

Herramienta para lograr ser un experto en Python



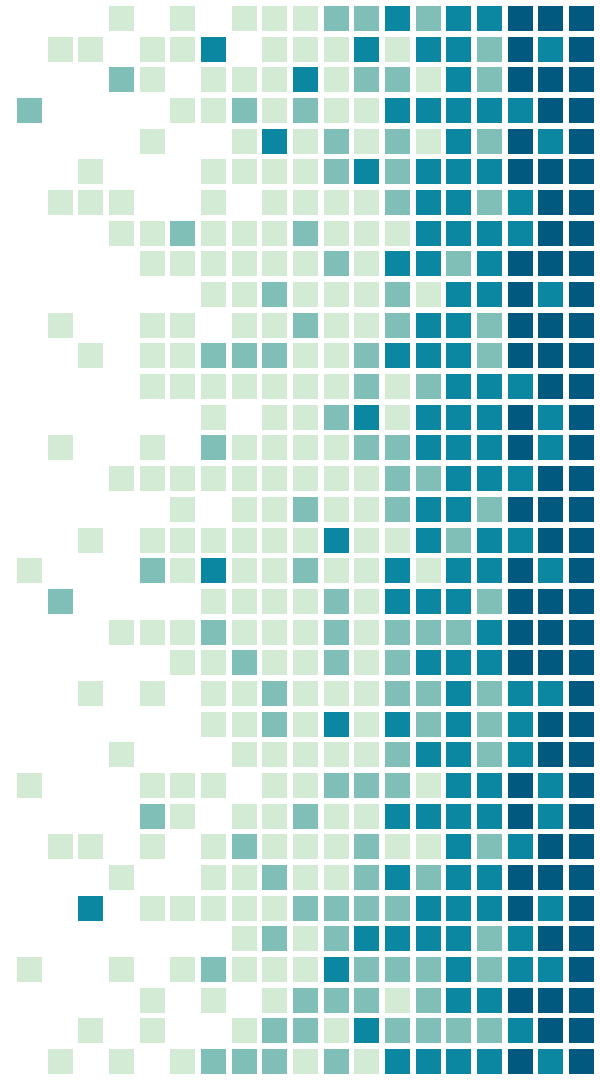
Universidad
Rey Juan Carlos

José Javier Merchante
Gregorio Robles

1.

INTRODUCCIÓN

¿Qué es el código Pythonico?



Introducción

Código Python

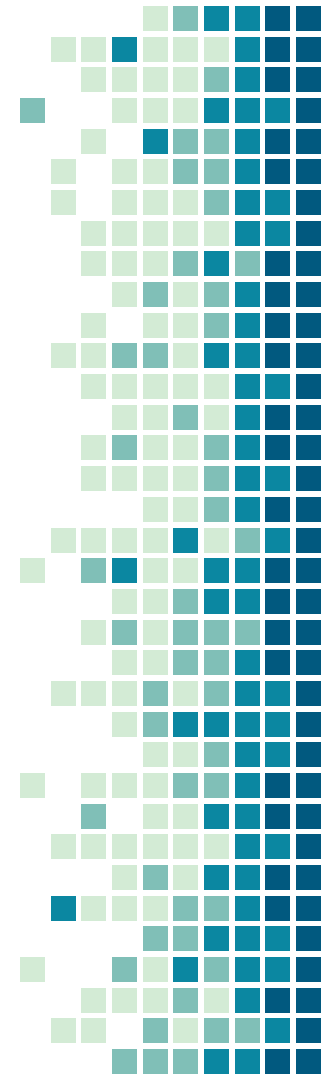
```
colors = ['red', 'blue', 'yellow', 'green', 'yellow', 'blue']  
  
d = {}  
for i in range(len(colors)):  
    color = colors[i]  
    if color not in d:  
        d[color] = 0  
    d[color] += 1
```



Introducción

Código Pythonico

```
d = defaultdict(int)
for color in colors:
    d[color] += 1
```

