

Ayudantía 2 ()

{

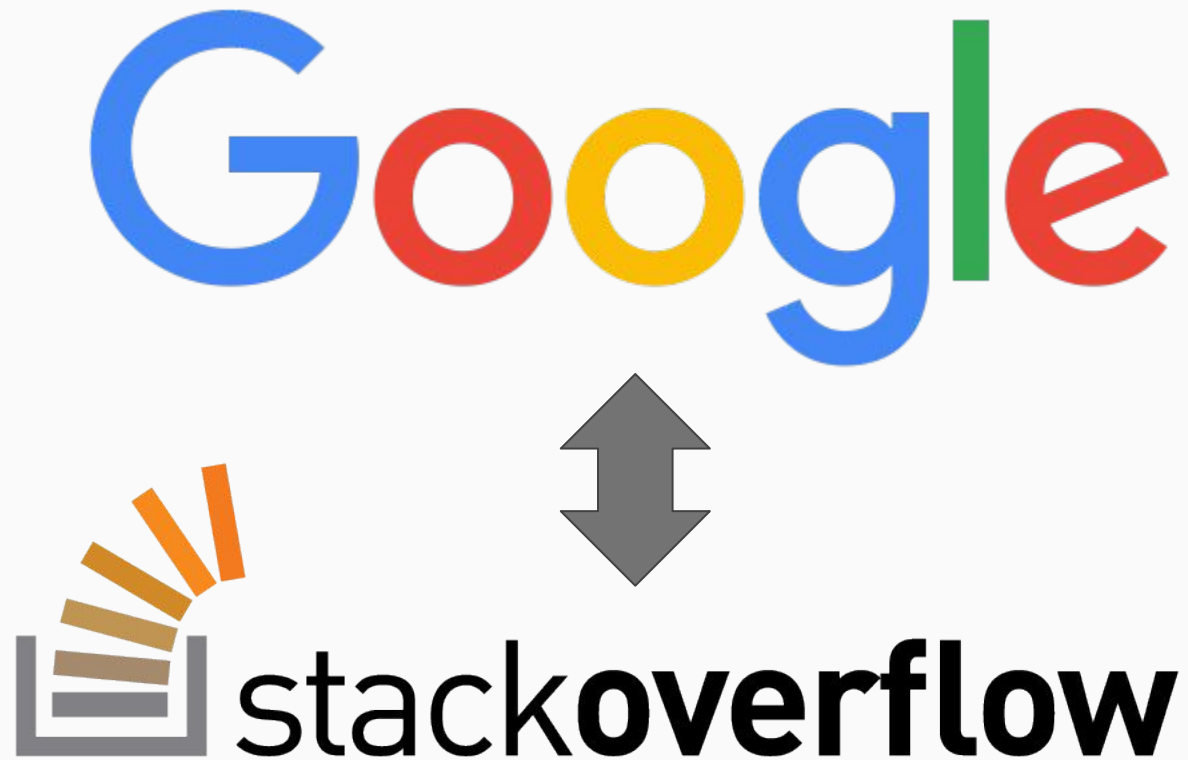
return IIC2233.HowToPass;

}

Agenda

- ¿Cómo buscar soluciones?
- Buenas prácticas
 - Modularización
 - PEP8
- Modelación

¡Sus únicos amigos durante este semestre!



¿Cómo buscar soluciones?

python [versión] [librería] [duda]



¡EN INGLÉS!

¿Cómo imprimir una cola con Python?



Python 3.4 collections print queue



¿Cómo buscar soluciones?

python [versión] [error]



