# LICENCIATURA EN CIENCIA DE DATOS

## PROGRAMACIÓN AVANZADA (189)

Lic. Felipe Morales Clase N. 1. Unidad I. Introducción a Programación Orientada a Objetos







#### **TEMARIO**

- T0: Revisión de formulario de encuesta
- T1: Introducción al Paradigma de Programación Orientada a Objetos.
- T2: Pilares del Paradigma Orientado a Objetos.
- T3: Encapsulamiento, abstracción, herencia y polimorfismo.
- T4: Componentes: Objetos, Clase, Atributos, Métodos
- T5: Ejemplos en Lenguajes Python
- Temas relacionados
- Practica
- Links



T0: Revisión de formulario de encuesta

https://forms.gle/2RMVi1uGKGm1Pfxf8



#### T1: Introducción al Paradigma de Programación Orientada a Objetos

#### ¿Qué es un paradigma?

"Teoría o conjunto de teorías cuyo núcleo central se acepta sin cuestionar y que suministra la base y modelo para resolver problemas y avanzar en el conocimiento"

## ¿Qué es un paradigma de programación?

Una posible definición es "Un paradigma de programación es una manera o estilo de programación de software. Existen diferentes formas de diseñar un lenguaje de programación y varios modos de trabajar para obtener los resultados que necesitan los programadores. Se trata de un conjunto de métodos sistemáticos aplicables en todos los niveles del diseño de programas para resolver problemas computacionales."

#### ¿Y orientada a Objetos?

Podemos decir que es algo así como "Algo que está focalizado o es representado por OBJETOS -elementos físicos o abstractos que existen en la vida real o imaginaria-"



## T1: Introducción al Paradigma de Programación Orientada a Objetos (Continuación)

Uniendo todo entonces nos surge la siguiente duda:

## ¿Qué es un paradigma de programación orientada a objetos?

Una posible definición es "En este modelo de paradigma se construyen modelos de objetos que representan elementos (objetos) del problema a resolver, que tienen características y funciones"

#### ¿Y qué características tiene?

"Permite separar los diferentes componentes de un programa, simplificando así su creación, depuración y posteriores mejoras. La programación orientada a objetos disminuye los errores y promociona la reutilización del código. Es una manera especial de programar, que se acerca de alguna manera a cómo expresaremos las cosas en la vida real.

Podemos definir un objeto como una estructura abstracta que, de manera más fiable, describe un posible objeto del mundo real y su relación con el resto del mundo que lo rodea a través de interfaces. Ejemplos de lenguajes de programación orientados a objetos serían Java, Python o C#."



## T1: Introducción al Paradigma de Programación Orientada a Objetos (Continuación)

#### En forma simplificada es:

En este modelo de programación se construyen modelos de objetos que representan elementos (objetos) del problema a resolver, que tienen características y funciones.

#### Y sus características son:

- Permite separar los componentes de un programa
- Simplifica la creación, depuración y posteriores mejoras del programa
- Ayuda a disminuir errores
- Promueve la reutilización del código



T1: Introducción al Paradigma de Programación Orientada a Objetos (Continuación)

¿Existen otros paradigmas de programación?:

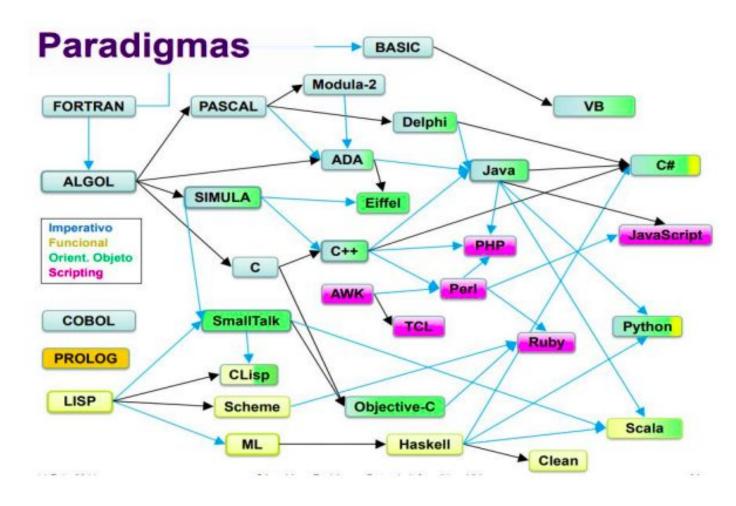
## SI!

## ¿Cuáles?

- Imperativo o Estructura o Procedural
- Declarativo
- Orientado a Objetos
- Reactivo
- Funcional
- Scripting
- Otros



T1: Introducción al Paradigma de Programación Orientada a Objetos (Continuación)



Paradigmas de programación en formato gráfico



## T2: Pilares del Paradigma de Programación Orientada a Objetos

Los pilares son los conceptos fundamentales sobre los cuales se sustenta el paradigma.

