

Utilizing functions' Concepts in Python Programming

- 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 requirement using the built-in function `print()`, `len()`, `type()`, `max()`, `min()`, `sorted()`, `reversed()` and `range()`.

Algorithm:

1. Start the program
2. Print a welcome program; 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 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: user `reversed()` to reverse the sorted list and converts it to a list.
8. Generate and print a range of grades 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]
    # Print a welcome message
    print("Welcome to the student grades analyze!\n")
    # 2. Determine and print the number of students
    num_students = len(student_name)
    print("Number of students:", num_students)

```

#3. Print the type of the student names list and the grade list.

```
print("In Type of Student-names list:", type(students_names))
print("Type of Student-grades list:", type(student_grades))
```

#4 Find and print the highest and lowest grades.

```
highest_grade = max(student_grades)
lowest_grade = min(student_grades)
```

```
print("In highest grade:", highest_grade)
print("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 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

```
Analyze_student_grades()
```

Output: *Hilisila*, *Gumtak*, *Concepcion* & *Mata* *Bohol*

Welcome to the Student grades analyzer! See you
number of students : 4
type of student names list : 2 class lists
(Aman, Omkar, Om, Omkar) each with 2 class lists
type of student-grades list : 2 class lists
(Computer and Chemistry)
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]

1. *On the 1st of May: see table to show the value of the*

of China and Ceylon 23201 became

Surficial of Obstruction 2000' Elevation To 1800' Elevation

28000 ft of 06280008 1924 - Part 1

• 520 . 29600 P to 1281 because there is
no 10 or 11 296000.

The board had a long conference with the members of the school board to discuss the proposed changes.

most of the time was suspended.

The College of Nursing
University of Michigan

Chestnut - Robert - Sponer 7th

Pomoxis, Shad, 15 lbs, doggin, Dab sign #

~~[OP, 85, SP, 25]~~ = emm. mehut?

~~(all) displays~~ ~~SP.C8~~ ~~spotted Instabili~~
~~Sporadic displays~~

~~2 bushels to 100 ft²~~ 30 ft² of Bougainvillea

6.2 You are asked 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 task: addition, subtraction, multiplication and 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 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 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:

sharp with highest priority finds 10 to sqrt with 10000

arithmetic operations

sum of 10 & 5 is 15 (sharp finds 10 & 5 to sum with 10000)

differences - find 10 & 5 (sharp finds 10 & 5 to diff with 10000)

product of 10 and 5 = 50 (sharp finds 10 & 5 to prod with 10000)

quotient of 10 and 5 = 2 (sharp finds 10 & 5 to quot with 10000)

greeting:

Hello Alice: welcome to the program (sharp reads "Hello friend" with 10000)

also print out of 10000 of sharp to t89 with 10000

(sharp finds 10000 to print with 10000)

(sharp reads 10000 to sharp with 10000)

also print out of sharp to t89 with 10000

(sharp finds 10000 to print with 10000)

out of 10000 of sharp to sharp with 10000

(sharp finds 10000 to read with 10000)

"i & sharp to read with 10000 of 1 out of sharp with 10000

(sharp reads

212000 with 10000

(sharp finds 212000 with 10000)

```

        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()

```

VEL TECH - CSE	
EX NO.	6
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' concepts was successfully executed and the output was verified.