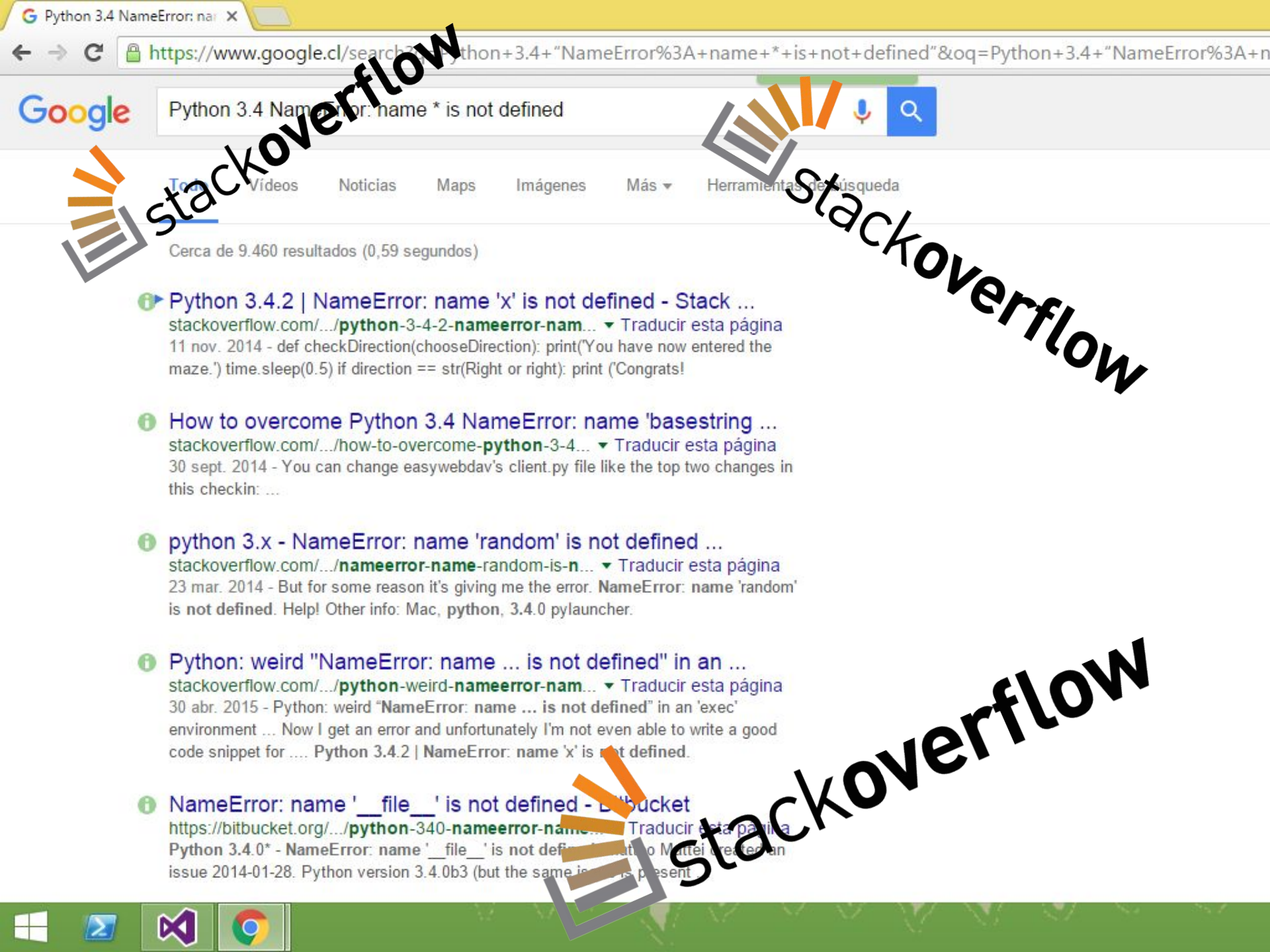
¡EN INGLÉS!

`NameError: name "MiVariable" is not defined`



`NameError: name * is not defined`





Python 3.4 NameError: name * is not defined

Todos Vídeos Noticias Maps Imágenes Más Herramientas de búsqueda

Cerca de 9.460 resultados (0,59 segundos)