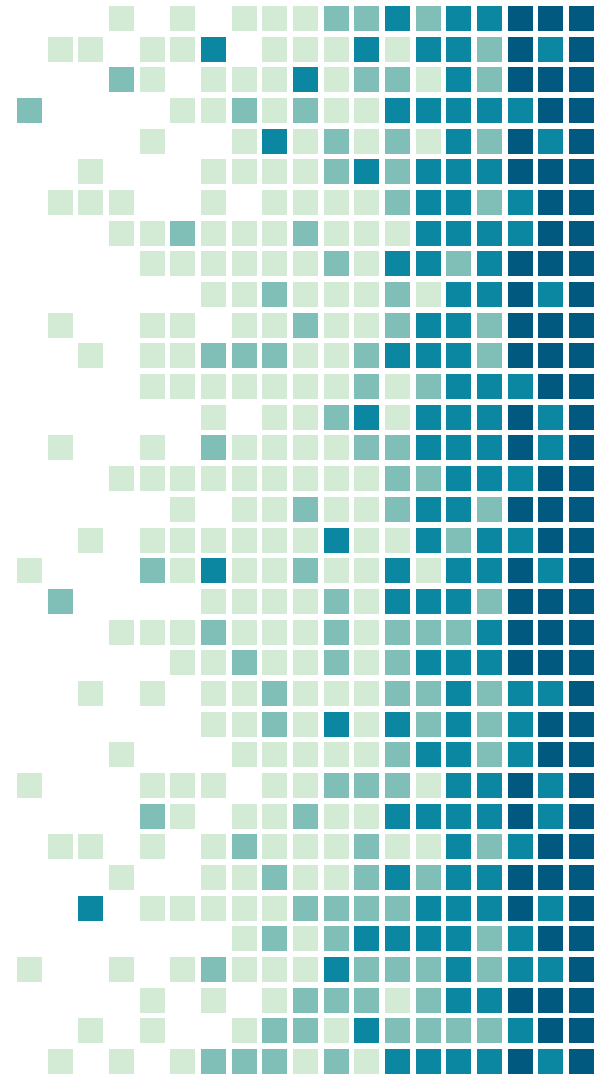


“*Existen muchas formas de programar una determinada tarea, pero SIEMPRE existe una que es preferible*”

2.

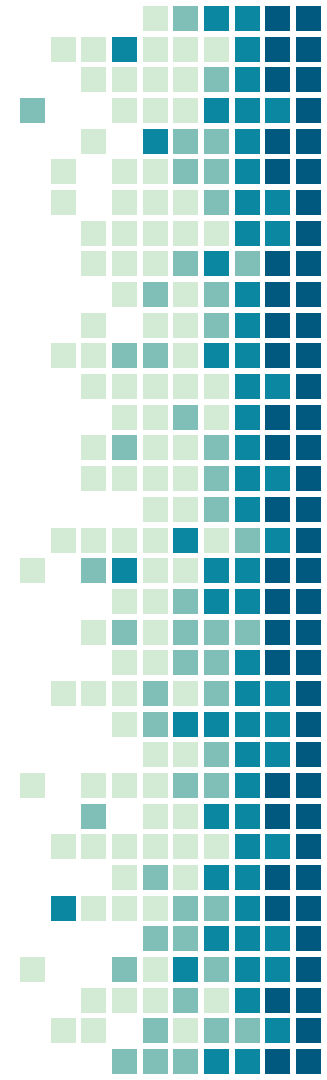
En busca del código

Pythonico



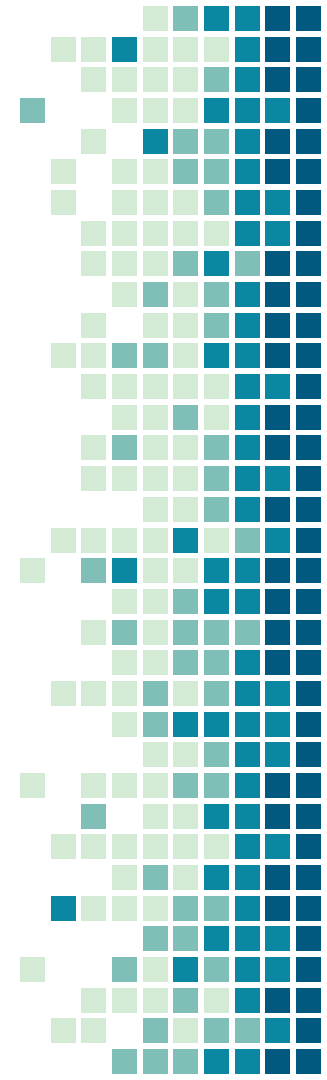
Objetivos principales

- Evaluar el nivel de “Pythonicidad” de un programador a partir de su código
- Ofrecer alternativas más “Pythónicas” de determinadas partes del código
- Incluir recursos para mejorar el aprendizaje

