**Write a python script to generate Fibonacci terms using generator function**

def fibonacci\_generator():

"""

A generator function to yield Fibonacci terms indefinitely.

"""

a, b = 0, 1

while True:

yield a

a, b = b, a + b

# Example usage

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

fib\_gen = fibonacci\_generator()

num\_terms = int(input("Enter the number of Fibonacci terms to generate: "))

for \_ in range(num\_terms):

print(next(fib\_gen))

**Write a python program to accept string and remove the characters which have odd index values of given string using user defined function.**

def remove\_odd\_index\_characters(input\_string):

"""

Function to remove characters with odd index values from a string.

"""

return ''.join(char for index, char in enumerate(input\_string) if index % 2 == 0)

# Main program

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

user\_input = input("Enter a string: ")

result = remove\_odd\_index\_characters(user\_input)

print(f"String after removing characters with odd index values: {result}")

Given a list of integers, create a Python function that returns the second largest number

def second\_largest(numbers):

"""

Function to return the second largest number in a list of integers.

"""

if len(numbers) < 2:

raise ValueError("List must contain at least two distinct numbers.")

unique\_numbers = list(set(numbers)) # Remove duplicates

if len(unique\_numbers) < 2:

raise ValueError("List must contain at least two distinct numbers.")

unique\_numbers.sort(reverse=True) # Sort in descending order

return unique\_numbers[1] # Return the second largest number

# Example usage

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

try:

nums = list(map(int, input("Enter a list of integers separated by spaces: ").split()))

print(f"The second largest number is: {second\_largest(nums)}")

except ValueError as e:

print(e)

**Write a Python function that accepts a string and calculate the number of upper-case letters and lower case letters. Sample String: 'The quick Brown Fox' Expected Output: No. of Upper case characters: 3 No. of Lower case characters: 13**

def count\_case\_characters(input\_string):

"""

Function to count the number of uppercase and lowercase letters in a string.

"""

upper\_case\_count = sum(1 for char in input\_string if char.isupper())

lower\_case\_count = sum(1 for char in input\_string if char.islower())

return upper\_case\_count, lower\_case\_count

# Example usage

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

sample\_string = 'The quick Brown Fox'

upper\_count, lower\_count = count\_case\_characters(sample\_string)

print(f"No. of Upper case characters: {upper\_count}")

print(f"No. of Lower case characters: {lower\_count}")

**Create a function that counts the frequency of each word in a given sentence using a dictionary.**

def word\_frequency(sentence):

words = sentence.split()

frequency = {}

for word in words:

frequency[word] = frequency.get(word, 0) + 1

return frequency

sentence = "the quick brown fox jumps over the lazy dog"

print(word\_frequency(sentence))

**. Create a Python function that takes a dictionary and returns a new dictionary with all the keys and values swapped**.

def swap\_dict\_keys\_values(d):

"""

Function to swap keys and values in a dictionary.

"""

return {value: key for key, value in d.items()}

# Example usage

sample\_dict = {'a': 1, 'b': 2, 'c': 3}

print(swap\_dict\_keys\_values(sample\_dict))

. **Given a tuple of integers, create a function that converts the tuple into a list and sorts the list.**

def tuple\_to\_sorted\_list(t):

"""

Function to convert a tuple of integers into a sorted list.

"""

return sorted(list(t))

# Example usage

numbers = (5, 3, 8, 1, 4)

print(tuple\_to\_sorted\_list(numbers))

**Create a Python function that takes a tuple of strings and returns a new tuple with the lengths of each string.**

def string\_lengths(t):

"""

Function to return a tuple with the lengths of each string in the input tuple.

"""

return tuple(len(s) for s in t)

# Example usage

strings = ("apple", "banana", "cherry")

print(string\_lengths(strings))

. **Create a Python function that takes a list of strings and returns a new list where all strings are converted to uppercase.**

def convert\_to\_uppercase(strings):

"""

Function to convert a list of strings to uppercase.

"""

return [s.upper() for s in strings]

# Example usage

string\_list = ["hello", "world", "python"]

print(convert\_to\_uppercase(string\_list))

**Using a tuple of numbers, create a function that calculates and returns the sum of all elements in the tuple.**

def sum\_tuple\_elements(t):

"""

Function to calculate the sum of all elements in a tuple.

"""

return sum(t)

# Example usage

numbers = (10, 20, 30, 40)

print(sum\_tuple\_elements(numbers))

**Create a function that takes two sets and returns the symmetric difference between them. Demonstrate the function with sample sets**

def symmetric\_difference(set1, set2):

"""

Function to return the symmetric difference between two sets.

"""

return set1.symmetric\_difference(set2)

# Example usage

set\_a = {1, 2, 3, 4}

set\_b = {3, 4, 5, 6}

print(symmetric\_difference(set\_a, set\_b))

**. Write a Python function that checks if a given set is a subset of another set**

def is\_subset(set1, set2):

"""

Function to check if a set is a subset of another set.

"""

return set1.issubset(set2)

set\_a = {1, 2}

set\_b = {1, 2, 3, 4}

print(is\_subset(set\_a, set\_b))