Session 6: Introduction to advanced tools

October 9th, 2017 | Wouter Klijn

Overview

- Versioning (GIT)
- Tests
  - Types
  - How to start testing
  - Unittests
- Debugging
  - pdb
- Interactive Development Invironments

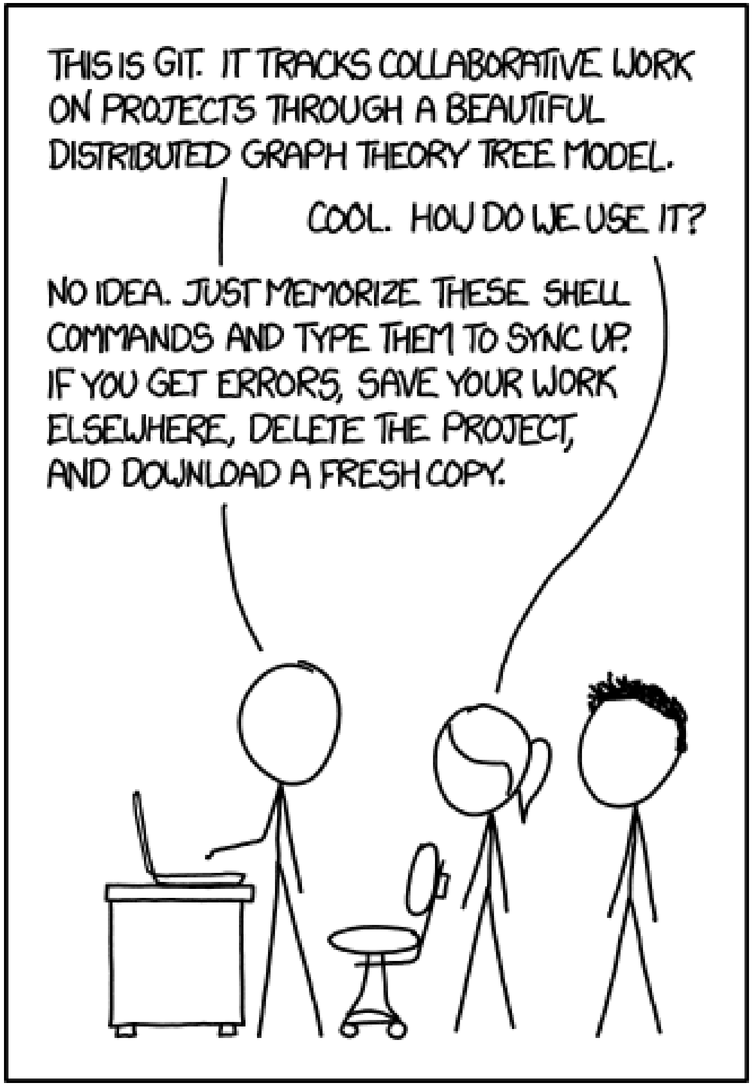
Git: Why

- Storage (backup) of source code file
- Who changed what when
- Undo / redo
- Facilitates working on multiple versions of a software
- Merge of changes from multiple developers

https://www.slideshare.net/phpcodemonkey/introduction-to-version-control-presentation