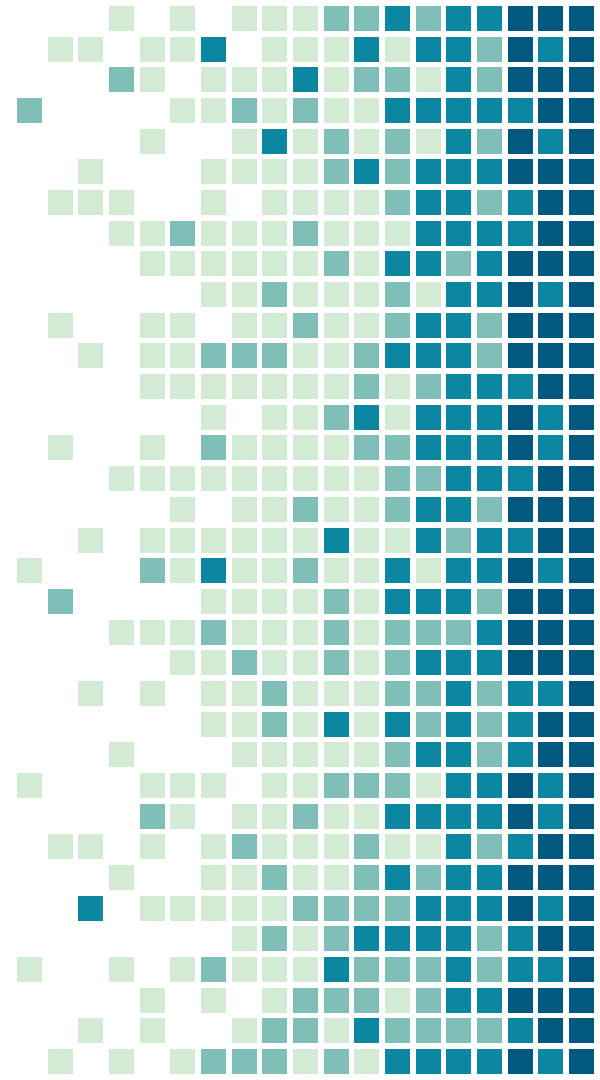


Estado actual

- Analiza repositorios procedentes de GitHub identificando *idioms*
- Analiza usuarios de GitHub para mostrarles una sencilla evaluación
- Muestra los *idioms* encontrados y ofrece sugerencias de otros con referencias externas



3. Breve demostración de su funcionamiento



Análisis de un repositorio



The screenshot displays a web interface for repository analysis. A modal window titled "Symmetrical Expressions" is open, showing a table with the following data:

File	Lines	Author	Link
pythons/clients/guestbook.py	1	@seanbeane@guest.com	View
pythons/clients/web.py	2408	@seanbeane@guest.com	View
pythons/clients/web.py	470	@seanbeane@guest.com	View
pythons/clients/web.py	731	@seanbeane@guest.com	View
pythons/clients/web.py	748	@seanbeane@guest.com	View
pythons/clients/web.py	791	@seanbeane@guest.com	View
pythons/clients/web.py	826	@seanbeane@guest.com	View
pythons/clients/web.py	877	@seanbeane@guest.com	View
pythons/clients/web.py	896	@seanbeane@guest.com	View
pythons/clients/web.py	922	@seanbeane@guest.com	View
pythons/clients/web.py	1444	@seanbeane@guest.com	View
identical_repository.py	17	@seanbeane@guest.com	View

