Programming Concepts: IDEs, Debug

Paulo Penteado

http://www.ppenteado.net/pc/

(http://phdcomics.com/comics/archive.php?comicid=1690)

IDEs

Interactive Development Environments

Exist for every language (even stuff like LaTeX, HTML, CSS).

Provide a single environment for

- editing
- compiling / packing
- running
- debugging
- file inspection (any file, not only source files)
- documentation access
- profiling (performance measurement)
- interactive use
- GUI (graphical interface) building

The IDE knows the language being use.

Very useful, often ignored tool (when one uses, instead, text editor + command line).

Some IDEs are multilanguage and multiplatform.

IDEs

Main examples:

- **Eclipse** – nearly any platform and language *(even obscure ones like IDL, R)*
- Spyder – multiplatform, Python only
- Enthought Canopy – multiplatform, Python only
- MS Visual Studio - Windows, C, C++, VB, C#, and several others (only the Express version, more limited, is free)
- NetBeans – any platform, Java, C, C++, e several others
- Xcode - Mac, C, C++, Objective C, Java, and several others
- Solaris Studio (old Sun Studio) - Linux, Unix, C, C++, Fortran
- C++ Builder (old Borland C++) - Windows, C, C++ (commercial)

The editors in the IDEs are the most capable, because they know the languages:

- **S*yntax highlighting***.
- **Context help***.*
- **Integrated debugger** – setting and using breakpoints, looking up variable values.
- In some languages, show error codes live errors, similarly to spellcheckers (not only syntax errors: might show errors in function arguments).

IDEs – example (Eclipse, integrated to IDL ≥7)

mylib/pp_lib/pp_editablecube__define.pro - IDL

File Edit Source Project Run Window Help

Open New File New Project Save Cut Copy Paste Undo Back Forward Compile Run Stop Step Step Over $MAIN$ Call Stack Reset

Proj... Outl... Vari...

pp_editablecube__define.pro
- ::init( )
- ::setproperty
- ::updatedatainfo
- ::updatesuffix
- ::updatecore
- ::updatelocations
- ::headeredit
- ::write
- ::cleanup
- pp_editablecube__define

pp_lorentz_from_fw... | histogram_pp.pro | pp_multiplot__defin... | pp_editablecube__d...

```
168 ;Update the bottomplanes if provided
169 if bottomchange||dimschanged[1]||typechanged then self->updatesuffix,suffix=bottomplanes,'LI
170
171 ;Fix suffix dimension information, if it changed
172 if (~array_equal(self.info.suffdims,oldsuffdims)) then begin
173   self.info.dims=self.info.coredims+self.info.suffdims
174   pp_setcubeheadervalue,*self.labels,'SUFFIX_ITEMS',self.info.suffdims
175 en
176
177 ;R
178 if
179
180
181
182
183
184
185
186
187
188
189   byteorder,raw,swap_if_big_endian=self.info.littleendian,$
190    swap_if_little_endian=~self.info.littleendian
191   self.raw=ptr_new(raw,/no_copy)
192 endif
```

Syntax

pp_setcubeheadervalue, header, key, value

Documentation

:Description:

Sets the value of the given key from the given header in ISIS cube format. If value is not provided, the key is erased from the header. If append is given, its contents are just inserted into the header, without affecting the rest of its contents (and value and key are ignored).

:Params:

header : in, required

A string array where each element is one line of an ISIS cube.

Press 'F2' for focus

Console | Command History | Problems — Current Directory: /home/penteado/idl

IDL

```
% Loaded DLM: PNG.
IDL> nx=201
IDL> x=(dindgen(nx)/(nx-1d0)-0.5d0)*5d0
IDL> yl=pp_lorentz_from_fwhm(x,fwhm=1d0)
IDL> pl=plot(x,yl,color='blue',name='Lorentzian, FWHM=1d0',thick=2.)
IDL> yg=pp_gauss_from_fwhm(x,fwhm=1d0)
IDL> pg=plot(x,yg,color='red',name='Gaussian, FWHM=1d0',thick=2.,/over)
IDL> l=legend(target=[pg,pl],position=[0.5,0.5]) ;Identify the two lines
IDL> pl.save,'pp_lorentz_from_fwhm.png',resolution=100
IDL>
```

Writable | Smart Insert | 182 : 24

IDEs – example (Spyder)

File Edit Search Source Run Interpreters Tools View ?