Git

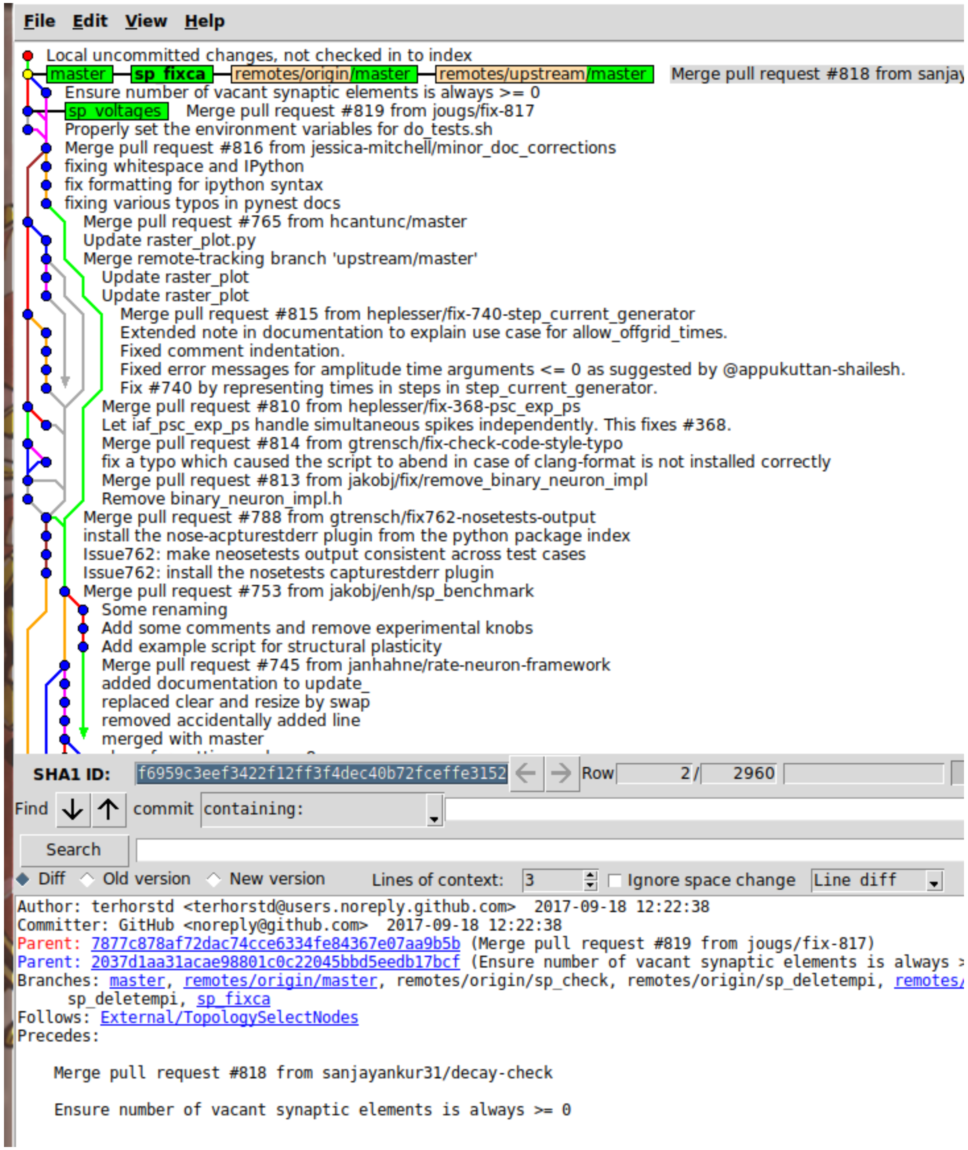
- 70s software interface
- Command line with ‘intuitive‘ arguments
- Graphical user interfaces: Tortoise git (Windows), GitKraken (Linux)
- Integration in mature IDE’s: PyCharm, Visual studio, Eclipse


THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

https://xkcd.com/1597/

Git

- clone
- checkout
- add
- commit
- fetch
- pull
- push
- remote
- branch



```
Local uncommitted changes, not checked in to index
master  sp_fixca  remotes/origin/master  remotes/upstream/master   Merge pull request #818 from sanjay
Ensure number of vacant synaptic elements is always >= 0
sp_voltages   Merge pull request #819 from jougs/fix-817
Properly set the environment variables for do_tests.sh
Merge pull request #816 from jessica-mitchell/minor_doc_corrections
fixing whitespace and IPython
fix formatting for ipython syntax
fixing various typos in pynest docs
Merge pull request #765 from hcantunc/master
Update raster_plot.py
Merge remote-tracking branch 'upstream/master'
Update raster_plot
Update raster_plot
Merge pull request #815 from heplesser/fix-740-step_current_generator
Extended note in documentation to explain use case for allow_offgrid_times.
Fixed comment indentation.
Fixed error messages for amplitude time arguments <= 0 as suggested by @appukuttan-shailesh.
Fix #740 by representing times in steps in step_current_generator.
Merge pull request #810 from heplesser/fix-368-psc_exp_ps
Let iaf_psc_exp_ps handle simultaneous spikes independently. This fixes #368.
Merge pull request #814 from gtrensch/fix-check-code-style-typo
fix a typo which caused the script to abend in case of clang-format is not installed correctly
Merge pull request #813 from jakobj/fix/remove_binary_neuron_impl
Remove binary_neuron_impl.h
Merge pull request #788 from gtrensch/fix762-nosetests-output
install the nose-acpturestderr plugin from the python package index
Issue762: make neosetests output consistent across test cases
Issue762: install the nosetests capturestderr plugin
Merge pull request #753 from jakobj/enh/sp_benchmark
Some renaming
Add some comments and remove experimental knobs
Add example script for structural plasticity
Merge pull request #745 from janhahne/rate-neuron-framework
added documentation to update_
replaced clear and resize by swap
removed accidentally added line
merged with master
```

