1. **SOLID – L – Liskov Substitution**

class MetodoPagoBase:

    def procesar\_pago():

        pass

class MetodoPagoAutomatico(MetodoPagoBase):

    def procesar\_pago(self, cantidad):

        pass

class MetodoPagoManual(MetodoPagoBase):

    def procesar\_pago(self, cantidad):

        pass

# Metodos de pago automaticos

class PagoTarjeta(MetodoPagoAutomatico):

    def \_\_init\_\_(self, numero\_tarjeta):

        self.numero\_tarjeta = numero\_tarjeta

    def procesar\_pago(self, cantidad):

        print(

            f"Procesando pago automatico de {cantidad} usando tarjeta { self.numero\_tarjeta}"

        )

class PagoPayPal(MetodoPagoAutomatico):

    def \_\_init\_\_(self, cuenta\_paypal):

        self.cuenta\_paypal = cuenta\_paypal

    def procesar\_pago(self, cantidad):

        print(

            f"Procesando pago automatico de {cantidad} usando PayPal cuenta { self.cuenta\_paypal}"

        )

class PagoBitcoin(MetodoPagoAutomatico):

    def \_\_init\_\_(self, direccion\_bitcoin):

        self.direccion\_bitcoin = direccion\_bitcoin

    def procesar\_pago(self, cantidad):

        print(

            f"Procesando pago automatico de {cantidad} usando Bitcoin { self.direccion\_bitcoin}"

        )

# Metodos de pago manual

class PagoCheque(MetodoPagoManual):

    def \_\_init\_\_(self, numero\_cheque):

        self.numero\_cheque = numero\_cheque

    def procesar\_pago(self, cantidad):

        print(

            f"Procesando pago manual de {cantidad} usando cheque {self.numero\_cheque}"

        )

def realizar\_pago\_automatico(metodo\_pago: MetodoPagoAutomatico, cantidad):

    metodo\_pago.procesar\_pago(cantidad)

def realizar\_pago\_manual(metodo\_pago: MetodoPagoManual, cantidad):

    metodo\_pago.procesar\_pago(cantidad)

# Instanciar las clases

pago\_tarjeta = PagoTarjeta("123 456 789 123")

pago\_paypal = PagoPayPal("mi\_cuenta@pago.com")

pago\_bitcoin = PagoBitcoin("892173912udfhie938")

pago\_cheque = PagoCheque("123456768")

realizar\_pago\_automatico(pago\_tarjeta, 1000)

realizar\_pago\_automatico(pago\_paypal, 400)

realizar\_pago\_automatico(pago\_bitcoin, 6000)

realizar\_pago\_manual(pago\_cheque, 3000)

class Figura:

    def obtener\_area(self):

        pass

class Rectangulo(Figura):

    def \_\_init\_\_(self, ancho, alto):

        self.ancho = ancho

        self.alto = alto

    def obtener\_area(self):

        return self.ancho \* self.alto

    def set\_ancho(self, ancho):

        self.ancho = ancho

    def set\_alto(self, alto):

        self.alto = alto

class Cuadrado(Figura):

    def \_\_init\_\_(self, lado):

        self.lado = lado

    def set\_lado(self, lado):

        self.lado = lado

    def obtener\_area(self):

        return self.lado \* self.lado

def imprimir\_area(figura: Figura):

    print(f"Area:{figura.obtener\_area()}")

rectangulo = Rectangulo(4, 5)

cuadrado = Cuadrado(4)

imprimir\_area(rectangulo)

imprimir\_area(cuadrado)

rectangulo.set\_alto(6)

rectangulo.set\_ancho(7)

imprimir\_area(rectangulo)

cuadrado.set\_lado(2)

imprimir\_area(cuadrado)