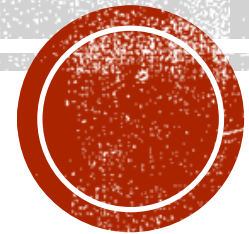
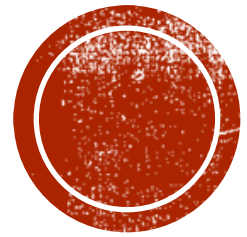


DEBUGGING TIPS AND TRICKS

Step 1: Don't Panic





TYPES OF EXCEPTIONS

Full listing: <https://docs.python.org/2/library/exceptions.html>

TRACING ERRORS

```
In [2]: %run "C:\Users\Chrisandra\Downloads\SUTD\Term 5\10.009 - Digital World (TA)\Week 6 codes.py"
-----
NameError                                Traceback (most recent call last)
C:\Users\Chrisandra\Downloads\SUTD\Term 5\10.009 - Digital World (TA)\Week 6 codes.py in <module>()
     11     '''
     12
--> 13 print forward(lol)
     14
     15 def check_password(pword):

NameError: name 'forward' is not defined
```

Read the traceback in the python kernel, see which line is causing the error, and what error is it
(Note: The traceback may reflect multiple lines to help you trace the error to its source)



ZERO-DIVISION-ERROR

What Does It Mean:

You are trying to divide by 0 somewhere

What Should I Do:

Look at all your division cases in your code, and look specifically for possibilities where a variable in the denominator of an equation can be 0



ATTRIBUTE-ERROR

What Does It Mean:

You are trying to access or call an attribute that a particular object type does not possess

What Should I Do:

Check documentation if the attribute exists for the object type (e.g. does `add()` exist for tuples?)



IMPORT-ERROR

What Does It Mean:

Import statement failed – Unable to find the module to be imported

What Should I Do:

Ensure you have installed the module or library into the right directories, and that you've spelt the name of the module or library correctly



INDEX-ERROR

What Does It Mean:

Index is not found in a sequence

What Should I Do:

Check if you are trying to access an item in an $(n+1)$ position when your list/tuple only contains n items



KEY-ERROR

What Does It Mean:

Key is not found in existing keys in a dictionary

What Should I Do:

Check if you spelt the name of the key correctly, or if you have included that key in the dictionary called



NAME-ERROR

What Does It Mean:

Identifier is not found in the local or global namespace

What Should I Do:

Check if you have spelt the variable name correctly, if you have defined your variable as a global variable (if required), and if you have defined the class before you call it (if required)



UNBOUND-LOCAL-ERROR

What Does It Mean:

You are trying to call a local variable in a function or method, but no value has been bound to that variable

What Should I Do:

Check if you have defined or initialised the variable before calling it



IO-ERROR

What Does It Mean:

Input/output operation failed

What Should I Do:

Check if you are trying to open a file that doesn't exist, or if the file is in the directory your code is trying to open it from



IO-ERROR

Unable to figure out where should you save your text files too? Do a write function, and use your python code to create a new text file with a unique name like “tagyoureit”, then search in your C drive for the file “tagyoureit.txt”

```
>>> f = open('test.txt', 'w')  
>>> f.write('line one \nline two \nline three')  
>>> f.close()
```



SYNTAX-ERROR

What Does It Mean:

Error in python syntax

What Should I Do:

Check for misspelling of python operators and functions (e.g. did you type a
';' at the end of any line? Did you type adds() instead of add()?)



