O que aprendi ensinando Python?

Uma breve história da minha vida

Como vai ser?

- Uma breve introdução, sem promoções, de eu
- O que são comunidades
- Comunidades de software
  - Um pouco de história
  - Como isso mudou o mundo
  - O que a gente ganha com isso?
  - Posso contribuir com isso?
- Tá, mas e daí?

- Versão: 1.0
- Alias: Minduim
- Bebida preferida: Leite

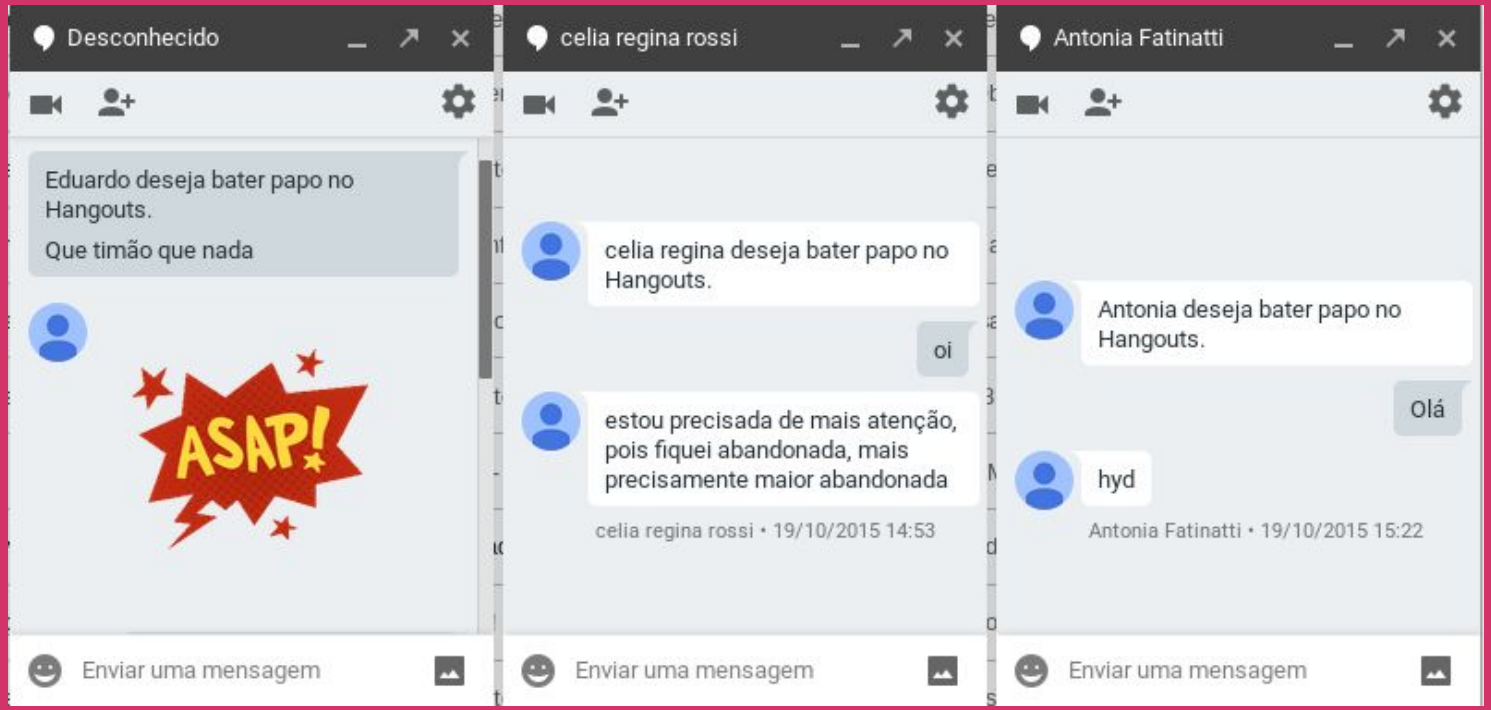
- Versão: 2.0
- Alias: Du
- Bebida preferida: Coca-Cola

