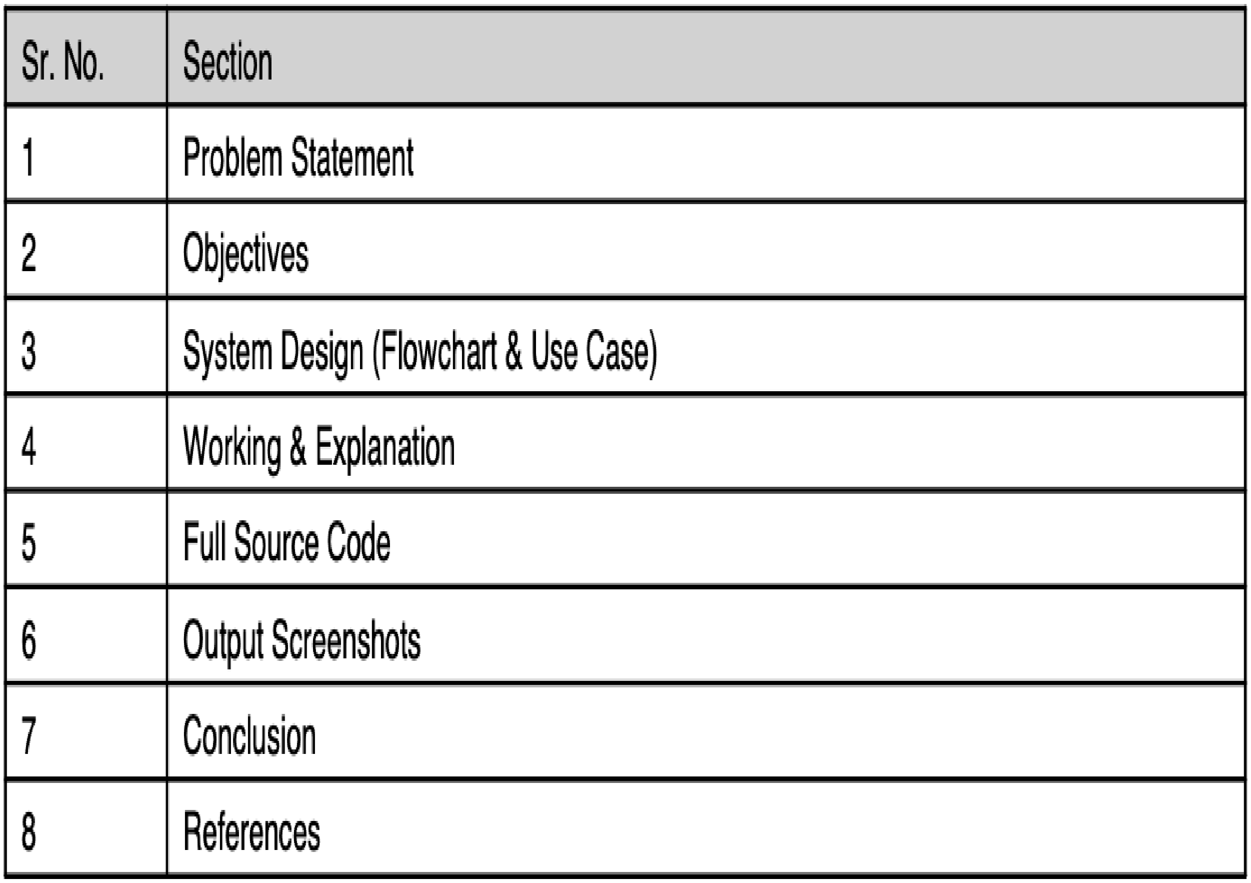
Simple Inventory and Billing Management System Using Python

Mini Project Report

Submitted By: Aarav Sharma
Reg. No.: 25BAI10617
Under The Guidance Of:
Dr MK Jayanthi

Index



1. Problem Statement

Small retail shops still use manual registers, leading to calculation errors, wrong GST, forgotten stock updates, and wasted time. A free, simple, offline digital solution was needed

