

Problem Set-01:

01_1:

```
def count_upper_lower(string):
    upper = 0
    lower = 0

    for char in string:
        if char.isupper():
            upper += 1
        elif char.islower():
            lower += 1
    print("No. of Uppercase characters:", upper)
    print("No. of Lowercase characters:", lower)

sample_string = str(input("Enter Somethings: "))
count_upper_lower(sample_string)
```

01_2:

```
def remove_duplicates(input_list):
    distinct_list = []
    for item in input_list:
        if item not in distinct_list:
            distinct_list.append(item)
    return distinct_list

original_list = [1, 2, 3, 3, 4, 4, 5, 6, 6]
new_list = remove_duplicates(original_list)
print(new_list)
```

01_3:

```
import math

def is_prime(number):
    if number <= 1:
        return False
    elif number <= 3:
        return True
    elif number % 2 == 0 or number % 3 == 0:
        return False
    else:
        sqrt_num = int(math.sqrt(number))
        for i in range(5, sqrt_num + 1, 6):
            if number % i == 0 or number % (i + 2) == 0:
                return False
        return True

num = int(input("Enter a number to check if it's prime:
"))
if is_prime(num):
    print(num, "is a prime number.")
else:
    print(num, "is not a prime number.")
```

Problem Set-02:

02_1:

```
def is_palindrome(s):
    s = ''.join(char.lower() for char in s if
char.isalnum())

    return s == s[::-1]

input_string = input("Enter a string to check if it's a
palindrome: ")
if is_palindrome(input_string):
    print("The string is a palindrome.")
else:
    print("The string is not a palindrome.")
```

02_2:

```
def factorial_iterative(n):
    if n == 0:
        return 1
    factorial = 1
    for i in range(1, n + 1):
        factorial *= i
    return factorial

num = int(input("Enter a non-negative integer: "))
if num < 0:
    print("Factorial is not defined for negative
numbers.")
else:
    print("Factorial of", num, "is:",
factorial_iterative(num))
```

02_3:

```
def is_within_range(number, lower_bound, upper_bound):  
    return lower_bound <= number <= upper_bound  
  
num = int(input("Enter a number: "))  
lower_bound = int(input("Enter the lower bound of the  
range: "))  
upper_bound = int(input("Enter the upper bound of the  
range: "))  
  
if is_within_range(num, lower_bound, upper_bound):  
    print(num, "falls within the range [{},  
{ }].".format(lower_bound, upper_bound))  
else:  
    print(num, "does not fall within the range [{},  
{ }].".format(lower_bound, upper_bound))
```

02_4:

```
def print_even_numbers(input_list):  
    even_numbers = [num for num in input_list if num % 2  
== 0]  
    return even_numbers  
  
sample_list = [1, 2, 3, 4, 5, 6, 7, 8, 9]  
result = print_even_numbers(sample_list)  
print("Even numbers from the list:", result)
```

Problem Set-03:

```
def is_prime(num):
    a=0
    for i in range(2, num):
        if(num % i == 0):
            a=1
            break

    if a==0:
        return True
    else:
        return False

num = int(input("Enter a number: "))

if num<=1:
    print("Enter a number greater than 1")
else:
    if is_prime(num):
        print(num, "is a prime number")
    else:
        print(num, "is not a prime number")
```

Problem Set-04:

```
def sum_Of_Number(n):  
    if n == 0:  
        return 0  
    else:  
        return n + sum_Of_Number(n - 1)  
  
num = int(input("Enter a non-negative integer: "))  
if num <= 0:  
    print("Please try again.")  
else:  
    print("The sum of the digits is:",  
sum_Of_Number(num))
```