SHA1 ID: f6959c3eef3422f12ff3f4dec40b72fceffe3152 Row 2 / 2960

Lines of context: 3

```
Author: terhorstd <terhorstd@users.noreply.github.com>  2017-09-18 12:22:38
Committer: GitHub <noreply@github.com>  2017-09-18 12:22:38
Parent: 7877c878af72dac74cce6334fe84367e07aa9b5b (Merge pull request #819 from jougs/fix-817)
Parent: 2037d1aa31acae98801c0c22045bbd5eedb17bcf (Ensure number of vacant synaptic elements is always >
Branches: master, remotes/origin/master, remotes/origin/sp_check, remotes/origin/sp_deletempi, remotes/
      sp_deletempi, sp_fixca
Follows: External/TopologySelectNodes
Precedes:

    Merge pull request #818 from sanjayankur31/decay-check

    Ensure number of vacant synaptic elements is always >= 0
```

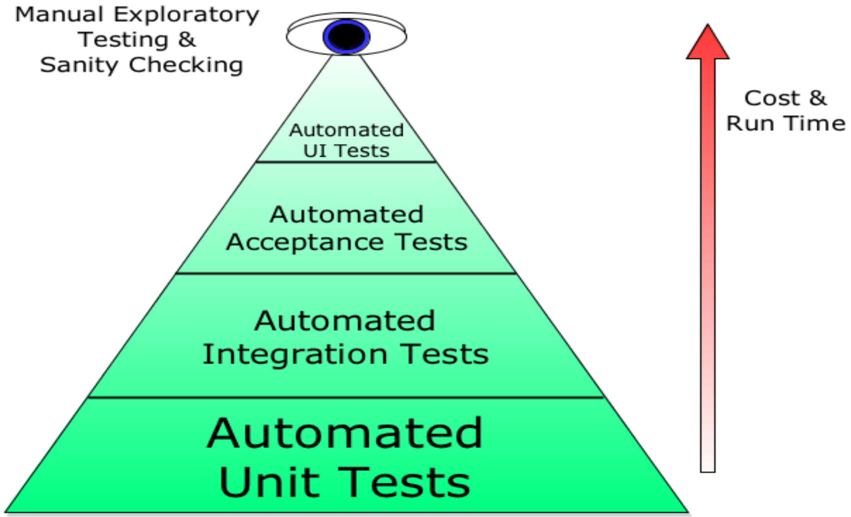
Testing

- Automatic programs or checklist assessing the correction functioning of software.
- Prevent introduction of errors when adding features.
- But also:
  - Tests as documentation
  - Leads to better design: loose coupling
  - In larger projects, improved development speed (mostly due to reduction in bugs to be solved)

Testing pyramid

- Major types of tests:
    - Manual testing
    - Data driven delta testing / regression testing
    - Component testing
    - Unit testing

The concepts are fuzzy and there is overlap and different names for the same thing


Manual Exploratory Testing & Sanity Checking
Automated UI Tests
Automated Acceptance Tests
Automated Integration Tests
Automated Unit Tests
Cost & Run Time

http://willhamill.com/2013/08/12/automated-testing-and-the-evils-of-ice-cream

How to start testing?

- Writing down the **manual tests** you already do
  - Doubles as documentation
- Create an **data driven delta** test
  - Create test data
  - Forces you to think about ‘user’ interactions
  - Doubles as introductory how-to
- Pick a single important **component** and disconnect it from the rest.
  - And continue doing this till you end up with:
- **Unit test** for small parts of the code that do one and only one thing.

