



# 10 Pines

Diseño a la Gorra  
Temporada 02 - Episodio 04



Hernán Wilkinson



hernan.wilkinson@10pines.com



@HernanWilkinson



<https://alagorra.10pines.com>

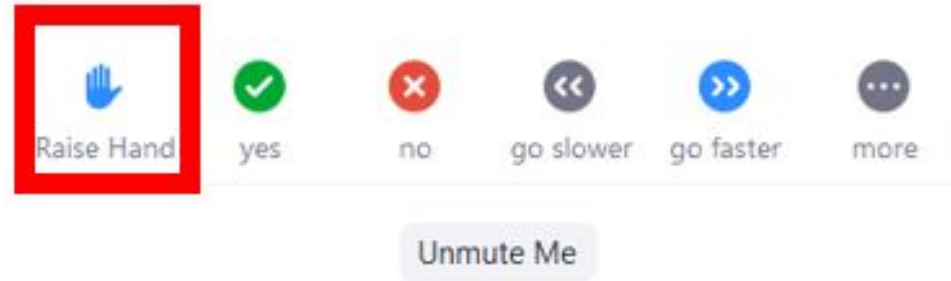


#diseño-a-la-gorra en Academia-10Pines  
(si no recibieron invitación avisen!)





Estar muteados a menos que sea necesario



No voy a poder leer el chat



La comunicación visual es importante. Usarla a discreción





# Diseño ¡a la gorra!

¡Bienvenidos a este exitoso Webinar que se llama "*Diseño a la Gorra*"!

Durante **el año 2020** hicimos **17 ediciones** donde exploramos qué es diseñar, cómo hacerlo mejor, cómo desarrollar un sistema haciendo TDD, etc.

En **el año 2022** tendremos nuevos episodios en los que tocaremos todo tipo de tema, desde diseño de lenguajes de programación y su impacto en nuestro trabajo hasta temas de diseño clásico. A todo esto le sumaremos sorteos a los cursos abiertos que estaremos dando en 10Pines.

Te esperamos todos los **Miércoles a las 19 Hrs GMT-3** a partir del Miércoles 2 de Marzo de 2022. Para poder participar [REGISTRARTE ACA](#).

Todo el código y presentaciones estarán disponibles para que lo puedan usar y consultar en cualquier momento [acá](#).

👉 Más abajo están los links a todos los videos 👉

## Pago voluntario



mercado  
pago

\$1.000 - Muy buena 🍺

Pagar

\$1.500 - Dos combos 🍔

Pagar

\$2.000 - Dos buenos 🍷

Pagar



Donate



[¿Querés abonar un monto diferente o en otra plataforma?](#)

<https://alagorra.10pines.com>

Online

# Diseño Avanzado de Software con Objetos I

📅 Empieza el **04/04**

📅 Del 04/04 al 18/04. Días: 3 días a la semana durante 2 semanas - Lunes, Miércoles y Viernes - 9:00 - 13:00 GMT-3.

💰 **AR\$ 64.800-** (IVA incluido) **USD 495-** (impuestos incluidos)

Hacer una consulta

Inscribirme ahora **10% Desc.**

Dictado por:  
**Máximo Prieto**



~~AR\$ 72.000-~~


**AR\$ 64.800-**

(IVA incluido)

~~USD 550-~~

**USD 495-**

(impuestos incluidos)

