Date: 10/9/25

Ex. No: 3.c Developing functions models within programming a banking transaction system.

Aim: To develop a python program using functions, that simulates basic banking operations deposit, withdraw, and checking the account.

Algorithm:-

1. Initialize account balance to zero.
2. Define a function to deposit money which increases the balance.
3. Define a function to withdraw money, checking if the balance is sufficient.
4. Define a function to display the current balance.
5. Use menu-driven options to perform deposit, withdraw and balance check actions.

Program:-

```
balance = 0
def deposit (amount)
  global balance
  balance += amount
    print(" Deposited:", amount)
def withdraw (amount):
   global balance:
   if amount <= balance:
     balance -= amount
```

output:

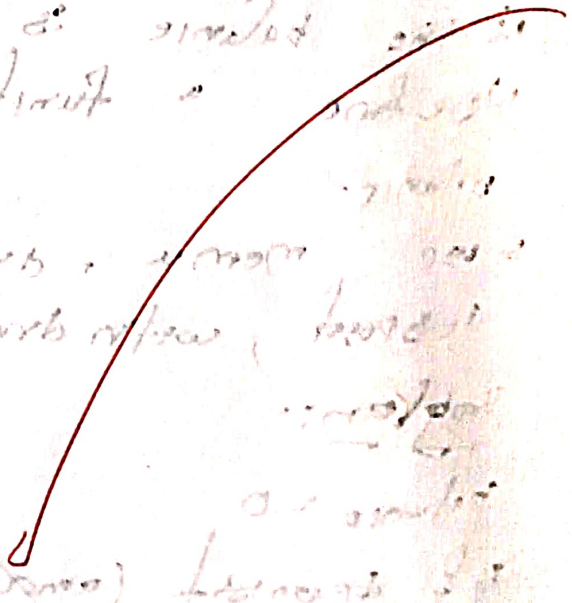
Deposite : 500

withdraw : 200

current Balance : 300

insufficient Balance

current balance : 300



```
        print("Insufficient balance")
    def check_balance(self):
        print("Current balance:", self.balance)
# Example usage
account = BankAccount(100)
account.deposit(50)
account.withdraw(30)
print(account.balance)
account.withdraw(200)
account.check_balance()
```

Result: The program performs banking transactions using functions and maintains the account balance accurately.

5. Student Result calculator

Aim:- To create a python program using functions to accept marks of three subjects. calculate total average, grade and display.

Algorithm:-

1) Define a function to accept marks for three subjects.
2. Define a function to calculate the total and average.
3. Define a function to determine the grade (A/B/C/Fail) based on average.
4. Define a separate function to display the result.

Program:-

```
def accept_marks():
    m1 = int(input("Enter marks for subject1:"))
    m2 = int(input("Enter marks for subject2:"))
    m3 = int(input("Enter marks for subject3:"))
    return m1, m2, m3
def calculate_result(m1, m2, m3):
    total = m1+m2+m3
    average = total/3
    if average >= 75
        grade = 'A'
    else if average >= 60
        grade = 'B'
```

output:-

Enter marks for subject 1: 80
Enter marks for subject 2: 70
Enter marks for subject 3: 60
total marks: 210
Average marks: 70.0
Grade: B

```
    elif average >= 40
        grade = 'c'
    else
        grade = 'fail'
    return total, average, grade,
def display_result(total marks:", total)
    print("total marks:", total)
    print("Average marks:", average)
    print("Grade:", grade)
marks = accept_marks()
total, average, grade = calculate_result(marks)
display_result(total, average, grade).
```

VEL TECH	
EX No.	7
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	25
SIGN WITH DATE	

Result :- The program uses functions to process student mark and displays a result, including total, average, and grade classification.