Python: unittest

- Based on the xunit standard

- Setup -> test -> teardown
  1. Create files, etc. needed to run the component

  1. Run individual function an test the correct output eg:
     - assertEqual
     - assertTrue
     - assertExceptionThrown

  1. Delete used resources

http://pythontesting.net/framework/unittest/unittest-introduction/

Python: unittest

```
import unittest

def function(parameter):
    return parameter

class TestSomething(unittest.TestCase):

    def setUp(self):
        pass

    def test_fail(self):
        self.assertEqual(function(13), 12)

    def test_succes(self):
        self.assertEqual(function(12), 12)

    def tearDown(self):
        pass

if __name__ == '__main__':
    unittest.main()
```

```
wouter@WKLIJNWORK:/mnt/c/work$ python3 unittester.py
F.
======================================================================
FAIL: test_something_fail (__main__.TestSomething)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "unittester.py", line 15, in test_fail
    self.assertEqual(function(13), 12)
AssertionError: 13 != 12


----------------------------------------------------------------------
Ran 2 tests in 0.001s


FAILED (failures=1)
```

Debugging

- Debug print statements
- Use binairy search to find the problem. If you know your program this is often the fastest
- If the program is big, or not your own, it’s a hard problem:

```
python –m pdb program.py
```

Debugging: pdb

Command	action
n	Execute the next command
enter	Repeat the last command
q	Hard exit (with a signal / exception)
p <var>,<var>	Print the value of the variable
c	Continue with program (until trace_point)
s	Step into a function
r	Continue till end of function
list <n1,n2>	Print surrounding code, include (n1, n2)

https://pythonconquerstheuniverse.wordpress.com/2009/09/10/debugging-in-python/

Debugging: pdb cont.

- PDB starts your program and halts at the first statement.

- For large programs you can add trace points:

  ```
  import pdb
  pdb.set_trace()
  ```

- Execution will drop into debugging mode

When doing interactive development:

- pdb.run('statement to evaluated')

https://pymotw.com/2/pdb/

Debugging: pdb advanced

Interactive development:

- pdb.run('statement to evaluated')

- Postmortem:
  - pdb.pm()
    "Debugging of the sys.last_backtrace"
  - Could be use in combination with except

- For more in-depth information:
  - https://pymotw.com/3/pdb/index.html

https://pymotw.com/2/pdb/

IDE

- The biggest difference between python and Matlab is the Integrated Development Environment (IDE)
- Python is typically interacted with via code or console.
- Selecting an IDE is an ‘important’ choice.
    - It takes time to get use to a IDE
    - Operating system
    - Features

IDE

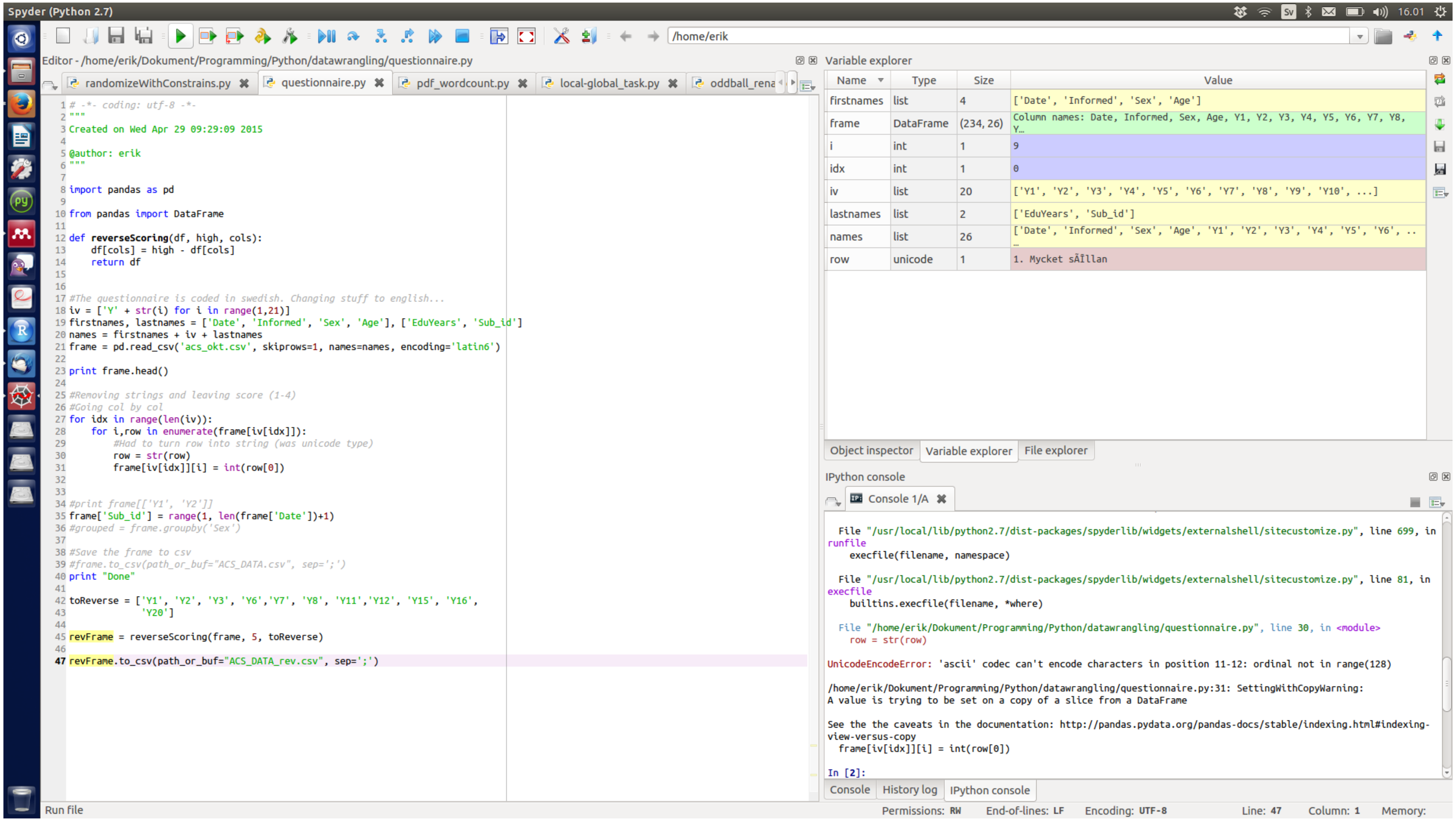
- Spyder: MATLAB like interface
    - Available on most operating systems
    - Python centric