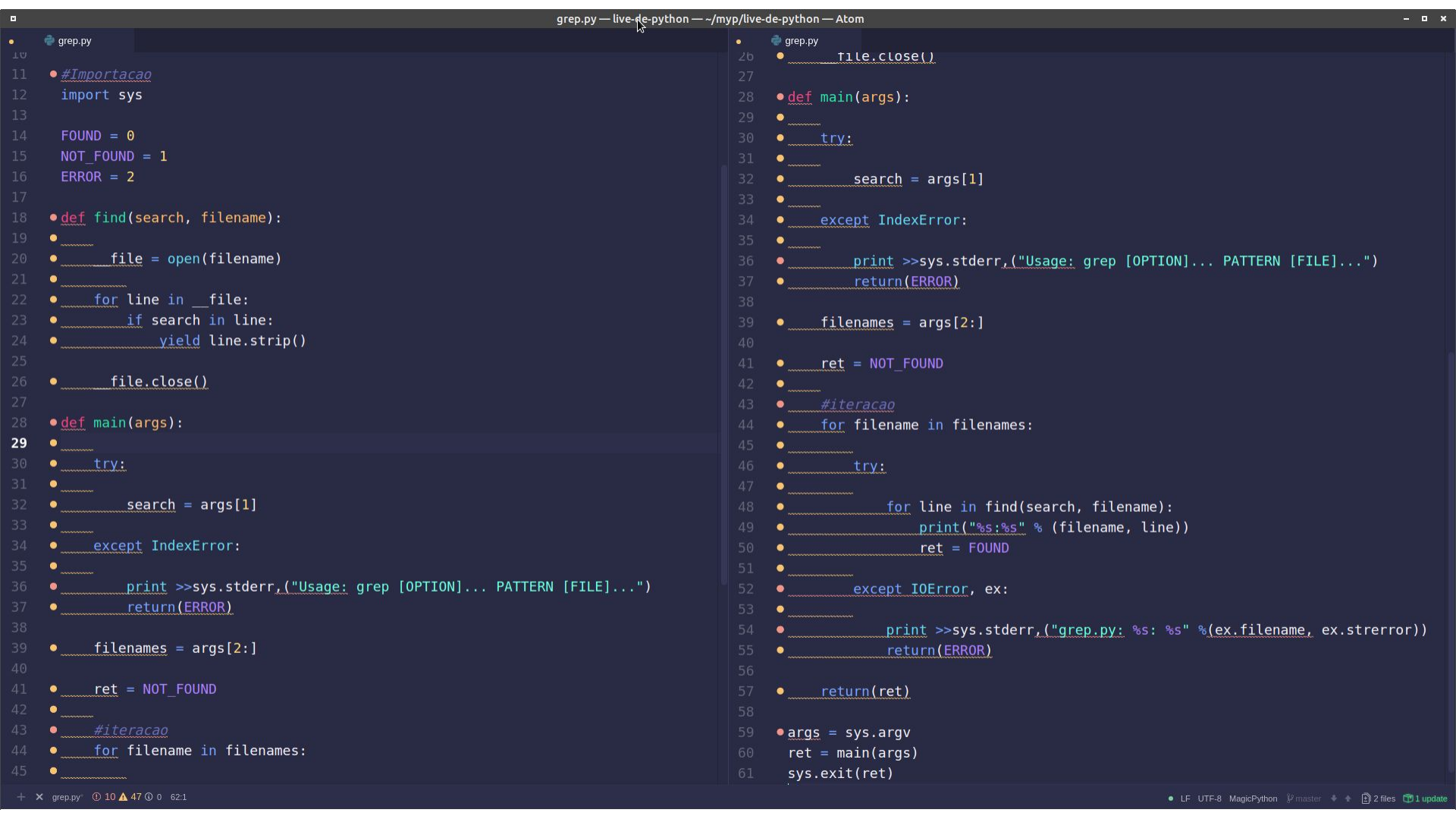
Comunidade!


