In [4]:
```python
# string_utils.py

def count_length(s):
    return len(s)

def to_upper(s):
    return s.upper()

def is_palindrome(s):
    s = s.lower().replace(" ", "")
    return s == s[::-1]

def reverse_string(s):
    return s[::-1]
```

Out[4]:
```
'NAMAN'
```

In [5]:
```python
import string_utils

user_string = input("Enter a string: ")

print("-" * 30)
print("Original string:", user_string)

length = string_utils.count_length(user_string)
print("Length of string:", length)

upper_case = string_utils.to_upper(user_string)
print("Uppercase string:", upper_case)

is_pal = string_utils.is_palindrome(user_string)
print("Is it a palindrome?", "Yes" if is_pal else "No")

reversed_str = string_utils.reverse_string(user_string)
print("Reversed string:", reversed_str)
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
Cell In[5], line 1
----> 1 import string_utils
      3 user_string = input("Enter a string: ")
      5 print("-" * 30)

ModuleNotFoundError: No module named 'string_utils'
```

In [ ]:
```python
import os

base_dir_name = "Python_Task_Dir"
sub_dir_name = "Sub_Directory"

try:
    path_to_create = os.path.join(base_dir_name, sub_dir_name)
    os.makedirs(path_to_create, exist_ok=True)
```

```
    print("Successfully created directory structure:", path_to_create)
except OSError as e:
    print("Error creating directories:", e)

current_working_dir = os.getcwd()
print("\n" + "=" * 30)
print("Current Working Directory Path:", current_working_dir)
print("=" * 30)

print("\nFiles and Directories in:", base_dir_name)
try:
    if os.path.exists(base_dir_name):
        files_and_dirs = os.listdir(base_dir_name)
        if files_and_dirs:
            for item in files_and_dirs:
                print("*", item)
        else:
            print("The directory is empty.")
    else:
        print("Directory does not exist:", base_dir_name)
except Exception as e:
    print("An error occurred while listing files:", e)
```

In [ ]:

```
# cal.py

def add(a, b):
    return a + b

def subtract(a, b):
    return a - b

def multiply(a, b):
    return a * b

def divide(a, b):
    if b == 0:
        return "Error: Cannot divide by zero"
    return a / b
```

In [ ]:

```
# main_498.py

import cal

num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))

print("-" * 30)

print("Addition (", num1, "+", num2, "):", cal.add(num1, num2))
print("Subtraction (", num1, "-", num2, "):", cal.subtract(num1, num2))
print("Multiplication (", num1, "*", num2, "):", cal.multiply(num1, num2))
print("Division (", num1, "/", num2, "):", cal.divide(num1, num2))
```

In [ ]:
```python
def get_first_word(input_string):
    words = input_string.strip().split()
    if words:
        return words[0]
    else:
        return ""
```

In [ ]:
```python
import first_word

input_text = 'This is Python Programming'
output_word = first_word.get_first_word(input_text)

print("--- Working of 'first_word' Module ---")
print("Input:", input_text)
print("Output:", output_word)

test_input = "  Hello world!   "
print("-" * 37)
print("Input:", test_input)
print("Output:", first_word.get_first_word(test_input))
```

In [ ]:
```python

```