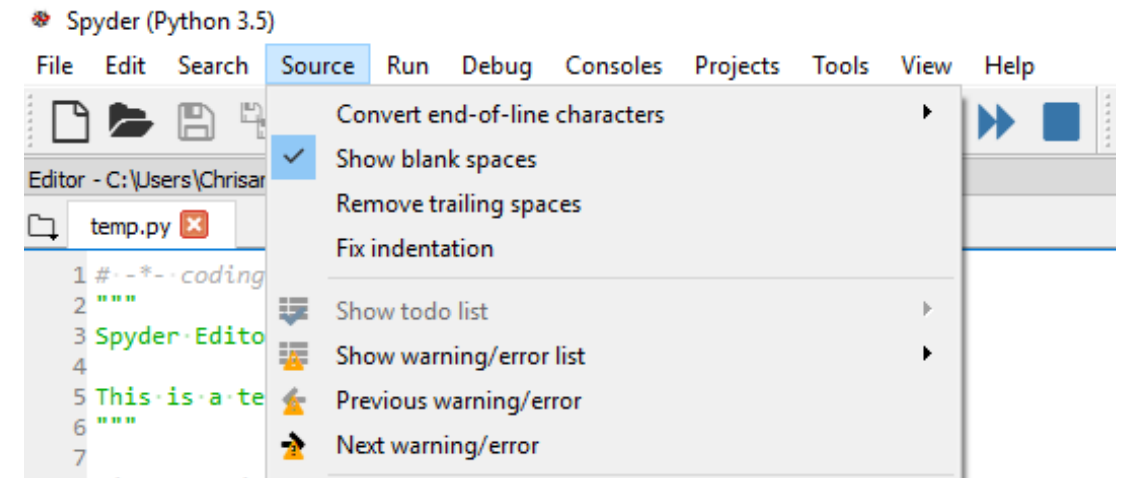
INDENTATION-ERROR

What Does It Mean:

Indentation not specified properly

What Should I Do:

Check for erroneous indentations



Sad fact: I can't get fix indentation to work, but remove trailing spaces works



TYPE-ERROR

What Does It Mean:

You are trying to attempt an invalid operation or function for a particular data type (e.g. dividing a string by an integer)

What Should I Do:

Follow the stack trace in the error message to see which line is causing the error, and what invalid operation or function you tried to perform on that data type



VALUE-ERROR

What Does It Mean:

Built-in operation or function received an argument that has the right data type, but an inappropriate value

What Should I Do:

Check for instances in your code where, for example, you are expecting to receive an integer, but receive a string instead



RUNTIME-ERROR: MAXIMUM RECURSION DEPTH EXCEEDED

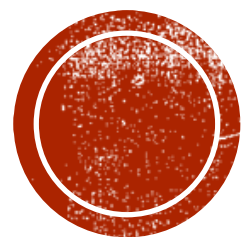
What Does It Mean:

Your code is running repeatedly and not ending at all

What Should I Do:

Check why the base case is not being met for recursion and while loops. Is there some condition or exception which you've neglected, causing the code to keep repeating itself? Did you forget to increment or decrement your counter?





TACKLING LONG QUESTIONS



COMMENT AND ANNOTATE

- If you have many functions, write a short but easily understandable comment (for yourself) above each function or on your question paper with these information:
 - What the function does
 - What input the function requires
 - What does the function output

```
# This function reads the file which contains the name, units, and values of  
# fundamental constants, and returns a dictionary mapping the name of the  
# constants to the value of the constants  
  
#Input: String of file name  
#Output: Dictionary mapping name of constants to value of constant  
def get_fundamental_constants(f):  
    fileRead=open(f,'r')  
    fileList=list(fileRead)|
```



COMMENT AND ANNOTATE

- If you have a really long solution for any question, or a really long function, segment your code into chunks, and leave line breaks between each chunk
 - Write a short comment on what each chunk does as well



USE INFORMATIVE NAMES

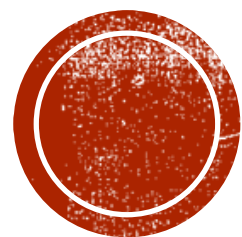
- Don't name your variables a, b, c. You're going to forget what a, b, and c are when you're rushing and stressed up!
- Use names which aptly describe the variable or function

```
def get_fundamental_constants(f):  
    fileRead=open(f, 'r')  
    a=list(fileRead)  
    print a  
    print "\n"  
    del a [0]  
    del a [0]  
    c=[]  
    d={}  
    for i in range(len(a)):  
        b=a[i].split(" ")  
        for j in range(len(b)):  
            if b[j]!='':  
                c.append(b[j])  
  
    print c  
    print "\n"  
    for k in range(0, len(c), 3):  
        d[c[k]]=float(c[k+1])  
    return d
```

V.S.

```
def get_fundamental_constants(f):  
    fileRead=open(f, 'r')  
    fileList=list(fileRead)|  
    print fileList  
    print "\n"  
    del fileList [0]  
    del fileList [0]  
    dictList=[]  
    constDict={}  
    for i in range(len(fileList)):  
        tempList=fileList[i].split(" ")  
        for j in range(len(tempList)):  
            if tempList[j]!='':  
                dictList.append(tempList[j])  
  
    print dictList  
    print "\n"  
    for k in range(0, len(dictList), 3):  
        constDict[dictList[k]]=float(dictList[k+1])  
    return constDict
```





MORE TIPS AND TRICKS



KNOW HOW TO USE DOCUMENTATION

- Download or bookmark pages in the documentation which may be useful, like the documentation page on lists, dictionary, etc.
- Unfamiliar syntax?
 - Use the kernel or create an empty python file and spend no more than 2-3 minutes trying to figure out how you can use it



