**Understand the Application**

Sherlock Holmes…acclaimed detective whose abilities are tauted as bordering on the fantastic was known for his astute ability to adopt disguise and use forensic science to solve the unsolvable.

Our task for this week is to wear our own Sherlock Holmes hat of sorts using our guise as a computer scientist to ferret out the unknown.   The next time the tall tale of “check is in the mail” you’ll be able to step in to verify this statement for yourself as you use your own forensic science to solve the question if indeed that check, that phonecall, (i.e. that identifier) is in the mail (i.e. in the record log).

Specifically we will write a Python3 program that will read in a Python source file producing an index of all identifiers in that file.  For each identifier, our solution intelligence will additionally print all lines in which the identifier occurs.

**The Program Spec**

Write a program that reads in a filename provided by the user, opens it and reads all of the lines.  Create a dictionary to hold the line numbers for each valid identifier found.  Your solution will need to process each line to determine whether a valid identifier exists on that line and if so, record both the line number and identifier found in the dictionary.

A valid identifier must be a string consisting of only letters, numbers and underscores.  If a line contains an invalid identifier no record is made in the dictionary. If a line contains a valid identifier already recorded in the dictionary append the line number to the dictionary log of where the identifier was found.

**Deliverable:  yournameLab4.py**  Your source code solution and a copy of the run pasted into your source submission file.  Be sure to comment out your run so that your **.py** file will still run in the grader test bed.

**Input Error Checking:**  For simplicity a python file is provided to feed into your dictionary generator.   The filename is **t4.py**provided in the link below.

**Test Run Requirements:**  Use the provided t4.py file as your test run validator.

Here are some other tips and requirements:

1.    Keep provided test file intact

2.    Use t4.py as your input test file to generate submission run output

3.    Verify identifiers are valid; ignore if invalid

Here is a sample run:

…

apple: [1, 11]

art: [5]