

Repaso P00





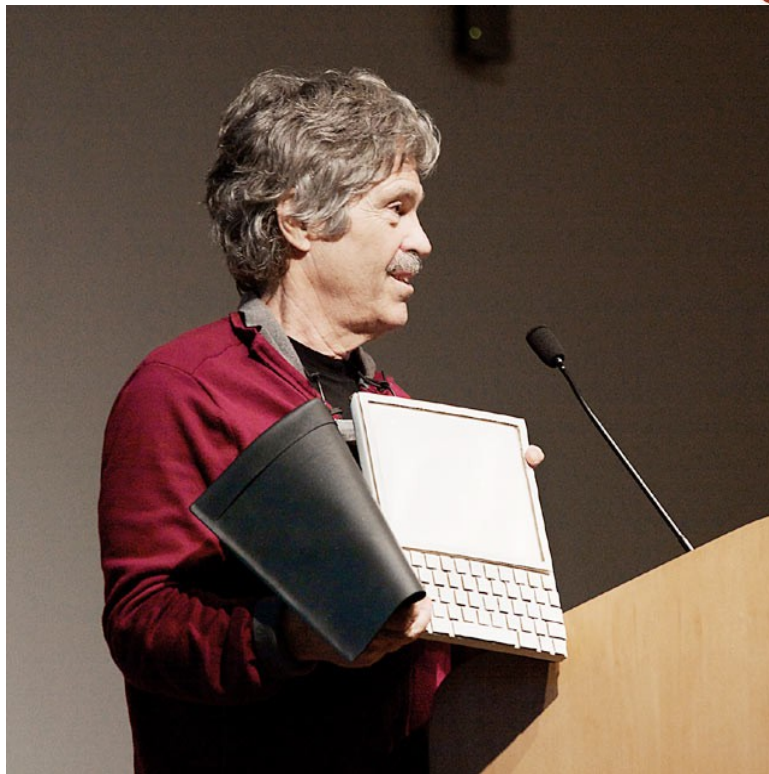
Tiempos Heroicos



- Código:
 - Datos
 - Funciones
- Interfaz: terminal

El PARC y Alan Kay

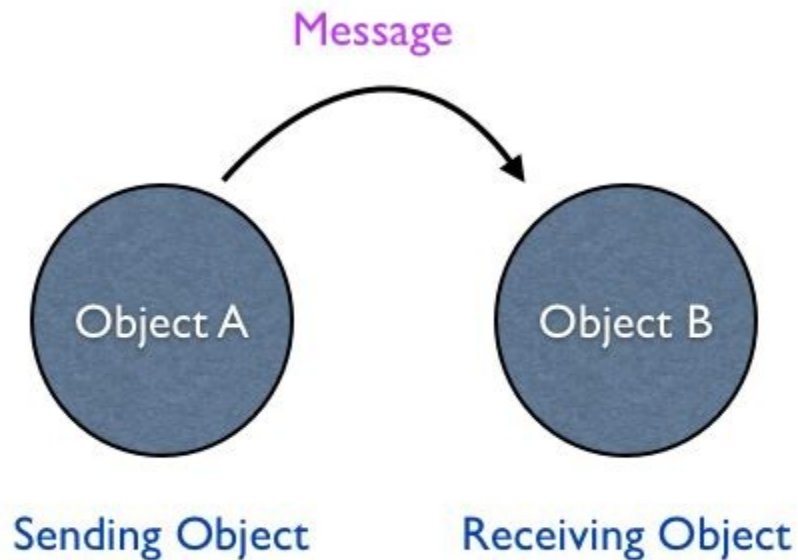
- Código:
 - POO
- Interfaces Gráficas
- Deberes: Ver "[*How to invent the future*](#)"