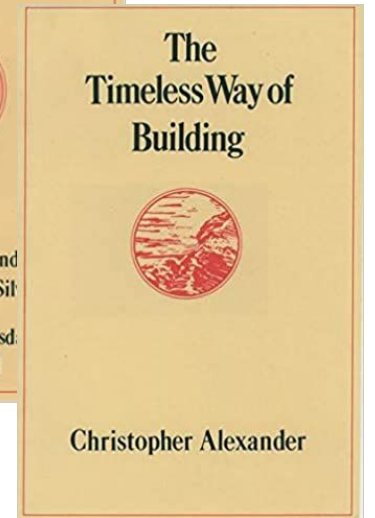
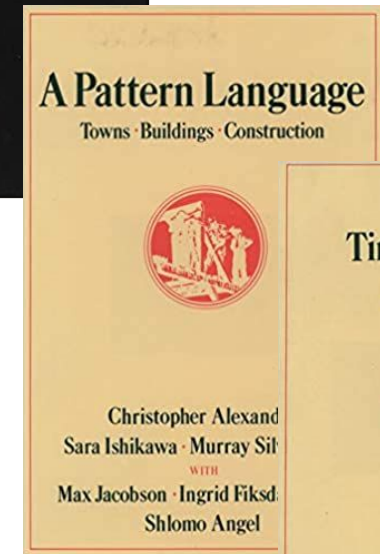
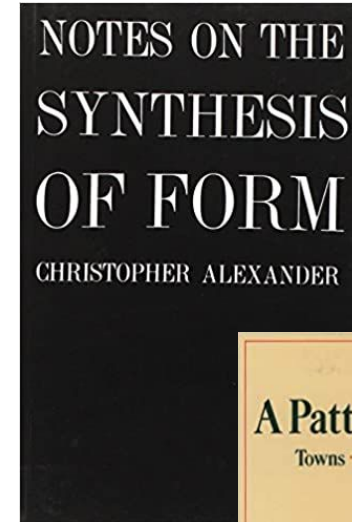
 **Early Bird** Inscripción Temprana hasta 21/03

<http://academia.10pines.com>





# RIP - Christopher Alexander



# Errores históricos: Default Parameters o Un poco de diversión





Pablo Nussembaum 

@baunax



Mirá [@HernanWilkinson](#)! gran ejemplo para tu charla de los errores de los lenguajes de programación!

[Translate Tweet](#)



(Ollie  ) @OliverCaldwell · Mar 17

I'm no Python expert, but is this a super common bug? How do people use defaults without this happening?

```
def x (y=[]):  
    return y
```

```
x() => []
```

```
x().append(10)
```

```
x() => [10]
```

[Show this thread](#)





¿¿Es Verdad??  
Ejemplo





**phil jones (he/him - ele)** @interstar · Mar 17



Replying to [@OliverCaldwell](#)

This isn't considered a "bug". It's a feature. (I.e. it's documented and fairly well known.)

People use it to get some kind of cheap "static" state that is shared between multiple calls of the function.

Workaround is:

```
def x(y=None) :  
    if y==None : y = []  
    return y
```



4



1



45





**phil jones (he/him - ele)** @interstar · Mar 17



Replying to @OliverCaldwell

This isn't considered a "bug". It's a feature. (I.e. it's documented and fairly well known.)

People use it to get some kind of cheap "static" state that is shared between multiple calls of the function.

Workaround is:

```
def x(y):  
    if y==  
    return
```



**Dave Warry** @davewarry · Mar 19



Replying to @OliverCaldwell

It's just one of those things that have always been that way, AFAIK. It's explicitly called out in the Python Tutorial ([docs.python.org/3/tutorial/con...](https://docs.python.org/3/tutorial/con...)) - bottom of 4.8.1 and in the language spec ([docs.python.org/3/reference/co...](https://docs.python.org/3/reference/co...))



2



**Important warning:** The default value is evaluated only once. This makes a difference when the default is a mutable object such as a list, dictionary, or instances of most classes. For example, the following function accumulates the arguments passed to it on subsequent calls:

```
def f(a, L=[]):  
    L.append(a)  
    return L  
  
print(f(1))  
print(f(2))  
print(f(3))
```

This will print

```
[1]  
[1, 2]  
[1, 2, 3]
```



¿¿Qué pasa con otros lenguajes??  
Ejemplo





## Conclusión:

- No es un feature!!
- Es claramente un error o simplificación de implementación que “convirtieron en feature”



# Errores Históricos: Clases que no son objetos



Ejemplo



## Conclusión:

- No puedo usar polimorfismo a nivel clase
- No hay self/this
- No hay super
- Método static no es objetos!!
- Puedo simularlo con Metaprogramación, pero es todo un lio



# Acoplamiento con Clases





Ejemplo



## Conclusión:

- Referencias a clases generan acoplamiento
  - A la implementación
  - A un recurso externo
- En lenguajes estáticos es peor aún si las clases no son objetos



# Singleton



---

# SINGLETON

---

## Object Creational

---

### Intent

Ensure a class only has one instance, and provide a global point of access to it.

