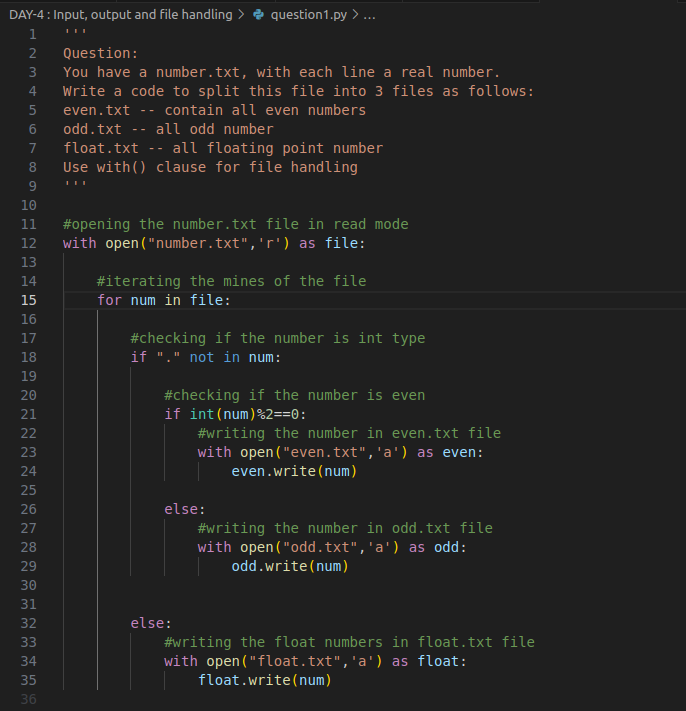
**INPUT-OUTPUT AND FILE HANDLING**

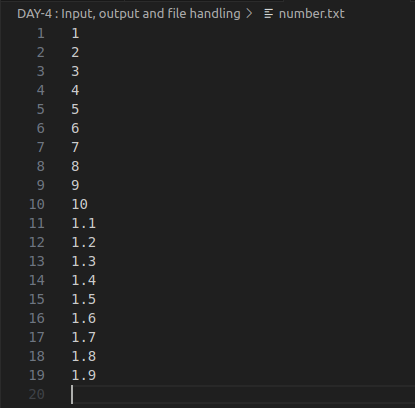
[**GITHUB LINK**](https://github.com/Nihal-Ahmad-TTN/python-exercise)

1. You have a number.txt, with each line a real number. Write a code to split this file into 3 files as follows: even.txt -- contain all even numbers odd.txt -- all odd number float.txt -- all floating point number Use with() clause for file handling