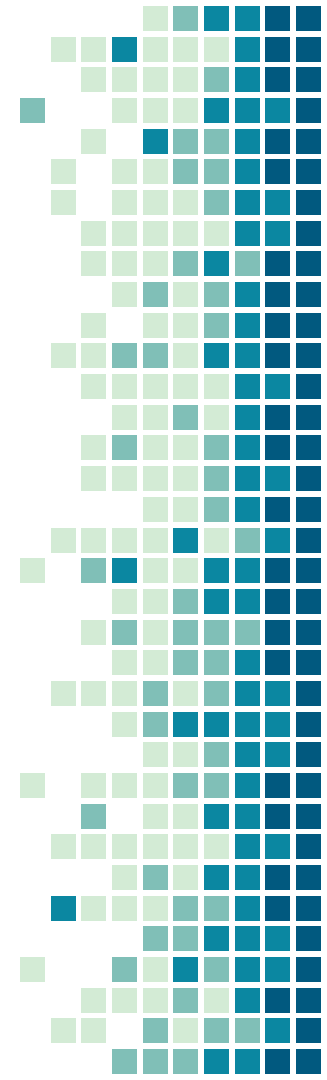
The background interface shows a sidebar with categories like "LINE CODE", "PRO FILE", and "FEATURE". The main content area displays several code snippets with titles such as "Assert", "Finally block", "deljpe collection", and "defaultdict collection".

Análisis de un usuario



The screenshot shows a web interface with a red header bar. The main content area is titled "Select your python repositories" and contains a table with the following columns: "Selected", "Name", "Forked", "Main language", and "Link". The table lists 15 repositories, with the first 14 having a radio button in the "Selected" column and the last one having a checked radio button. A mouse cursor is visible over the "Name" column of the 14th row.

Selected	Name	Forked	Main language	Link
<input type="radio"/>	python/python	✖	python	View
<input type="radio"/>	python/awesome-python		url	View
<input type="radio"/>	python/awesome-python-3000k-ideas	✖	Text	View
<input type="radio"/>	python/awesome-python-3000k-ideas		Text	View
<input type="radio"/>	python/awesome-python-3000k-ideas		url	View
<input type="radio"/>	python/awesome-python-3000k-ideas		Text	View
<input type="radio"/>	python/awesome-python-3000k-ideas		Text	View
<input type="radio"/>	python/awesome-python-3000k-ideas	✖	Text	View
<input type="radio"/>	python/awesome-python-3000k-ideas	✖	url	View
<input type="radio"/>	python/awesome-python-3000k-ideas	✖	url	View
<input type="radio"/>	python/awesome-python-3000k-ideas	✖	url	View
<input checked="" type="radio"/>	python/awesome-python-3000k-ideas	✖	Python	View
<input type="radio"/>	python/awesome-python-3000k-ideas	✖	Python	View
<input type="radio"/>	python/awesome-python-3000k-ideas	✖	python	View
<input type="radio"/>	python/awesome-python-3000k-ideas	✖	python	View
<input checked="" type="radio"/>	python/awesome-python-3000k-ideas	✖	Python	View