Automação por comando de
voz via Raspberry Pi

Voluntariado

Voluntariado




grep.py — live-de-python — ~/myp/live-de-python — Atom
grep.py
#Importacao
import sys
FOUND = 0
NOT_FOUND = 1
ERROR = 2
def find(search, filename):
file = open(filename)
for line in __file:
if search in line:
yield line.strip()
file.close()
def main(args):
try:
search = args[1]
except IndexError:
print >>sys.stderr,("Usage: grep [OPTION]... PATTERN [FILE]...")
return(ERROR)
filenames = args[2:]
ret = NOT_FOUND
#iteracao
for filename in filenames:
grep.py
file.close()
def main(args):
try:
search = args[1]
except IndexError:
print >>sys.stderr,("Usage: grep [OPTION]... PATTERN [FILE]...")
return(ERROR)
filenames = args[2:]
ret = NOT_FOUND
#iteracao
for filename in filenames:
try:
for line in find(search, filename):
print("%s:%s" % (filename, line))
ret = FOUND
except IOError, ex:
print >>sys.stderr,("grep.py: %s: %s" %(ex.filename, ex.strerror))
return(ERROR)
return(ret)
args = sys.argv
ret = main(args)
sys.exit(ret)
grep.py* 10 47 0 62:1
LF UTF-8 MagicPython master 2 files 1 update

Luciano Ramalho

- Versão: 3.0
- Alias: Zara
- Bebida preferida: Cerveja

G.E.C.A.

GRUPO DE ESTUDOS CRIPTOGRAFICOS DE AMERICANA

As quatro liberdades

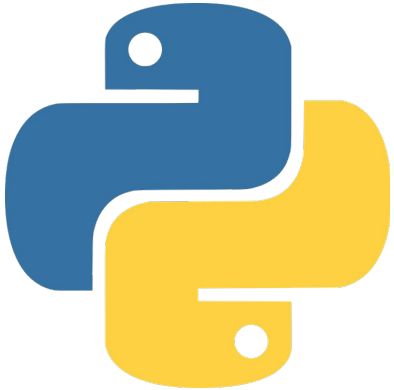
0. Executar o programa como você desejar, para qualquer propósito

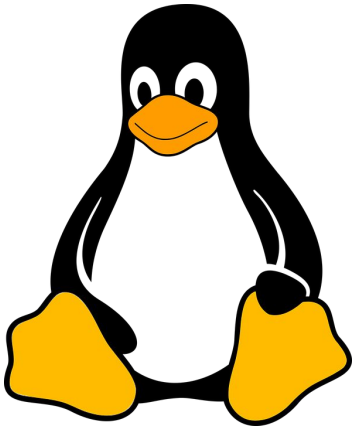
1. Estudar como o programa funciona, e adaptá-lo às suas necessidades

2. Redistribuir cópias de modo que você possa ajudar ao próximo

3. Redistribuir cópias de suas versões modificadas a outros

O que a gente ganha com isso?





Posso contribuir com isso?

- Não sou programador
- Ainda não estou preparado
- Eles vão falar mal do meu código
- Tenho dificuldades com inglês
- Não tenho tempo
- Não sei por onde começar

Posso contribuir com isso?

