Task 7 :-

Utilize 'functions' concepts in python programming

Aim:- To write the python program using 'functions' concepts in python programing

7.1 you are developing a small python script to analyze and manipulate a list of student grades for a class project. write a python program that satisfies the above requirements using the built-in functions print(), len(), type(), max(), min(), sorted(), reversed(), and range().

Algorithm:-

1. start the program
2. print a welcome message: outputs a simple greeting.
3. determine and print the number of students: uses len() to find the number of elements in the student_names list.
4. print the type of lists: uses type() to show the type of the student_names and student_grades lists.
5. find and print highest and lowest grades: uses max() and min() to determine the highest and lowest values in student_grades.
6. print sorted list of grades: uses sorted() to sort the grades
7. print reversed list of grades: uses reversed() to reverse the sorted list and converts it to a list.
8. Generate the print a range of grade indices: uses range() to create a list of indices from 1 to the number of students.
9. stop

Program:-

```
def analyze_student_grades():
#sample data
student_names = ["Alice", "Bob", "Charlie", "Diana"]
student_grades = [85, 92, 78, 90]
```

output:-

welcome to the student grades Analyzer!

number of students: 4

Type of student_names list: <class 'list'>

Type of student_grades list: <class 'list'>

Highest grade: 92

Lowest grade: 78

Sorted grades: [78, 85, 90, 92]

Reversed grades: [92, 90, 85, 78]

Grad indices from 1 to number of students: (1, 2, 3, 4)

```
#1. print a welcome message
print("welcome to the student Grades Analyzer\n")
# 2 determine and print the number of students
num students = len (student_names)
print("Number of students:", num_students)
#3 print the type of the students names list and the
grades list.
print("\n type of student_names list:", type (student_names))
print ("Type of student_grades list:", type (student_grades))
#4 find and print the highest and lowest grade
highest grade = max (student_grades)
lowest grade = min (student_grades)
print("\n highest grade:", highest_grade)
print("lowest grade:", lowest_grade)
#5. print the list of grades sorted in ascending order
sorted grades = sorted (student_grades)
print("\n sorted grades:", sorted_grades)
# 6. print the list of grades in reverse order
reversed_grades = list (reversed ( sorted_grades))
print("reversed grades:", reversed_grades)
#7 generate and print a range of grades indices
from 1 to the number of students.
grade_indices = list(range(1, num_students+1))
print("\n Grade indices from 1 to number of students:",
grade_indices)
#Run the analysis
analyze_student_grades()
```

7.2 you are tasked with creating a small calculator application to help users perform basic arithmetic operations and greet them with a personalized message. your application should perform the following tasks: addition, subtraction, multiplication, division.

Algorithm:—

1. Start the program
2. user input for numbers: The program prompts the user to enter two numbers.
3. user input for operations: the program prompts the user to choose an arithmetic operations (addition, subtraction, multiplication, division),
4. perform operation: Based on the user's choice, the program performs the chosen arithmetic operation using the defined functions.
5. Display result: the program displays the result of the operation.
6. stop.

7.2 Program:—

```
def add (a,b):
 """ Return the sum of two numbers. """

 return a+b
def subtract (a,b):
""" return the difference between two numbers. "
return a-b
def multiply(a,b):
" " return the product of two numbers. """
return a*b
def divide (a,b):
""" return the quotient of two numbers Handles
    division by zero""
if b!=0
   return a/b
else,
   return "Error: Division by zero"
def greet (name):
```

Output:—

Arithmetic operations:

Sum of 10 and 5 : 15

Difference between 10 and 5 : 5

product of 10 and 5 : 50

Quotient of 10 and 5 : 2.0

Greeting:

Hello, Alice : welcome to the program.

```
"""Return a greeting message for the user."""
return f"Hello, {name}! welcome to the program".
def main():
    # demonstrating the use of user-defined functions.
    # Arithmetic operations.
    num1 = 10
    num2 = 5
    print("Arithmetic operations:")
    print(f"Sum of {num1} and {num2}:", add(num1, num2))
    print(f" difference between (num1) and (num2) subtract
    (num1, num2))
    print(f" product of {num1} and {num2}:" multiply
    (num1, num2))
    print(f"quotient of {num1} and {num2}:" divide (num1,
    num2)
    # Greeting the user
    user_name = "Alice"
    print("\nGreeting:")
    print(greet(user_name))
    #Run the main function
    if __name__ == "__main__":
        main()
```

Result:— Thus the Python program using 'functions' concepts was successfully executed and the output was verified.

VEL TECH

EX No.	7
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	