- Python 3.4.2 | NameError: name 'x' is not defined - Stack ...**
stackoverflow.com/.../python-3-4-2-nameerror-nam... Traducir esta página
11 nov. 2014 - def checkDirection(chooseDirection): print("You have now entered the maze.") time.sleep(0.5) if direction == str(Right or right): print ("Congrats!
- How to overcome Python 3.4 NameError: name 'basestring ...**
stackoverflow.com/.../how-to-overcome-python-3-4... Traducir esta página
30 sept. 2014 - You can change easywebdav's client.py file like the top two changes in this checkin: ...
- python 3.x - NameError: name 'random' is not defined ...**
stackoverflow.com/.../nameerror-name-random-is-n... Traducir esta página
23 mar. 2014 - But for some reason it's giving me the error. NameError: name 'random' is not defined. Help! Other info: Mac, python, 3.4.0 pylauncher.
- Python: weird "NameError: name ... is not defined" in an ...**
stackoverflow.com/.../python-weird-nameerror-nam... Traducir esta página
30 abr. 2015 - Python: weird "NameError: name ... is not defined" in an 'exec' environment ... Now I get an error and unfortunately I'm not even able to write a good code snippet for Python 3.4.2 | NameError: name 'x' is not defined.
- NameError: name '__file__' is not defined - Bitbucket**
https://bitbucket.org/.../python-340-nameerror-name... Traducir esta página
Python 3.4.0* - NameError: name '__file__' is not defined. Matteo Creteran
issue 2014-01-28. Python version 3.4.0b3 (but the same is present