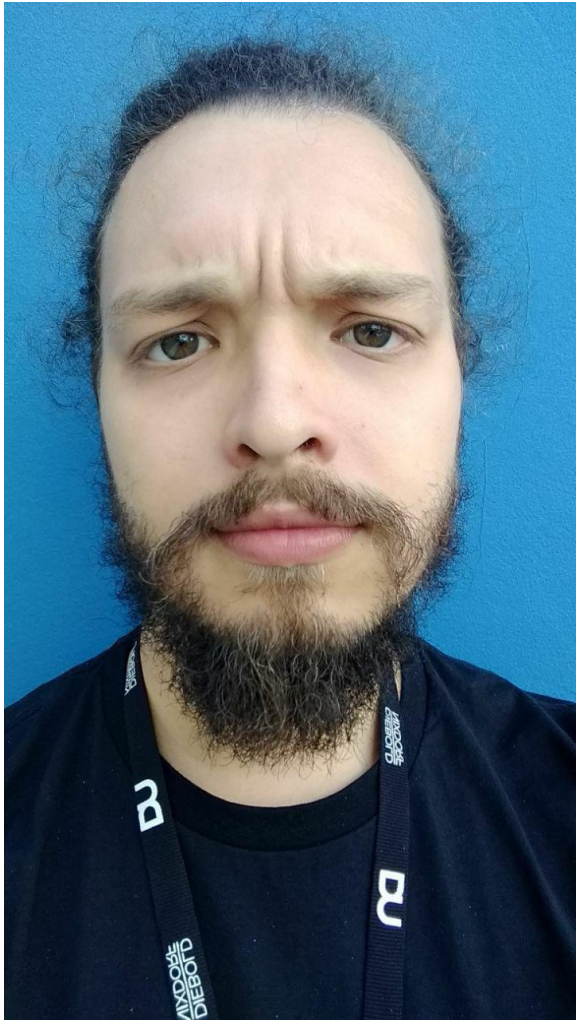
- Não sou programador
- Ainda não estou preparado
- Eles vão falar mal do meu código
- Tenho dificuldades com inglês
- Não tenho tempo
- Não sei por onde começar