- Visual Studio: python development tools
    - Windows
    - Prepared for later C++ development (Cython)

- Eclipse JAVA based but supports most languages
    - Available on most operating systems
    - Prepared for later C++ development

- PyCharm. Python centric IDE

IDE: Spyder



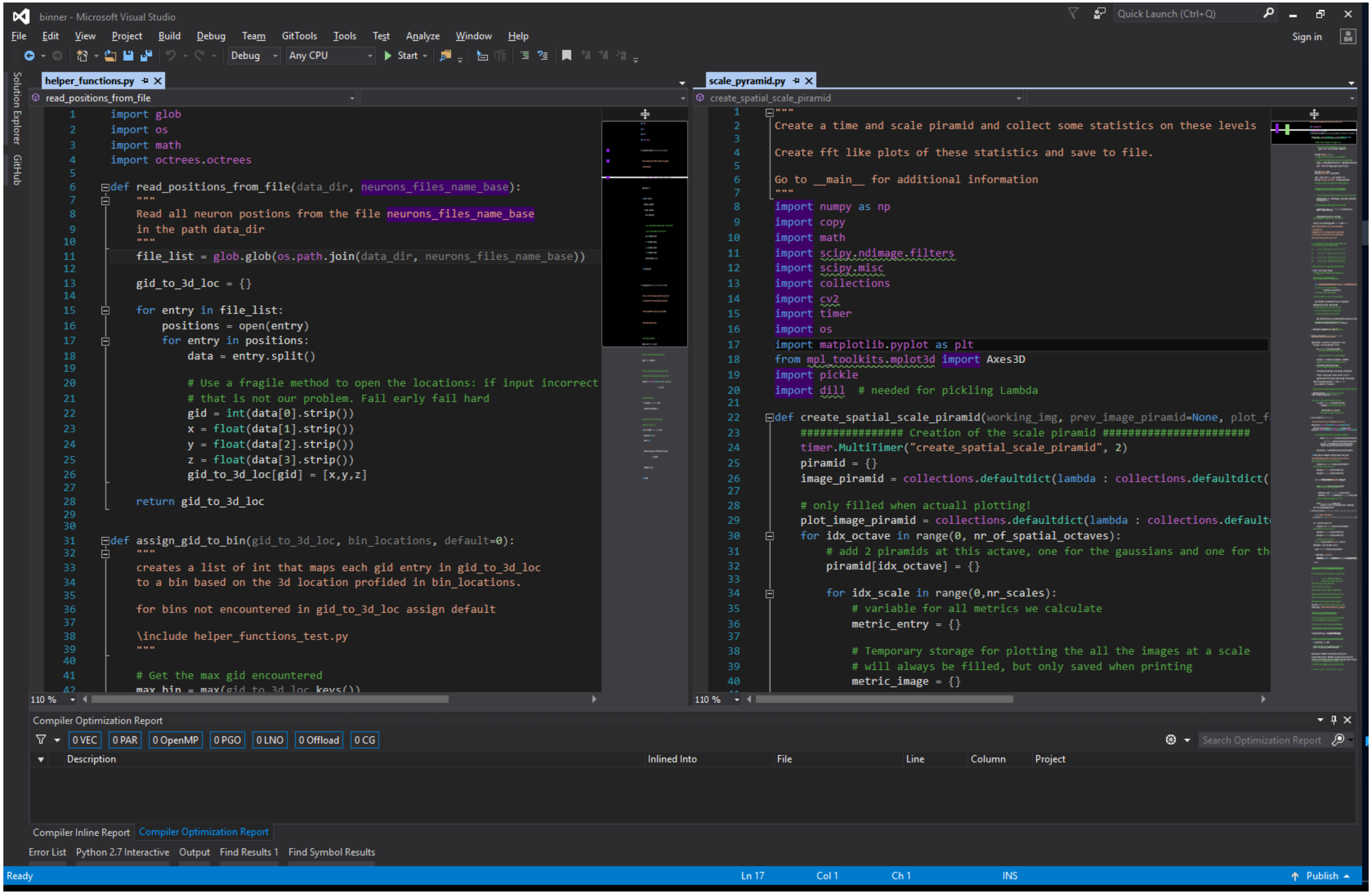
```python
# -*- coding: utf-8 -*-
"""
Created on Wed Apr 29 09:29:09 2015

@author: erik
"""

import pandas as pd

from pandas import DataFrame

def reverseScoring(df, high, cols):
    df[cols] = high - df[cols]
    return df


#The questionnaire is coded in swedish. Changing stuff to english...
iv = ['Y' + str(i) for i in range(1,21)]
firstnames, lastnames = ['Date', 'Informed', 'Sex', 'Age'], ['EduYears', 'Sub_id']
names = firstnames + iv + lastnames
frame = pd.read_csv('acs_okt.csv', skiprows=1, names=names, encoding='latin6')

print frame.head()

#Removing strings and leaving score (1-4)
#Going col by col
for idx in range(len(iv)):
    for i,row in enumerate(frame[iv[idx]]):
        #Had to turn row into string (was unicode type)
        row = str(row)
        frame[iv[idx]][i] = int(row[0])


#print frame[['Y1', 'Y2']]
frame['Sub_id'] = range(1, len(frame['Date'])+1)
#grouped = frame.groupby('Sex')

#Save the frame to csv
#frame.to_csv(path_or_buf="ACS_DATA.csv", sep=';')
print "Done"