Objetos y Mensajes

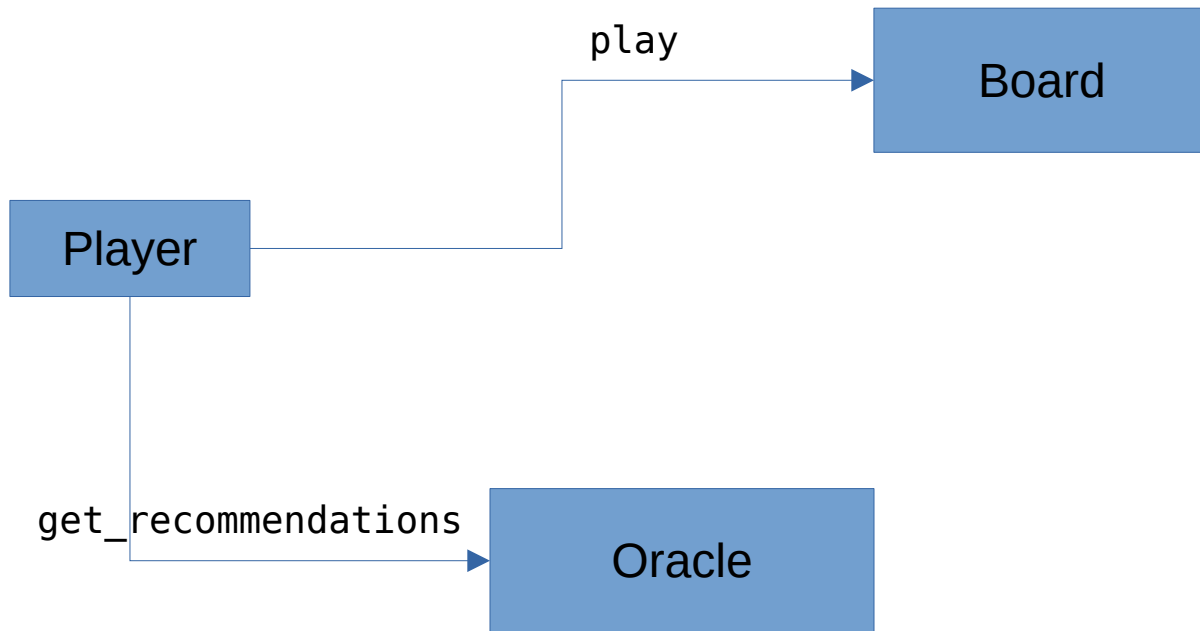
- Modelamos el mundo como “objetos que reciben mensajes” y responden a ellos.
- Ejemplos
 - Toda cerradura entiende los mensajes abrir y cerrar
 - Todo microondas entiende los mensajes, tiempo, potencia, empezar y parar
 - Todo aparato eléctrico entiende los mensajes enciende y apaga

Objetos y Mensajes



Message Passing

Objetos y Mensajes



¿Cómo se representa todo esto en código?

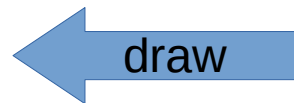
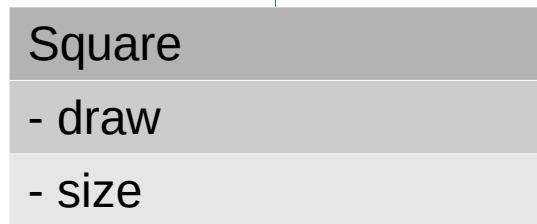
- Combinando datos y funciones relacionadas en una sola estructura
 - Datos
 - Funciones (métodos)

