

25

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 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 element in the student_names list.
- 4) print the print types 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 and print a range of grade indices: use `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 student-name list:", type(student_names))
```

```
    print("Type of student-grades list:", type(student_grades))
```

Output:-

welcome to the student Grade 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]

#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 grades", 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("Reverse grades:", reversed_grades)

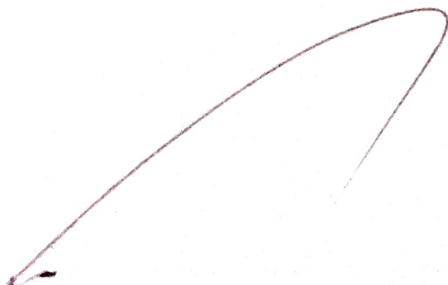
#7. Generate and print a range of grade indices from 1 to the number of students

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.

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 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):  
    """ 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))
```

Output:-

Arithmetic Operations:

Sum of 10 and 5: 15

Difference of 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.

o/f

28

```

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("\n Greeting:")
print(greet(user_name))
# Run the main function
if __name__ == "__main__":

```

Result:- Thus, the python program using 'Functions' concepts was successfully executed and the output was verified.

EX No.	
PERFORMANCE (5)	7
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	27
SIGN WITH DATE	