Assignment (I/P-O/P & File Handling)

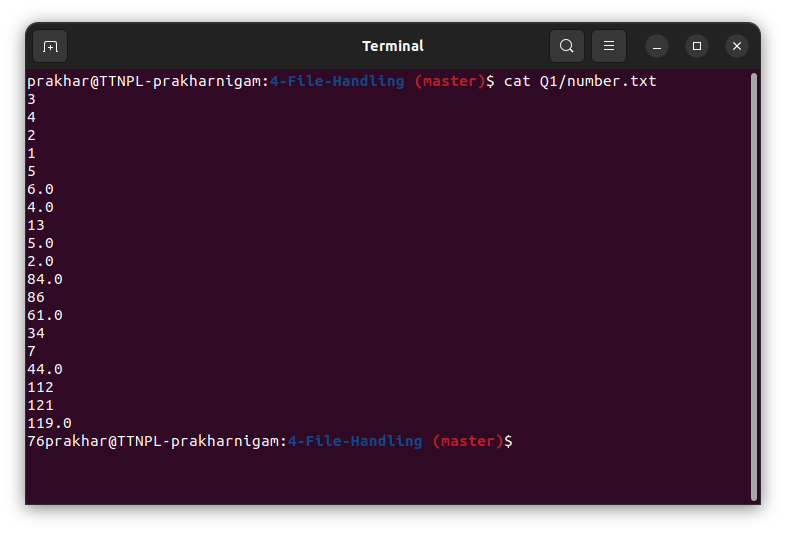
**Q1. You have a number.txt, with each line a real number. Write a code to split this file into 3 files as follows:**

