

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "Hello, world!"
5 print(text.replace("world", " #Python Program"))
```

```
Hello, #Python Program!
```

```
=== Code Execution Successful ===
```

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 words = ["Hello", " #Python Program"]
5 print("".join(words))
```

```
Hello #Python Program
```

```
=== Code Execution Successful ===
```

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "Hello123"
5 print(text.isalnum())
```

True

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "apple,banana,orange"
5 print(text.split(","))
```

```
['apple', 'banana', 'orange']
```

```
=== Code Execution Successful ===
```

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = " #Python Program  "
5 print(text.rstrip())
```

```
#Python Program

=== Code Execution Successful ===
```

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = " Hello"
5 print(text.lstrip())
```

Hello

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = " #Python Program  "
5 print(text.strip())
```

#Python Program

=== Code Execution Successful ===

Programiz Python Online Compiler

Programiz PRO >

main.py

Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "hello #Python Program"
5 print(text.capitalize())
```

```
Hello #python program

=== Code Execution Successful ===
```


main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "hello #python program"
5 print(text.title())
```

Hello #Python Program

=== Code Execution Successful ===



main.py



Share

Run

Output

Clear



JS

TS

GO

php



```
1 # Function to add two numbers
2 def add_numbers(a, b):
3     return a + b
4
5 # Example usage
6 num1 = 5
7 num2 = 7
8 result = add_numbers(num1, num2)
9 print("The sum is:", result)
10
```

The sum is: 12

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "42"
5 print(text.zfill(5))
```

00042

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "Hello #Python Program"
5 print(text.swapcase())
```

```
hELLO #pYTHON pROGRAM
```

```
=== Code Execution Successful ===
```

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 num = "12345"
5 print(num.isdigit())
```

True

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "Hello"
5 print(text.isalpha())
```

True

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "is fun!"
5 print(text.endswith("fun!"))
```

True

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "Hello, #Python Program!"
5 print(text.startswith("Hello"))
```

True

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "banana banana"
5 print(text.count("banana"))
```

```
2
=== Code Execution Successful ===
```

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "programming"
5 print(text.find("program"))
```

```
0
=== Code Execution Successful ===
```

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = " #Python Program"
5 print(text.lower())
6
```

#python program

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "hello"
5 print(text.upper()) # Output: HELLO
6
```

HELLO

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Python Program
2 # Simple string program using built-in function
3
4 text = "Hello, #Python"
5 print(len(text)) # Output: 14
6
```

14

=== Code Execution Successful ===

main.py



Share

Run

Output

Clear

```
1 # Function to swap two numbers
2 def swap_numbers(a, b):
3     return b, a
4
5 # Example usage
6 num1 = 10
7 num2 = 20
8 print("Before swapping: num1 =", num1, ", num2 =", num2)
9
10 num1, num2 = swap_numbers(num1, num2)
11 print("After swapping: num1 =", num1, ", num2 =", num2)
12
```

```
Before swapping: num1 = 10 , num2 = 20
After swapping: num1 = 20 , num2 = 10
```

```
=== Code Execution Successful ===
```

main.py



Share

Run

Output

Clear

```
1 # Function to find power of a number
2 def power(base, exponent):
3     return base ** exponent
4
5 # Example usage
6 base = 2
7 exponent = 3
8 result = power(base, exponent)
9 print(f"{base} raised to the power of {exponent} is {result}")
10
```

2 raised to the power of 3 is 8

=== Code Execution Successful ===

main.py



Share

Run

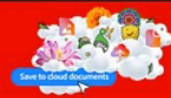
Output

Clear

```
1 # Function to check if a number is even or odd
2 def check_even_odd(number):
3     if number % 2 == 0:
4         return "Even"
5     else:
6         return "Odd"
7
8 # Example usage
9 num = 7
10 result = check_even_odd(num)
11 print(f"The number {num} is {result}.")
12
```

The number 7 is Odd.

=== Code Execution Successful ===



main.py



Share

Run

Output

Clear

```
1 # Function to find factorial of a number
2 def factorial(n):
3     if n < 0:
4         return "Factorial is not defined for negative numbers"
5     elif n == 0 or n == 1:
6         return 1
7     else:
8         result = 1
9         for i in range(2, n + 1):
10             result *= i
11         return result
12
13 # Example usage
14 num = 5
15 print("The factorial of", num, "is:", factorial(num))
16
```

The factorial of 5 is: 120

=== Code Execution Successful ===

JS

TS

GO

PHP

