1. Write a Python program to find words which are greater than given length k?

ANS: def find\_words\_greater\_than\_k(words, k):

return [word for word in words if len(word) > k]

# Test the function

if \_\_name\_\_ == "\_\_main\_\_":

try:

word\_list = input("Enter space-separated words: ").split()

k = int(input("Enter the value of k: "))

words\_greater\_than\_k = find\_words\_greater\_than\_k(word\_list, k)

print(f"Words greater than length {k}:", words\_greater\_than\_k)

except ValueError:

print("Invalid input. Please enter space-separated words and an integer for k.")

1. Write a Python program for removing i-th character from a string?

ANS: def remove\_ith\_character(input\_string, i):

if i < 0 or i >= len(input\_string):

return "Invalid index. Please provide a valid position within the string."

return input\_string[:i] + input\_string[i+1:]

# Test the function

if \_\_name\_\_ == "\_\_main\_\_":

try:

input\_str = input("Enter a string: ")

index\_to\_remove = int(input("Enter the index of the character to remove: "))

result\_string = remove\_ith\_character(input\_str, index\_to\_remove)

print("String after removing the character:", result\_string)

except ValueError:

print("Invalid input. Please enter a valid string and an integer for the index.")

1. Write a Python program to split and join a string?

ANS: def split\_and\_join(input\_string):

# Split the string into a list of words using whitespace as the delimiter

words\_list = input\_string.split()

# Join the words in the list using a space as the separator

joined\_string = ' '.join(words\_list)

return words\_list, joined\_string

# Test the function

if \_\_name\_\_ == "\_\_main\_\_":

try:

input\_str = input("Enter a string: ")

words, joined\_str = split\_and\_join(input\_str)

print("Original string:", input\_str)

print("Split words:", words)

print("Joined string:", joined\_str)

except ValueError:

print("Invalid input. Please enter a valid string.")

1. Write a Python to check if a given string is binary string or not?

ANS: def is\_binary\_string(input\_string):

# A binary string should contain only '0' and '1' characters

for char in input\_string:

if char not in ['0', '1']:

return False

return True

# Test the function

if \_\_name\_\_ == "\_\_main\_\_":

input\_str = input("Enter a string: ")

if is\_binary\_string(input\_str):

print("The string is a binary string.")

else:

print("The string is not a binary string.")

1. Write a Python program to find uncommon words from two Strings?

ANS: def find\_uncommon\_words(string1, string2):

# Convert the strings to sets of words

set1 = set(string1.split())

set2 = set(string2.split())

# Find uncommon words by taking the symmetric difference of the sets

uncommon\_words = set1.symmetric\_difference(set2)

return uncommon\_words

# Test the function

if \_\_name\_\_ == "\_\_main\_\_":

try:

string1 = input("Enter the first string: ")

string2 = input("Enter the second string: ")

uncommon\_words = find\_uncommon\_words(string1, string2)

if len(uncommon\_words) == 0:

print("No uncommon words found.")

else:

print("Uncommon words:", uncommon\_words)

except ValueError:

print("Invalid input. Please enter valid strings.")

1. Write a Python to find all duplicate characters in string?

ANS: def find\_duplicate\_characters(input\_string):

# Create an empty dictionary to store character frequencies

char\_freq = {}

# Iterate through each character in the string and update its frequency

for char in input\_string:

char\_freq[char] = char\_freq.get(char, 0) + 1

# Filter out characters with frequency greater than 1 (i.e., duplicates)

duplicates = [char for char, freq in char\_freq.items() if freq > 1]

return duplicates

# Test the function

if \_\_name\_\_ == "\_\_main\_\_":

input\_str = input("Enter a string: ")

duplicate\_chars = find\_duplicate\_characters(input\_str)

if len(duplicate\_chars) == 0:

print("No duplicate characters found.")

else:

print("Duplicate characters:", duplicate\_chars)

1. Write a Python Program to check if a string contains any special character?

ANS: import string

def contains\_special\_character(input\_string):

special\_characters = set(string.punctuation)

for char in input\_string:

if char in special\_characters:

return True

return False

# Test the function

if \_\_name\_\_ == "\_\_main\_\_":

input\_str = input("Enter a string: ")

if contains\_special\_character(input\_str):

print("The string contains special characters.")

else:

print("The string does not contain any special characters.")