# Task:7 Utilizing 'Functions' concepts in Python Program.

**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 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 is 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: use 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 and print a range of grade indices: uses range() to create a list of indices from 1 to the number of

the 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 grades list.
Print("\nType 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:",lowest_grade)
#5. Print the list of grades sorted in ascending order.
Sorted_grades=
Sorted(student_grades)
```

Output:

Welcome to the student Grades Analyzer!

Number of Students: 4

Type of student_name 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("\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
grades_indices = list(range(1, num_students+1))
Print("\nGrade 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 operation: The program prompts the user to choose an arithmetic operation (addition, subtraction, multiplication, division).
4. Perform operation: Based on the user's choice, the operation

Program Performs the Choosen arithmetic operation using the defined functions.

5. Display Result: The Program displays the results of the operation

6. Stop.

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."""
```

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 f"Hello, {name}! Welcome to the Program".
def main():
    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()
```

| VEL TECH | |
|---|---|
| EX No. | 7 |
| PERFORMANCE (5) | 5 |
| RESULT AND ANALYSIS (5) | 5 |
| VIVA VOCE (5) | 5 |
| RECORD (5) | 5 |
| TOTAL (20) | 20 |
| SIGN WITH DATE | 10/9 |

Result: Thus, the Python Program using 'Functions Concepts was successfully executed and the output was verified.