Utilizing 'Functions' concepts in python programming.

TASK 7

Date: 10-9-25

Aim: To write the python program using 'Functions' concepts in python programming

7.1 you are developing a small python script to analyze and manipulate 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
2. print a welcome message:
3. Determine and print number of students. Uses len() to find the number of elements in the student names list
4. Print sorted list of grades.
5. Print reversed list of grades. Uses reversed() to reverse the sorted list and convert it to a list
6. Generate and print a range indices. Uses range() to create a list of
7. Stop

output

Welcome to student grades Analyzer!

Number of student: 4

Type of student_names list: <class 'list'>

Type of student_grades list: <class 'list'>

Highest grade: 94

Lowest grade: 78

Sorted grade: [78, 85, 90, 94]

Reversed grades: [94, 90, 85, 78]

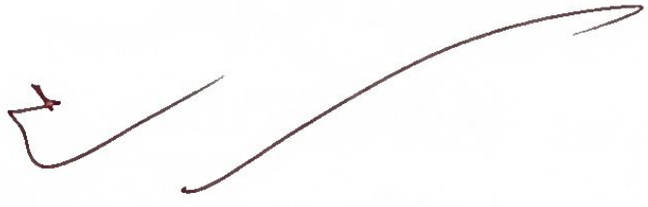
Grade indices from 1 to number of students

: [1, 2, 3, 4]

Program:

```
def analyze_student_grades():
    student_names = ["Alice", "Bob", "Charlie", "Diana"]
    student_grades = [85, 92, 72, 90]
print("Welcome to student Grade Analyzer!")
num_students = len(student_names)
print("Number of students:", num_students)
print("In Type of student names list:", type(student_
                                        name))
print("Type of student_grades list:", type(student_g
                                        ))
highest_grade = max(student_grade)
lowest_grade = min(student_grade)
print("In Highest grade:", highest_grade)
print("lowest grade:", lowest_grade)
reversed_grades = list(reversed(sorted_grades))
print("Reversed grades:", reversed_grades)
grade_indices = list(range(1, num_students+1))
print("In Grade indices from 1 to number of
        students:", grade_indices)

analyze_student_grades().
```



Result: Thus the python program for the function concept is successfully executed.

7.2 You are tasked with creating a simple calculator application to help users perform basic arithmetic operations and display them with a personalized message. Your application should perform the following tasks: addition, subtraction, multiplication, division.

Algorithm:

1. Start
2. User input for numbers.
3. User input for operation
4. Perform operation: Based on user's choice the program performs the chosen arithmetic operation using the defined function
5. Display result
6. Stop

Program:

```
def add(a,b):
    return a+b
def subtract(a,b):
    return a-b
def multiply(a,b):
    return a*b
def divide(a,b):
    if b!=0:
        return a/b
```

```
else:
    return "Error: Division by zero"

def greet(name):
    return f"Hello, {name}! Welcome to the Program"

def main():
Arithmetic operation
num1 = 10
num2 = 5
print("Arithmetic Operations:")
print(f"Sum of {num1} and {num2}:", add{num1, num2})
print(f"Difference of {num1} and {num2}:", subtract{num1,
print(f"multiply of {num1} and {num2}:", multiply{num1, num
print(f"Current {num1} and {num2}:", divide(num1, n

#Greeting
User_name = "Alice"
print("In Greeting:")
print(greet(User_name))

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

VEL TECH - CSE	
EX NO.	7
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	3
VIVA VOCE (3)	3
RECORD (4)	4
TOTAL (15)	15
SIGN WITH DATE	

Result:

Thus the python program using functions concept was successfully executed and output was verified