Buenas Prácticas

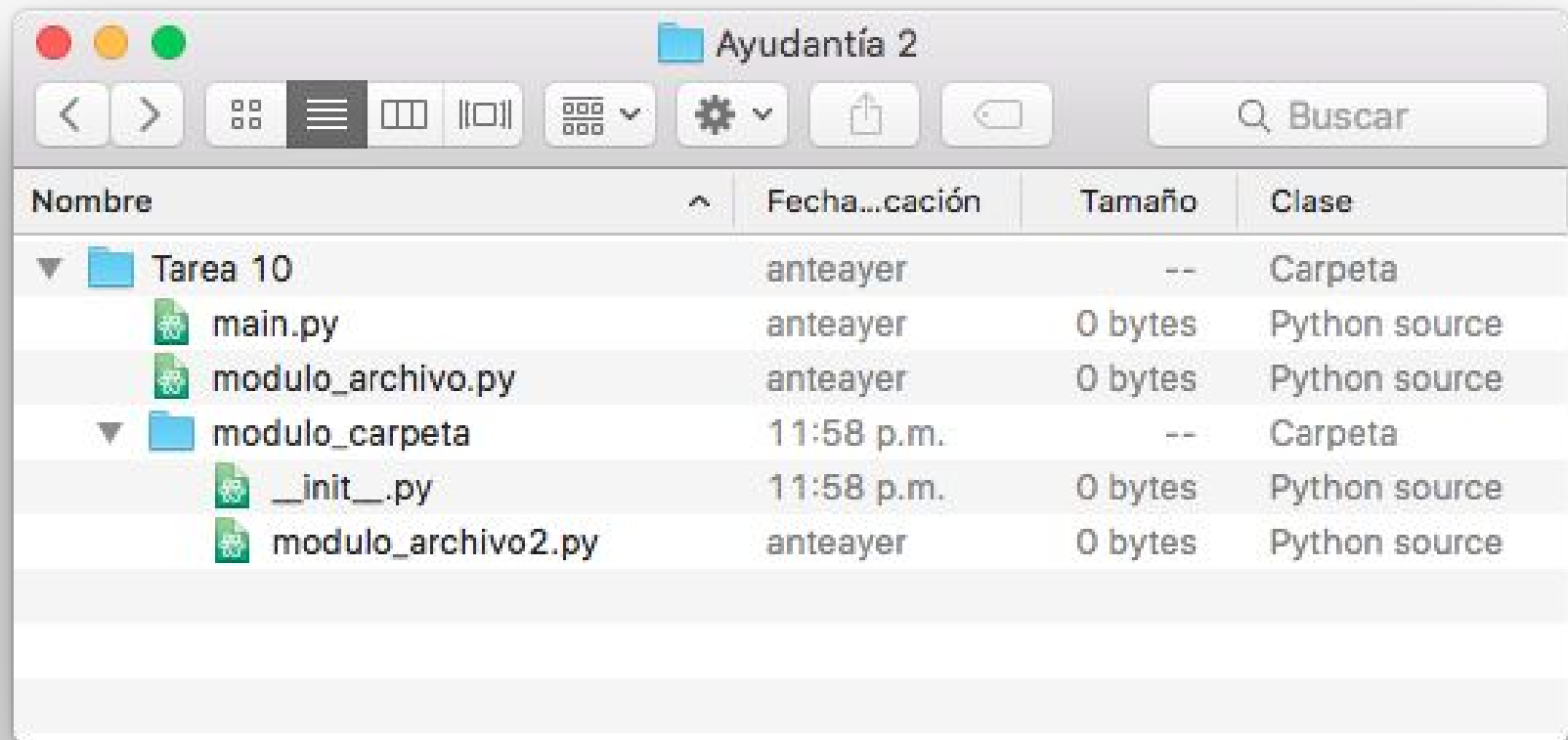
1 módulo = 1 archivo Python

Modularización

Modularización: ¿Por qué?

- Cuando un programa crece, se hace inviable mantenerlo en un solo archivo:
 - El mantenimiento es difícil
 - El trabajo en equipo es difícil
 - Es desordenado
- Como es un archivo de python normal puede tener:
 - Variables
 - Métodos
 - Clases

Modularización



Cómo usar módulos

- Importándolo entero:

```
import modulo_archivo
```

```
if __name__ == "__main__":  
    variable_tipica = modulo_archivo.VALOR_FIJO  
    objeto_tipico = modulo_archivo.Clase()  
    modulo_archivo.funcion()
```

Cómo usar módulos

- Importándolo entero con pseudónimo:

```
import modulo_archivo as ma
```

```
if __name__ == "__main__":  
    variable_tipica = ma.VALOR_FIJO  
  
    objeto_tipico = ma.Clase()  
  
    ma.funcion()
```

Cómo usar módulos

- Importando lo necesario

```
from modulo_archivo import (VALOR_FIJO, Clase, function)
```

```
if __name__ == "__main__":  
    variable_tipica = VALOR_FIJO  
  
    objeto_tipico = Clase()  
  
    function()
```

Cómo usar módulos

- Importando un package
- Un package es una carpeta que tiene tener el archivo vacío “__init__.py”

```
import modulo_carpeta as mc
```

```
if __name__ == "__main__":
```

```
    variable_tipica = mc.modulo_archivo2.VALOR_FIJO
```

```
    objeto_tipico = mc.modulo_archivo2.Clase()
```

```
    mc.modulo_archivo2.funcion()
```

Cómo usar módulos

- Cuando se importa un módulo se ejecuta todo el código en él
- Para evitar que se ejecute código de un módulo al ser importado se utiliza el siguiente if:

```
if __name__ == "__main__":  
    # Mucho código escrito
```

Cómo NO usar módulos

- Importando todo sin referencia al módulo

```
from modulo_archivo import *
```

```
if __name__ == "__main__":  
    variable_tipica = VALOR_FIJO  
    objeto_tipico = Clase()  
    funcion()
```



Cómo NO usar módulos

- Evita usar módulos que se llamen igual a los que vienen incluidos en python
- Cómo busca los módulos python:
 - Módulo integrado
 - Módulo en la misma carpeta
 - Módulo en el directorio de instalación

PEP8

- Python Enhancement Proposal 8 es la guía de estilo de Python
- Se usa para hacer más legible y consistente el código
- <https://www.python.org/dev/peps/pep-0008/>

PEP8

- Importaciones al comienzo del módulo
- Nombres de variables descriptivos
- Espacios entre líneas
 - 2 líneas después de los imports
 - 2 líneas alrededor de las clases y funciones
 - 1 línea entre métodos de clase
 - 1 espacio después de “,” y a cada lado de los operadores
- Líneas de máximo 80 caracteres (incluyendo espacios)
- NO mezclar tabs y espacios

CamelCase y snake_case

```
class ClaseMuyBakanosa:  
  
    def __init__(self, hola):  
        self.best_variable_ever = hola  
  
    def metodo_muy_weno(self):  
        return 1 + 1 == 2
```

PEP8: En resumen

Siempre recuerda que el código se lee más veces de lo que se escribe y que otro lo va a leer.

La mejor manera de escribir buen código es:

“Piensa que la persona que leerá tu código es un psicópata asesino que sabe donde vives.”

Actividad: Arreglar un desastre

Modelación: Actividad 2