UNIVERSIDAD DE LIMA | 60 años

Paradigma Orientado a Objetos (p1)

Lenguajes de Programación

Historia

- Nace a partir de la necesidad de crear programas más complejos, de fácil mantenibilidad, con menos bugs.
- El primer lenguaje orientado a objetos fue el SmallTalk (70s).
- Fue un lenguaje que planteó los principios fundamentales del paradigma orientado a objetos.
- Creado en Xerox PARC por Alan Kay, fue utilizado para la creación de la primera GUI en el sistema Macintosh.
- A finales de los 80s (debido a los avances en hardware) estos principios se volvieron populares y fueron adoptados por la industria.

Pirates of Silicon Valley (1999)

https://www.youtube.com/watch?v=pPsfWRzdK_k

UNIVERSIDAD DE LIMA

Definición OBJETO

Un objeto es una representación en memoria de cualquier estructura del lenguaje de programación.

Diferencias entre Clases y Objetos

Una clase es un **TIPO** de dato especificado por el usuario en tiempo de ejecución.

Un objeto es una **INSTANCIA** de una clase.

Persona

Alan Turing es de tipo Persona.
Turing es una instancia de la clase Persona.

Alan Kay es de tipo Persona.
Kay es una instancia de la clase Persona.

James Gosling es de tipo Persona.
Gosling es una instancia de la clase Persona.

Relación entre Clases y Objetos

Un **objeto** al ser de cierta **clase**, tiene que definir o implementar las **propiedades** que la clase ha definido.

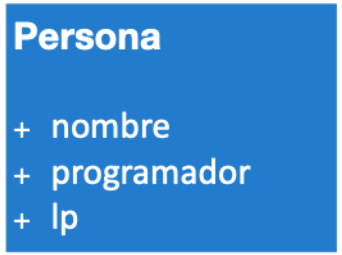
Turing, Kay y Gosling son objetos del tipo Persona (de la clase Persona).

¿La clase Persona es un objeto?

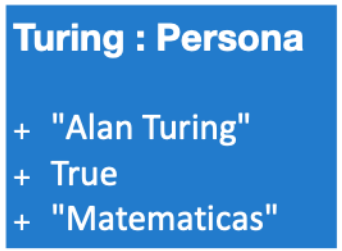
Si, también es un objeto de un tipo especial que suele llamarse Class.

UNIVERSIDAD DE LIMA

Todo es un objeto

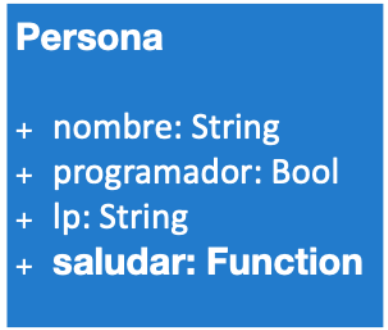


Persona (clase) es una instancia de Class.
nombre, programador, lp es una instancia de una clase (Field, Property, Method, etc). En terminología OOP se les conoce como **propiedades**.



Turing es una instancia de Persona
"Alan Turing" es un objeto de tipo String
True es un objeto de tipo Bool
"Matematicas" es un objeto de tipo String

Las clases pueden definir métodos



Las funciones que son propiedades de una clase se conocen como **métodos**.

Al ser propiedades también pueden ser tratadas como objetos.

Los métodos definen el comportamiento que pueden tener los objetos de una clase determinada.

Tipos de Propiedades

Propiedades de una clase: Estas propiedades se definen a nivel de clase y no se necesita instanciar la clase para poder accederlos. También se le conoce como **variables o atributos de clase** y en el caso de los métodos se les conoce como **métodos de clase**.

Propiedades de una instancia: Estas propiedades se definen en la clase pero solamente se pueden acceder mediante una instancia de la clase. También se le conocen como **variables o atributos de instancia** y en el caso de los métodos se le conoce como **métodos de instancia.**

Las variables que se definen **dentro de una función (método)** se les conoce como **variables temporales**, o sea, que su vida se limita a la ejecución de una función.

UNIVERSIDAD DE LIMA

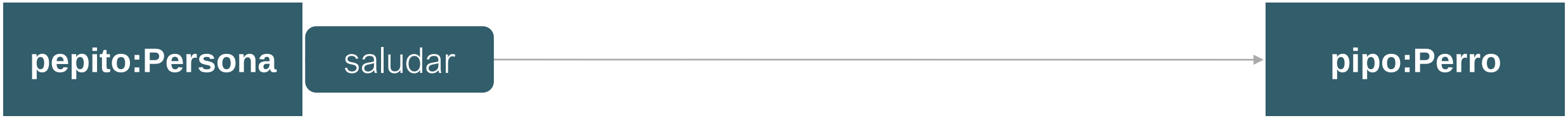
Comunicación entre objetos (Mensajes)

La relación entre objetos se da mediante el envío de mensajes entre estos.

El objeto pepito le envía un mensaje al objeto pipo.

Comunicación entre objetos (Mensajes)

En código, se vería de la siguiente forma:



Los mensajes se implementan utilizando los métodos. El objeto que inicia la comunicación llama un método del receptor del mensaje.

```
class Persona:
  def __init__(self, nombre, mascota):
    self.nombre = nombre
    self.mascota = mascota

  def saludar(self):
    self.mascota.saludar()
```

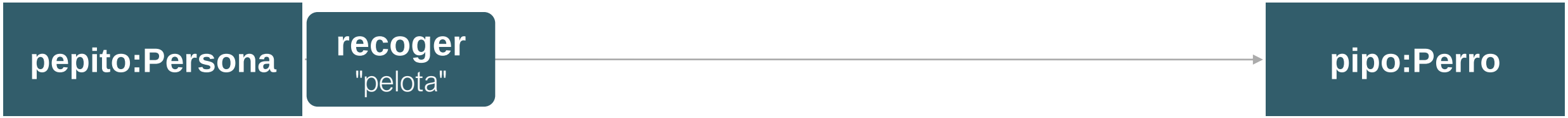
```
class Perro:
  def __init__(self, nombre):
    self.nombre = nombre

  def saludar(self):
    print("Perro dice: Guau guau!”)
```

UNIVERSIDAD DE LIMA

Los mensajes pueden contener parámetros

Cuando el objeto emisor envía un mensaje al objeto receptor, suele enviarle parámetros que el objeto receptor utilizará para reaccionar ante el mensaje.




```
class Persona:
  # metodo que envia el mensaje
  def jugar(self)
    self.mascota.recoger("pelota")
```

```
class Perro:
  # Metodo del mensaje
  def recoger(self, objeto):
    if objeto == "pelota":
      print("Recogio la pelota")
    else:
      print("No hizo nada")
```

UNIVERSIDAD DE LIMA