

Task 7: Utilizing 'functions' concepts in Python Programming

Aim:-

To write the Python program using 'functions' concepts in Python programming.

7.11 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 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. 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 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")
```

```
print("Number of Students:", num_students)
```

```
# 2. Determine and print the number of students
```

```
num_students = len(student_names)
```

```
print("Number of students:", num_students)
```

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'>

high grade: 92

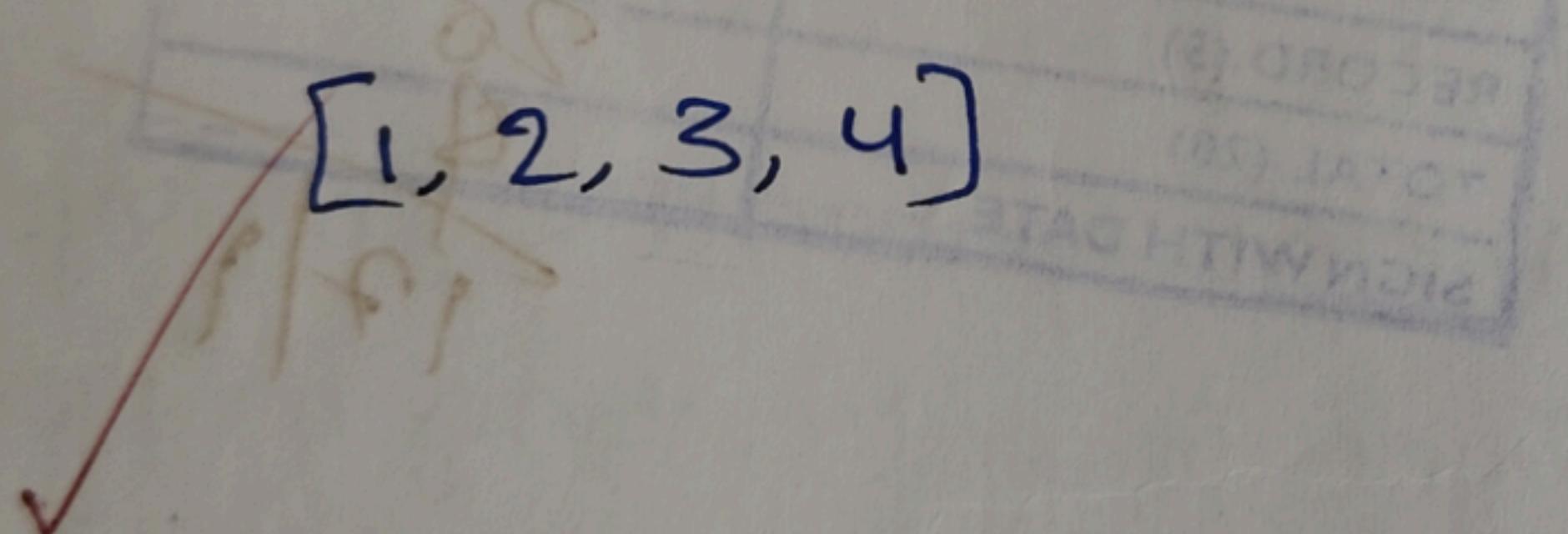
lowest grade: 78

sorted grades: [78, 85, 90, 92]

reversed -grades: [90, 78, 92, 85]

grade indicates from 1 to number of students:

[1, 2, 3, 4]



```

def analyse_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 Analyser!\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 student names list and the grades list
    print("In type of student names list:", type(Student_names))
    print("In 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("In highest grade:", highest_grade)
    print("In lowest grade:", lowest_grade)

    # 5. Print the list of grades sorted in ascending order
    sorted_grades = sorted(Student_grades)
    print("In sorted 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("In Grade indices from 1 to number of students:", grade_indices)

# Run the analysis
analyse_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.
4. perform operations: 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.

Program:-

```
def fadd(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"
```

Output:-

Arithmetical 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

```

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))
    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("In Greeting:")
    print(greet(user_name))

# Run the main function
if __name__ == "__main__":
    main()

```

YEL TECH	
PERFORMANCE (5)	75
RESULT AND ANALYSIS (5)	55
PHYSICAL FITNESS (5)	55
RECORD (5)	20
TOTAL (20)	175
WITH DATE	17/07/2023

Result:-

Thus, the Python program using 'functions' concepts was successfully executed and the output was verified.