Não tenho vontade de contribuir

Eventos de Python

Eventos de Python

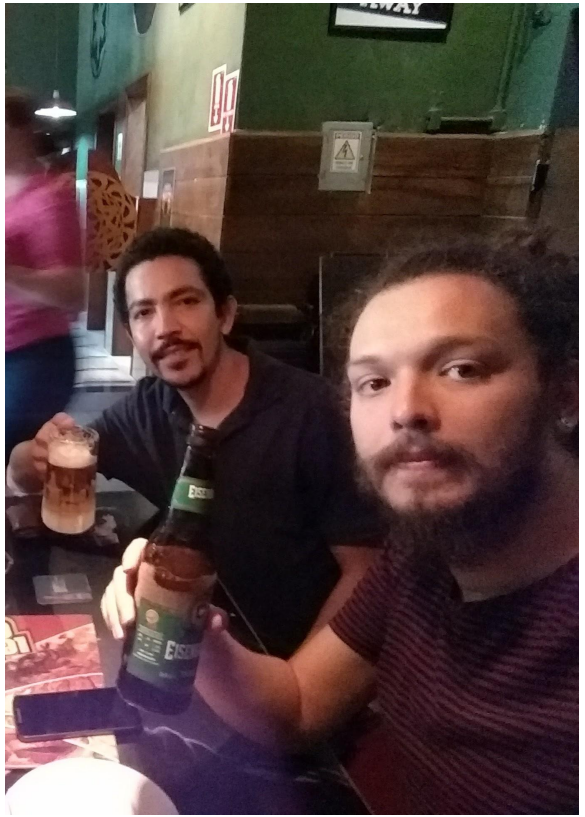
Eventos de Python



- Versão: unstable
- Alias: Du
- Bebida preferida: Cerveja

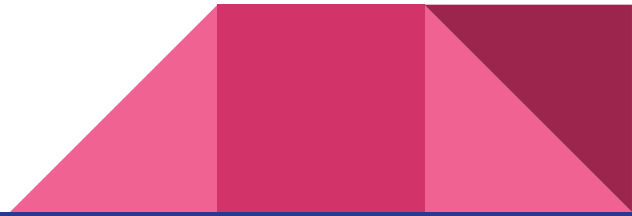

DIEBOLD NIXDORF
AÉROPOSTALE
NYC

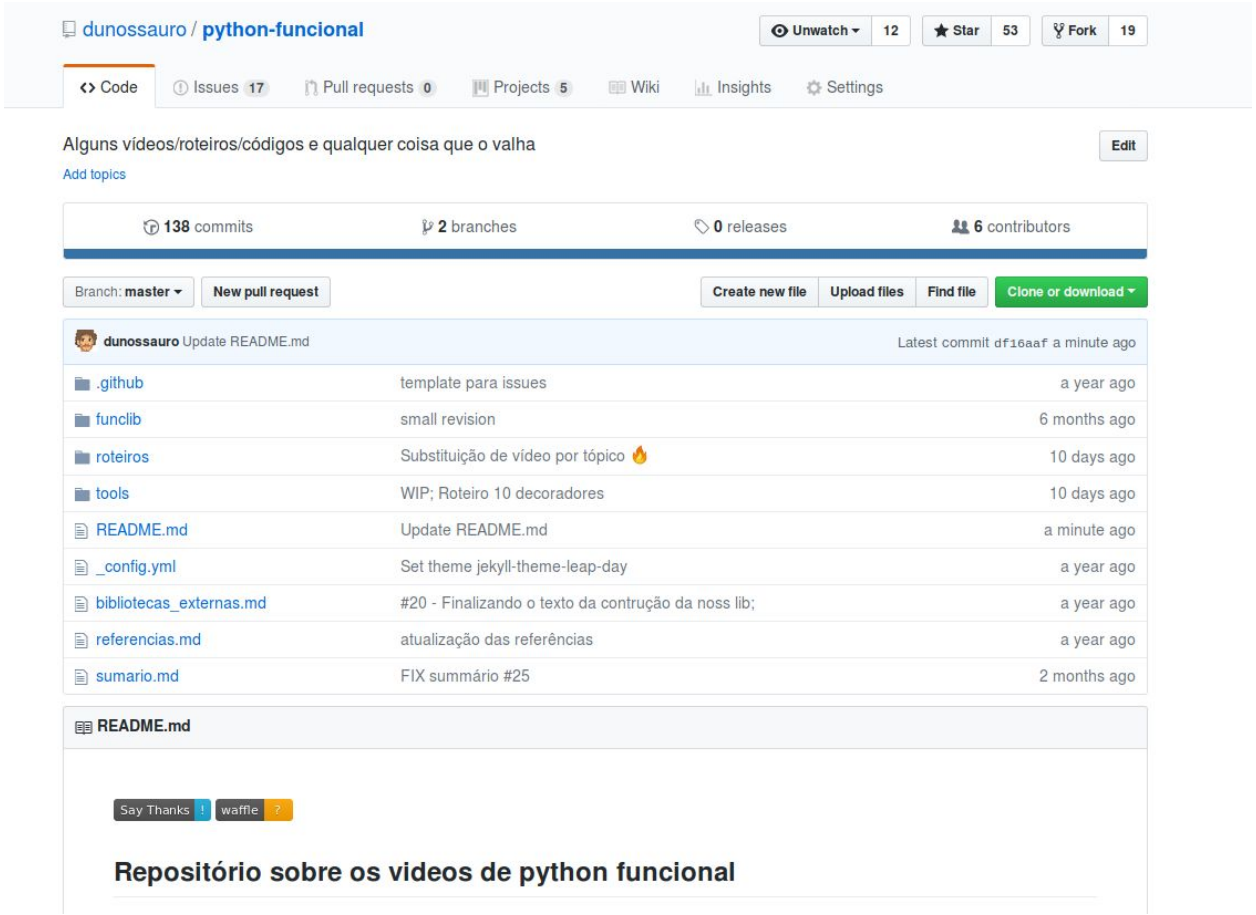
QA



A decisão

- Como vou contribuir se não sou um bom programador?
- Que raios eu posso fazer se não documentações?




