Syntax errors

In effect, syntax errors represent *grammar errors* in the use of the programming language. Common examples are:

- Misspelled variable and function names
- Missing semicolons
- Improperly matches parentheses, square brackets, and curly braces
- Incorrect format in selection and loop statements

Runtime errors

Runtime errors occur when a program with no syntax errors asks the computer to do something that the computer is unable to reliably do. Common examples are:

- Trying to divide by a variable that contains a value of zero
- Trying to open a file that doesn't exist

There is no way for the compiler to know about these kinds of errors when the program is compiled.

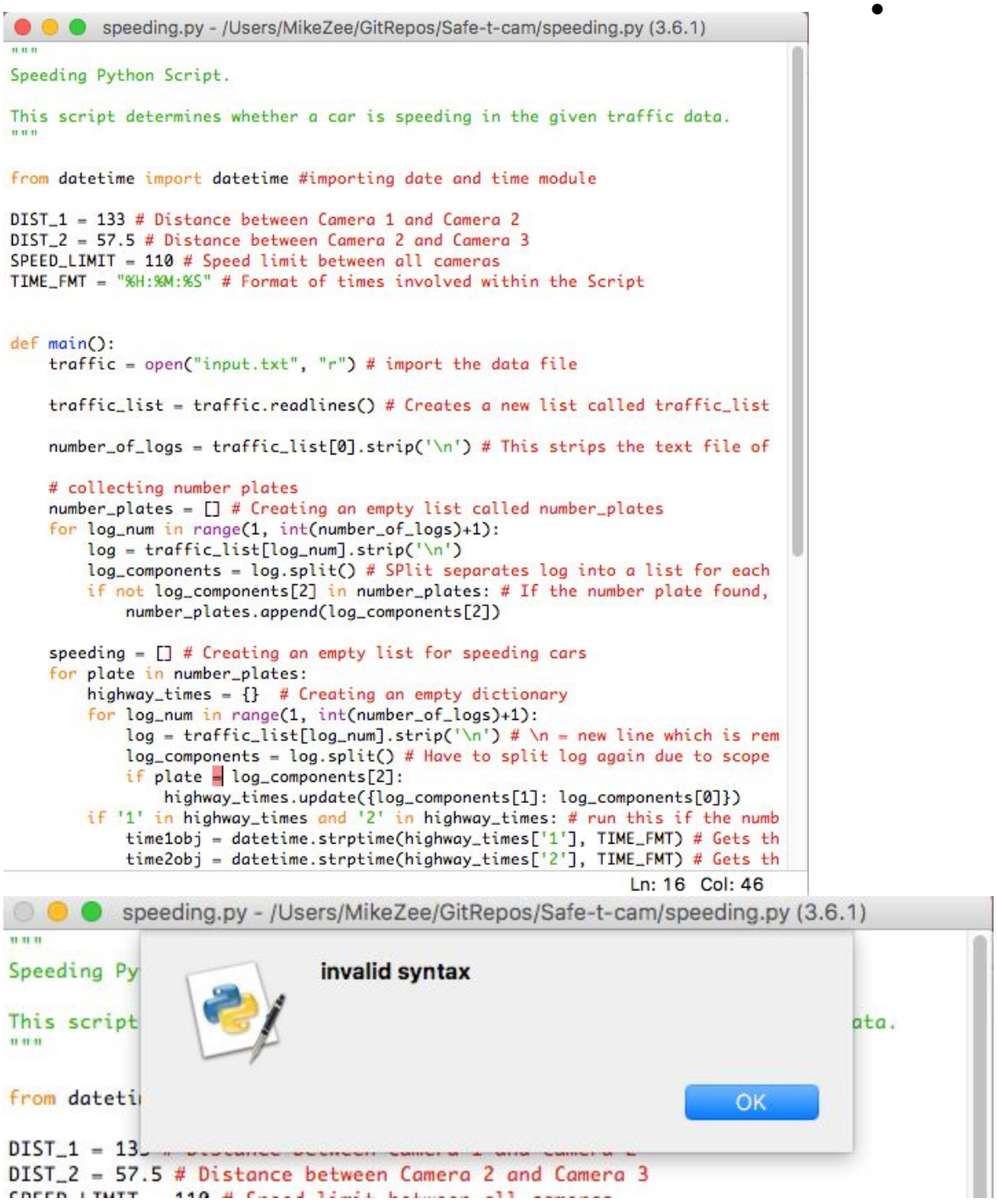
Logic errors

Logic errors occur when there is a design flaw in your program. Common examples are:

