K.RAMAKRISHNAN

COLLEGE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTION)

SAMAYAPURAM, TRICHY-621 112

Practical Record Note

Name : NANTHINI.S

Register Number : 2303811710422101

Subject code/name : Laboratory

Programme :

Certified that this is a bonafide record of work done by **NANTHINI.S** of _____ Semester in **Python Programming - I Year - II Sem - Project Module** Laboratory during the academic year 2023-2024

His/Her University Register Number is **2303811710422101**

Aim:

Project Module.

Program:

CTP28132.py

```
class PharmacyManagementSystem:
    def __init__(self):
        self.inventory = {}
        self.prescriptions = []
        self.customers = {}
        self.invoices = []
        self.reports = {}

    # Inventory Management
    def add_medication(self):
        medication_name = input("Enter medication name: ")
        quantity = int(input("Enter quantity: "))
        if medication_name in self.inventory:
            self.inventory[medication_name] += quantity
        else:
            self.inventory[medication_name] = quantity
        print(f"{quantity} units of {medication_name} added to inventory.")

    # Prescription Handling
    def process_prescription(self):
        patient_name = input("Enter patient name: ")
        medication_name = input("Enter medication name: ")
        quantity = int(input("Enter quantity: "))
        if medication_name not in self.inventory or
self.inventory[medication_name] < quantity:
            print("Insufficient stock to fulfill the prescription.")
            return
        prescription = {"patient_name": patient_name, "medication":
medication_name, "quantity": quantity}
        self.prescriptions.append(prescription)
        self.inventory[medication_name] -= quantity
        print("Prescription processed successfully.")
   # Customer Management
    def add_customer(self):
        customer_id = input("Enter customer ID: ")
        name = input("Enter customer name: ")
        address = input("Enter customer address: ")
        self.customers[customer_id] = {"name": name, "address": address}
        print("Customer added successfully.")

    # Billing and Invoicing
    def generate_invoice(self):
        if not self.prescriptions:
            print("No prescriptions to generate invoice for.")
            return
        prescription = self.prescriptions[-1]
        invoice = {
            "patient_name": prescription["patient_name"],
            "medication": prescription["medication"],
            "quantity": prescription["quantity"],
            "total_cost": prescription["quantity"] * 10  # Assuming each unit
costs $10 for simplicity
        }
        self.invoices.append(invoice)
        print(f"Invoice generated successfully:\n{invoice}")

    # Reporting and Analytics
```

```
    def generate_report(self):
        report_type = input("Enter report type (Sales/Inventory/Customer): ")
        if report_type == "Sales":
            report = {"total_prescriptions": len(self.prescriptions),
"total_invoices": len(self.invoices)}
        elif report_type == "Inventory":
            report = {"inventory_status": self.inventory}
        elif report_type == "Customer":
            report = {"total_customers": len(self.customers)}
        else:
            print("Invalid report type.")
            return
        self.reports[report_type] = report
        print(f"{report_type} report generated successfully:\n{report}")
# Main menu for user interaction
def main_menu():
    pharmacy_system = PharmacyManagementSystem()
    while True:
        print("\nPharmacy Management System")
        print("1. Add Medication")
        print("2. Process Prescription")
        print("3. Add Customer")
        print("4. Generate Invoice")
        print("5. Generate Report")
        print("6. Exit")
        choice = input("Enter your choice: ")

        if choice == '1':
            pharmacy_system.add_medication()
        elif choice == '2':
            pharmacy_system.process_prescription()
        elif choice == '3':
            pharmacy_system.add_customer()
        elif choice == '4':
            pharmacy_system.generate_invoice()
        elif choice == '5':
            pharmacy_system.generate_report()
        elif choice == '6':
            break
        else:
            print("Invalid choice. Please try again.")

if __name__ == "__main__":
    main_menu()
```


Output:

Test case - 1
User Output
Hello World
Hello World

Result:

Thus the above program is executed successfully and the output has been verified