the famous iris dataset[2] and download the file `iris.data`. If you read the website carefully, you will find some problems with the data set:

> This data differs from the data presented in Fishers article

Which is followed by descriptions of the differences. Your task is to correct these differences using a script `iris_correction.py` and save a corrected version named `iris.csv`. (You don't need to submit the corrected file, just write the code to correct it.)

You can take a look into the `csv` module to read and write the files, but it is also possible to solve this exercise without.

Now that we have some corrected data, let's write another script `iris_statistics.py`. It should output (at least) the following simple statistics:

- How many rows are there in total?
- How many instances of each of the three iris classes are in the data set?
- What is the mean sepal length
  - of all samples?
  - of iris setosa?
- What is the median sepal width
  - of all samples?
  - of iris virginica?
- What is the mode of petal lengths of iris versicolor?

Remember to use functions where appropriate, e.g. to calculate the mean, median and mode. Add tests (i.e. `asserts`) for your statistics functions if you wrote your own. Make sure to add sufficient documentation.

[1] https://archive.ics.uci.edu/ml/
[2] https://archive.ics.uci.edu/ml/datasets/Iris