## ¿Cuáles?

- Encapsulamiento
- Abstracción
- Herencia
- Polimorfismo

#### Otros conceptos asociados:

- Cohesión
- Acoplamiento



## T3: Pilares del Paradigma de Programación Orientada a Objetos



Fuente: ed.team/cursos/poo



T4: Componentes del Paradigma de Programación Orientada a Objetos

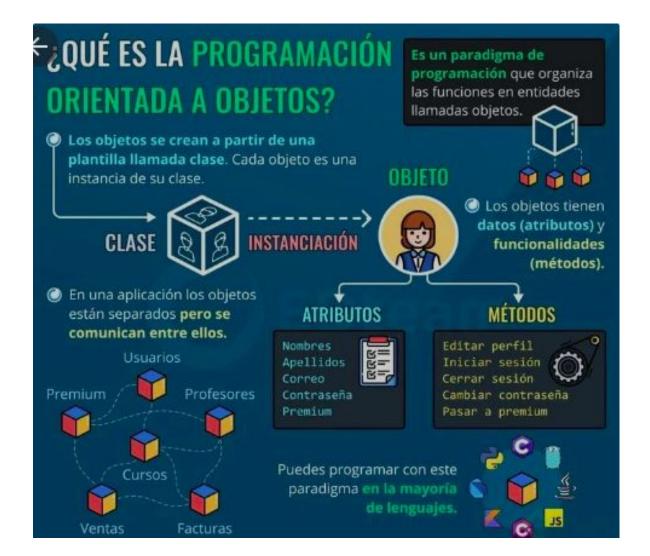
Los **componentes** están dados por los siguientes elementos constitutivos de este paradigma:

#### ¿Cuáles?

- Objetos
- Clase
- Atributos
- Métodos



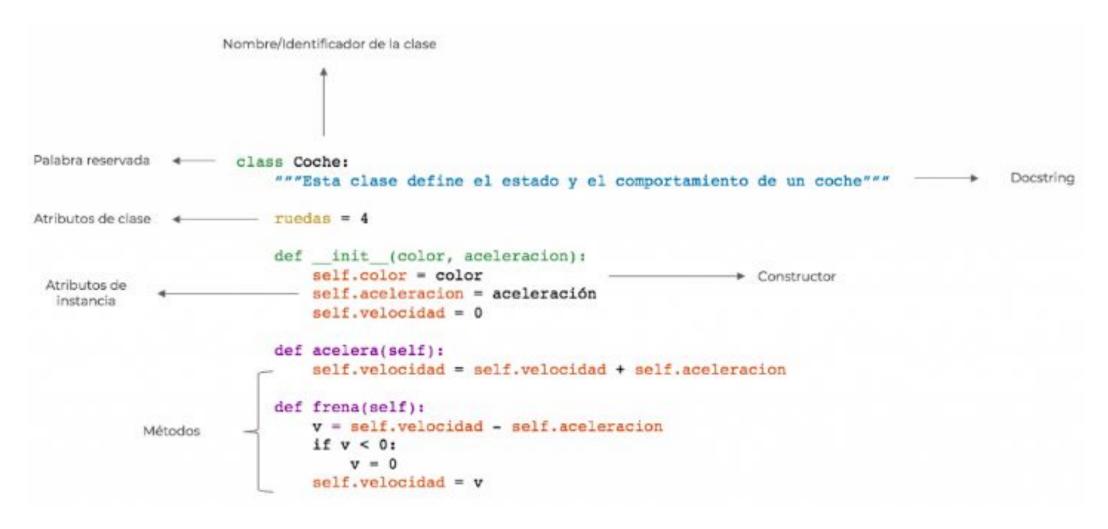
#### T4: Componentes del Paradigma de Programación Orientada a Objetos



Fuente: platzi.com



## T5: Ejemplo en Python





#### Práctica 1 - Unidad I

#### 1) Acceda al repositorio privado en GitHub:

- Registrarse en caso de con contar con una cuenta propia
- Link <a href="https://github.com/felipemoralesquerol/UNAB.2024.PA.Clase01">https://github.com/felipemoralesquerol/UNAB.2024.PA.Clase01</a>
- Adecuar su IDE favorito para ejecutar los ejemplos

#### 1) Sobre el repositorio privado en GitHub:

- Busqué el texto TODO en el repositorio.
- ¿Cómo resolvería los interrogantes que plantea?
- Seleccionar 5 TODOs y resolverlos (en forma individual o grupal)

#### 1) (Opcional) Seleccionar 2 (dos) de los pilares (o pilares asociados) y arme un breve resumen con:

- Breve explicación con sus palabras
- o Programar un ejemplo en Python y explicar cómo aplica este concepto



#### T5: Links interesantes:

#### Aprenda a pensar como un programado con Python

https://argentinaenpython.com/quiero-aprender-python/aprenda-a-pensar-como-un-programador-con-python.pdf

#### Documentación oficial:

https://www.python.org/doc/

#### **Curso de Python desde 0 (pildorasinformaticas)**

https://www.youtube.com/playlist?list=PLU8oAlHdN5BlvPxziopYZRd55pdqFwkeS

#### **Otros links:**

https://ellibrodepython.com/polimorfismo-en-programacion

https://parzibyte.me/blog/2019/06/30/clases-constructores-python-poo/





¡Muchas gracias por tu atención!