G51FSE Python Tutorial Summary Sheet

What is python?

Python is an interpreted programming language. It is similar to Java in many ways, including letting you do object orientation. You can actually compile python programs to be java byte code (its clever like that).

It is **dynamically typed** which means that the compiler/interpreter decides what type variables are from their usage, and so no predeclaration is necessary.

Python unlike C and Java is **sensitive to whitespace**. This means that no braces "{ }" are required around control statements. But it does mean that you have to be very precise with your indentation. You use spaces not tabs to make code simple, clear and readable.

While python is fully object oriented it can also work functionally like haskell. It has many advanced features including using lambdas - we wont need to use them in this course, but it is a fact worth knowing.

Python allows the use of numerous basic types, including float, int, str, and more complicated **collections** including lists and dictionaries.

A more unusual feature is that the python interpreter can be used like an interactive shell where you can run code one line at a time.

The Eclipse IDE and Vi(m) are both equally good for developing python. If you would like to use vi or vim, then a python friendly vimrc can be obtained by typing:

```
$cp ../nxb/.vimrc .
```

From your home directory using a UNIX terminal.

Additionally we feel it is an excellent tool for teaching because it is very well documented online.

Why are we learning another language in addition to C and Java?

Hopefully some knowledge of python will be useful to you for stuff like your second year group project

It is also important that you understand the principles of how to program independent of a language.

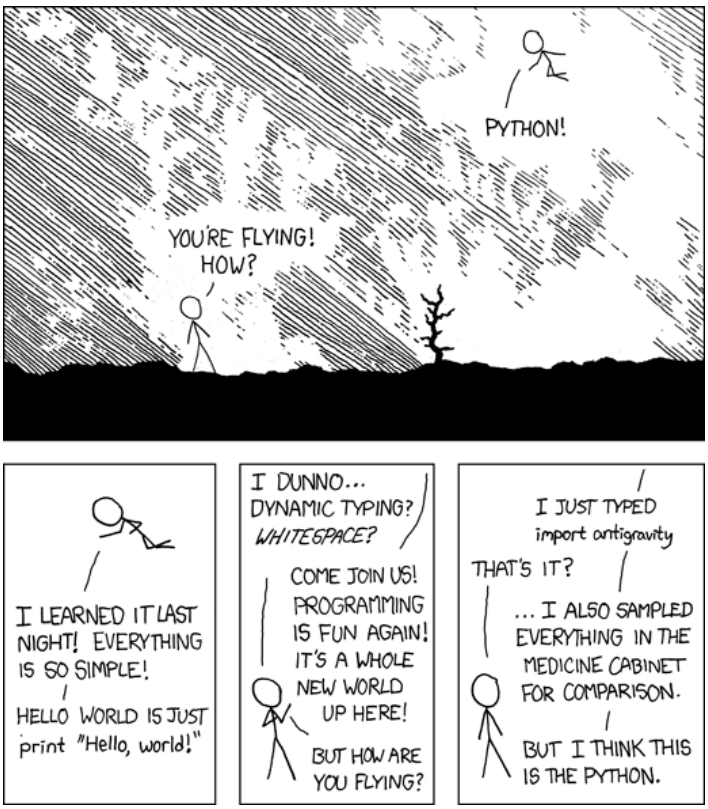
Python has a large collection of libraries which are dead easy to use. This includes the pygame system which you will be expected to be able to deploy and use for your FSE coursework.

We also think that it is a fun language to code in with its "batteries included" mentality - it lets you get on with the high level fun without having to worry about low level stuff so much.

Its pretty nifty as its interpreted - this means that it can run cross platform.

Also, employers are increasingly hiring python developers, and its used by many companies including Google, where it is used for the majority of their inhouse apps.

Your Task Today: Get to Grips With Python.



http://xkcd.com/353/

What we would like you to do is to try and reimplement some of the stuff you have been doing in OOP in python to help you work through a python tutorial. You may work in pairs or by yourself. If you are happy to go ahead and learn some python by yourself then please follow the code and instructions on the official python tutorial. The python tutorial contains a lot of help - dont forget to use as much information from online sources as you can to help get to grips with python.

Complete the following four tasks:

1. Produce a “hello world” program
2. Write a program where you input a distance in km and output the time taken to travel that distance assuming a hard-coded constant speed.
3. Modify your code from part two to accept user specified speeds
4. Further modify your code to calculate multiple distances in the same program by adding a while loop.

Follow the specification details given under the assignments tab http://www.cs.nott.ac.uk/~cah/G51OOP/public_html/index.html

Python Tutorial: http://docs.python.org/tutorial/