DEALING WITH FILE IO

- Read the file out into a manipulatable format (like a string or a list)
- Clean the list or string by stripping unwanted characters and spaces (if necessary)
- Use list or string to obtain the output you require





fileIOeg.py

```

5 #Input: String of file name
6 #Output: Dictionary mapping name of constants to value of constant
7 def get_fundamental_constants(f):
8     fileRead=open(f,'r')
9     fileList=list(fileRead)
10    print fileList
11    print "\n"
12    del fileList [0]
13    del fileList [0]
14    dictList=[]
15    constDict={}
16    for i in range(len(fileList)):
17        tempList=fileList[i].split(" ")
18        for j in range(len(tempList)):
19            if tempList[j]!='':
20                dictList.append(tempList[j])
21    print dictList
22    print "\n"
23    for k in range(0,len(dictList),3):
24        constDict[dictList[k]]=float(dictList[k+1])
25    return constDict

```

Python

C:\Users\Chrisandra

```
In [5]: %run "c:\users\chrisa~1\appdata\local\temp\tmpfaonot.py"
```

```

['name of constant      value      dimension\n', '-----\n', 'speedoflight      299792458.0
m/s\n', 'gravitationalconstant      6.67259e-11      m**3/kg/s**2\n', 'Planckconstant      6.6260755e-34      J*s\n', 'elementarycharge      1.60217733e-19
C\n', 'Avogadronumber      6.0221367e23      1/mol\n', 'Boltzmannconstant      1.380658e-23      J/K\n', 'electronmass      9.1093897e-31
kg\n', 'protonmass      1.6726231e-27      kg\n']

```

```

['speedoflight', '299792458.0', 'm/s\n', 'gravitationalconstant', ' 6.67259e-11', 'm**3/kg/s**2\n', 'Planckconstant', '6.6260755e-34', 'J*s\n', 'elementarycharge',
'1.60217733e-19', ' C\n', 'Avogadronumber', '6.0221367e23', ' 1/mol\n', 'Boltzmannconstant', ' 1.380658e-23', ' J/K\n', 'electronmass', '9.1093897e-31', 'kg\n',
'protonmass', '1.6726231e-27', 'kg\n']

```

```

{'Boltzmannconstant': 1.380658e-23, 'speedoflight': 299792458.0, 'protonmass': 1.6726231e-27, 'gravitationalconstant': 6.67259e-11, 'Avogadronumber': 6.0221367e+23,
'elementarycharge': 1.60217733e-19, 'Planckconstant': 6.6260755e-34, 'electronmass': 9.1093897e-31}

```

Cursor pos

9: 28

Python

C:/Users/Chrisandra/Downloads/SUTD/Term 5/10.009 - Digital World (TA)/fileIOeg.py



DON'T KNOW HOW TO START?

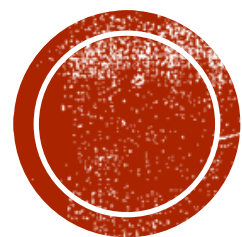
- Look at what are the inputs and outputs required of the function
 - Take note of the type
- Plan out on paper how are you going to get from the input(s) to the output(s)
 - How do I need to manipulate the input?
 - What formulae do I need to get from the input to the output?
- Remember that in programming, you are writing a set of instructions for your computer to follow, to get the desired output(s) from the input(s)



WRONG OUTPUT?

- If you can't figure out which part of your code is causing the wrong output, use print statements to print variable that can affect your output
- Narrow down your scope of search for the buggy part of your code until you locate it
- Figure out why this is wrong





COMMON MISTAKES



TYP0 IN FUNCTION OR VARIABLE NAMES

- Be careful (e.g. `functionName` v.s. `FunctionName`, `funcName` vs `funcNames`)
 - Similar, but not the same
- Tell tale signs:
 - `NameError: name 'functionName' is not defined`
 - Changing something in the code that most definitely will change the output of the code, doesn't result in a change in the output of the code
- What Should I do?
 - Locate the line reflected in the error message in the kernel, and see what you've typed wrongly
 - Check if you called the right function in the kernel or code



INT DIVISION VS FLOAT DIVISION

- Keep getting 0s or values that are way off? Check if you used integer division instead of float division
 - E.g. $1 \div 10 = 0$ v.s. $1.0 \div 10 = 0.1$



IF...ELSE...

```
If (x.len()==1):
```

```
    ...
```

```
Else if (x=="1"):
```

```
    ...
```

If `x="1"`, this code will only run the statements under `'If (x.len()==1):'`, the else if statement won't run



IF...ELSE...