toReverse = ['Y1', 'Y2', 'Y3', 'Y6','Y7', 'Y8', 'Y11','Y12', 'Y15', 'Y16',
             'Y20']

revFrame = reverseScoring(frame, 5, toReverse)

revFrame.to_csv(path_or_buf="ACS_DATA_rev.csv", sep=';')
```

Name	Type	Size	Value
firstnames	list	4	['Date', 'Informed', 'Sex', 'Age']
frame	DataFrame	(234, 26)	Column names: Date, Informed, Sex, Age, Y1, Y2, Y3, Y4, Y5, Y6, Y7, Y8, Y…
i	int	1	9
idx	int	1	0
iv	list	20	['Y1', 'Y2', 'Y3', 'Y4', 'Y5', 'Y6', 'Y7', 'Y8', 'Y9', 'Y10', ...]
lastnames	list	2	['EduYears', 'Sub_id']
names	list	26	['Date', 'Informed', 'Sex', 'Age', 'Y1', 'Y2', 'Y3', 'Y4', 'Y5', 'Y6', ..…
row	unicode	1	1. Mycket sÃĪllan

```
  File "/usr/local/lib/python2.7/dist-packages/spyderlib/widgets/externalshell/sitecustomize.py", line 699, in
runfile
    execfile(filename, namespace)

  File "/usr/local/lib/python2.7/dist-packages/spyderlib/widgets/externalshell/sitecustomize.py", line 81, in
execfile
    builtins.execfile(filename, *where)

  File "/home/erik/Dokument/Programming/Python/datawrangling/questionnaire.py", line 30, in <module>
    row = str(row)

UnicodeEncodeError: 'ascii' codec can't encode characters in position 11-12: ordinal not in range(128)

/home/erik/Dokument/Programming/Python/datawrangling/questionnaire.py:31: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-
view-versus-copy
  frame[iv[idx]][i] = int(row[0])

In [2]:
```

