Task No :- 7. Utilizing 'Functions' concepts in python

7.1 You are developing a small python script to analyze and manipulate a list of student grades for a class project.

Aim :- TO write the python program using Functions concepts in python programming.

Algorithm :-

1. Start the program.
2. Print 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. 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 and 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]
    # 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 student names list and
                                        the grades list
```

Print ("\n-Type of stu

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]

Grade indices from 1 to number of students:
[1, 2, 3, 4]

```
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("\nHighest grade:", highest_grade)
print("Lowest grade:", sorted_grade).
#5. Print the list of grades sorted in ascending order
sorted_grades = sorted(student_grades)
print("\nSorted 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 grade 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 tasks.

Algorithm:

1. Start the program.
2. User Input for Numbers: The program prompts the user to enter two numbers.
3. User Input for operation: The program prompts the user to choose an arithmetic operations.

4. perform operation: Based on the user's choice, the program performs the chosen arithmetic 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):
    """ 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
```

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

```
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.

VELTECH	
EX No.	1
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	
TOTAL (20)	20
SIGN WITH DATE	19/9