

Programación Orientada a Objetos (POO)

Herencia

Uso de *super()*

`super()` en Python es una función que se utiliza para llamar a un método de una clase base (superclase) desde una subclase. Es útil en situaciones donde una subclase sobrescribe un método de la superclase, pero aún se desea mantener la funcionalidad de la superclase en combinación con la nueva funcionalidad definida en la subclase.

El siguiente código muestra cómo se utiliza la función `super()` en Python para invocar métodos de la clase base desde una clase derivada. La función `super()` es útil para extender o modificar el comportamiento de los métodos heredados sin reescribir el código de la clase base.

Clase Base: `Animal`

- `__init__(self, nombre)`
 - **Propósito:** Constructor de la clase `Animal`.
 - **Parámetro:**
 - * `nombre`: Nombre del animal.
 - **Comportamiento:** Inicializa el atributo privado `__nombre`.
- `def hablar(self)`
 - **Propósito:** Método que define el comportamiento general de hablar para los animales.
 - **Comportamiento:** Imprime un mensaje que indica que el animal hace un sonido. El atributo `__nombre` es utilizado, pero como es privado, no se accede a él directamente en la clase derivada.

Clase Derivada: `Perro`

- `__init__(self, nombre, raza)`

- **Propósito:** Constructor de la clase `Perro`, que hereda de `Animal`.
 - **Parámetros:**
 - * `nombre`: Nombre del perro.
 - * `raza`: Raza del perro.
 - **Comportamiento:**
 - * `super().__init__(nombre)`: Llama al constructor de la clase base `Animal` para inicializar el atributo `__nombre`.
 - * Inicializa el atributo privado `__raza` específico de la clase `Perro`.
- `def hablar(self)`
 - **Propósito:** Sobrecribir el método `hablar` de la clase base `Animal` para proporcionar una implementación específica para los perros.
 - **Comportamiento:**
 - * `super().hablar()`: Llama al método `hablar` de la clase base `Animal` para imprimir el mensaje general.
 - * Imprime el sonido específico “Guau”, que es característico de los perros.

```
# Clase base
class Animal:
    def __init__(self, nombre):
        self.__nombre = nombre

    def hablar(self):
        print(f"{self.__nombre} hace el siguiente sonido: ")

# Clase Derivada
class Perro(Animal):
    def __init__(self, nombre, raza):
        super().__init__(nombre)
        self.__raza = raza

    def hablar(self):
        super().hablar()
        print("Guau")

mi_perro = Perro("Helena", "Pastor Alemán")

mi_perro.hablar()
```

Helena hace el siguiente sonido:
Guau