### Motivation

It's important for some classes to have exactly one instance. Although there can be many printers in a system, there should be only one printer spooler. There should be only one file system and one window manager. A digital filter will have one A/D converter. An accounting system will be dedicated to serving one company.

How do we ensure that a class has only one instance and that the instance is easily accessible? A global variable makes an object accessible, but it doesn't keep you from instantiating multiple objects.

A better solution is to make the class itself responsible for keeping track of its sole instance. The class can ensure that no other instance can be created (by intercepting requests to create new objects), and it can provide a way to access the instance. This is the Singleton pattern.



Ejemplo





## Conclusión:

- Si es Singleton, que no se note! (a la Smalltalk)
- No hay que confundir Singleton con “variable global”
- Un Singleton no me permite usar objetos polimórficos
- Si existe el mensaje “setInstance” no es singleton
- Dependency Injection
  - Problema forzado por frameworks
  - No tiene ningún sentido usar frameworks de dependency injection



# Métodos largos



Ejemplo



## Conclusión:

- Dificultan la comprensión
- No me dejan ver el QUÉ
- No me dejan ver la posibilidad de reuso



# Conclusiones







- No acoplarse a clases
- No usar Singletons
- No usar frameworks de Dependency Injection, modelar bien!
- Tener métodos cortos y bien declarativos



## Agenda

- Miércoles 30: “*TDD en la UI*” - Sorteo para Construcción de Software Robusto con TDD





# Diseño ¡a la gorra!

¡Bienvenidos a este exitoso Webinar que se llama "*Diseño a la Gorra*"!

Durante el año 2020 hicimos 17 ediciones donde exploramos qué es diseñar, cómo hacerlo mejor, cómo desarrollar un sistema haciendo TDD, etc.

En el año 2022 tendremos nuevos episodios en los que tocaremos todo tipo de tema, desde diseño de lenguajes de programación y su impacto en nuestro trabajo hasta temas de diseño clásico. A todo esto le sumaremos sorteos a los cursos abiertos que estamos dando en 10Pines.

Te esperamos todos los Miércoles a las 19 Hrs GMT-3 a partir del Miércoles 2 de Marzo de 2022. Para poder participar [REGISTRARTE ACA](#).

Todo el código y presentaciones estarán disponibles para que lo puedan usar y consultar en cualquier momento [acá](#).

👉 Más abajo están los links a todos los videos 👉

## Pago voluntario



\$1.000 - Muy buena 🍺

Pagar

\$1.500 - Dos combos 🍔

Pagar

\$2.000 - Dos buenos 🍷

Pagar



Donate



[¿Querés abonar un monto diferente o en otra plataforma?](#)

<https://alagorra.10pines.com>

¡Ahora si el sorteo!



# ¡Sorteo de Entradas!

- 1 entrada gratis, 1 al 50%, 2 al 25%
- Son transferibles.
- No son de uso obligatorio

Online

## Diseño Avanzado de Software con Objetos I


📅 Empieza el **04/04**

📅 Del 04/04 al 18/04. Días: 3 días a la semana durante 2 semanas — Lunes, Miércoles y Viernes — 9:00 a 13:00 GMT-3.


💰 **AR\$ 64.800-** (IVA incluido) **U\$D 495-** (impuestos incluidos)

[Hacer una consulta](#)

[Inscribirme ahora \*\*10% Dto.\*\*](#)

 Dictado por:  
**Máximo Prieto**

~~AR\$ 72.000-~~  
**AR\$ 64.800-**  
(IVA incluido)  
~~U\$D 550-~~  
**U\$D 495-**  
(impuestos incluidos)

 **Early Bird** Inscripción Temprana hasta 21/03



Muchas gracias





# 10 Pines

Creative Software Development



[10pines.com](http://10pines.com)



[info@10pines.com](mailto:info@10pines.com)



+54 (011) 6091-3125 / 4893-2057



Av. Leandro N. Alem 896 6° - Bs. As. - Argentina



@10pines