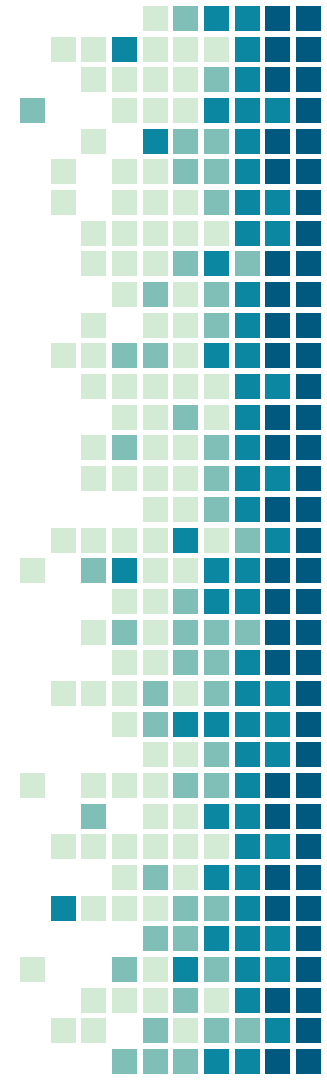


4. Metodología utilizada para el desarrollo de la herramienta



Listar los “idioms” de Python

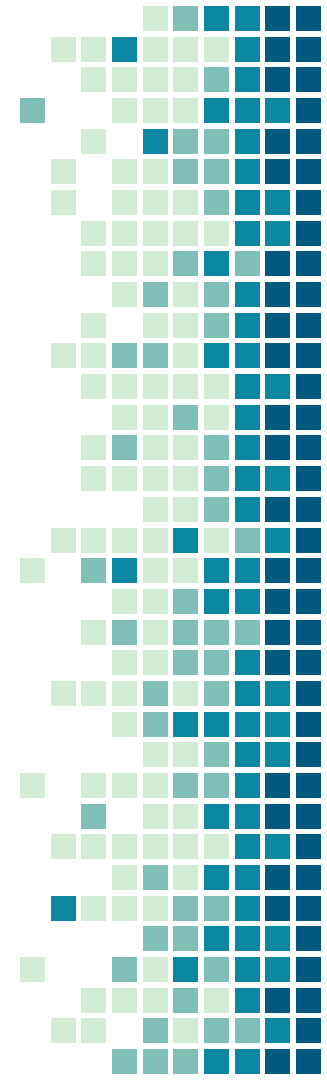
- List comprehensions
- Generator expressions
- Magic methods
- Decorators
- Lambda functions
- Structures from collections
- ...



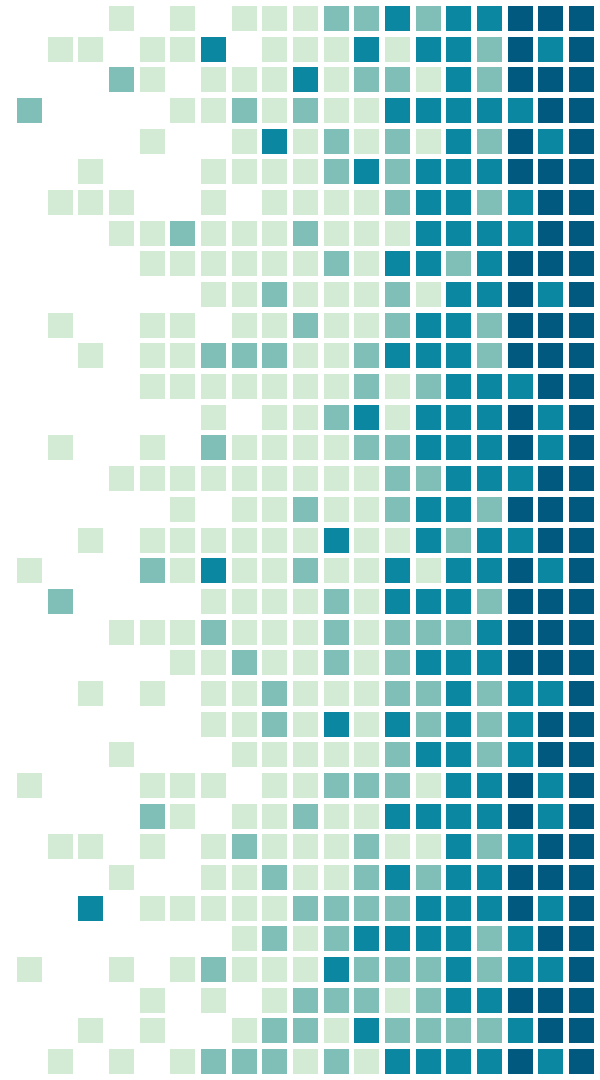
Implementación del analizador

```
adult_list = [person
               for person in people
               if is_adult(person)]
```

