

## PARADIGMAS DE PROGRAMACIÓN PARA INTELIGENCIA ARTIFICIAL Y ANÁLISIS DE DATOS

### PARADIGMAS AVANZADOS APLICADOS A IA

Eduardo Yepez Montenegro

2 Julio 2025





## Clase 2: Introducción

---

# Agenda

### 3 Paradigmas Avanzados Aplicados a IA

1. Programación Funcional en Python
  2. Programación Orientada a Objetos (POO) en Python
  3. Programación Declarativa en IA
-



## Objetivos de la Clase

---

¿Qué aprenderemos en esta clase?

1. Conocer diferentes enfoques de programación usados en inteligencia artificial.
  2. Aprender cómo escribir código sencillo y eficiente usando funciones.
  3. Entender cómo organizar el código usando objetos y clases.
  4. Ver cómo resolver problemas de IA enfocándose en el "qué" y no en el "cómo".
-



# Introducción a Paradigmas Avanzados Aplicados a IA

---

## ¿Qué son los paradigmas avanzados?

Los paradigmas de programación son estilos para resolver problemas con código. En IA, los paradigmas avanzados ayudan a crear modelos eficientes y flexibles.

Beneficios:

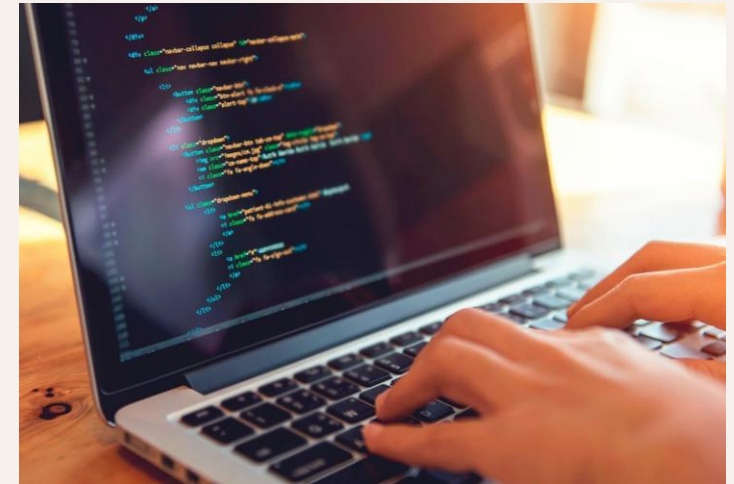
- Modularidad: Divide el código en partes reutilizables.
- Escalabilidad: Facilita la adaptación a más datos.
- Eficiencia: Mejora el rendimiento y reduce errores.

Aplicaciones:

- Programación Funcional: Útil en análisis de datos y procesamiento paralelo.
- Programación Orientada a Objetos: Ideal para crear modelos predictivos y software de machine learning.
- Programación Declarativa: Aplicada en sistemas expertos y algoritmos de optimización.

Estos enfoques permiten escribir código más adaptable y optimizado para proyectos de IA.

---





# Introducción a la Programación Funcional

## ¿Qué es la Programación Funcional?

- Paradigma de programación basado en el uso de funciones.
- Enfocado en el uso de funciones puras y datos inmutables.
- Ventajas: código más predecible, fácil de probar y paralelizable.
- Uso de funciones de orden superior (map, filter, reduce).



## Ejemplo: Python

# Función pura para calcular el retorno de inversión (ROI)

```
def calculate_roi(profit, investment):  
    return (profit - investment) / investment
```

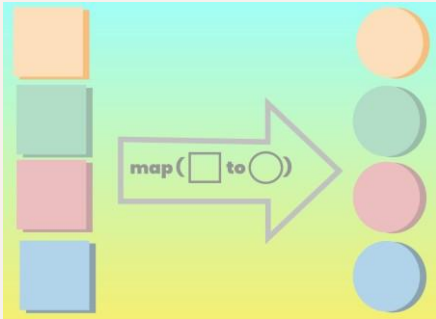
# Uso de la función pura

```
roi = calculate_roi(50000, 25000)  
print(f"El ROI es: {roi:.2%}") # Resultado: El ROI es: 100.00%
```

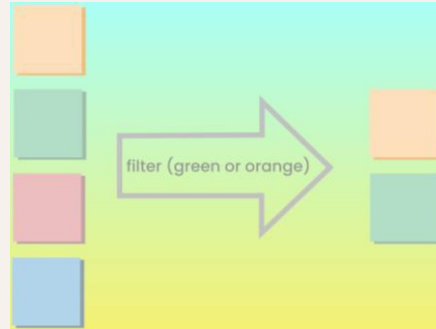


# Uso de map, filter y reduce

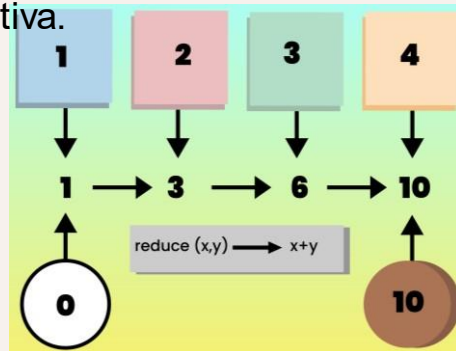
`map()`: Aplica una función a cada elemento de una lista.



`filter()`: Filtra elementos que cumplen una condición.



`reduce()`: Reduce una lista a un único valor aplicando una función acumulativa.