dunossauro / python-funcional
Unwatch 12
Star 53
Fork 19
Code
Issues 17
Pull requests 0
Projects 5
Wiki
Insights
Settings
Alguns vídeos/roteiros/códigos e qualquer coisa que o valha
Edit
Add topics
138 commits
2 branches
0 releases
6 contributors
Branch: master
New pull request
Create new file
Upload files
Find file
Clone or download
dunossauro Update README.md
Latest commit df16aaf a minute ago
.github template para issues a year ago
funclib small revision 6 months ago
roteiros Substituição de vídeo por tópico 10 days ago
tools WIP; Roteiro 10 decoradores 10 days ago
README.md Update README.md a minute ago
_config.yml Set theme jekyll-theme-leap-day a year ago
bibliotecas_externas.md #20 - Finalizando o texto da contrução da noss lib; a year ago
referencias.md atualização das referências a year ago
sumario.md FIX summário #25 2 months ago
README.md
Say Thanks !
waffle
Repositório sobre os videos de python funcional

Eduardo Mendes

1.852 inscritos

PERSONALIZAR O CANAL | ESTÚDIO DE CRIAÇÃO

INÍCIO | VÍDEOS | PLAYLISTS | CANAIS | DISCUSSÃO | SOBRE

Uploads | REPRODUZIR TODOS | CLASSIFICAR POR

Live de Python #45 - Rows (Com Turicas)

438 visualizações • 2 dias atrás

Live de Python #44 - Criando um bot para telegram (Com

466 visualizações • 1 semana atrás

Live de Python #43 - Gerenciadores de contexto

313 visualizações • 3 semanas atrás

Live de Python #42 - Testes de mutação

238 visualizações • 1 mês atrás

Live de Python #41 - Cobertura e testrunners

288 visualizações • 1 mês atrás

Live de Python #40 - Análise estática de código (linters)

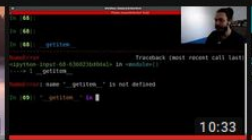
305 visualizações • 1 mês atrás

Live de Python #39 - Guias de estilo e padronização de

407 visualizações • 1 mês atrás

Live de Python #38 - Introdução a qualidade de

472 visualizações • 2 meses atrás



Rapidinha Pythonica #6 - Slice

292 visualizações • 2 meses atrás

Live de Python #37 - Arquivos de configuração

464 visualizações • 2 meses atrás

Live de Python #36 - Argumentos em linha de

315 visualizações • 2 meses atrás

Live de Python #35 - Redes neurais usando a biblioteca

1,2 mil visualizações • 2 meses atrás

Live de Python #34 - Trabalhando com arquivos

501 visualizações • 3 meses atrás

Rapidinha Pythonica #5 - Testando stdout

348 visualizações • 3 meses atrás

Live de Python #33 - Python para arquitetos, designers e

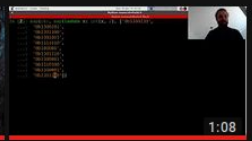
512 visualizações • 3 meses atrás

Qual o melhor sistema operacional para programar?

662 visualizações • 3 meses atrás

Editor de texto ou IDE?

599 visualizações • 3 meses atrás



Rapidinha Pythonica #4 - Especial de natal

267 visualizações • 3 meses atrás

teste.py

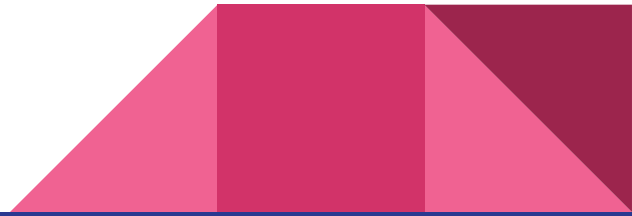
```python
from re import sub  #NOQA
from functools import partial
from collections import Counter
from multiprocessing.dummy import Pool

map_loko = Pool(4)


def pipe(*funcs):
    def wrapper(arg):
        result = arg
        for func in funcs:
            result = func(result)
        return result
    return wrapper


remove_blank_lines = partial(filter, lambda
remove_new_lines = partial(map_loko.map, lar
lower = partial(map, str.lower)
remove_punctuation = partial(map, lambda x:
join = partial(str.join, ' ')
split = partial(str.split, sep=' ')

parse = pipe(open, remove_blank_lines,
             remove_new_lines, lower, remove
```