**Python file:**

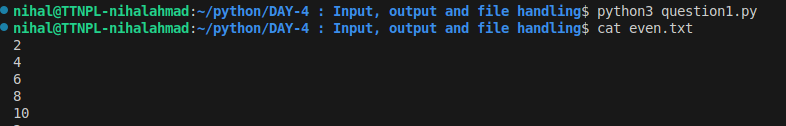


**Input file : number.txt**

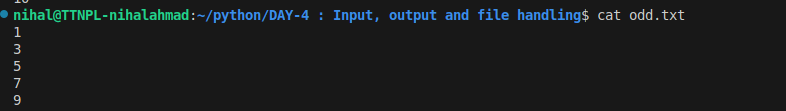
****

**OUTPUT**

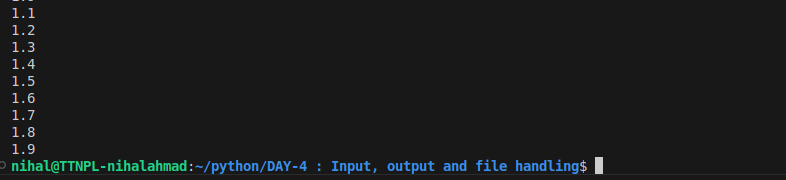
1. **Content of even.txt file**



1. **Content of odd.txt file**

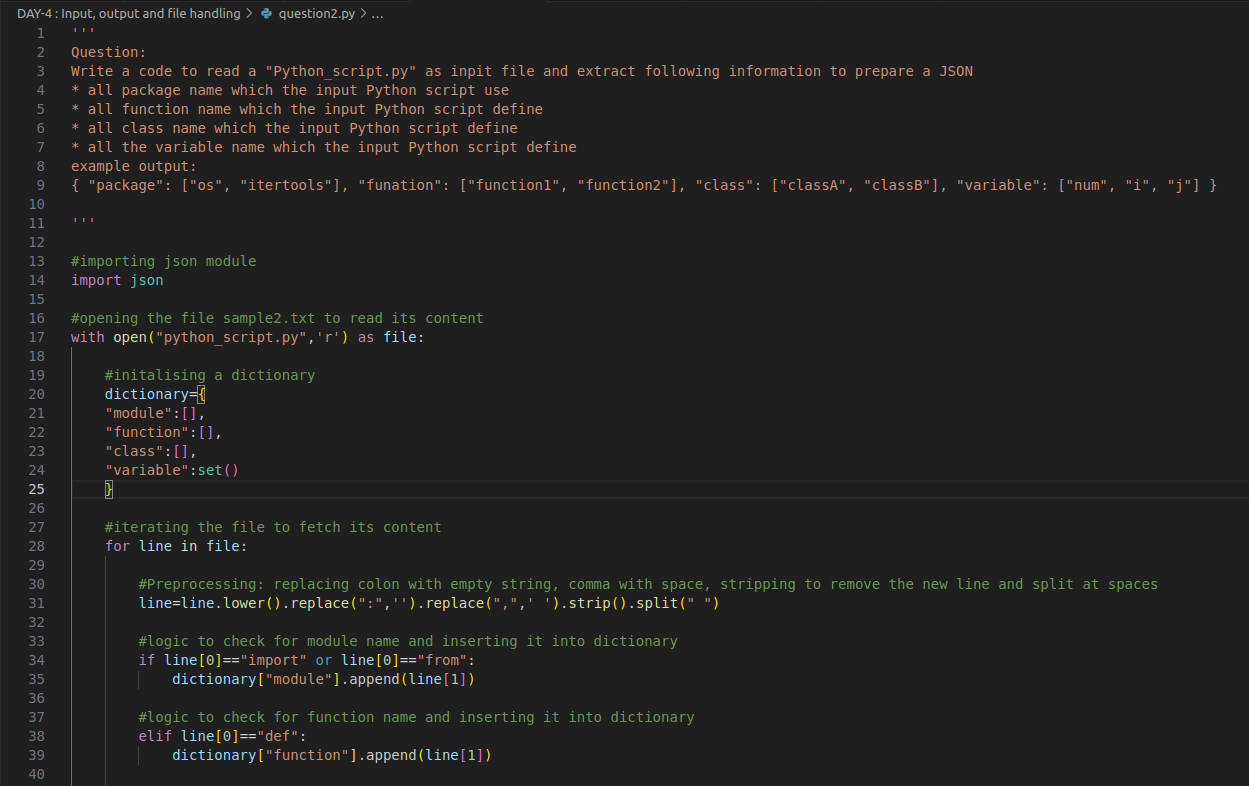


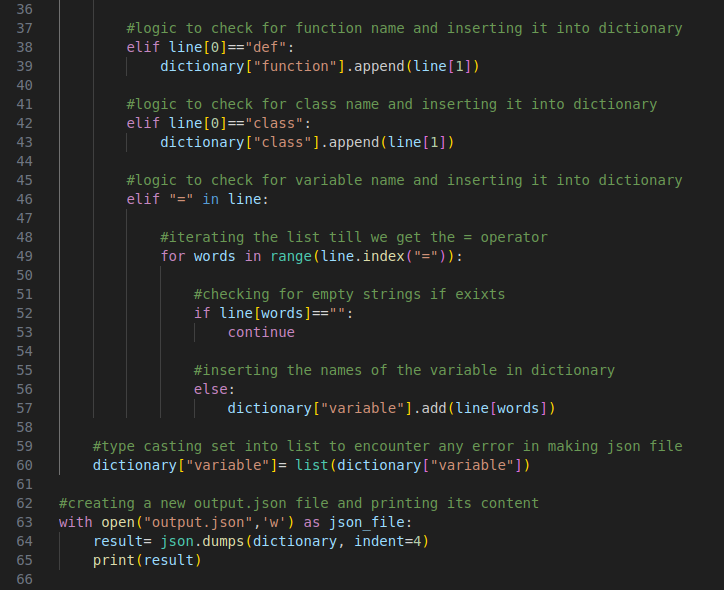
1. **Content of float.txt file**



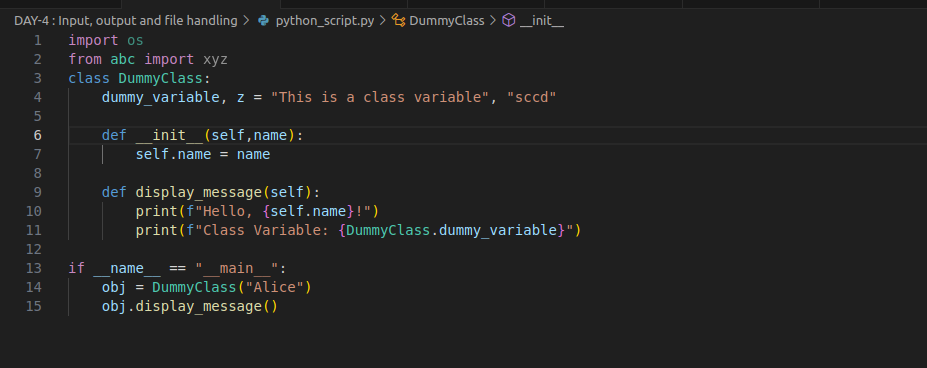
1. Write a code to read a "Python\_script.py" as input file and extract following information to prepare a JSON \* all package name which the input Python script use \* all function name which the input Python script define \* all class name which the input Python script define \* all the variable name which the input Python script define example output: { "package": ["os", "itertools"], "funation": ["function1", "function2"], "class": ["classA", "classB"], "variable": ["num", "i", "j"] }

