Exercises: Introduction

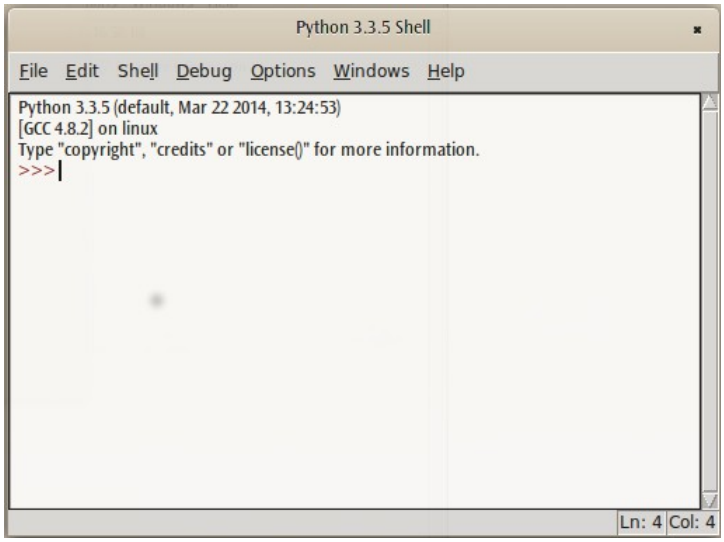
Part A

Ensure that Python is correctly installed by starting a Python interpreter in interactive mode, i.e. start a Python REPL (read–evaluate–print loop): on Linux, Solaris, Mac OS X, or any other Unix-style system, type 'python3' (or 'python') in a terminal, on Windows use the 'Python command line' 'start menu' item. You should see something like:

```
|> python3
Python 3.4.0 (default, Mar 22 2014, 12:55:21)
[GCC 4.8.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Part B

See if IDLE is correctly installed by trying to start it: on Linux, Solaris, Mac OS X, or any other Unix-style system, type 'idle' in a terminal, on Linux, Solaris, Mac OS X, or Windows use the IDLE menu item. You should see something like:



Part C

Using a Python REPL (either direct from the command line or via IDLE), enter and execute the following Python program:

```
print('Hello World.')
```

The expected result should be fairly obvious. 

Part D

Using the IDLE editor or some other text editor or IDE, create a file *helloWorld.py* with the program:

```
print('Hello World.')
```

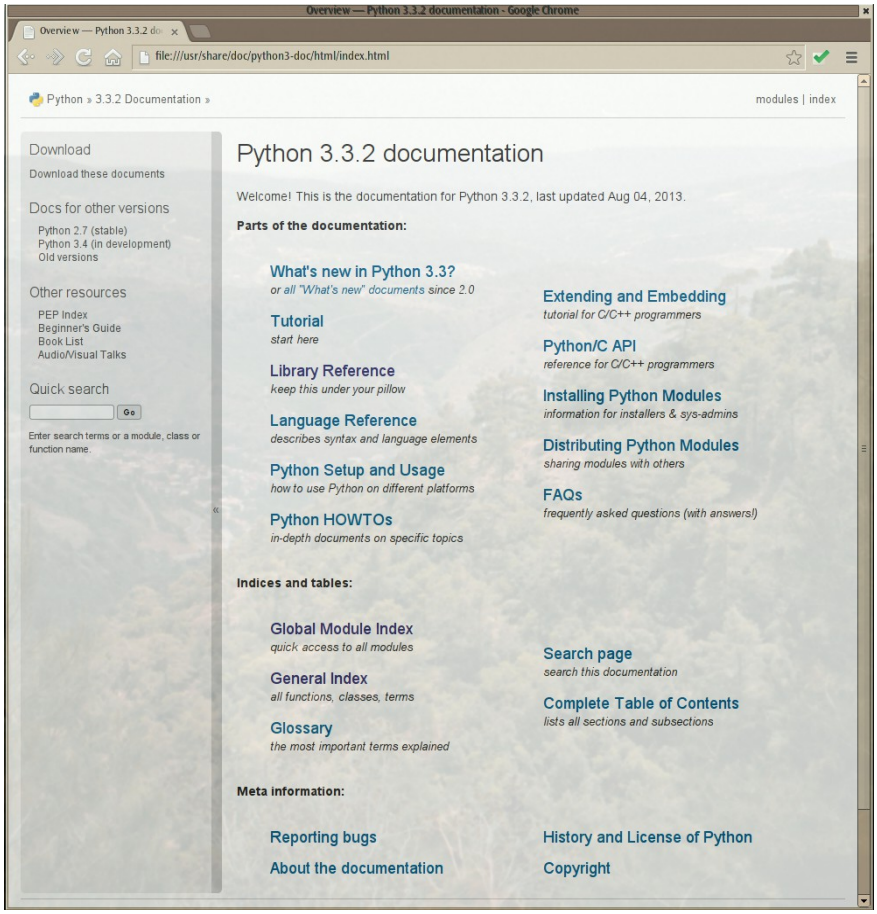
and execute it from a command line prompt. (This is really just to ensure that the path etc., are set correctly for later working.)

Part E

Ensure the Python documentation is loaded and accessible. If you have IDLE running, then F1 should bring up the documentation (or use the Help | Python docs menu entry). Alternatively start the browser manually pointing at the correct HTML file. For example on Debian and Ubuntu:

epiphany /usr/share/doc/python-doc/html/index.html

which should result in a window looking something like:



Using Windows the help is accessible via the standard help system.

Part F

Discover whether any or all of Tkinter, PyQt, Turtle, Tornado and Django are installed by seeing if that they can be imported and that documentation can be generated from them. In a REPL type:

```
import tkinter
import PyQt5.Qt
import turtle
import tornado
import django.http
```

The result should look like the following if all are installed correctly for the Python you are using. An exception will be raised for any that are not installed.

```
>>> import tkinter
>>> import PyQt5.Qt
>>> import turtle
>>> import tornado
>>> import django.http
>>>
```

Then type one (or more) of:

```
help(turtle)
help(tkinter.Radiobutton)
help(PyQt5.Qt.QRadioButton)
help(django.http)
```

and scroll through the API document – which is generated directly from the code and so is guaranteed to be correct, albeit somewhat arcane at times.

Part G

If you are going to use an IDE (e.g. Eclipse with PyDev, Aptana, PyCharm, Wing IDE,…), start your IDE, create a Python project, if needed, and add a file called helloWorld.py to the project containing the code:

```
print('Hello World.')
```

Execute the code. Using Eclipse/PyDev, with Python 2, you should then have a frame looking something like: