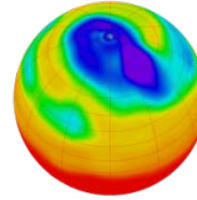




**National Centre for
Atmospheric Science**

NATURAL ENVIRONMENT RESEARCH COUNCIL



**Centre for Environmental
Data Archival**

SCIENCE AND TECHNOLOGY FACILITIES COUNCIL
NATURAL ENVIRONMENT RESEARCH COUNCIL

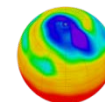
Python basics

Thanks to all contributors:

Alison Pamment, Sam Pepler, Ag Stephens, Stephen Pascoe,
Kevin Marsh, Anabelle Guillory, Graham Parton, Esther
Conway, Eduardo Damasio Da Costa, Wendy Garland, Alan
Iwi and Matt Pritchard.



**National Centre for
Atmospheric Science**
NATURAL ENVIRONMENT RESEARCH COUNCIL



**Centre for Environmental
Data Archival**
SCIENCE AND TECHNOLOGY FACILITIES COUNCIL
NATURAL ENVIRONMENT RESEARCH COUNCIL

BEFORE WE START (1)

I cannot teach you Python in a day...
...but hopefully I can show you that it is:

- Human-readable (relatively)
- Useful (even if you only know a bit)
- Flexible (you can use it in many places/ways)

BEFORE WE START (2)

Teaching materials courtesy of:

<http://www.software-carpentry.org/>



Overview

- Introduction – why we recommend Python
- Basics and control flow
- Lists, tuples and slicing
- Input/output
- Strings and text processing
- Functions, libraries and scripts
- Error handling and logging
- Sets, dictionaries and OOP

What is Python?

- A simple interpreted language
- Very human-readable with clean syntax
- Making it a very good “learn to programme” language

Why do we recommend Python?

- It is **open source** and **free**
- It is **cross-platform** (including Windows)
- It can be used for simple scripting through to writing full-blown complex applications
- Many libraries/tools to tackle all kinds of problems
- In the environmental science community it continues to grow in popularity...so **we can share code!**

Python version?

Python has multiple personalities!

- **Python 3** – new; different

But not used much yet

- **Python 2.6+** – established,

Used extensively in our community.

Used in this course!

But what changes?

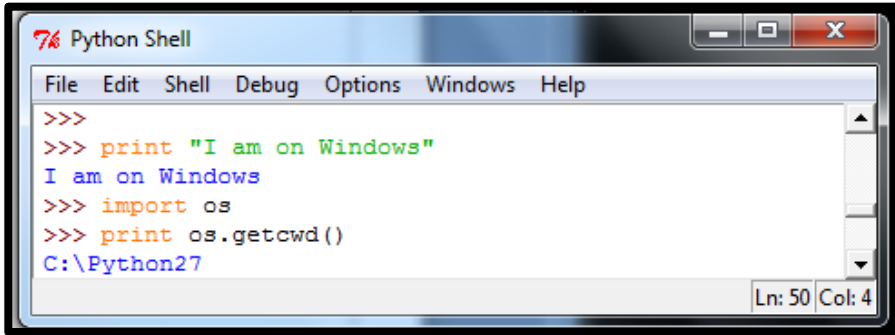
E.g. `print` changes from a statement to a function...

```
>>> print "hello" # in Python 2.* to...
```

```
>>> print ("hello") # in Python 3.*
```



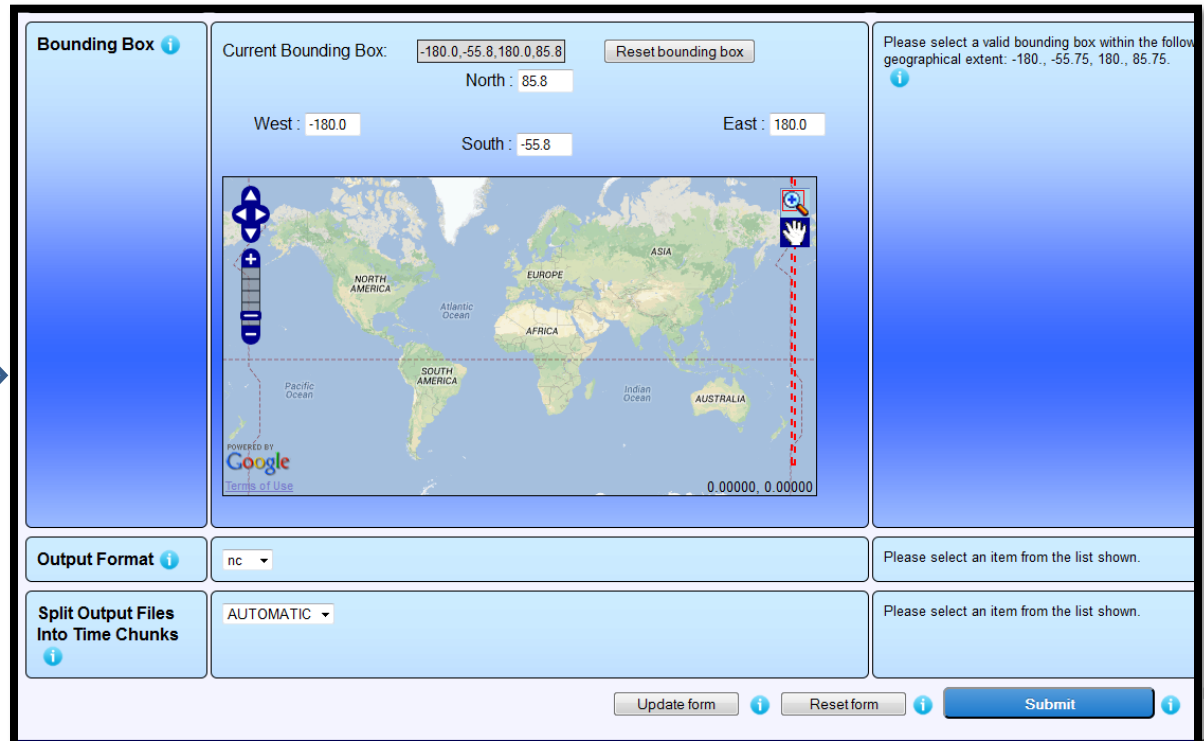
What can you do with python?



```
Python Shell
File Edit Shell Debug Options Windows Help
>>>
>>> print "I am on Windows"
I am on Windows
>>> import os
>>> print os.getcwd()
C:\Python27
Ln: 50 Col: 4
```

Interactive “shell”: allows quick learning/testing/use.


Web-programming: frameworks make this easy.



The web application interface includes the following elements:

- Bounding Box:** A section with a map of the world. The current bounding box is defined by coordinates: North: 85.8, South: -55.8, West: -180.0, East: 180.0. A "Reset bounding box" button is present.
- Output Format:** A dropdown menu set to "nc".
- Split Output Files Into Time Chunks:** A dropdown menu set to "AUTOMATIC".
- Instructions:** "Please select a valid bounding box within the following geographical extent: -180., -55.75, 180., 85.75."
- Buttons:** "Update form", "Reset form", and "Submit".

What can you do with python?



Iris 1.6

[home](#) | [examples](#) | [gallery](#) | [contents](#) | [Iris examples](#) »

[Visualisation examples](#) »

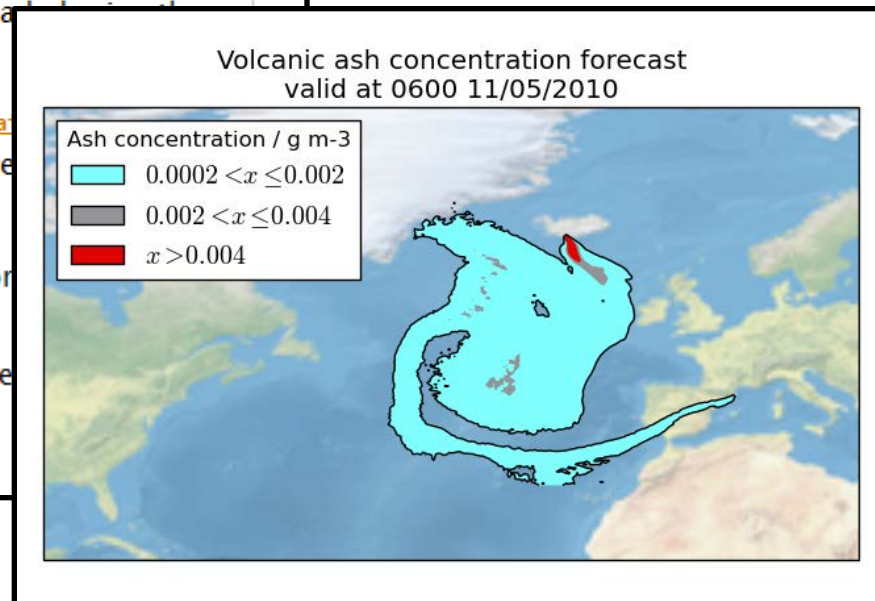
Loading a cube from a custom file format

This example shows how a custom text file can be loaded using the standard Iris load mechanism.