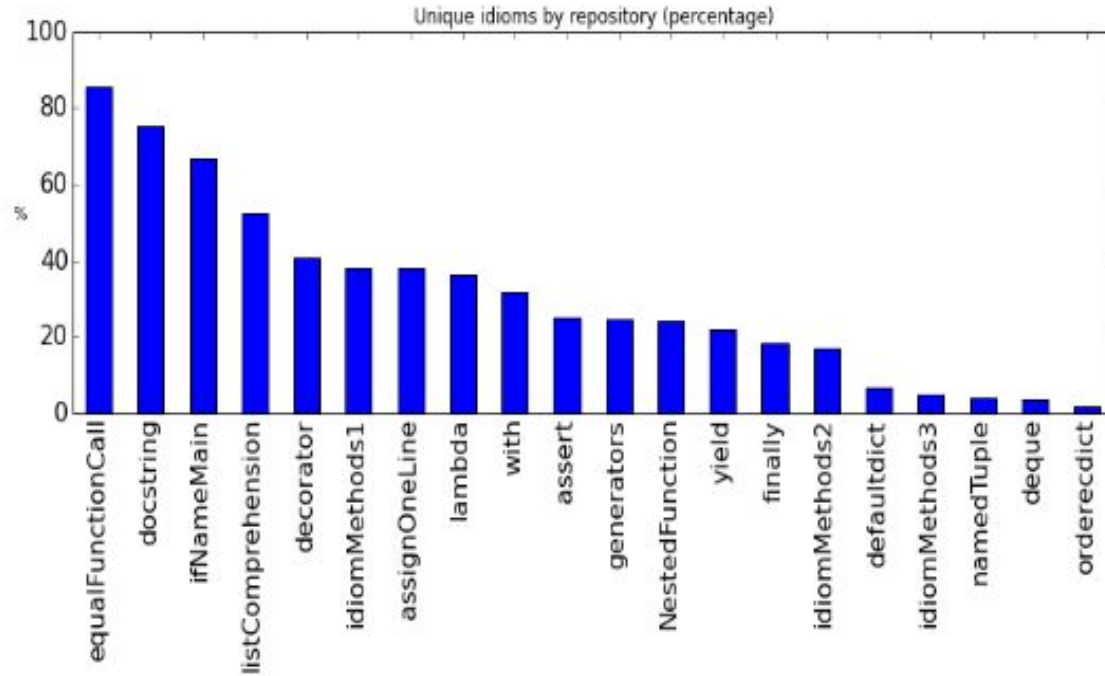
```
(Token.Name, u'adult_list')
(Token.Text, u' ')
(Token.Operator, u'=' )
(Token.Text, u' ')
(Token.Punctuation, u'[')
(Token.Name, u'person')
(Token.Text, u' ')
(Token.Keyword, u'for')
(Token.Text, u' ')
(Token.Name, u'person')
...
```



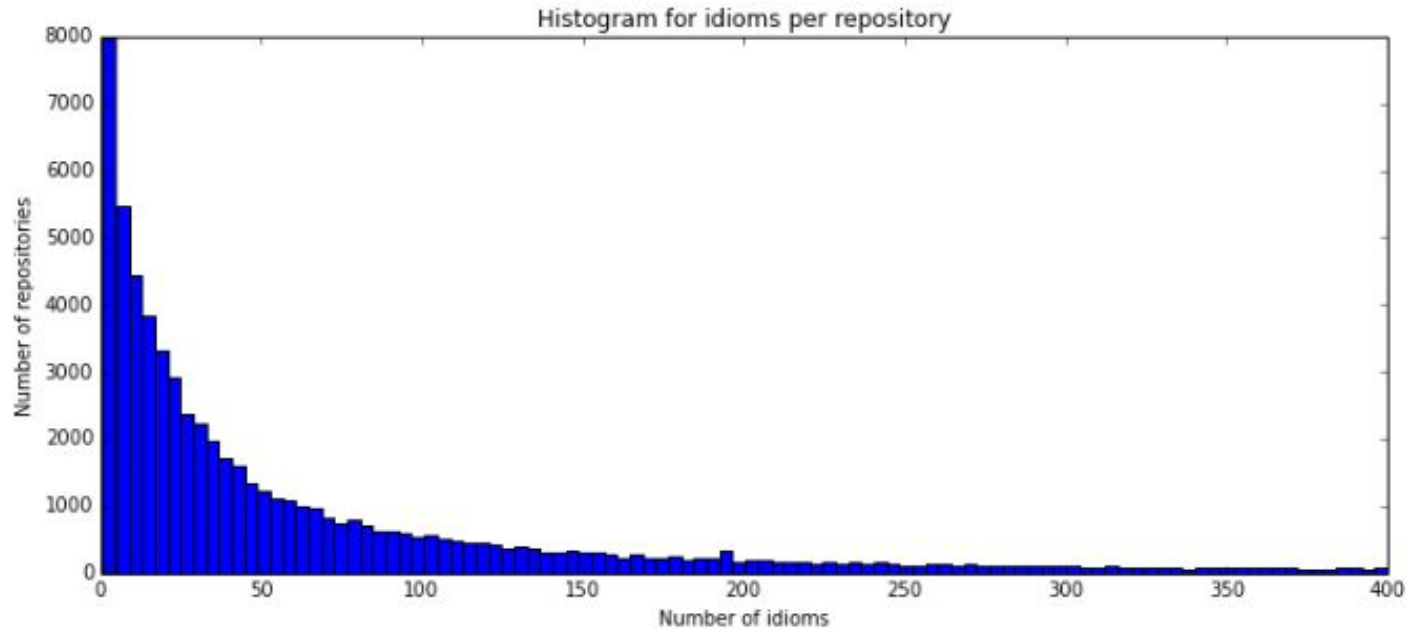
5. Análisis sobre repositorios de GitHub



