

John Rosedel R. Cortan

IT0011 – TW23

Act 6: OOP and Exception

Source Code:

```
CortanAct6.py U X
ACT6 > CortanAct6.py > ...
1  class Item:
2      def __init__(self, item_id, name, description, price):
3          self.item_id = item_id
4          self.name = name
5          self.description = description
6          self.price = price
7
8  class ItemManager:
9      def __init__(self):
10         self.items = {}
11
12     def create_item(self, item_id, name, description, price):
13         if item_id in self.items:
14             print("Error: Item ID already exists.")
15             return
16         if price < 0:
17             print("Error: Price cannot be negative.")
18             return
19         self.items[item_id] = Item(item_id, name, description, price)
20         print("Item added successfully.")
21
22     def read_item(self, item_id):
23         item = self.items.get(item_id)
24         if item:
25             print(f"ID: {item.item_id}, Name: {item.name}, Description: {item.description}, Price: {item.price}")
26         else:
27             print("Error: Item not found.")
28
29     def update_item(self, item_id, name=None, description=None, price=None):
30         item = self.items.get(item_id)
31         if not item:
32             print("Error: Item not found.")
33             return
34         if price is not None and price < 0:
35             print("Error: Price cannot be negative.")
36             return
37         if name:
38             item.name = name
39         if description:
40             item.description = description
41         if price is not None:
42             item.price = price
43         print("Item updated successfully.")
44
45     def delete_item(self, item_id):
46         if item_id in self.items:
47             del self.items[item_id]
48             print("Item deleted successfully.")
49         else:
```

```

50         print("Error: Item not found.")
51
52     def list_items(self):
53         if not self.items:
54             print("No items available.")
55             return
56         for item in self.items.values():
57             print(f"ID: {item.item_id}, Name: {item.name}, Description: {item.description}, Price: {item.price}")
58
59
60     def menu():
61         manager = ItemManager()
62         while True:
63             print("\nItem Management System")
64             print("1. Add Item")
65             print("2. View Item")
66             print("3. Update Item")
67             print("4. Delete Item")
68             print("5. List All Items")
69             print("6. Exit")
70
71             choice = input("Enter your choice: ")
72
73             if choice == '1':
74                 try:
75                     item_id = input("Enter item ID: ")
76                     name = input("Enter item name: ")
77                     description = input("Enter item description: ")
78                     price = float(input("Enter item price: "))
79                     manager.create_item(item_id, name, description, price)
80                 except ValueError:
81                     print("Error: Invalid input for price.")
82             elif choice == '2':
83                 item_id = input("Enter item ID: ")
84                 manager.read_item(item_id)
85             elif choice == '3':
86                 item_id = input("Enter item ID: ")
87                 name = input("Enter new name (or leave blank): ") or None
88                 description = input("Enter new description (or leave blank): ") or None
89                 try:
90                     price_input = input("Enter new price (or leave blank): ")
91                     price = float(price_input) if price_input else None
92                 except ValueError:
93                     print("Error: Invalid input for price.")
94                     continue
95                 manager.update_item(item_id, name, description, price)
96             elif choice == '4':
97                 item_id = input("Enter item ID: ")
98                 manager.delete_item(item_id)
99             elif choice == '5':
100                 manager.list_items()
101             elif choice == '6':
102                 print("Exiting program.")
103                 break
104             else:
105                 print("Invalid choice. Please enter a number between 1 and 6.")
106
107     menu()
108

```

Result:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

PS C:\Users\John John\OneDrive\Documents\GitHub\it0011_Cortan> python -u "c:\Users\John John\OneDrive\Documents\GitHub\it0011_Cortan\ACT6\CortanAct6.py"

Item Management System

1. Add Item
2. View Item
3. Update Item
4. Delete Item
5. List All Items
6. Exit

Enter your choice: 1

Enter item ID: 88

Enter item name: Paranoise Airforce 1

Enter item description: Gdragon Shoes

Enter item price: 101

Item added successfully.

Item Management System

1. Add Item
2. View Item
3. Update Item
4. Delete Item
5. List All Items
6. Exit

Enter your choice: 1

Enter item ID: 23

Enter item name: Fragment Jordan 1

Enter item description: Travis Scott Shoes

Enter item price: 202

Item added successfully.

Item Management System

1. Add Item
2. View Item
3. Update Item
4. Delete Item
5. List All Items
6. Exit

Enter your choice: 1

Enter item ID: 35

Enter item name: KD 4 ThunderStorm

Enter item description: Kevin Durant Shoes

Enter item price: 350

Item added successfully.

Item Management System

1. Add Item
2. View Item
3. Update Item
4. Delete Item
5. List All Items

```
Enter your choice: 2
Enter item ID: 88
ID: 88, Name: Paranoise Airforce 1, Description: Gdragon Shoes, Price: 101.0

Item Management System
1. Add Item
2. View Item
3. Update Item
4. Delete Item
5. List All Items
6. Exit
Enter your choice: 3
Enter item ID: 35
Enter new name (or leave blank): KD 4 MVP
Enter new description (or leave blank): Kevin Durant Shoes
Enter new price (or leave blank): 355
Item updated successfully.

Item Management System
1. Add Item
2. View Item
3. Update Item
4. Delete Item
5. List All Items
6. Exit
Enter your choice: 4
Enter item ID: 35
Item deleted successfully.

Item Management System
1. Add Item
2. View Item
3. Update Item
4. Delete Item
5. List All Items
6. Exit
Enter your choice: 5
ID: 88, Name: Paranoise Airforce 1, Description: Gdragon Shoes, Price: 101.0
ID: 23, Name: Fragment Jordan 1, Description: Travis Scott Shoes, Price: 202.0

Item Management System
1. Add Item
2. View Item
3. Update Item
4. Delete Item
5. List All Items
6. Exit
Enter your choice: 6
Exiting program.
PS C:\Users\John John\OneDrive\Documents\GitHub\it0011_Cortan>
```