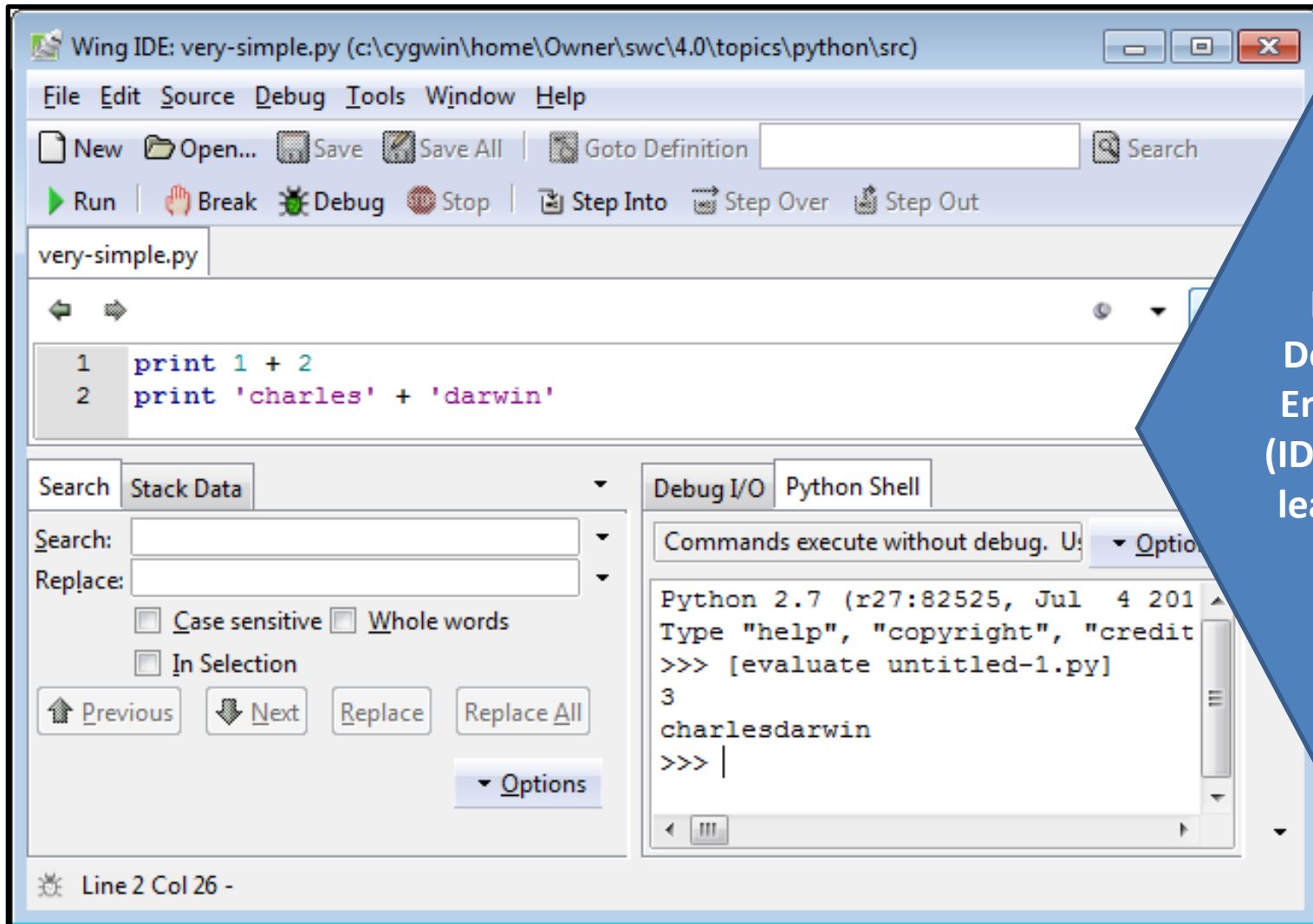
The first stage in the process is to define an Iris [Format](#) for the file format. To create a format specification we define the following:

- `format_name` – Some text that describes the format specification we are creating
- `file_element` – FileElement object describing the file which identifies this FormatSpecification.