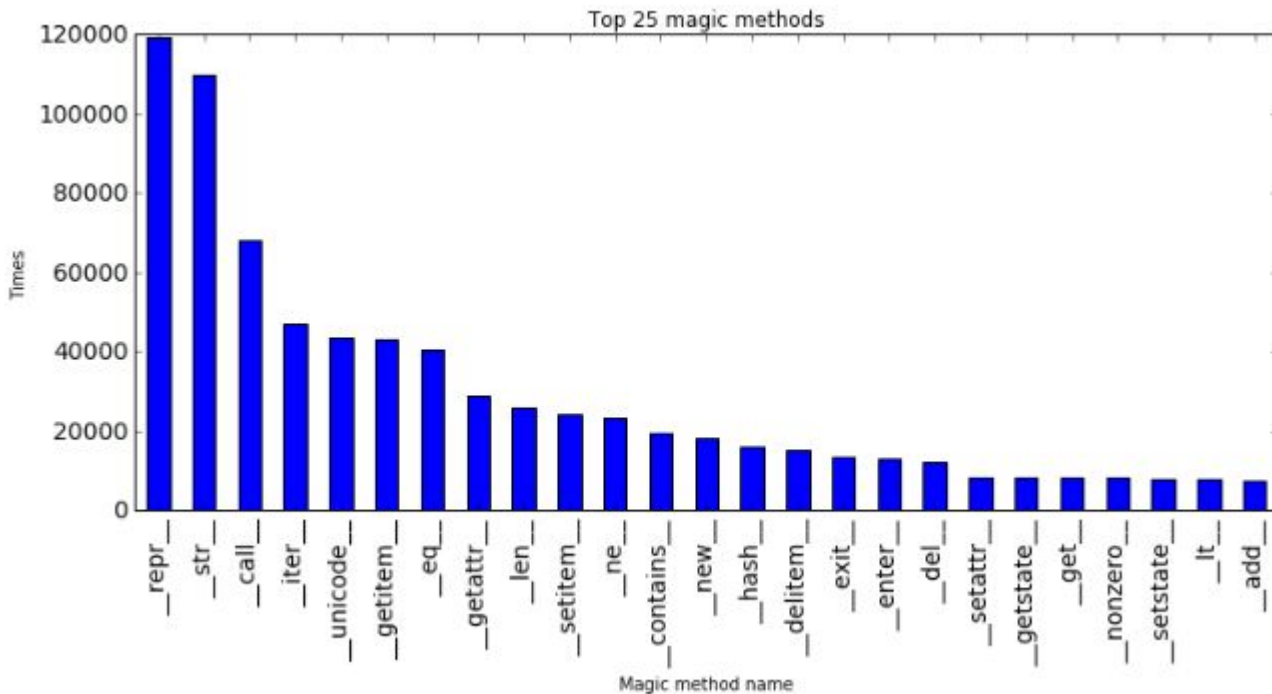
¿Cuales son los *idioms* más utilizados?



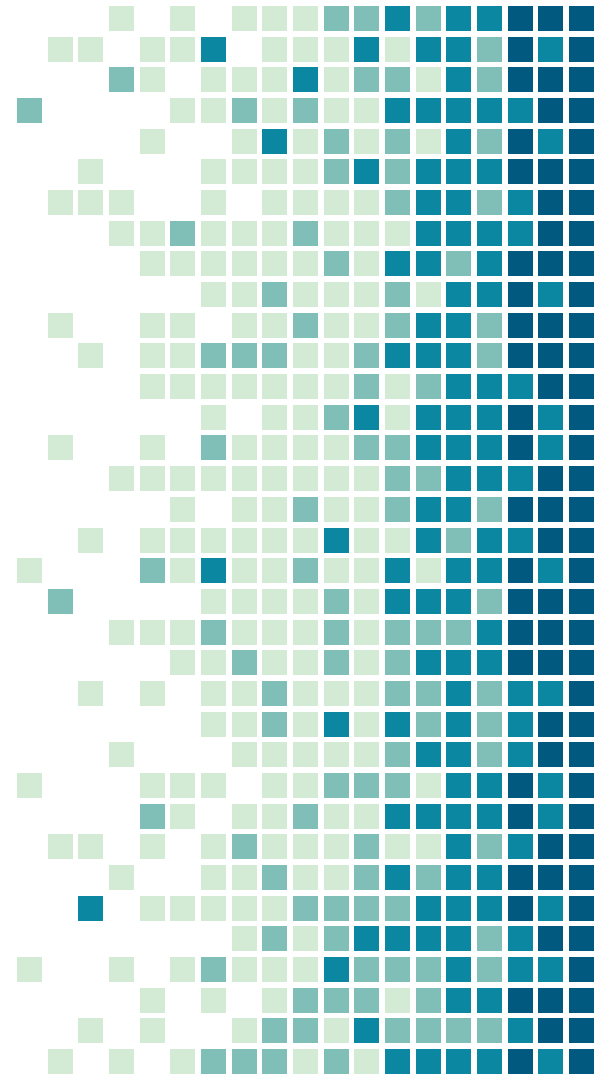
¿Cuántos *idioms* hay por repositorio?