## Ejemplo Python

```
from functools import reduce
```

```
# Lista de ingresos diarios  
ingresos = [500, 1200, 300, 1500, 800]
```

```
# Usando map para aplicar un impuesto del 10%  
ingresos_ajustados = list(map(lambda x: x * 0.9, ingresos))
```

```
# Filtrando ingresos mayores a $1,000  
ingresos_filtrados = list(filter(lambda x: x > 1000, ingresos_ajustados))
```

```
# Calculando el ingreso total con reduce  
ingreso_total = reduce(lambda x, y: x + y, ingresos_filtrados)  
print(f"Ingreso total ajustado: ${ingreso_total}")
```



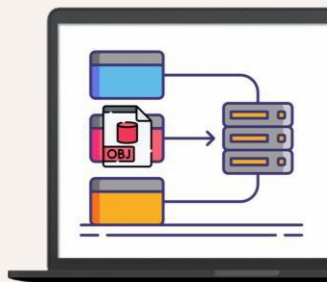
# Programación Orientada a Objetos (POO) en Python

---

## Clases y Objetos en Python para Modelos de IA:

### Definición de Clases y Objetos:

- Clases: Son plantillas que definen las propiedades y comportamientos de los objetos.
- Objetos: Son instancias de una clase, con datos y métodos específicos.



Aplicación en IA: En modelos de IA las clases ayudan a organizar datos y métodos, facilitando , la creación y el manejo de modelos predictivos.

### Ejemplo práctico

```
class Model:
    def __init__(self, data, model_type):
        self.data = data
        self.model_type = model_type

    def train(self):
        print(f"Training {self.model_type} model.")

    def predict(self):
        print("Making predictions...")

# Uso de la clase
my_model = Model(data="Dataset", model_type="Random Forest")
my_model.train()
my_model.predict()
```



# Programación Orientada a Objetos (POO) en Python



## ¿Qué es la Herencia?

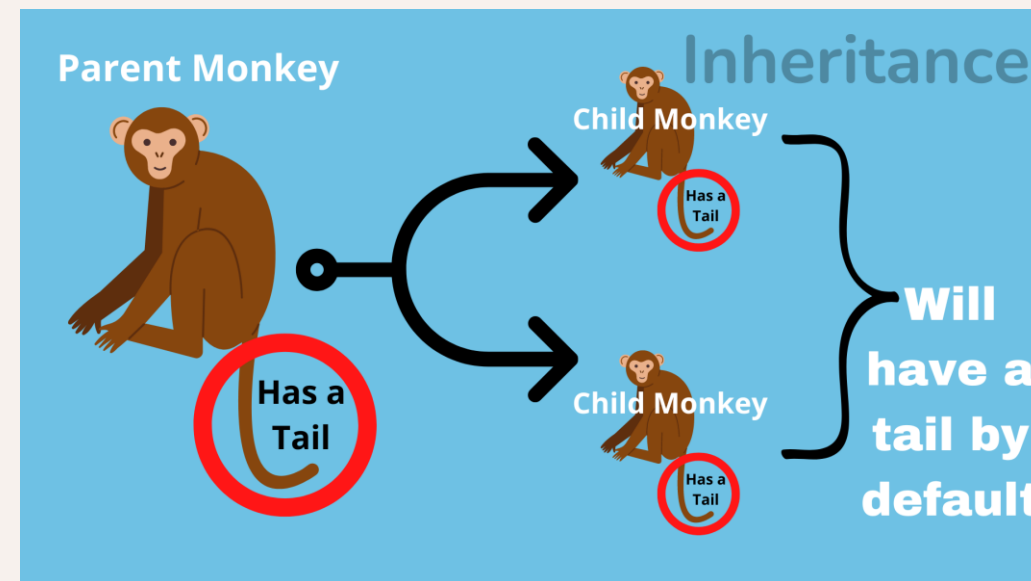
Es un pilar de la POO que permite a una clase (hija) adquirir los atributos y métodos de otra clase (padre). Esto fomenta la reutilización de código y crea una jerarquía lógica.

**Clase Base (Padre):** Define una interfaz y comportamiento común. Por ejemplo, una clase BaseModel con métodos train() y predict().

**Clase Derivada (Hija):** Hereda de la clase base y puede añadir o modificar funcionalidades. Por ejemplo, LinearRegressionModel y NeuralNetworkModel pueden tener implementaciones distintas del método train().

### Aplicación en IA:

Permite construir una "familia" de modelos de machine learning que comparten una estructura común, pero cada uno con su lógica de entrenamiento y predicción específica. Esto hace que el código sea más limpio, modular y fácil de extender.



Fuente: <https://python-hub.com/inheritance-in-python-oop/>





# Programación Declarativa en IA

## Introducción a la Programación Declarativa en IA

¿Qué es la Programación Declarativa?

La programación declarativa se centra en describir el resultado que deseamos lograr sin especificar los pasos para obtenerlo.

Objetivo:

Comprender este enfoque permite enfocarse en el resultado deseado, facilitando la creación de soluciones eficientes en IA.

Aplicaciones en IA:

Este paradigma es útil para desarrollar sistemas expertos, bases de datos inteligentes y procesos de automatización.

Ejemplo Práctico:

Ejemplo conceptual:



*"En lugar de programar los pasos para encontrar la mejor ruta en un mapa, definimos las reglas de tráfico y dejamos que el sistema decida la mejor ruta basada en esas reglas."*

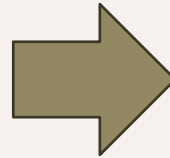


# Programación Declarativa en IA

## Lenguajes Declarativos en IA: SQL

Introducción a SQL:

- SQL es ampliamente usado en bases de datos relacionales para realizar consultas complejas, facilitando el análisis y extracción de patrones en proyectos de IA.



Ejemplo de consulta en SQL

```
SELECT nombre, puntuacion FROM  
estudiantes WHERE puntuacion > 90;
```



Objetivo:

Aprender a aplicar estos lenguajes para resolver problemas complejos y manejar datos con precisión.



Gracias por su atención

---



Preguntas

---