even.txt -- contain all even numbers

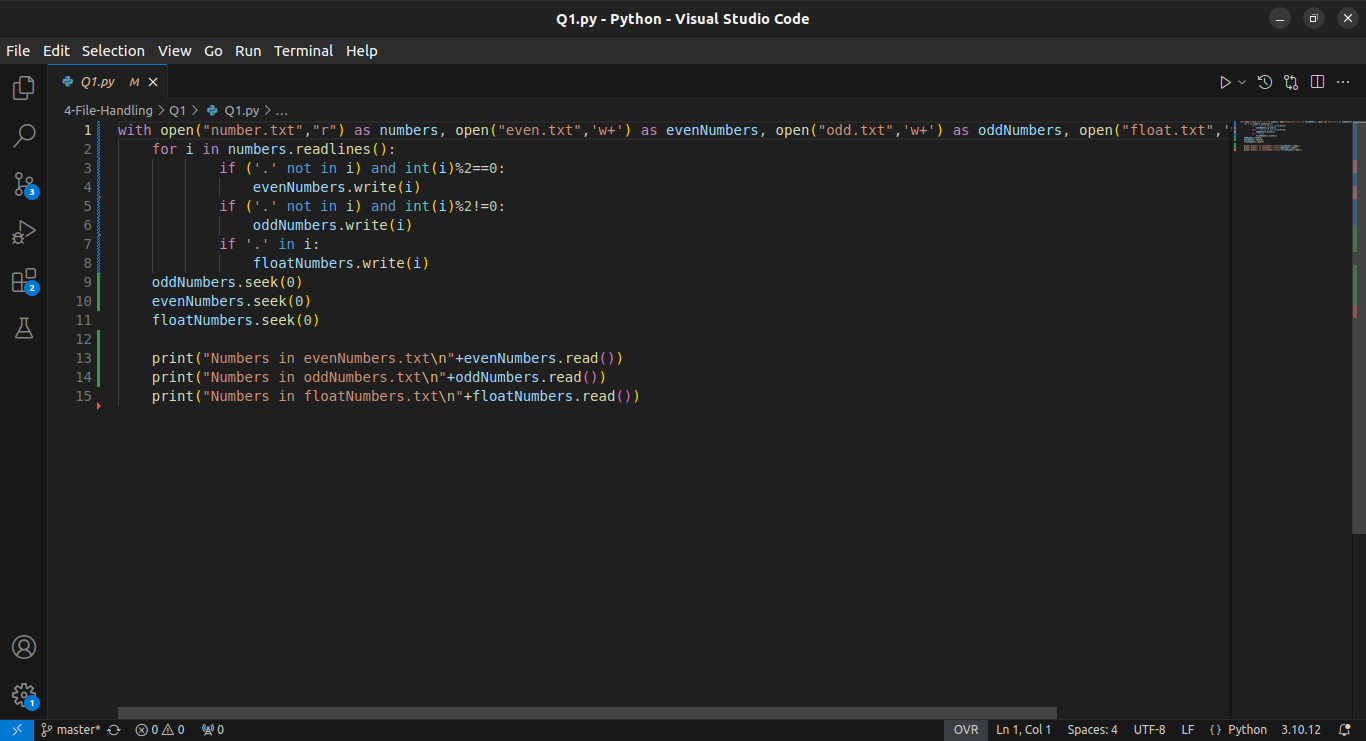
odd.txt -- all odd number

float.txt -- all floating point number

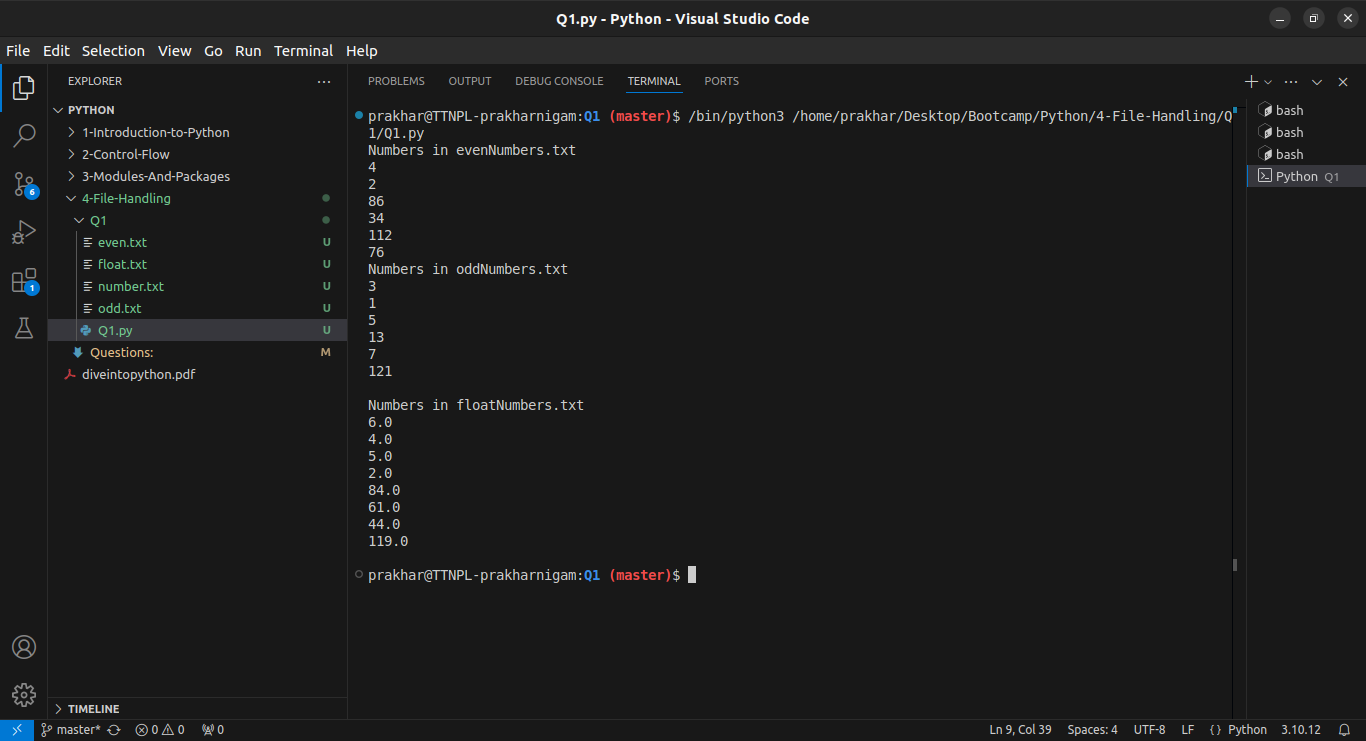
**Use with() clause for file handling**



Number.txt



Code before executing



After executing the code, we see different files were created and each number in number.txt was written to their respective txt files.