/tmp

Object inspector

Source Console Object .zeros Options

zeros

Definition : zeros(shape, dtype=float, order='C')

Type : Function of numpy.core.multiarray module

Return a new array of given shape and type, filled with zeros.

Parameters

Editor - /homec/penteado/cpc/cpcc/new/pp_comparecolors.py

pp_comparecolors.py* acepairs.py fakes.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Apr  2 17:18:08 2014
4
5 @author: penteado
6 """
7
8 import numpy as np
9 def pp_comparecolors(color1,color2):
10   return np.amax(np.abs(color1-color2))
11
12 if __name__ == '__main__':
13
14   color1=np.array((200,190,0),dtype='u1')
15   f=np.array()
16   color2=np.array((198,190,0),dtype='u1')
17   color3=np.array((201,190,0),dtype='u1')
```

Object ins... Variable ex... O

abs
absolute
add
add_docstring
add_newdoc
add_newdocs
alen
all
allclose
ALLOW_THREADS
alltrue
alterdot

Console

Python 1 00:06:29

```
Python 2.7.3 (default, Apr 30
[GCC 4.7.0 20120416 (Red Hat
Type "help", "copyright", "cr

Imported NumPy 1.6.2, SciPy 0
Type "scientific" for more de
>>> import numpy as np
>>> a=np.zeros((12),dtype=np.)
```

Internal console Console History log

Permissions: RW End-of-lines: LF Encoding: UTF-8 Line: 15 Column: 14 Memory:

Not quite an IDE – IPython notebook

File Edit View History Bookmarks Tools Help

IPy IPython Dashboard | IPy example1

localhost:8888/609cd1b4-660b-40e9-b13e-4b0aee7437d6

IP[y]: Notebook

example1 | Save | QuickHelp

Notebook

Actions: New | Open | Download | ipynb | Print

Cell

Actions: Delete

Format: Code | Markdown

Output: Toggle | ClearAll

Insert: Above | Below

Move: Up | Down

Run: Selected | All

Autoindent: ✓

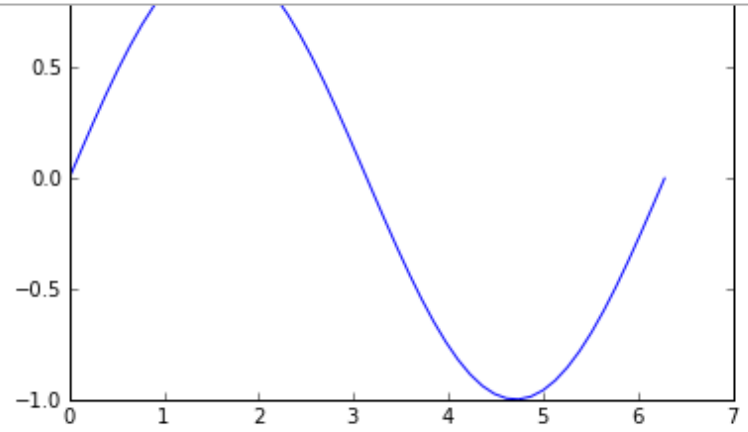
Kernel

Actions: Interrupt | Restart

Kill kernel upon exit: ☐

Help

Links: Python | IPython | NumPy | SciPy



In [9]:
```
import urllib
urllib.urlretrieve('http://phy49.phy.nau.edu/frontier/pp_ff_index.fits','p
```

Out[9]:
```
('pp_ff_index.fits', <httplib.HTTPMessage instance at 0x2ad0170>)
```

In [10]:
```
import pyfits
f=pyfits.open('pp_ff_index.fits')
```

In [11]:
```
f.info()
```