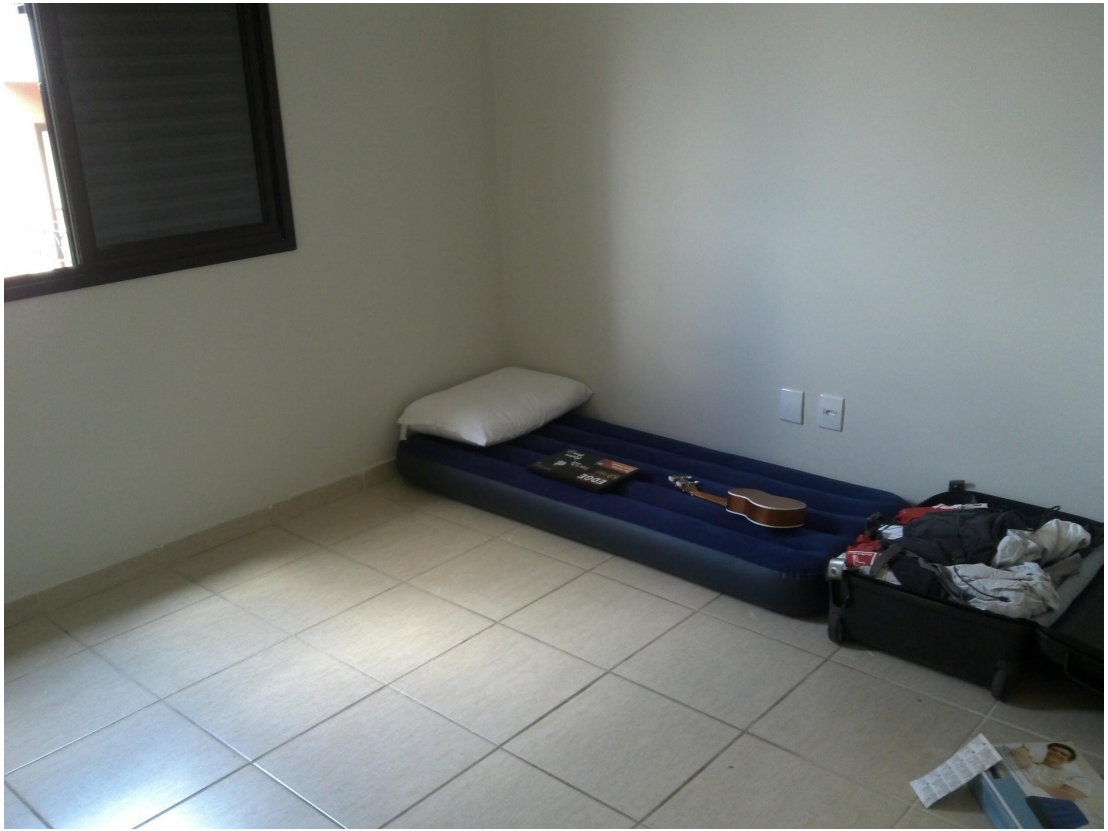
teste.py | tabacaria.txt

```python


def pipe(*funcs):
    def wrapper(arg):
        result = arg
        for func in funcs:
            result = func(result)
        return result
    return wrapper


remove_blank_lines = partial(filter, lambda x: x != '\n')
remove_new_lines = partial(map_loko.map, lambda x: str.strip(x, '\n'))
lower = partial(map, str.lower)
remove_punctuation = partial(map, lambda x: sub(r'[\.,?!\-\();]', '', x))
join = partial(str.join, ' ')
split = partial(str.split, sep=' ')

parse = pipe(open, remove_blank_lines,
             remove_new_lines, lower, remove_punctuation,
             join, split)

count_parse = pipe(parse, Counter)

xpto = count_parse('tabacaria.txt')

```

Um papo sobre educação

- Educação como forma de ativismo
- Educação para transformar





XOXO

Dúvidas/sugestões

agora é a hora de dizer algo