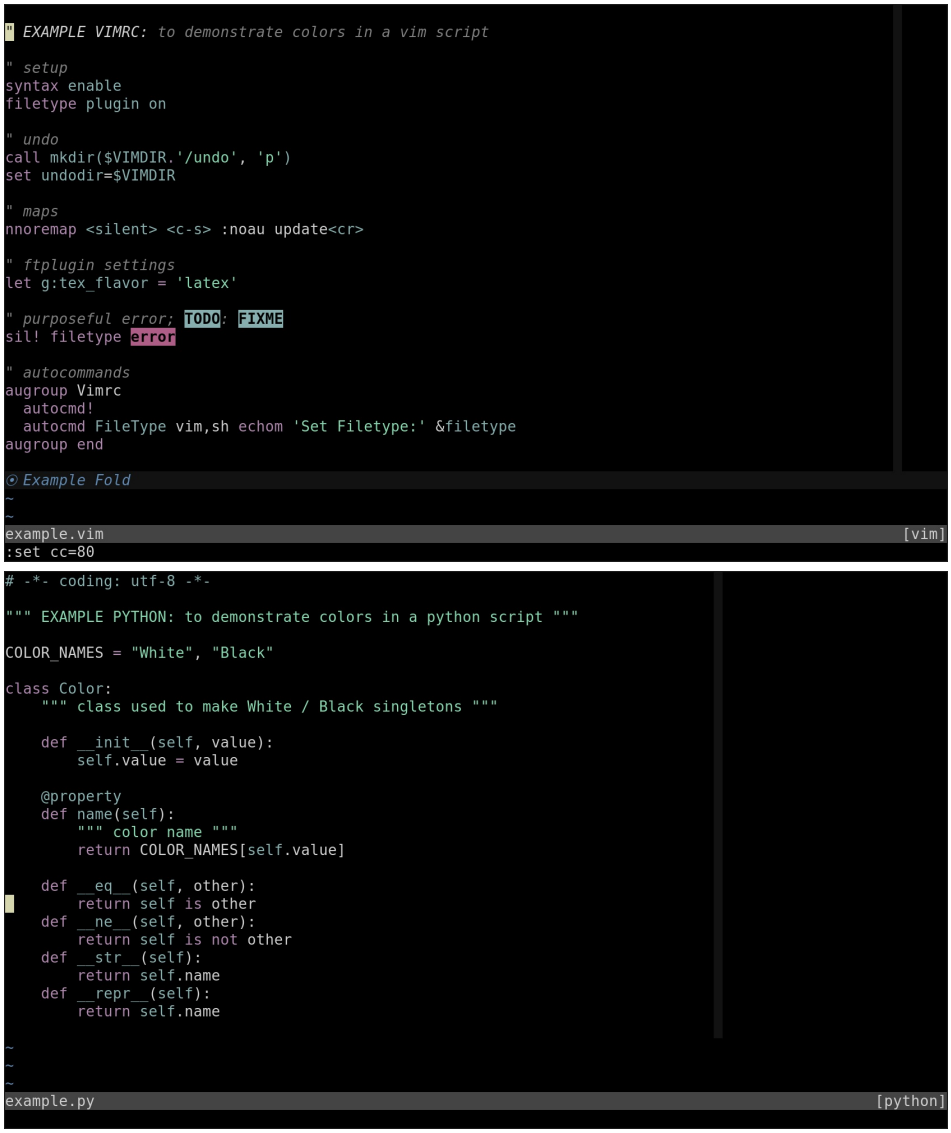
Nokto


" EXAMPLE VIMRC: to demonstrate colors in a vim script
" setup
syntax enable
filetype plugin on
" undo
call mkdir($VIMDIR.'/undo', 'p')
set undodir=$VIMDIR
" maps
nnoremap <silent> <c-s> :noau update<cr>
" ftplugin settings
let g:tex_flavor = 'latex'
" purposeful error; TODO: FIXME
sil! filetype error
" autocommands
augroup Vimrc
autocmd!
autocmd FileType vim,sh echom 'Set Filetype:' &filetype
augroup end
Example Fold
example.vim
[vim]
:set cc=80

-*- coding: utf-8 -*-
""" EXAMPLE PYTHON: to demonstrate colors in a python script """
COLOR_NAMES = "White", "Black"
class Color:
""" class used to make White / Black singletons """
def __init__(self, value):
self.value = value
@property
def name(self):
""" color name """
return COLOR_NAMES[self.value]
def __eq__(self, other):
return self is other
def __ne__(self, other):
return self is not other
def __str__(self):
return self.name
def __repr__(self):
return self.name
example.py
[python]

```c
/* EXAMPLE C: to demonstrate colors in a C program */

#include <stdio.h>

static char* author = "Nova Senco";

int main(int argc, char* argv[])
{
  int i;
  printf("author: %s\n", author);
  /* print number of arguments */
  for (i = 1; i < argc; ++i) {
    /* print each argument in a loop */
    printf("[%d]: %s\n", i - 1, argv[i]);
  }

  return 0;
}

~
~
~
~
~
~
~
~
~
example.c                                                    [c]
:set cc=80
```

```ruby
module VimColors
  class RubyExample
    CONSTANT = /^[0-9]+ regex awesomes$/

    attr_reader :colorscheme

    # TODO: Bacon ipsum dolor sit amet
    def initialize(attributes = {})
      @colorscheme = attributes[:colorscheme]
    end

    def self.examples
      # Bacon ipsum dolor sit amet
      ['string', :symbol, true, false, nil, 99.9, 1..2].each do |value|
        puts "it appears that #{value.inspect} is a #{value.class}"
      end

      {:key1 => :value1, key2: 'value2'}.each do |key, value|
        puts "the #{key.inspect} key has a value of #{value.inspect}"
      end

      %w[One Two Three].each { |number| puts number }
    end

    private

    def heredoc_example
      <<-SQL
        SELECT *
example.rb                                                   [ruby]
```