https://www.marsja.se/rstudio-like-python-ides-rodeo-spyder/

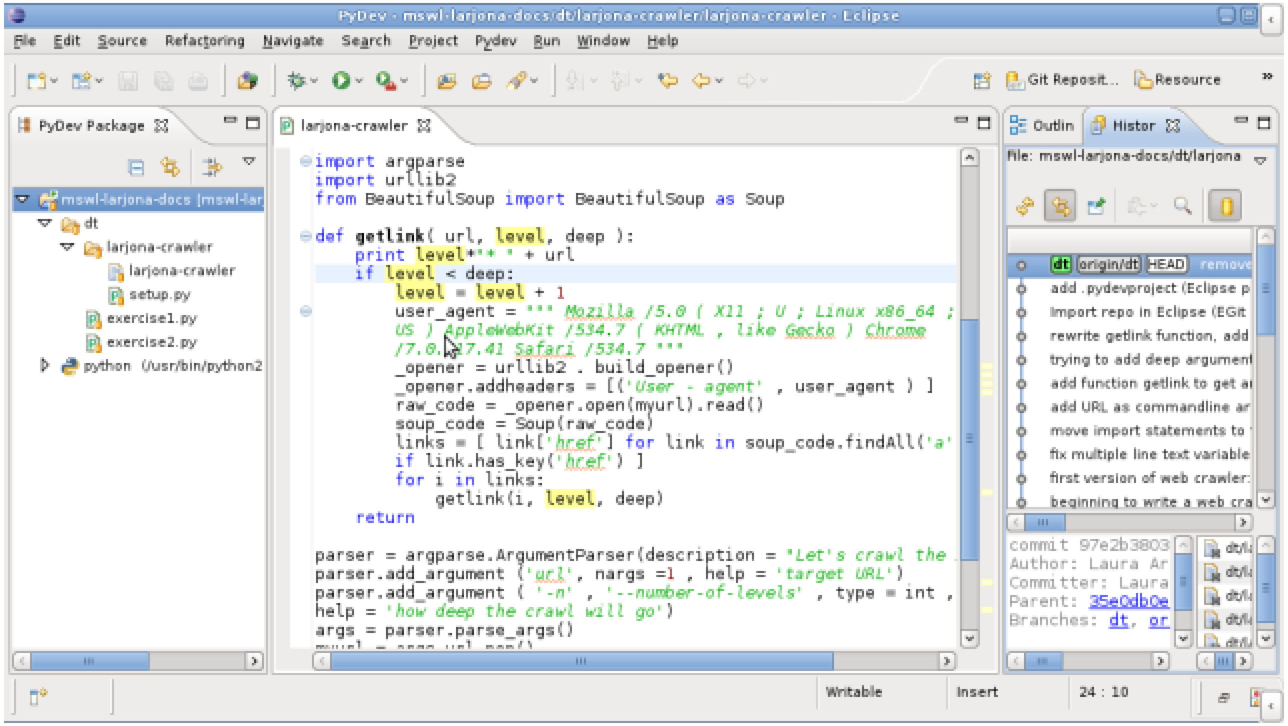
IDE: Visual Studio


binner - Microsoft Visual Studio
File Edit View Project Build Debug Team GitTools Tools Test Analyze Window Help
Debug
Any CPU
Start
Quick Launch (Ctrl+Q)
Sign in
Solution Explorer
GitHub
helper_functions.py
read_positions_from_file
scale_pyramid.py
create_spatial_scale_piramid
110 %
110 %

```python
import glob
import os
import math
import octrees.octrees

def read_positions_from_file(data_dir, neurons_files_name_base):
    """
    Read all neuron postions from the file neurons_files_name_base
    in the path data_dir
    """
    file_list = glob.glob(os.path.join(data_dir, neurons_files_name_base))

    gid_to_3d_loc = {}

    for entry in file_list:
        positions = open(entry)
        for entry in positions:
            data = entry.split()

            # Use a fragile method to open the locations: if input incorrect
            # that is not our problem. Fail early fail hard
            gid = int(data[0].strip())
            x = float(data[1].strip())
            y = float(data[2].strip())
            z = float(data[3].strip())
            gid_to_3d_loc[gid] = [x,y,z]

    return gid_to_3d_loc


def assign_gid_to_bin(gid_to_3d_loc, bin_locations, default=0):
    """
    creates a list of int that maps each gid entry in gid_to_3d_loc
    to a bin based on the 3d location profided in bin_locations.

    for bins not encountered in gid_to_3d_loc assign default

    \include helper_functions_test.py
    """

    # Get the max gid encountered
    max_bin = max(gid_to_3d_loc.keys())
```

```python
"""
Create a time and scale piramid and collect some statistics on these levels

Create fft like plots of these statistics and save to file.

Go to __main__ for additional information
"""
import numpy as np
import copy
import math
import scipy.ndimage.filters
import scipy.misc
import collections
import cv2
import timer
import os
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import pickle
import dill  # needed for pickling Lambda

def create_spatial_scale_piramid(working_img, prev_image_piramid=None, plot_f
    ################ Creation of the scale piramid ######################
    timer.MultiTimer("create_spatial_scale_piramid", 2)
    piramid = {}
    image_piramid = collections.defaultdict(lambda : collections.defaultdict(

    # only filled when actuall plotting!
    plot_image_piramid = collections.defaultdict(lambda : collections.defaultd