**Q2. 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:**

{

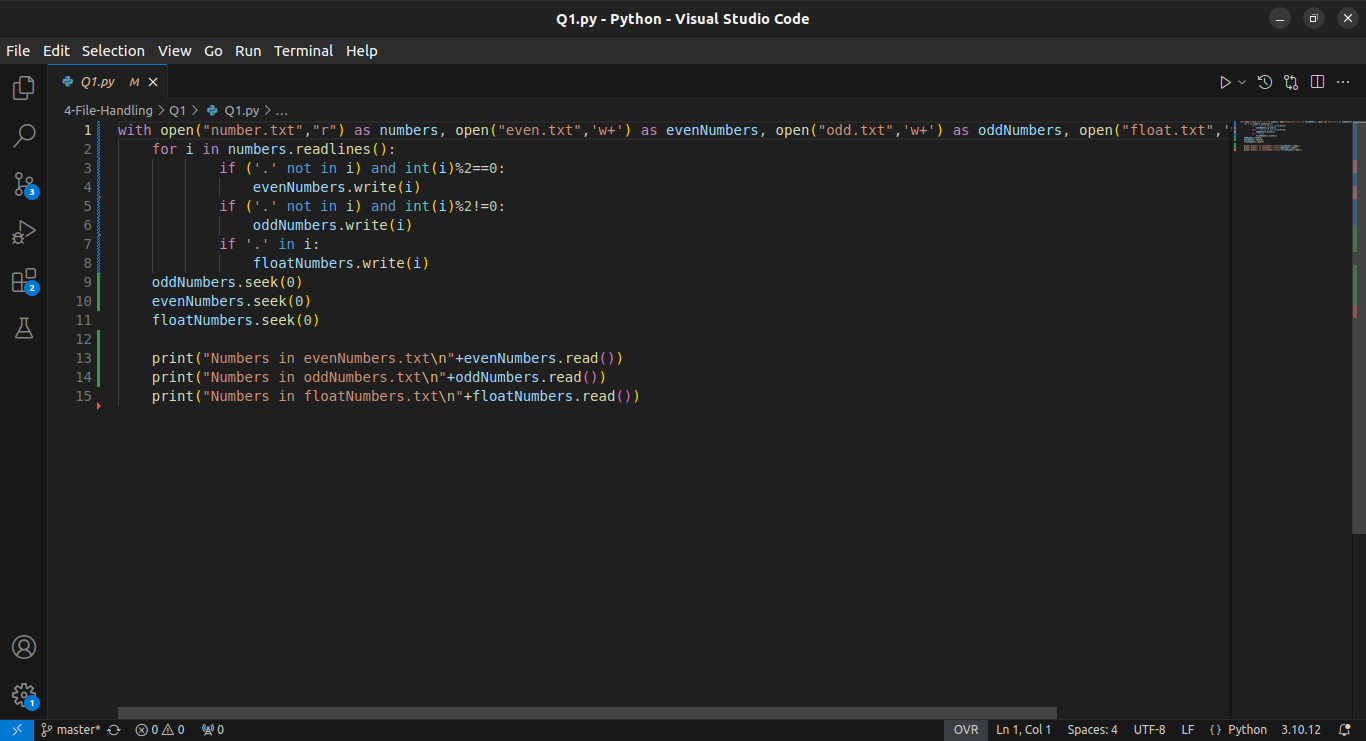
"package": ["os", "itertools"],

"funation": ["function1", "function2"],

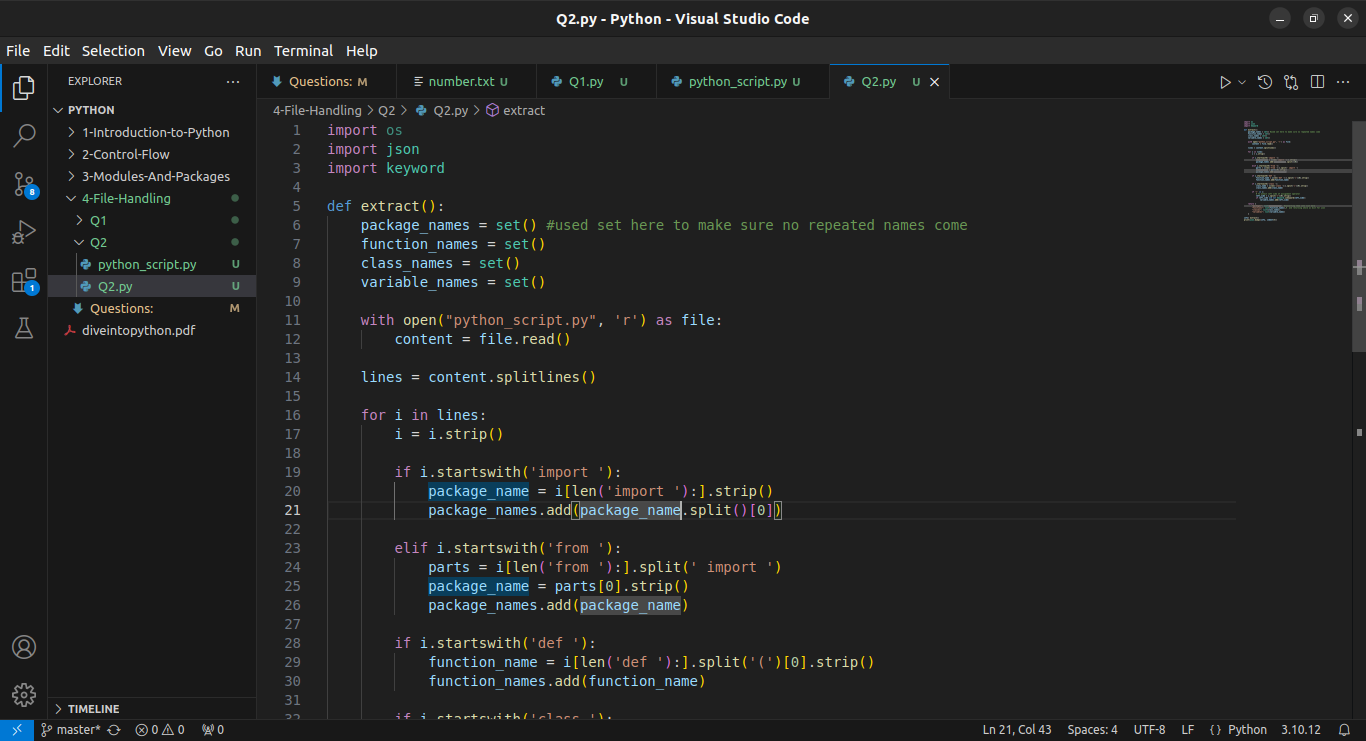
"class": ["classA", "classB"],

"variable": ["num", "i", "j"]

