**Activity: Identifying and Refactoring Code to Follow SRP**

**Instructions:**

You are given a Python script that contains two implementations of an Order class:

1. One version **violates the Single Responsibility Principle (SRP)**.
2. The other version **follows SRP correctly**.

✅ **Your Task:**

* Analyze the code and **identify which part violates SRP**.
* Refactor the violating code to ensure that each class has a **single responsibility**.
* Submit your **refactored version** with an explanation of how it improves the design.

💡 **Hint:** Each class should have **only one reason to change**.

**📂 Filename:** python\_activity\_15.py

class Order:

def \_\_init\_\_(self, order\_id, items):

self.order\_id = order\_id

self.items = items

def calculate\_total(self):

total = sum(price for \_, price in self.items)

return total

def print\_invoice(self):

print(f"Invoice for Order ID: {self.order\_id}")

for item, price in self.items:

print(f"{item}: ${price}")

print(f"Total: ${self.calculate\_total()}")

def save\_to\_database(self):

print(f"Saving order {self.order\_id} to database...")

class OrderSRP:

def \_\_init\_\_(self, order\_id, items):

self.order\_id = order\_id

self.items = items

class OrderCalculator:

@staticmethod

def calculate\_total(order):

return sum(price for \_, price in order.items)

class InvoicePrinter:

@staticmethod

def print\_invoice(order):

print(f"Invoice for Order ID: {order.order\_id}")

for item, price in order.items:

print(f"{item}: ${price}")

print(f"Total: ${OrderCalculator.calculate\_total(order)}")

class OrderRepository:

@staticmethod

def save\_to\_database(order):

print(f"Saving order {order.order\_id} to database...")

print("=== Testing SRP-Violating Code ===")

order1 = Order(1, [("Laptop", 1200), ("Mouse", 50)])

order1.print\_invoice()

order1.save\_to\_database()

print("\n=== Testing SRP-Compliant Code ===")

order2 = OrderSRP(2, [("Keyboard", 100), ("Monitor", 300)])

InvoicePrinter.print\_invoice(order2)

OrderRepository.save\_to\_database(order2)