2. Objectives

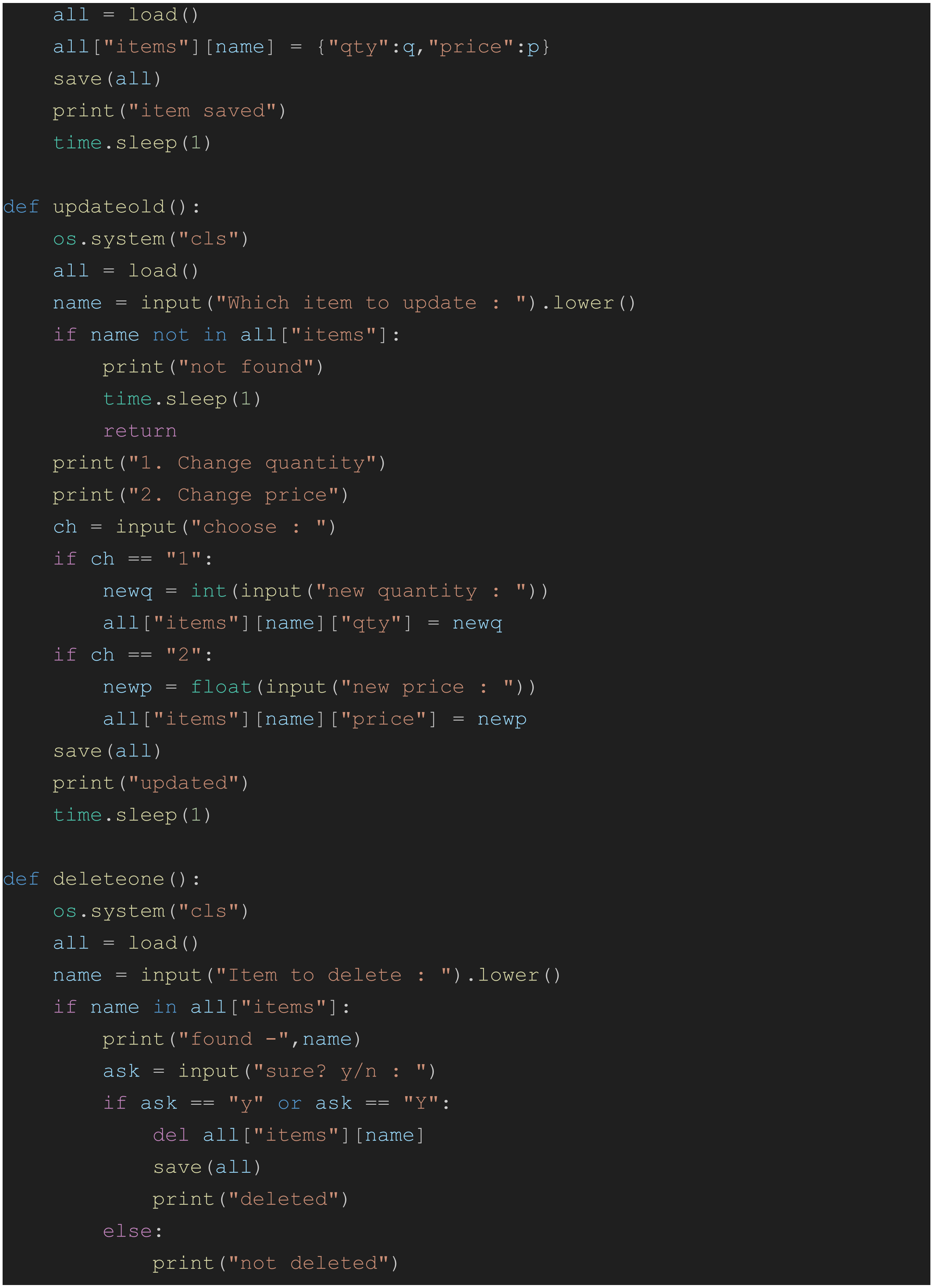
Create a lightweight Python program to add/update/delete/search items, generate bills with 18% GST, auto-reduce stock after sales, and permanently save data in a JSON file with zero external dependencies