**Python file:**

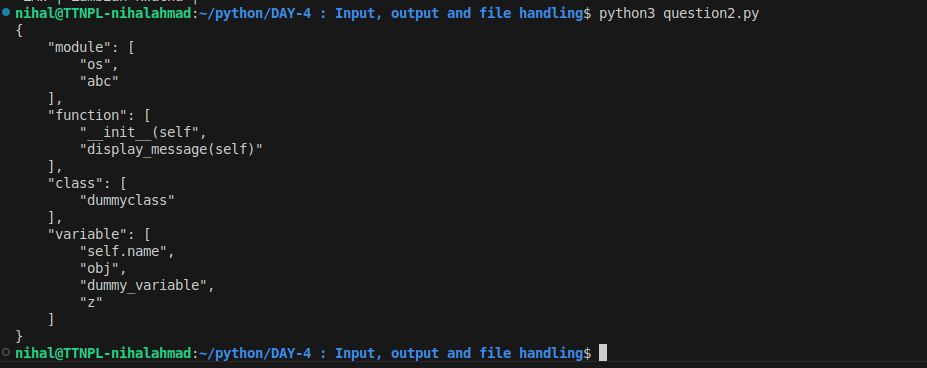
****

****

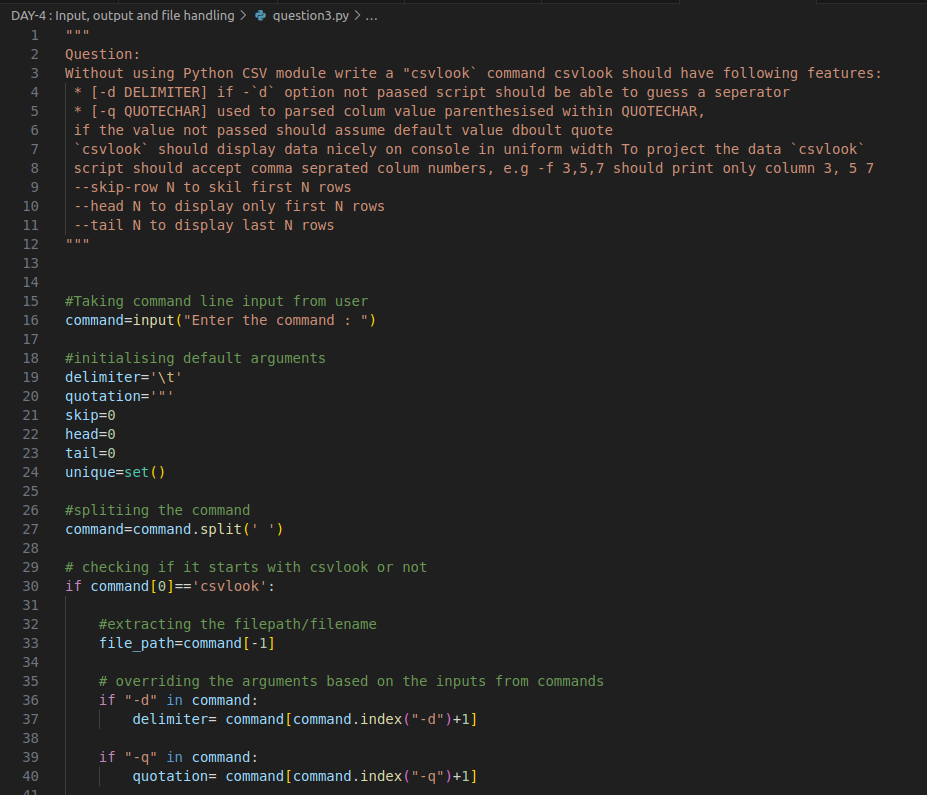
**INPUT FILE: python\_script.py**

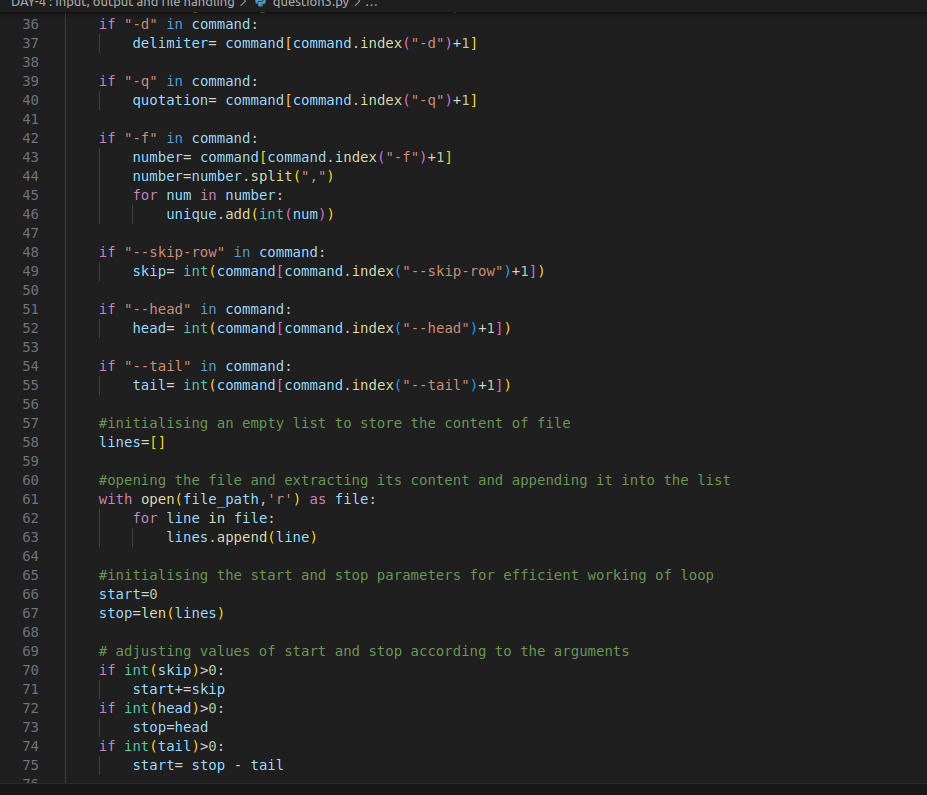
****

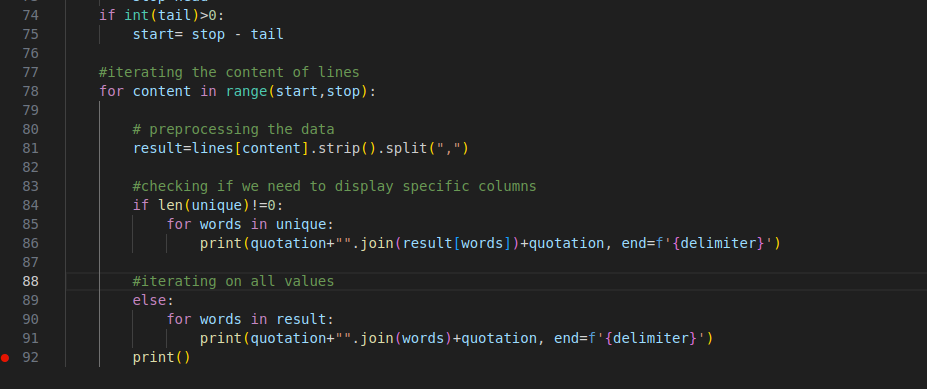
**OUTPUT:**

****

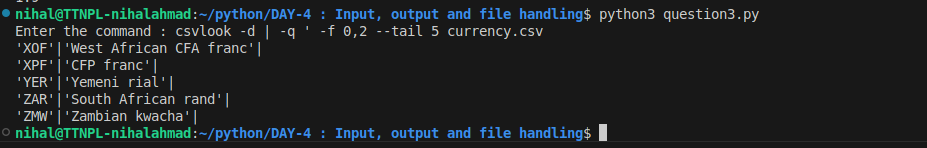
1. Without using Python CSV module write a "csvlook` command csvlook should have following features: \* [-d DELIMITER] if -`d` option not passed script should be able to guess a separator \* [-q QUOTECHAR] used to parsed colum value parenthesised within QUOTECHAR, if the value not passed should assume default value double quote `csvlook` should display data nicely on console in uniform width To project the data `csvlook` script should accept comma separated column numbers, e.g -f 3,5,7 should print only column 3, 5 7 --skip-row N to skip first N rows --head N to display only first N rows --tail N to display last N rows







**OUTPUT:**

****