¿Qué *magic methods* son los más usados?

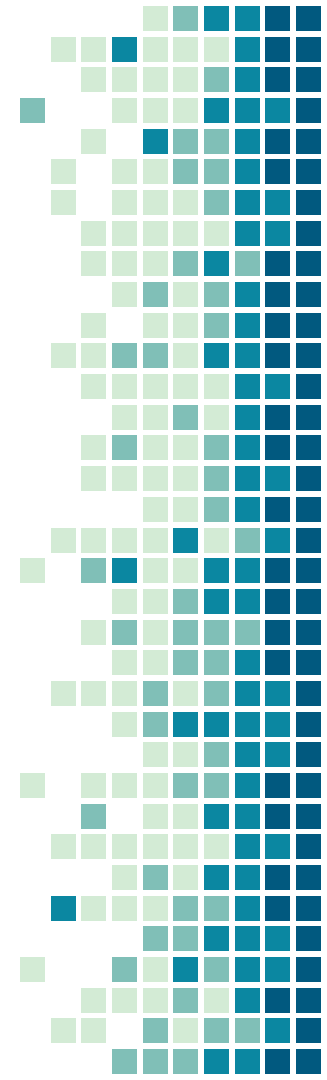


6. Conclusiones



Conclusiones

- El código “pythonico” ayuda al mantenimiento del mismo
- La mayor parte de los repositorios contiene muy pocos *idioms*
- Python es un lenguaje rico en *idioms*
- Es necesario mejorar la “pythonicidad” del código

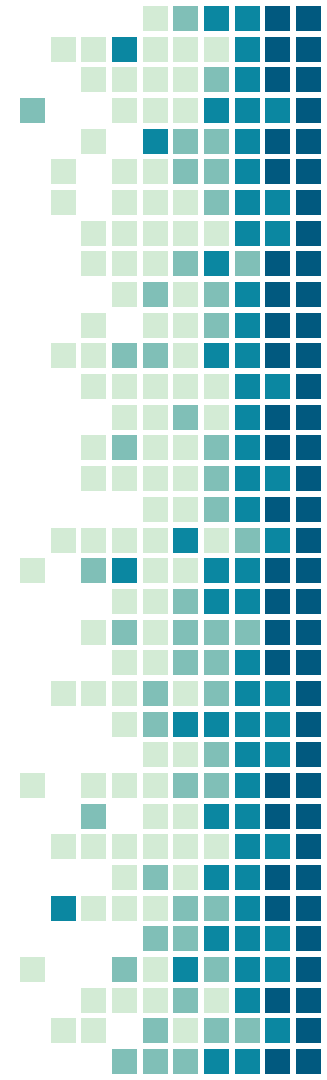




Voluntarios para una entrevista

Necesitamos mejorar la precisión del programa para ser de más ayuda

Comprobar que el uso de “idioms” es la manera adecuada de conocer el nivel en Python



“ Always code as if the guy
who ends up maintaining
your code will be a violent
psychopath who knows
where you live”

-- John Woods

¡GRACIAS POR SU ATENCIÓN!

¿Preguntas?

Pueden contactar con nosotros en:

- jj.merchante@gmail.com
- grex@gsyc.urjc.es

Web:

- pythonic.libresoft.info