4. Working & Explanation

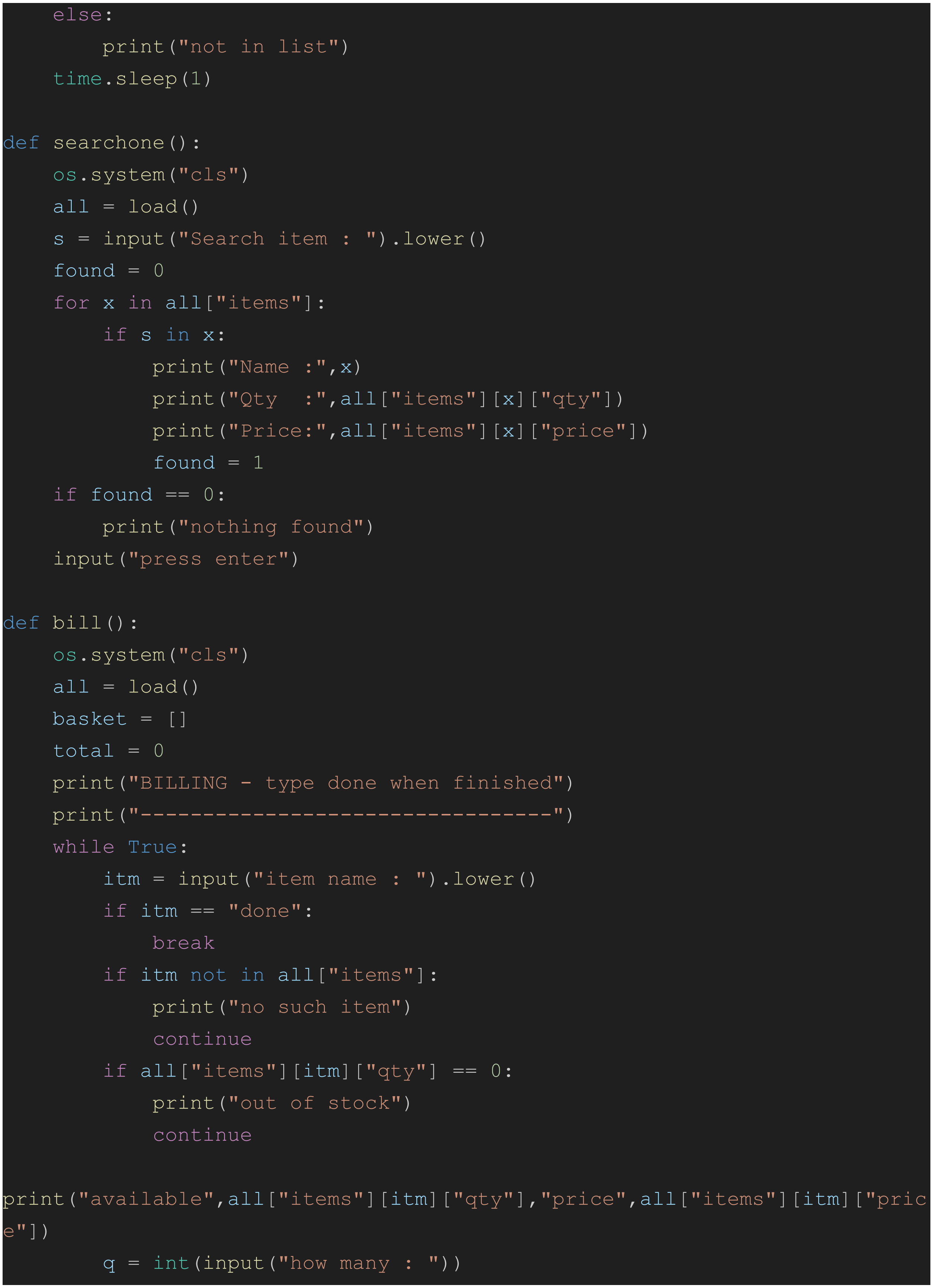
The program loads data from inventory.json (creates if missing), displays a menu, calls the chosen function, updates stock during billing, and saves everything back to the JSON file instantly. Entire logic uses only built-in Python features.