```
Filename: pp_ff_index.fits
No.    Name         Type       Cards   Dimensions   Format
0    PRIMARY     PrimaryHDU      4    ()           uint8
1                BinTableHDU    73    417R x 28C   [K, 5A, 13A, K, 4A, D,
```

Debug – what is a debugger?

The most helpful tool to diagnose and fix bugs. (specially if used in an IDE).

Provides a “live” and interactive analysis of the program execution.

- Like seeing what goes on inside a machine, while it runs. Much easier to see where the problem is.

The most common debugging approach:

- If the program crashes, see which line of code caused it.
- Inspect the program's variables, and how they change during program execution.

A common, slow, tiresome alternative is fill the program with ***prints***, to know where it is, and the values / characteristics of some variables.

- **Debuggers allow interactive inspection, much faster and more flexible than prints.**

Debug – what does a debugger do?

Runs the code, observing where it goes and what it does.

When an interruption happens (error, interactively, or reaching a breakpoint), the debugger provides:

- Contents, type and dimensions of **every variable** in scope.
- Evaluation / visualization / function calling / analysis of expressions with the variables.
- **Arguments used** when the current function was called.
- *Call stack* – **the sequence of function calls that led to the current point.**
- ***Call stack*** navigation – change the current scope among different points in the call stack.
- **Incremental code execution –** take gradual steps, to see the changes that happen when each [line / function / code fragment] is executed. Possibly skipping lines that cause trouble.
- Change the values in some variables, before resuming execution.

Debug – example (Eclipse, integrated to IDL ≥7)

mylib/pp_lib/pp_multiplot__define.pro - IDL

File Edit Source Project Run Window Help

Open New File New Project Save Cut Copy Paste Undo Back Forward Compile Resume Stop Step Step Over PP_MULTIPLOT::S Call Stack Reset

Project Ex... Outline Variables

Name	Value
DSS	<Undefined>
EL	<Undefined>
I	<Undefined>
LAYOUT	0
LINE	0
MINDEX	1
NEWXRANGES	<Undefined>
NEWYRANGES	<Undefined>
PROPS	<Undefined>
SEL	<Undefined>
SELF	PP_MULTIPLOT (Id=10
TO_UPDATE	<Undefined>
XR	Double[2]
YR	<Undefined>

pp_lorentz_from_... pp_multiplot__de... idlitgrworld__de...

```
1028      foreach el,self.oplots,i do if isa(el) then begin
1029        ds=stregex(el.getfullidentifier(),'.*(/DATA SPACE).*/',/extract)
1030        if dss eq ds then break
1031      endif
1032      mindex=i
1033    endif
1034  endif
1035  if (n_elements(mindex) eq 0) then mindex=0
1036  ;Get the ranges from the given index
1037  mindex=0>mindex<(self.ncolumns*self.nlines-1)
1038  column=mindex mod self.ncolumns
1039  [illegible]f.ncolumns
1040  [illegible]s: Updating axes to those of index '+strtrim(mindex,2)
1041  [illegible](mindex) eq 1) then begin
1042    if isa((self.oplots)[mindex]) then begin
1043      xr=((self.oplots)[mindex]).xrange
1044      yr=((self.oplots)[mindex]).yrange
1045      self.setendticks,(self.xendticks)[mindex],(self.oplots)[mindex],'x'
1046      self.setendticks,(self.yendticks)[mindex],(self.oplots)[mindex],'y'
1047    endif else begin
1048      print,'sync_axes: Selected index does not yet contain a plot; doing nothing'
1049      return
```

column = 1
Press 'F2' for focus

Console Command History Problems Current Directory: /home/penteado/idl

```
IDL
% Compiled module: PP_MULTIPLOT::SYNC_AXES.
% Compiled module: PP_MULTIPLOT::CLOSE.
% Compiled module: PP_MULTIPLOT::SAVE.
% Compiled module: PP_MULTIPLOT__DEFINE.
IDL> m.sync_axes,1
sync_axes: Updating axes to those of index 1
% Breakpoint at: PP_MULTIPLOT::SYNC_AXES 1044 /software/idl/pp_lib/pp_multiplot__define.pro
IDL>
```

Debug – example (Spyder)

File Edit Search Source Run Interpreters Tools View ?

/tmp

Variable explorer

Name	Type	Size	Value
color1	uint8	(3,)	array([200, 190, 0], dtype=uint8)
color2	uint8	(3,)	array([198, 190, 0], dtype=uint8)
color3	uint8	(3,)	array([201, 190, 0], dtype=uint8)
e	float	1	2.718281828459045
pi	float	1	3.141592653589793

Object ins... Variable ex... Onlin... File ex... Break...

Editor - /homec/penteado/cpc/cpcc/new/pp_comparecolors.py

pp_comparecolors.py acepairs.py fakes.p

