

Problem Statement 1:

Design a pseudocode algorithm for a simple inventory management system. The system should allow users to add new items, update quantities, and generate reports. Implement functions for each operation, and incorporate error handling using exceptions.

Solution :

//Initialize an empty inventory dictionary to store items and quantities

inventory = {}

//Function to add a new item to the inventory

function add_item(item_name, quantity):

try:

//check if the item already exists in the inventory

if item_name in inventory:

// If yes, update the quantity

inventory[item_name] +=quantity

else:

// If no, add the new item to the inventory

inventory[item_name] = quantity

print(f"{quantity} units of {item_name} added to the inventory)

except Exception as e:

print(f"Error: {e}")

//Function to update the quantity of an existing item in the inventory

function update_quantity(item_name,new_quantity):

try:

//check if the item exists in the inventory

if item_name in inventory:

//Update the quantity

inventory[item_name] = new_quantity

print(f"Quantity of {item_name} updated to {new_quantity}.")

else:

print(f"Error: {item_name} not found in the inventory.")

except Exception as e:

print(f"Error: {e}")

//Function to generate a report of the current inventory

function generate_report():

try:

//check if the inventory is not empty

if inventory:

print("Current Inventory Report: ")

```
        for item, quantity in inventory.items():
            print(f"{item}: {quantity} units")
    else:
        print("Inventory is empty.")
except Exception as e:
    print(f"Error: {e}")
```

```
/* Example
add_item("ProductA",100)
add_item("ProductB",200)
generate_report()
update_quantity("ProductA",50)
generate_report() */
```