Objeto: Una estructura de huecos con nombres



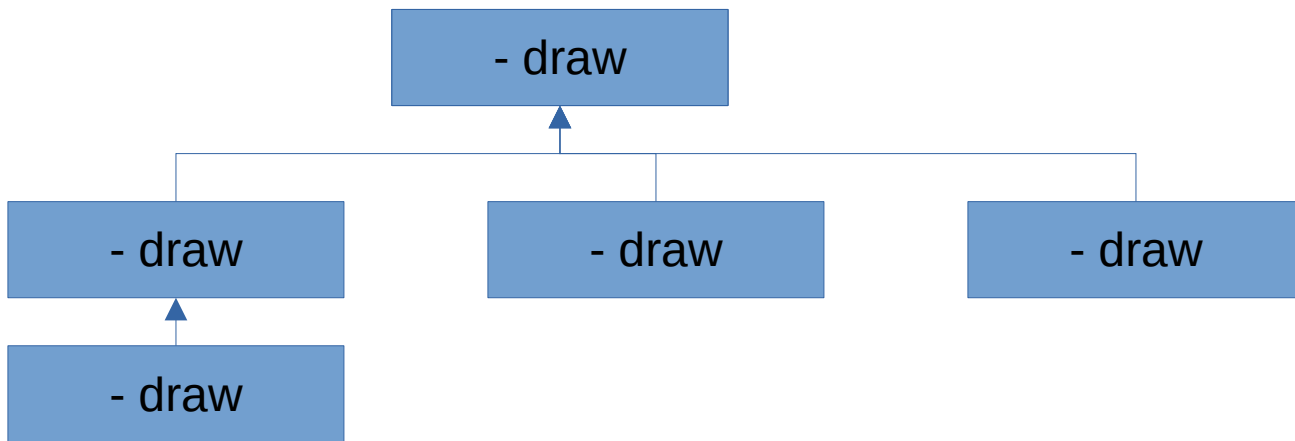
- Cada hueco (slot) tiene un nombre
- Cada hueco contiene o bien un dato o una función (llamada método)
- Los nombres son los mensajes que entiende el objeto
- Cuando a un objeto le mandas un mensaje, mira a ver si tiene un hueco con ese nombre, y si lo tiene
 - Devuelve el valor en caso de ser un dato
 - Ejecuta la función, en caso de ser un método.

Los objetos se organizan en jerarquías (herencia)



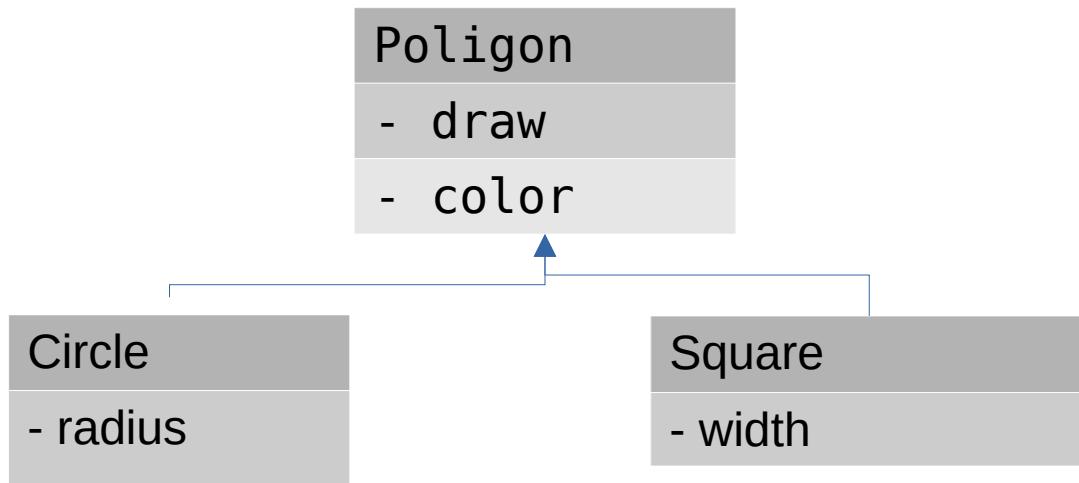
Polimorfismo

- Varios objetos de una jerarquía entienden el mismo mensaje.
- Cuando lo reciben, cada uno hace lo que le sale el moño. La etiqueta del slot (el mensaje) es el mismo, pero el contenido es código propio de cada caso.



Principio de Sustitución de Liskov

- Debería de poder sustituir una subclase por otra sin problemas.
- Dicho de otra forma: todas las subclases deben de entender un mismo conjunto de mensajes



Encapsulación

- Que cada perro se lama su cipote.
- O dicho de otra manera: cuando te mando un mensaje, espero que hagas tu trabajo, y me importa un bledo los detalles de cómo lo haces.
- Es una forma de delegar y no “micro gestionar”.

