Ejercicios de Herencia y Polimorfismo en Python

A continuación, se presentan 5 ejercicios de herencia y polimorfismo en Python, con diferentes niveles de dificultad: básico, intermedio y avanzado.

Básico: Clase Animal y Perro

Crea una clase 'Animal' con un método 'hablar()' que imprima 'El animal hace un sonido'. Crea una clase 'Perro' que herede de 'Animal' y sobrescriba el método 'hablar()' para que imprima 'El perro ladra'.

Ejemplo:

```
class Animal:
    def hablar(self):
        print('El animal hace un sonido')


class Perro(Animal):
    def hablar(self):
        print('El perro ladra')


animal = Animal()
animal.hablar()
perro = Perro()
perro.hablar()
```

Intermedio: Clases Vehículo y Carro

Crea una clase 'Vehiculo' con los atributos 'marca', 'modelo' y 'año'. Crea una clase 'Carro' que herede de 'Vehiculo' y añada el atributo 'tipo de combustible'. Sobrescribe el método '__str__()' para que imprima una descripción completa del carro.

Ejemplo:

```
class Vehiculo:
    def __init__(self, marca, modelo, año):
        self.marca = marca
        self.modelo = modelo
        self.año = año

class Carro(Vehiculo):
    def __init__(self, marca, modelo, año, combustible):
        super().__init__(marca, modelo, año)
        self.combustible = combustible
    def __str__(self):
        return f'{self.marca} {self.modelo} ({self.año}), Combustible: {self.combustible}'

carro = Carro('Toyota', 'Corolla', 2020, 'Gasolina')
print(carro)
```

Avanzado: Clases Figura y Círculo

Crea una clase base llamada 'Figura' que tenga un método 'area()'. Luego, crea una clase 'Círculo' que herede de 'Figura' y sobrescriba el método 'area()' para calcular el área del círculo (usa el valor de pi = 3.14). Crea también una clase 'Rectángulo' que calcule el área de un rectángulo.

Ejemplo:

```
import math

class Figura:
    def area(self):
        pass
```

```
class Circulo(Figura):
    def __init__(self, radio):
        self.radio = radio
    def area(self):
        return 3.14 * (self.radio ** 2)

class Rectangulo(Figura):
    def __init__(self, largo, ancho):
        self.largo = largo
        self.ancho = ancho
    def area(self):
        return self.largo * self.ancho

circulo = Circulo(5)
print('Área del círculo:', circulo.area())
rectangulo = Rectangulo(4, 6)
print('Área del rectángulo:', rectangulo.area())
```

Intermedio: Clases Persona y Estudiante

Crea una clase 'Persona' con los atributos 'nombre' y 'edad'. Luego, crea una clase 'Estudiante' que herede de 'Persona' y añada el atributo 'curso'. Sobrescribe el método '__str__()' para mostrar todos los detalles del estudiante.

Ejemplo:

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

```
class Estudiante(Persona):
    def __init__(self, nombre, edad, curso):
        super().__init__(nombre, edad)
        self.curso = curso
    def __str__(self):
        return f'{self.nombre}, {self.edad} años, Curso: {self.curso}'

estudiante = Estudiante('Juan', 21, 'Matemáticas')
print(estudiante)
```

Básico: Clase Vehículo con Polimorfismo

Crea una clase base 'Vehiculo' con un método 'conducir()'. Luego, crea dos clases derivadas 'Coche' y 'Moto' que sobrescriban este método para imprimir mensajes específicos de cada tipo de vehículo.

Ejemplo:

```
class Vehiculo:
    def conducir(self):
        print('El vehículo está siendo conducido')

class Coche(Vehiculo):
    def conducir(self):
        print('El coche está siendo conducido')

class Moto(Vehiculo):
    def conducir(self):
        print('La moto está siendo conducida')
```

vehiculo = Vehiculo()

vehiculo.conducir()

coche = Coche()

coche.conducir()

moto = Moto()

moto.conducir()