5. Full Source Code

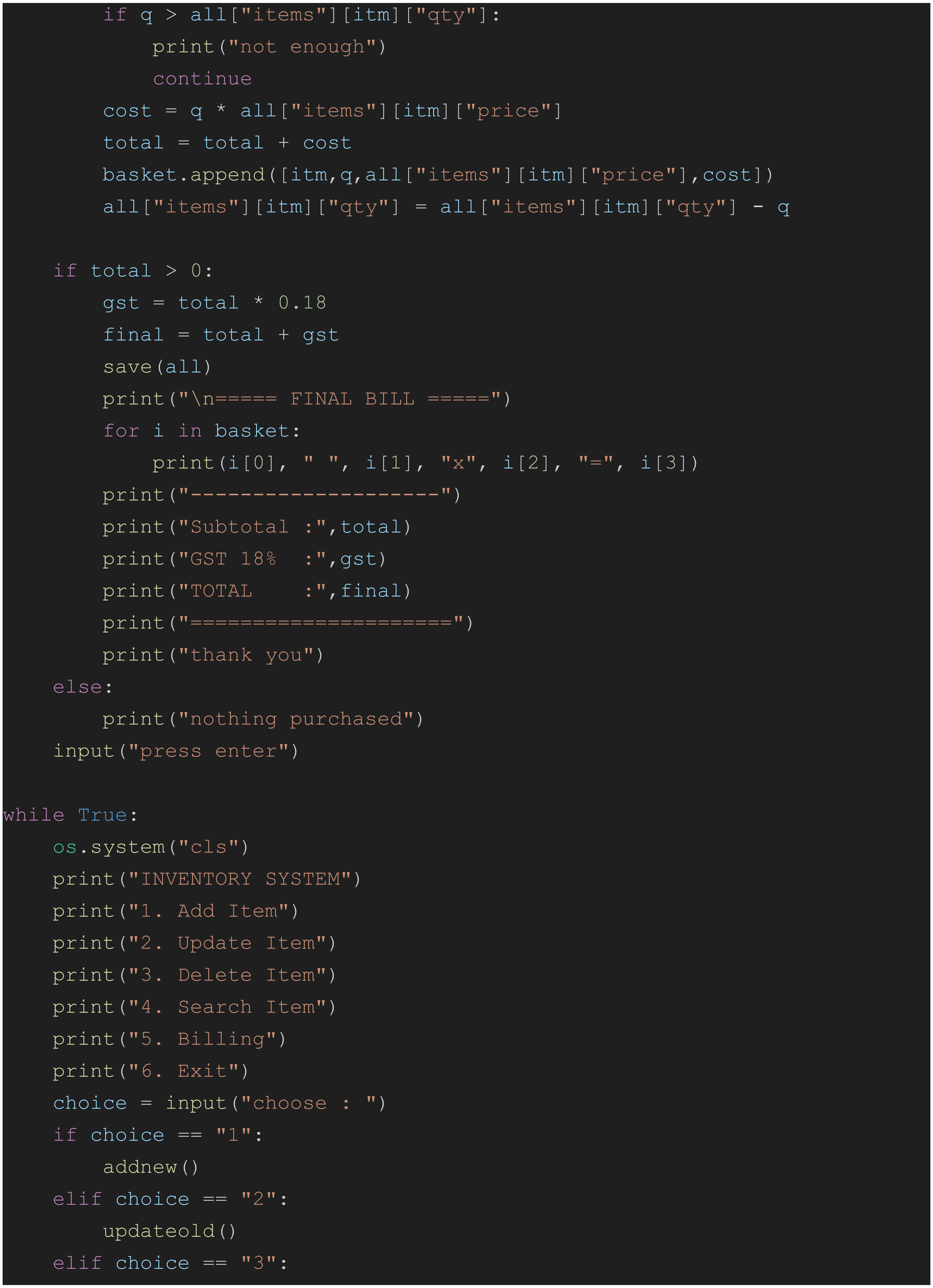
```
import json
import os
import time

datafile = "inventory.json"

def load():
    if os.path.exists(datafile):
        f = open(datafile)
        x = json.load(f)
        f.close()
        return x
    else:
        return {"items":{}}

def save(alldata):
    f = open(datafile,"w")
    json.dump(alldata,f,indent=2)
    f.close()

def addnew():
    os.system("cls")
    print("ADD NEW ITEM")
    print("------------")
    name = input("Item name : ").lower().strip()
    if name == "":
        print("cant be blank")
        time.sleep(1)
        return
    q = int(input("Quantity  : "))
    p = float(input("Price     : "))
```



```
    all = load()
    all["items"][name] = {"qty":q,"price":p}
    save(all)
    print("item saved")
    time.sleep(1)

def updateold():
    os.system("cls")
    all = load()
    name = input("Which item to update : ").lower()
    if name not in all["items"]:
        print("not found")
        time.sleep(1)
        return
    print("1. Change quantity")
    print("2. Change price")
    ch = input("choose : ")
    if ch == "1":
        newq = int(input("new quantity : "))
        all["items"][name]["qty"] = newq
    if ch == "2":
        newp = float(input("new price : "))
        all["items"][name]["price"] = newp
    save(all)
    print("updated")
    time.sleep(1)

def deleteone():
    os.system("cls")
    all = load()
    name = input("Item to delete : ").lower()
    if name in all["items"]:
        print("found -",name)
        ask = input("sure? y/n : ")
        if ask == "y" or ask == "Y":
            del all["items"][name]
            save(all)
            print("deleted")
        else:
            print("not deleted")
```



```
    else:
        print("not in list")
    time.sleep(1)

def searchone():
    os.system("cls")
    all = load()
    s = input("Search item : ").lower()
    found = 0
    for x in all["items"]:
        if s in x:
            print("Name :",x)
            print("Qty  :",all["items"][x]["qty"])
            print("Price:",all["items"][x]["price"])
            found = 1
    if found == 0:
        print("nothing found")
    input("press enter")

def bill():
    os.system("cls")
    all = load()
    basket = []
    total = 0
    print("BILLING - type done when finished")