¿Qué es una clase?

- La clase es la fábrica que construye los objetos.
- Un objeto es siempre la instancia de una clase.
- Para que algo sea un objeto, tiene que tener una clase.
- Para crear objetos,
 - Creamos una clase
 - Le pedimos a la clase que cree un nuevo objeto
- En Python, las clases también son objetos y entienden mensajes y tienen métodos y
tienen una clase.
- Calma.

Métodos de instancia y métodos de clase

- Las instancias (uséase, objetos) tienen métodos (slots que almacenan una función).
 - Por ejemplo, los objetos de tipo str, tienen métodos como:
 - lower
 - find, etc...
- Las clases también pueden tener métodos y se llaman.... Métodos de clase. Se usan casi siempre para modificar la forma de crear objetos.



Clase Base



- La clase de la cual descienden todas las demás.

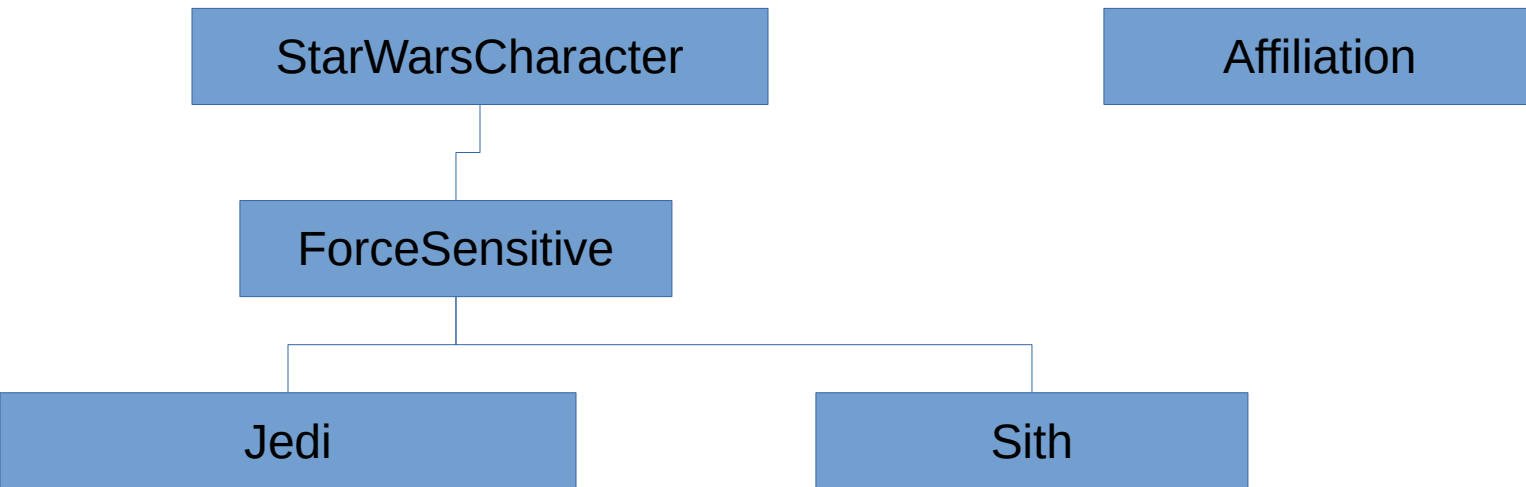
El TAO de cualquier lenguaje orientado a objetos

- Es la interfaz (conjunto de mensajes que entiende) de la clase base.
- En el caso de Python, esos son los dunders.
- Domina los dunders, y dominarás a Python

Herencia Sencilla y Herencia Múltiple

- Herencia sencilla: una clase sólo tiene una superclase
- Herencia múltiple: una clase puede heredar de varias.
- Python permite la herencia múltiple, pero NO la uses hasta que tengas mucha más soltura

Personajes de StarWars





KEEPCODING

Tech School

Madrid | Barcelona | Bogotá

Datos de contacto