

Untitled

```
# Practica 5. Patrones de diseño

class Logger:
    _instancia = None

    def __new__(cls):
        if cls._instancia is None:
            cls._instancia = super().__new__(cls)
            cls._instancia.archivo = open("app.log", "a")
        return cls._instancia

    def log(self, mensaje):
        self.archivo.write(mensaje)
        self.archivo.flush()

logger1 = Logger()
logger2 = Logger()

logger1.log("\nInicio de sesion en la aplicacion.")
logger2.log("\nEl usuario se autentico")

print(logger1 is logger2)  # True → es la misma instancia

class Presidente:
    _instancia = None

    def __new__(cls, nombre):
        if cls._instancia is None:
            cls._instancia = super().__new__(cls)
            cls._instancia.historial = []
            # inicializamos nombre la primera vez
            cls._instancia.nombre = nombre
        return cls._instancia

    def __init__(self, nombre):
        # Solo actualiza si nunca se puso
        if not hasattr(self, "nombre"):
            self.nombre = nombre

    def accion(self, accion):
        evento = f"{self.nombre} {accion}"
```

```

        self.historial.append(evento)
        print(evento)

p1 = Presidente("Amlo")
p2 = Presidente("Peña Nieto")
p3 = Presidente("Fox")

p1.accion("Firmo decreto")
p2.accion("Visito España")
p3.accion("Aprobo un presupuesto")

print("\nHistorial del presidente")
print(p1.historial)

```

```

True
Amlo Firmo decreto
Amlo Visito España
Amlo Aprobo un presupuesto

Historial del presidente
['Amlo Firmo decreto', 'Amlo Visito España', 'Amlo Aprobo un presupuesto']

```