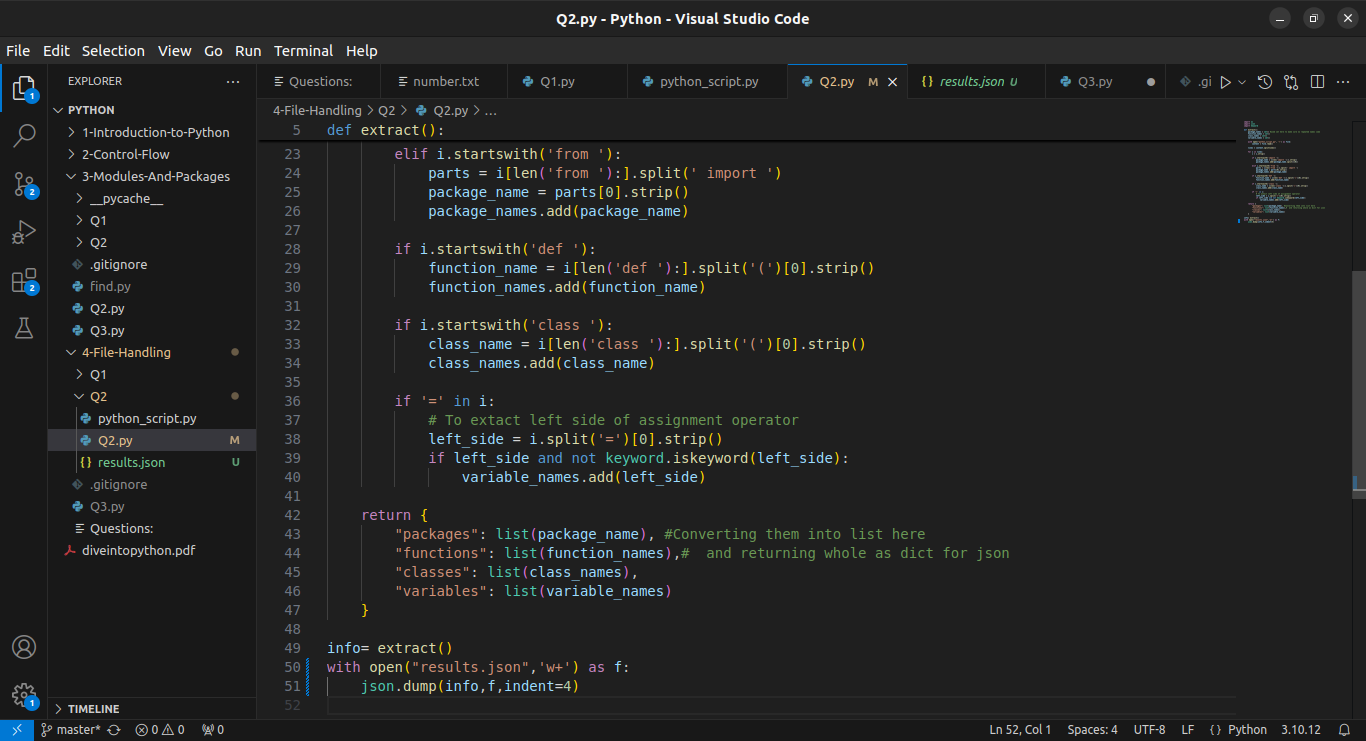
}



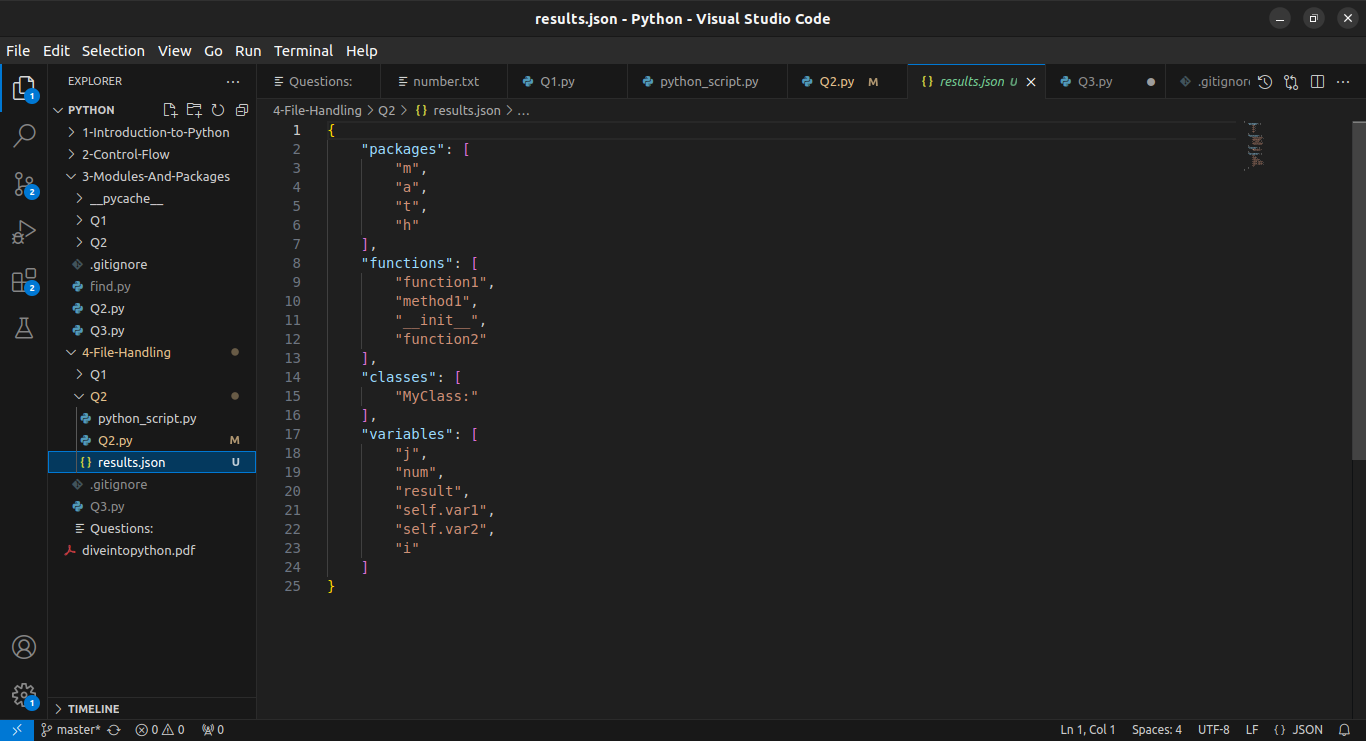
Python\_script.py



Code 1/2



Code 2/2



Output After Execution