- Multiplying when you should be dividing
- Adding when you should be subtracting
- Opening and using data from the wrong file
- Displaying the wrong message

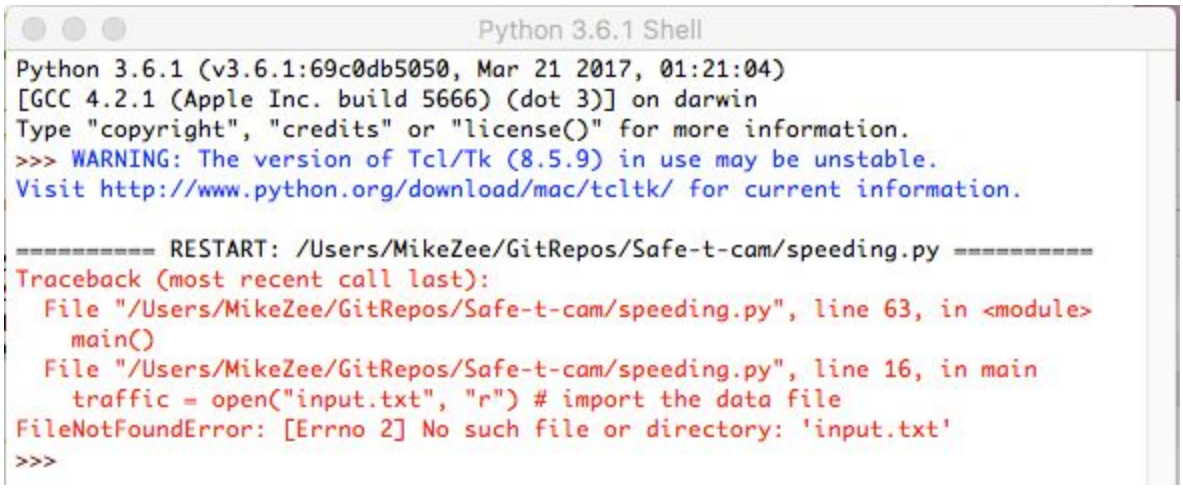
Syntax Error:

- “IF plate = logcomponmenets 2, instead of doing if plate == log component 2
-



Runtime Error:

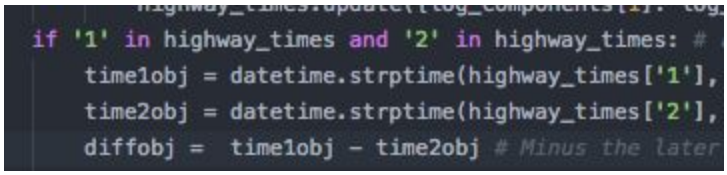
- Try to open up the file but it isn't i the right folder input.txt




```
def main():
    traffic = open("input.txt", "r") # import the data file
```

Logic Error:

- To get the difference in time I do the first time - the second time on the highway times giving me a negative number and causing the output of my code to be incorrect and only showing people speeding through checkpoint 3



```
if '1' in highway_times and '2' in highway_times: #
    time1obj = datetime.strptime(highway_times['1'],
    time2obj = datetime.strptime(highway_times['2'],
    diffobj =  time1obj - time2obj # Minus the later
```

```
14:58:16 3 JX64XY 114.4
14:53:54 3 ZOOM99 120.3
14:58:26 3 LEMANS 112.7
15:05:54 3 LINFOX 112.5
15:05:50 3 THSTIG 110.3
15:15:33 3 OZ26WC 112.6
15:17:09 3 H8COPS 113.6
15:35:29 3 KK09LY 113.7
14:38:39 3 GB72NM 114.2
>>>
```