If (x.len()==1):

Else if (x=="1"):

...

If (x.len()==1):

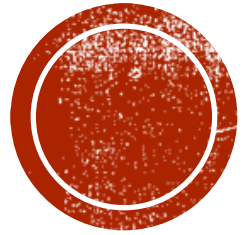
...

If (x=="1"):

...

If you want your code to check both conditions and run the codes under both conditions if both conditions are met, do this instead

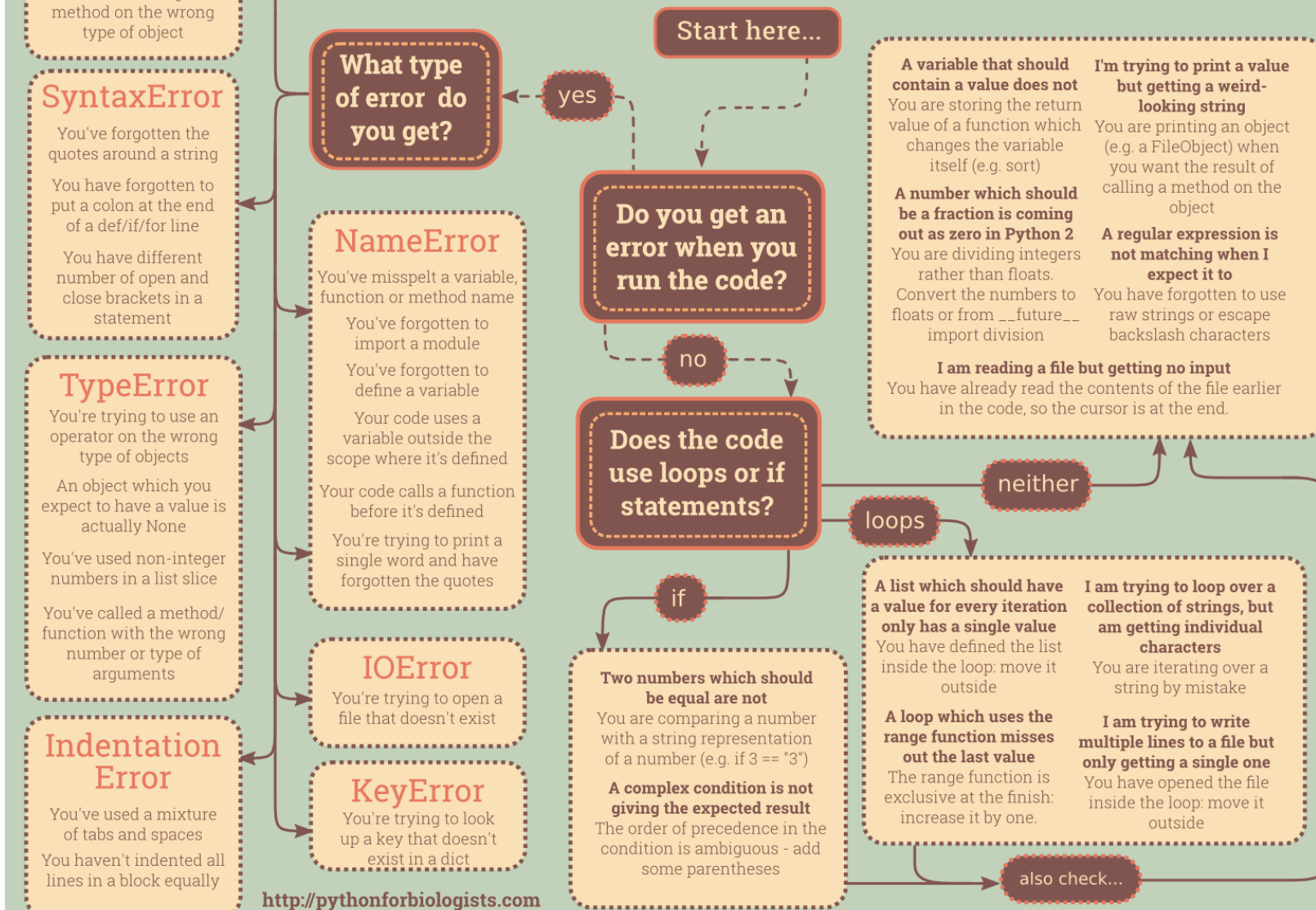




COOL SUMMARY

Bonus: It looks nice as a wallpaper too
(<http://pythonforbiologists.com/index.php/29-common-beginner-python-errors-on-one-page/>)

My code isn't working :-)



<http://pythonforbiologists.com>