**Q3. 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

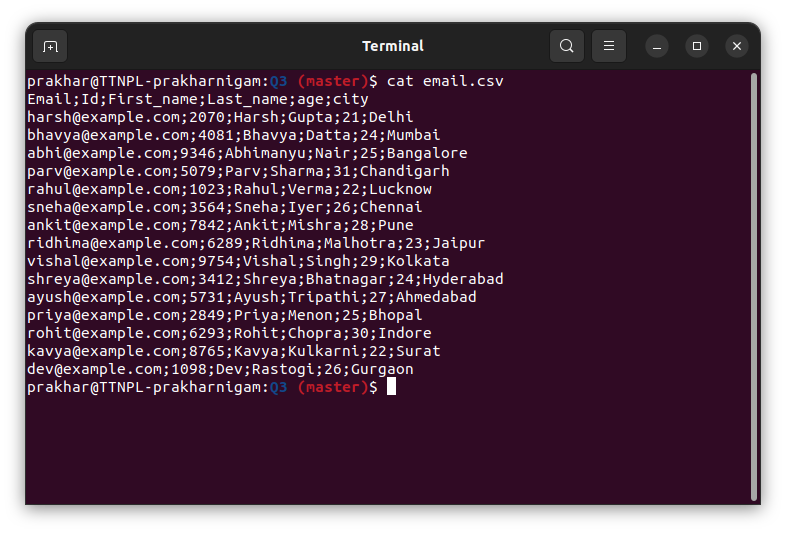
To project the data `csvlook` script should accept comma separated column numbers, e.g

