

# Curso: Programación Orientada a Objetos

Tema:  
Poliforfismo





¿Qué entiende por  
polimorfismo?

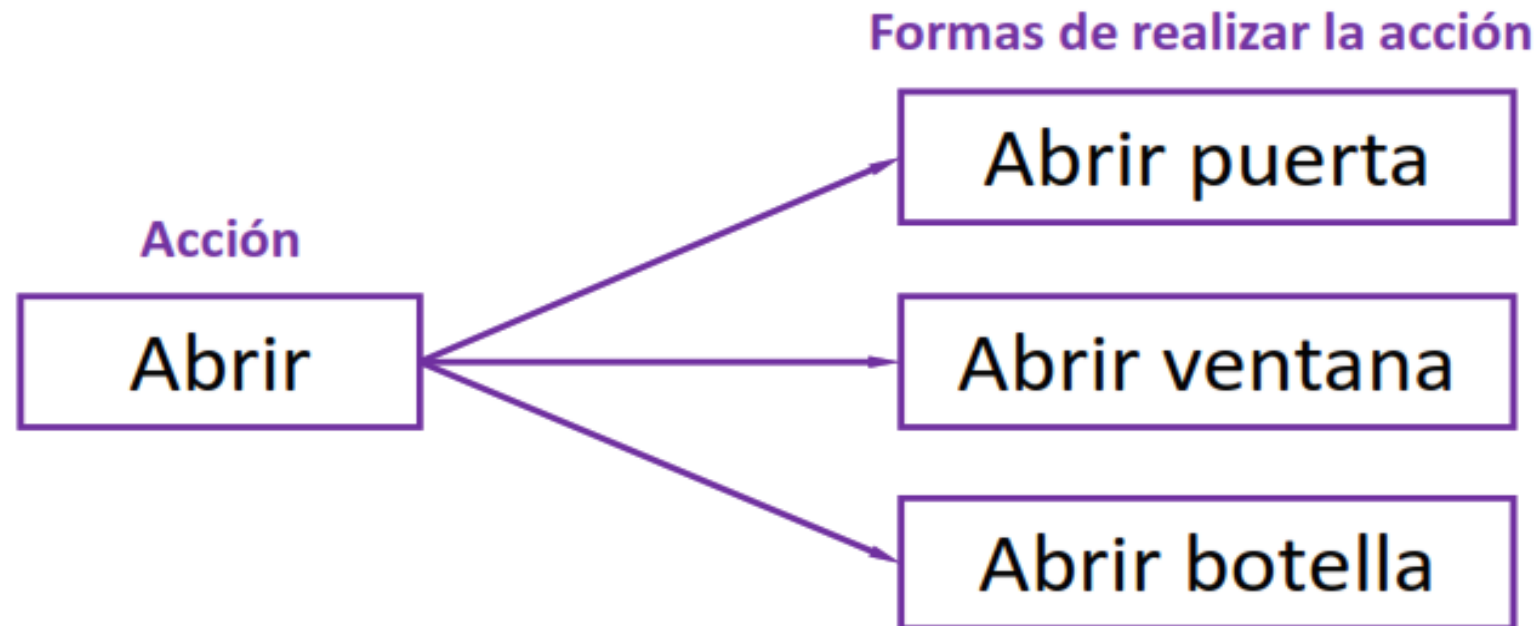


# ¿Qué es polimorfismo?

Unis



El polimorfismo en programación, especialmente en Python, se refiere a la capacidad de objetos de diferentes tipos de responder al mismo método o función. Esto significa que puedes usar el mismo nombre de método o función en diferentes tipos de objetos y obtener resultados específicos para cada uno.



# Ejemplo

```
class Perro:
    def hablar(self):
        print('Guau, Guau')

class Gato:
    def hablar(self):
        print('Miau, Miau')

class Vaca:
    def hablar(self):
        print('Muuu, Muuu')

def llama_hablar(x):
    x.hablar()

p=Perro()
p.hablar()
g=Gato()
g.hablar()
v=Vaca()
v.hablar()

llama_hablar(p)
llama_hablar(g)
llama_hablar(v)

llama_hablar(Perro())
llama_hablar(Gato())
llama_hablar(Vaca())

lista=[Perro(),Gato(),Vaca()]
for animal in lista:
    animal.hablar()
```

```
Guau, Guau
Miau, Miau
Muuu, Muuu
```