    for idx_octave in range(0, nr_of_spatial_octaves):
        # add 2 piramids at this actave, one for the gaussians and one for th
        piramid[idx_octave] = {}

        for idx_scale in range(0,nr_scales):
            # variable for all metrics we calculate
            metric_entry = {}

            # Temporary storage for plotting the all the images at a scale
            # will always be filled, but only saved when printing
            metric_image = {}
```

IDE: Eclipse



PyDev - mswl-larjona-docs/dt/larjona-crawler/larjona-crawler - Eclipse
File Edit Source Refactoring Navigate Search Project Pydev Run Window Help
Git Reposit... Resource
PyDev Package
mswl-larjona-docs [mswl-lar
dt
larjona-crawler
larjona-crawler
setup.py
exercise1.py
exercise2.py
python (/usr/bin/python2
larjona-crawler
Outlin Histor
File: mswl-larjona-docs/dt/larjona
dt origin/dt HEAD remove
Writable
Insert
24 : 10

```
import argparse
import urllib2
from BeautifulSoup import BeautifulSoup as Soup

def getlink( url, level, deep ):
    print level*"* " + url
    if level < deep:
        level = level + 1
        user_agent = """ Mozilla /5.0 ( X11 ; U ; Linux x86_64 ;
        US ) AppleWebKit /534.7 ( KHTML , like Gecko ) Chrome
        /7.0.517.41 Safari /534.7 """
        _opener = urllib2 . build_opener()
        _opener.addheaders = [('User - agent' , user_agent ) ]
        raw_code = _opener.open(myurl).read()
        soup_code = Soup(raw_code)
        links = [ link['href'] for link in soup_code.findAll('a'
        if link.has_key('href') ]
        for i in links:
            getlink(i, level, deep)
    return

parser = argparse.ArgumentParser(description = 'Let's crawl the
parser.add_argument ('url', nargs =1 , help = 'target URL')
parser.add_argument ( '-n' , '--number-of-levels' , type = int ,
help = 'how deep the crawl will go')
args = parser.parse_args()
myurl = args.url.pop()
```

```
add .pydevproject (Eclipse p
Import repo in Eclipse (EGit
rewrite getlink function, add
trying to add deep argument
add function getlink to get a
add URL as commandline ar
move import statements to
fix multiple line text variable
first version of web crawler:
beginning to write a web cra
```

```
commit 97e2b3803
Author: Laura Ar
Committer: Laura
Parent: 35e0db0e
Branches: dt, or
```

https://larjona.wordpress.com/2011/09/27/first-steps-with-python-and-eclipse-ide/

Thank you for your attention

References and further reading: