

```

1 class Item:
2     def __str__(self):
3
4
50
51     def __repr__(self):
52         """Method __repr__() - แสดงข้อมูลสำหรับ debugging"""
53         return f"Item(name='{self.__name}', price={self.__price}, quantity={self.__quantity})"
54
55 # ตัวอย่างการใช้งาน Class Item
56 if __name__ == "__main__":
57     print("=== LAB 2: Enhanced Class Item Demo ===")
58
59     # สร้าง Object จาก Class Item
60     item1 = Item("iPhone 15")
61     item2 = Item("Samsung Galaxy")
62     item3 = Item() # ใช้ชื่อเริ่มต้น
63
64     print("=== 1. การตั้งค่าพื้นฐาน ===")
65     print(f"Item 1: {item1}")
66     print(f"Item 2: {item2}")
67     print(f"Item 3: {item3}")
68
69     print("\n=== 2. การตั้งค่าราคาและจำนวน ===")
70     item1.setPrice(25000.00)
71     item1.setQuantity(2)
72     item2.setPrice(22000.00)
73     item2.setQuantity(1)
74     item3.setName("iPad Pro")
75     item3.setPrice(35000.00)
76     item3.setQuantity(1)
77
78     print(f"Item 1: {item1}")
79     print(f"Item 2: {item2}")
80     print(f"Item 3: {item3}")
81
82     print("\n=== 3. การคำนวณราคารวม ===")
83     print(f"iPhone 15 ราคารวม: {item1.getTotalPrice():.2f} บาท")
84     print(f"Samsung Galaxy ราคารวม: {item2.getTotalPrice():.2f} บาท")
85     print(f"iPad Pro ราคารวม: {item3.getTotalPrice():.2f} บาท")
86
87     print("\n=== 4. การลดราคา ===")
88     print(f"iPhone 15 ก่อนลดราคา: {item1.getPrice():.2f} บาท")
89     item1.applyDiscount(10) # ลด 10%
90     print(f"iPhone 15 หลังลดราคา 10%: {item1.getPrice():.2f} บาท")
91
92     item2.applyDiscount(15) # ลด 15%
93     print(f"Samsung Galaxy หลังลดราคา 15%: {item2.getPrice():.2f} บาท")
94
95     print("\n=== 5. การแสดงผลด้วย __str__ และ __repr__ ===")
96     print(f"__str__: {item1}")
97     print(f"__repr__: {repr(item1)}")

```

```

56 if __name__ == "__main__":
57
92     item2.applyDiscount(15) # ลด 15%
93     print(f"Samsung Galaxy หลังลดราคา 15%: {item2.getPrice():.2f} บาท")
94
95     print("\n=== 5. การแสดงผลด้วย __str__ และ __repr__ ===")
96     print(f"__str__: {item1}")
97     print(f"__repr__: {repr(item1)}")
98
99     print("\n=== 6. การทดสอบ Error Handling ===")
100     try:
101         item1.setPrice(-100) # ราคาติดลบ
102     except ValueError as e:
103         print(f"Error: {e}")
104
105     try:
106         item1.setQuantity(-5) # จำนวนติดลบ
107     except ValueError as e:
108         print(f"Error: {e}")
109
110     try:
111         item1.applyDiscount(150) # เปอร์เซ็นต์เกิน 100
112     except ValueError as e:
113         print(f"Error: {e}")
114
115     print("\n=== 7. การทดสอบ Private Attribute ===")
116     try:
117         print(f"พยายามเข้าถึง __price โดยตรง: {item1.__price}")
118     except AttributeError as e:
119         print(f"Error: {e}")
120     print("ไม่สามารถเข้าถึง private attribute __price ได้โดยตรง")
121     print("ต้องใช้ getPrice() method แทน")
122
123     print("\n=== 8. สรุปข้อมูลสินค้าทั้งหมด ===")
124     items = [item1, item2, item3]
125     total_value = 0
126     for i, item in enumerate(items, 1):
127         print(f"สินค้า {i}: {item}")
128         total_value += item.getTotalPrice()
129
130     print(f"\nมูลค่ารวมทั้งหมด: {total_value:.2f} บาท")
131

```

```

1 class Item:
2     def __init__(self, name="Unknown Item"):
3         """Constructor - กำหนดค่าเริ่มต้น price เป็น 0"""
4         self._price = 0.0 # private attribute
5         self._name = name # private attribute กำหนดชื่อสินค้า
6         self._quantity = 1 # private attribute กำหนดจำนวน
7
8     def getPrice(self):
9         """Method getPrice() - return ค่า price กลับมา"""
10        return self._price
11
12    def setPrice(self, p):
13        """Method setPrice(float p) - รับ float และกำหนด price"""
14        if p < 0:
15            raise ValueError("กำหนดค่าเป็นบวก")
16        self._price = float(p)
17
18    def getName(self):
19        """Method getName() - return ชื่อสินค้า"""
20        return self._name
21
22    def setName(self, name):
23        """Method setName() - ตั้งชื่อสินค้า"""
24        self._name = str(name)
25
26    def getQuantity(self):
27        """Method getQuantity() - return จำนวนสินค้า"""
28        return self._quantity
29
30    def setQuantity(self, qty):
31        """Method setQuantity() - ตั้งจำนวนสินค้า"""
32        if qty < 0:
33            raise ValueError("จำนวนต้องเป็นจำนวน")
34        self._quantity = int(qty)
35
36    def getTotalPrice(self):
37        """Method getTotalPrice() - คำนวณรวม (ราคา x จำนวน)"""
38        return self._price * self._quantity
39
40    def applyDiscount(self, discount_percent):
41        """Method applyDiscount() - ลดราคาเป็นเปอร์เซ็นต์"""
42        if 0 <= discount_percent <= 100:
43            self._price = self._price * (1 - discount_percent / 100)
44        else:
45            raise ValueError("เปอร์เซ็นต์ลดราคาต้องอยู่ระหว่าง 0-100")
46
47    def __str__(self):
48        """Method __str__() - แสดงข้อมูลสินค้า"""
49        return f"Item: {self._name}, Price: {self._price:.2f}, Quantity: {self._quantity}, Total: {self.getTotalPrice():.2f}"

```

LAB : สร้าง Class (ด้วย AI)

จงสร้าง Class ใหม่ชื่อ Item โดยมีข้อกำหนดดังต่อไปนี้:

ฉันจะช่วยให้คุณสร้าง Class Item ตามข้อกำหนดที่ใหมาในไฟล์ lab2.py ครับ