## Ejemplo 2

```
class Empleado:
```

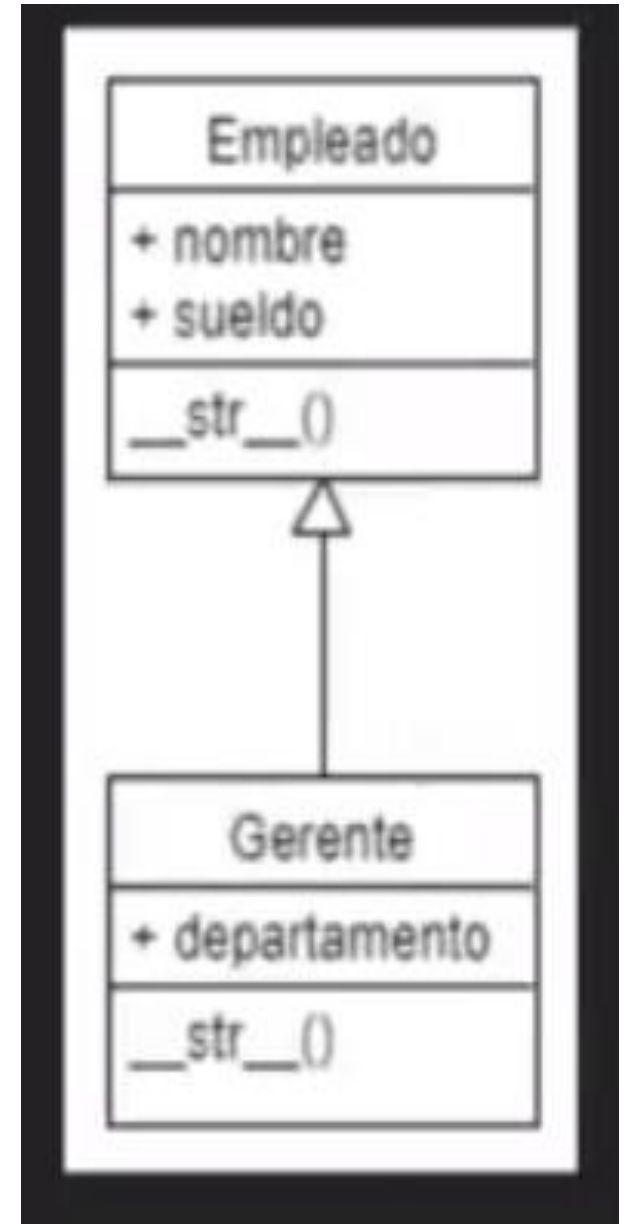
```
    def __init__(self, nombre, sueldo):  
        self.nombre = nombre  
        self.sueldo = sueldo
```

```
    def __str__(self):  
        return f'Empleado: [Nombre: {self.nombre}, Sueldo: {self.sueldo}]'
```

```
class Gerente(Empleado):
```

```
    def __init__(self, nombre, sueldo, departamento):  
        super().__init__(nombre, sueldo)  
        self.departamento = departamento
```

```
    def __str__(self):  
        return f'Gerente [Departamento: {self.departamento}] {super().__str__()}'
```



```
def imprimir_detalle(objeto):  
    print(type(objeto))  
    print(objeto)
```

```
empleado = Empleado('Juan', 5000)  
imprimir_detalle(empleado)
```

```
gerente = Gerente('Karla', 6000, 'Sistemas')  
imprimir_detalle(gerente)
```

```
<class '__main__.Empleado'>
```

```
Empleado: [Nombre: Juan, Sueldo: 5000]
```

```
<class '__main__.Gerente'>
```

```
Gerente [Departamento: Sistemas] Empleado: [Nombre: Karla, Sueldo: 6000]
```

```
if __name__ == '__main__':
```

# Ejercicios

1. Declarar la clase “Figura” con los métodos:

- Perímetro
- Área

Implementando los métodos de la clase “Figura” en las clases:

- **Triangulo**, con los atributos de instancia:
  - Lado\_01
  - Lado\_02
  - Lado\_03

Nota: Los valores de instancia deben inicializarse en el constructor con valores ingresados por teclado

- El método **perímetro** se implementa retornando como valor la suma de los lados ingresados por teclado
- El método **Área** se implementa mediante la siguiente expresión:

Donde p es la semisuma de los lados a, b y c de un triangulo  $A = \sqrt{p(p - a)(p - b)(p - c)}$

- **Circulo**, con el atributo de instancia:

- Radio

Nota: El valor de la instancia “Radio” debe de inicializarse en el constructor asignándole un valor ingresado por teclado

- El método perímetro se implementa retornando como valor  $2\pi r$ , donde “r” es el radio del circulo.

- El método Área se implementa mediante la siguiente expresión:

$$A = 2\pi r^2$$

Declarar y crear un objeto de cada uno, luego imprimir un método que reporte todos los datos de los objetos.