|9|2025

Task No. 7 Utilizing 'Functions' Concepts in Python Programming

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 wellcome message: outputs a simple greeting
3. Determine & print the number of students: Uses len() to find the number of elements in the student_names list.
4. Print the type of lists: User type() to show the type of Student_names and student_grades lists.
5. Find and print highest and lowest grades: User max() and min() to determine the highest and lowest values in student_grad
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 ranges() 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 & the 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)
      lowes_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)
      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("\nGrade indices from 1 to number of students:",
grade_indices)

      #Run the analysis
      analyze_student_grades()
```

Output:

```
Welcome to the student 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]
grades indices from 1 to number of students:
[1, 2, 3, 4]
```

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 users to enter two numbers
3. User Input for Operation: The program prompts the user to choose an arithmetic Operation
4. Perform Operation: Based on the user's choice, the program performs the chosen arithmetic Operation Using the defined fun.
5. Display Result:-The program displays the result of 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."""
~~def~~ 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 great(name):
   """Return a greeting message for the user."""
   return f"Hello,{name}!welcome to the program."
def main ():
   #Demonstrating the user of user-defined functions
```

```
#Arithmetic Operations
num 1 =10
num 2 =5
Print ("Arithmetic Operations:")
Print (f"Sum of {num 1} and {num 2} :", add (num1 ,num 2))
Print (f'Difference between {num 1} and {num 2}:", Subtract (num1, num2))
Print(f"Product of {num 1} and {num 2} :", multiply (num1,num 2))
Print(f"Quotient of {num 1} and {num 2}:", divide (num 1 ,num 2))
#Greeting the user
user_name ="Abbie"
Print ("\nGreeting :")
Print (greet(user_name))
# Run the main function
if _name_ = = "_main_":
  main ()
```

VEL TECH - CSE	
EX NO.	
PERFORMANCE (5)	7
RESULT AND ANALYSIS (5)	8
VIVA VOCE (5)	8
RECORD (5)	8
TOTAL (20)	
SIGN WITH DATE	18

RESULT :

Thus, the python program using 'Function' concepts was successfully executed and the output was verified .

output :-

```
Arithmatic Operations
Sum of 10 & 5 is :15
Difference between 10&5 is : 5
product of 10 & 5 : 50
Quotient of 10 & 5 :2.0
```

Greeting :-

```
Hello, Abbie ! welcome to the program !
```