    print("---------------------------------")
    while True:
        itm = input("item name : ").lower()
        if itm == "done":
            break
        if itm not in all["items"]:
            print("no such item")
            continue
        if all["items"][itm]["qty"] == 0:
            print("out of stock")
            continue

print("available",all["items"][itm]["qty"],"price",all["items"][itm]["price"])
        q = int(input("how many : "))
```



```python
        if q > all["items"][itm]["qty"]:
            print("not enough")
            continue
        cost = q * all["items"][itm]["price"]
        total = total + cost
        basket.append([itm,q,all["items"][itm]["price"],cost])
        all["items"][itm]["qty"] = all["items"][itm]["qty"] - q

    if total > 0:
        gst = total * 0.18
        final = total + gst
        save(all)
        print("\n===== FINAL BILL =====")
        for i in basket:
            print(i[0], " ", i[1], "x", i[2], "=", i[3])
        print("--------------------")
        print("Subtotal :",total)
        print("GST 18%  :",gst)
        print("TOTAL    :",final)
        print("=====================")
        print("thank you")
    else:
        print("nothing purchased")
    input("press enter")

while True:
    os.system("cls")
    print("INVENTORY SYSTEM")
    print("1. Add Item")
    print("2. Update Item")
    print("3. Delete Item")
    print("4. Search Item")
    print("5. Billing")
    print("6. Exit")
    choice = input("choose : ")
    if choice == "1":
        addnew()
    elif choice == "2":
        updateold()
    elif choice == "3":
        deleteone()
    elif choice == "4":
        searchone()
    elif choice == "5":
        bill()
    elif choice == "6":
        print("bye")
        time.sleep(1)
        break
    else:
        print("wrong input")
        time.sleep(1)
```

7. Conclusion

The project fully achieves its goals. A single small Python script successfully replaces the traditional register, making inventory and billing accurate, fast, and completely free for any small shop owner

8. References

- Python Official Documentation — https://docs.python.org
- NumPy Documentation — https://numpy.org
- ReportLab PDF Library — https://www.reportlab.com
- Stack Overflow (general debugging help)