```
 8 import numpy as np
 9 def pp_comparecolors(color1,color2):
10   return np.amax(np.abs(color1-color2))
11
12 def main():
13   color1=np.array((200,190,0),dtype='u1')
14   color2=np.array((198,190,0),dtype='u1')
15   color3=np.array((201,190,0),dtype='u1')
16
17
18   print pp_comparecolors(color1,color2)
19   print pp_comparecolors(color1,color3)
20
21 if __name__ == '__main__':
22
23   main()
24
```

Console

Python 1 00:12:53

```
-> print pp_comparecolors(color1,color3)
(Pdb) bt
  /usr/lib64/python2.7/bdb.py(387)run()
-> exec cmd in globals, locals
  <string>(1)<module>()
  /usr/lib/python2.7/site-packages/spyderlib/widgets/externalshell/sitecustomize.py(523)runfile()
-> execfile(filename, namespace)
  /homec/penteado/cpc/cpcc/new/pp_comparecolors.py(23)<module>()
-> main()
> /homec/penteado/cpc/cpcc/new/pp_comparecolors.py(19)main()
-> print pp_comparecolors(color1,color3)
(Pdb)
```

Internal console Console History log

Permissions: RW End-of-lines: LF Encoding: UTF-8 Line: 19 Column: 35 Memory:

Debug – example (IPython)

```
[penteado@javelina r4]$ python -m pdb /home/penteado/MOD/acepairs.py -f
56570.8292426.sex
> /home/penteado/MOD/acepairs.py(4)<module>()
-> import sys, re, string, os, getopt, math, numpy
(Pdb) b /home/penteado/MOD/acepairs.py:890
Breakpoint 1 at /home/penteado/MOD/acepairs.py:890
(Pdb) c
[ 56570.8292426  56570.8311639  56570.8330852  56570.8350065  56570.8369278
  56570.8388491  56570.8407704]
starting to find pairs
> /home/penteado/MOD/acepairs.py(890)find_pairs()
-> T1 = KDTree(self.pos[group])
(Pdb) dir()
['group', 'i', 'llim', 'pair_1st', 'pair_2nd', 'rad', 'self', 't', 'ulim']
(Pdb) ulim
56570.831163899995
(Pdb) bt
  /share/apps/local/Ureka/python/lib/python2.7/bdb.py(400)run()
-> exec cmd in globals, locals
  <string>(1)<module>()
  /home/penteado/MOD/acepairs.py(940)<module>()
-> process()
  /home/penteado/MOD/acepairs.py(82)process()
-> pairs = dataset.find_pairs(opt.maxpairsep)
> /home/penteado/MOD/acepairs.py(890)find_pairs()
-> T1 = KDTree(self.pos[group])
(Pdb) dir(self)
['__doc__', '__init__', '__module__', 'clubs', 'data', 'diff', 'filename',
'tbins', 'times']
```

Debug – programming pearls

Don't debug standing up. It cuts your patience in half, and you need all you can muster,
Dave Storer, Cedar Rapids,Iowa

Testing can show the presence of bugs, but not their absence.
Edsger W. Dijkstra, University of Texas

Each new user of a new system uncovers a new class of bugs.
Brian Kernighan, Bell Labs

The first step in fixing a broken program is getting it to fail repeatably.
Tom Duff, Bell Labs

http://users.erols.com/blilly/programming/Programming_Pearls.html

Some references

Tutorial: Debugging your Python Applications with pdb
https://www.youtube.com/watch?v=bZZTeKPRSLQ

The IPython Notebook (including video demo):
http://ipython.org/notebook.html

The Notebook Gallery (including the book *Lectures on Scientific Computing with Python*, written as IPython Notebooks):
https://github.com/ipython/ipython/wiki/A-gallery-of-interesting-IPython-Notebooks

Spyder video tutorial:
https://www.youtube.com/watch?v=4Iu7UrZXzEY
(one of several from Firebox Training, http://www.fireboxtraining.com/blog/python-tutorials)