-f 3,5,7 should print only column 3, 5 7

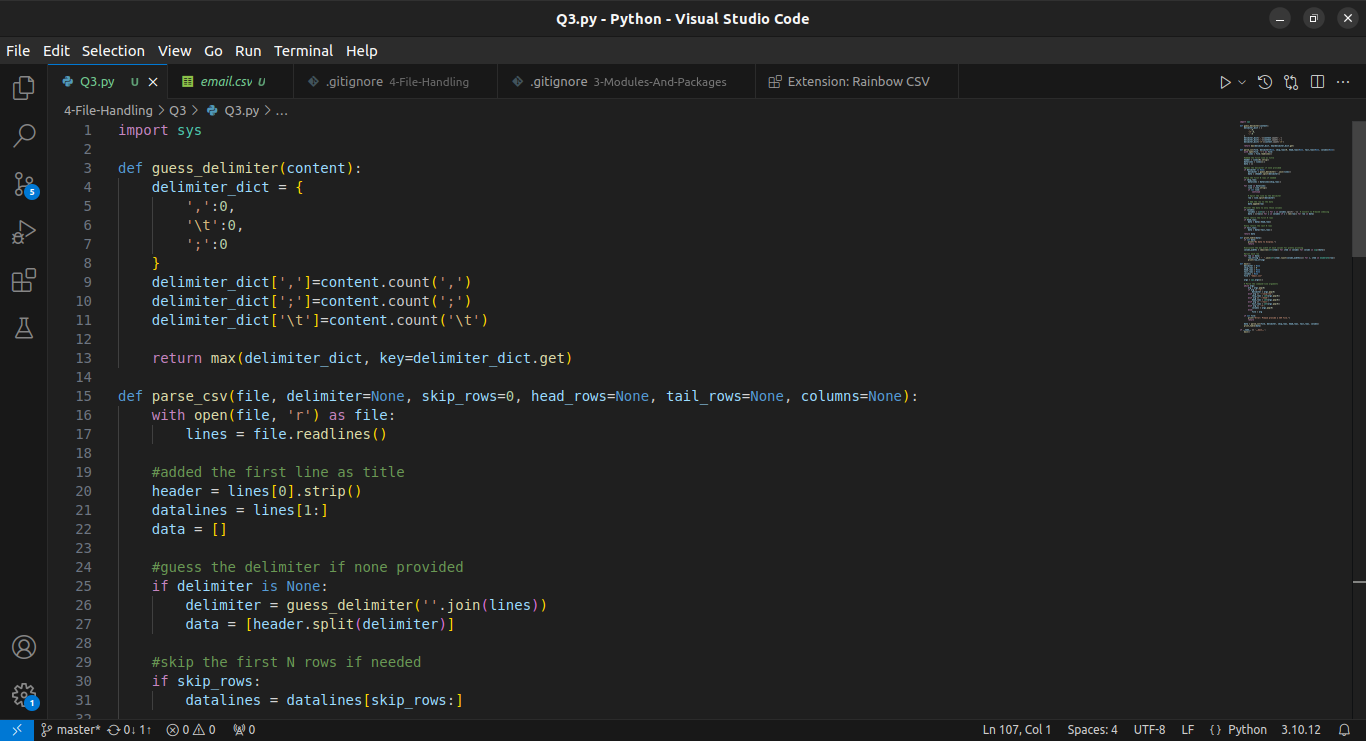
--skip-row N to skip first N rows

--head N to display only first N rows

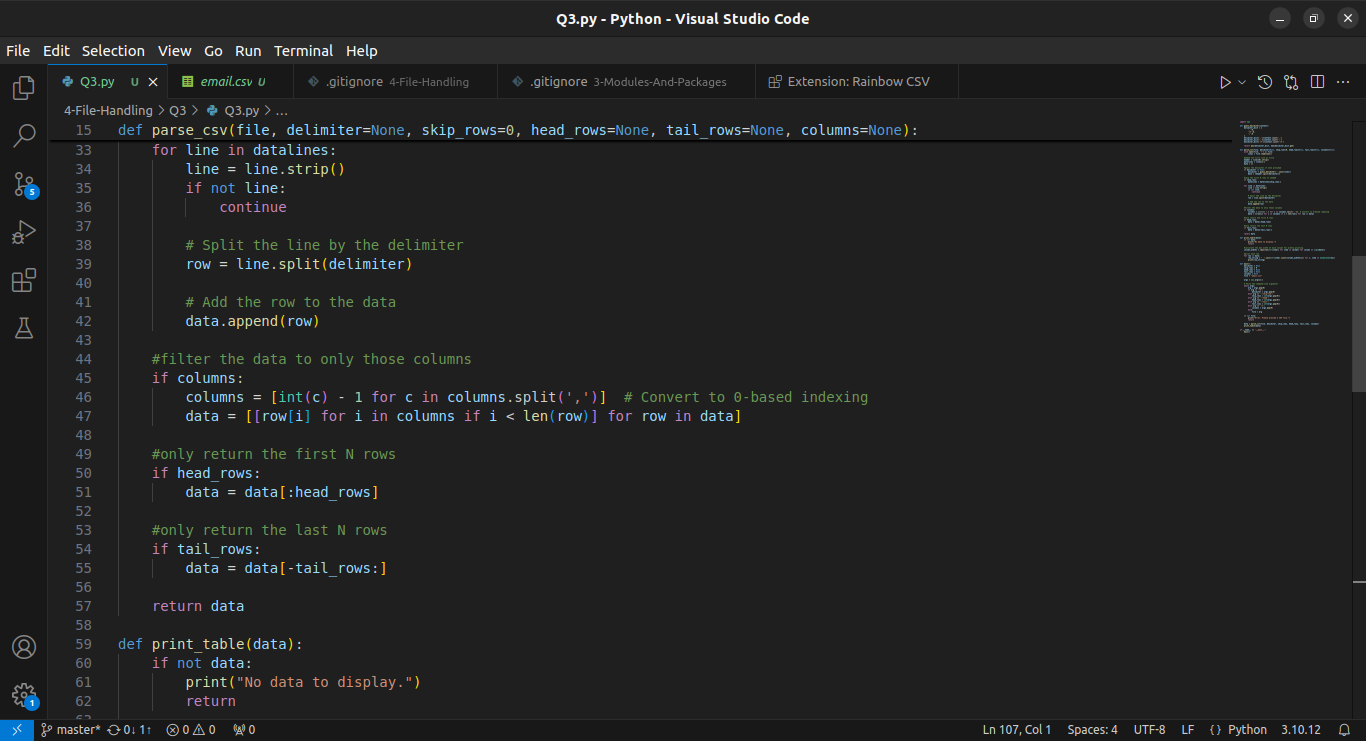