Open source packages for data analysis and visualisation.

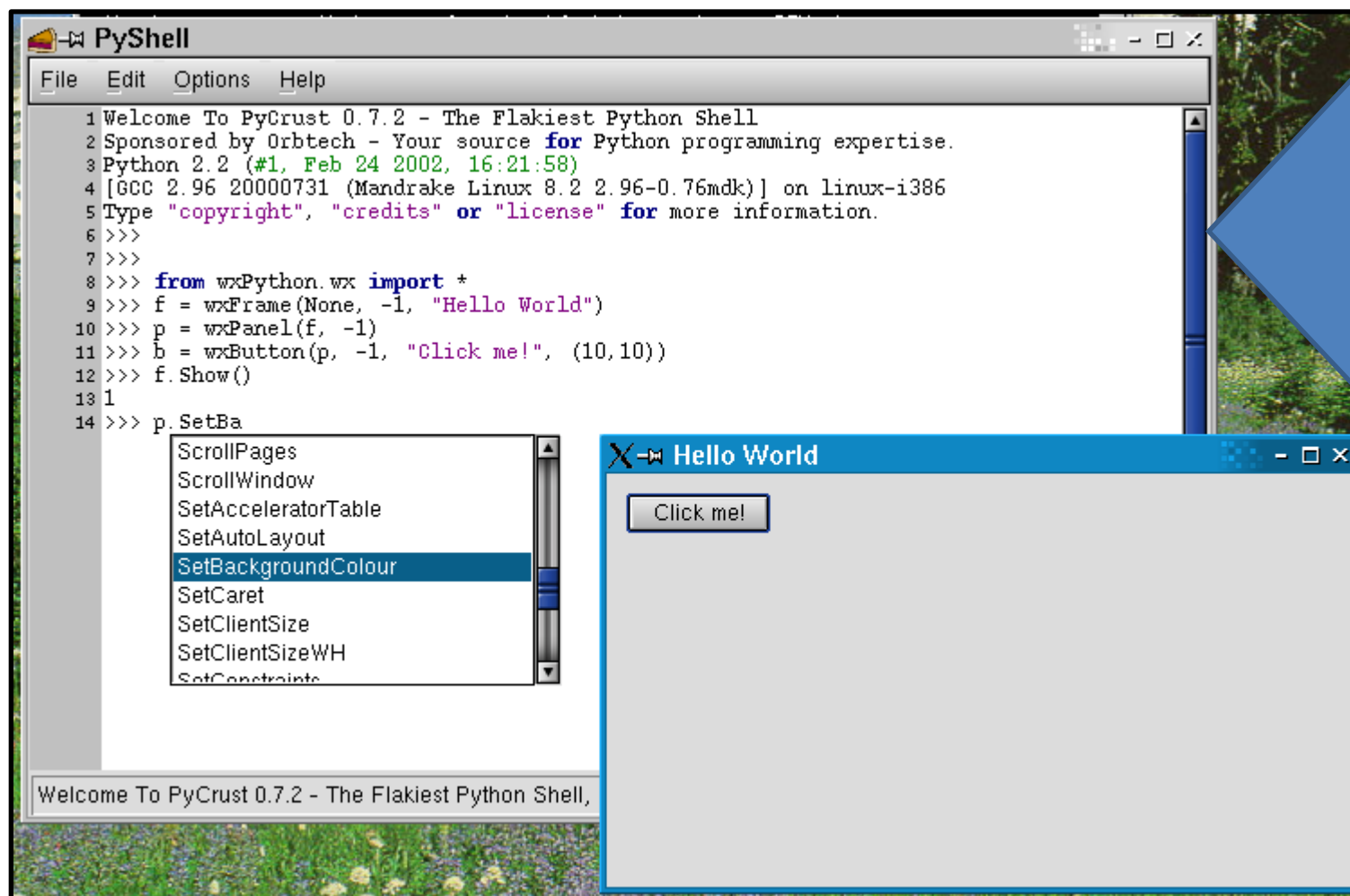


What can you do with python?



Integrated Development Environments (IDEs): help you learn/manage coding.

What can you do with python?



Tools for
building GUIs:
e.g. wxpython

Let's get to work...