--tail N to display last N rows



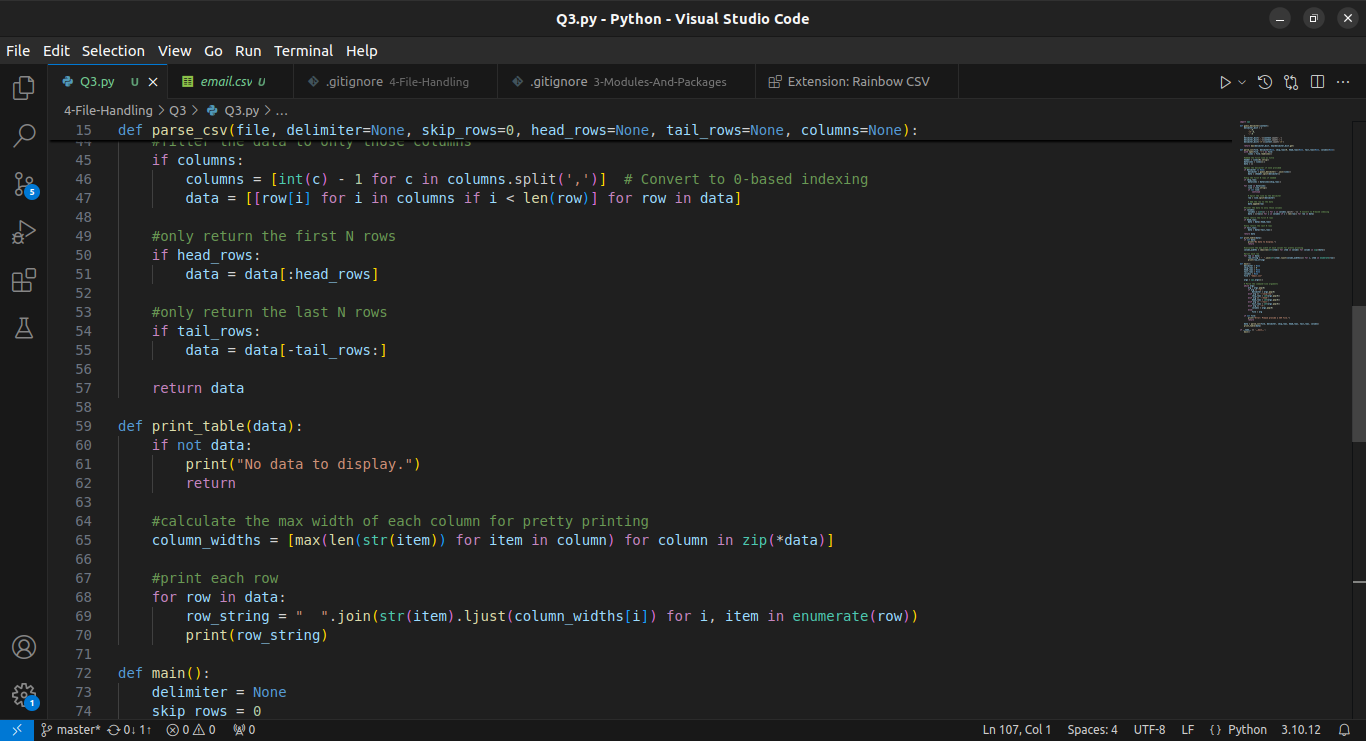
Example CSV File



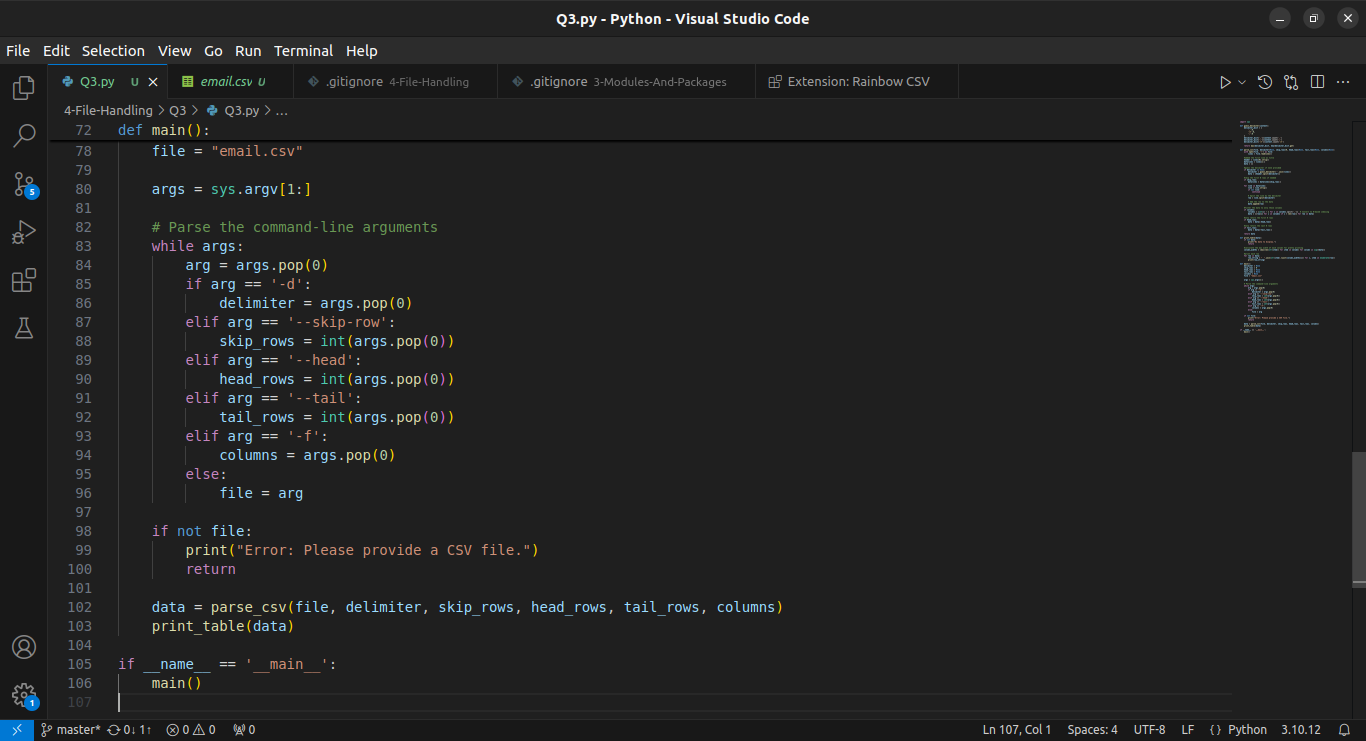
Code 1/4



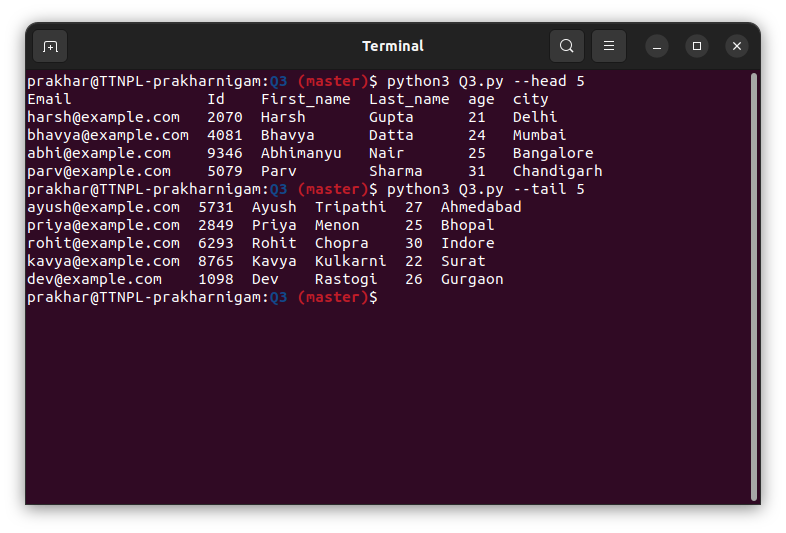
Code 2/4



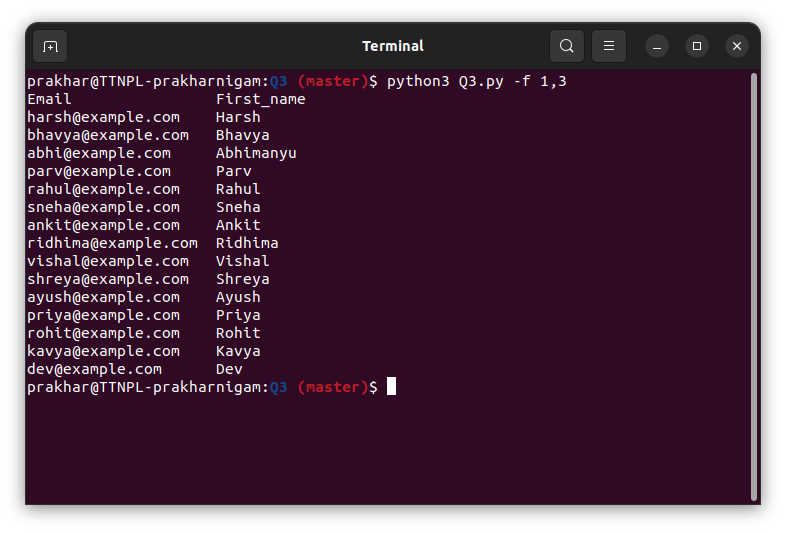
Code3/4



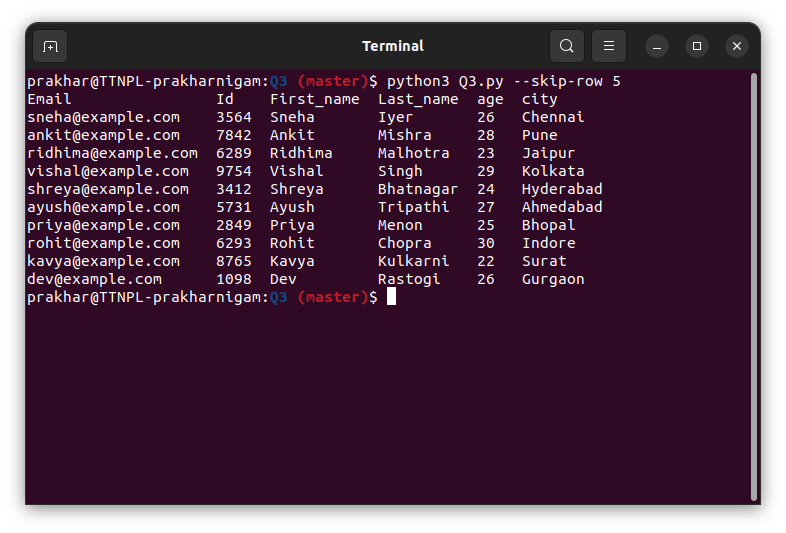
Code 4/4



Output used head and tail



Used -f to filter columns



Used skip rows