Day 7 (2/9/25): String Operators and Methods in Python

1. Introduction

Strings in Python support **various operators and built-in methods** to perform operations like concatenation, repetition, searching, and formatting.

2. String Operators

A. Concatenation (+)

Joins two or more strings together.

```
str1 = "Hello"
str2 = "World"
result = str1 + " " + str2
print(result)  # Hello World
```

B. Repetition (*)

Repeats a string multiple times.

```
str1 = "Hi! "
print(str1 * 3)  # Hi! Hi! Hi!
```

C. Membership (in, not in)

Checks if a substring exists in a string.

```
text = "Python Programming"
print("Python" in text)      # True
print("Java" not in text)    # True
```

D. Comparison (==, !=, <, >)

Compares strings lexicographically (ASCII-based).

```
print("apple" < "banana")    # True
print("apple" == "Apple")    # False
```

3. String Methods

Python provides **many built-in string methods**. Here are the most commonly used:

Method	Description
upper()	Converts string to uppercase
lower()	Converts string to lowercase
capitalize()	Capitalizes the first character
title()	Capitalizes the first character of each word
strip()	Removes leading/trailing spaces
replace(old, new)	Replaces a substring with another
split(separator)	Splits string into a list of substrings
join(iterable)	Joins elements of iterable into a string
find(sub)	Returns index of first occurrence of substring (-1 if not found)
count(sub)	Returns number of occurrences of substring

4. Example Program – String Operators and Methods

```
# String Operators
str1 = "Hello"
str2 = "World"

# Concatenation
concat = str1 + " " + str2

# Repetition
repeat = str1 * 3

# Membership
check1 = "Hello" in concat
check2 = "Python" not in concat
```

```
# String Methods
text = "   python programming   "
upper_text = text.upper()
lower_text = text.lower()
capital_text = text.capitalize()
title_text = text.title()
strip_text = text.strip()
replace_text = text.replace("python", "Java")
split_text = text.split()
join_text = "-".join(split_text)

print("Concatenation:", concat)
print("Repetition:", repeat)
print("'Hello' in concat?", check1)
print("'Python' not in concat?", check2)
print("Uppercase:", upper_text)
print("Lowercase:", lower_text)
print("Capitalized:", capital_text)
print("Title Case:", title_text)
print("Stripped:", strip_text)
print("Replace 'python' with 'Java':", replace_text)
print("Split into list:", split_text)
print("Join list with '-':", join_text)
```

5. Sample Output

```
Concatenation: Hello World
Repetition: HelloHelloHello
'Hello' in concat? True
'Python' not in concat? True
Uppercase:    PYTHON PROGRAMMING
Lowercase:    python programming
Capitalized:    Python programming
Title Case:    Python Programming
Stripped: python programming
Replace 'python' with 'Java':    Java programming
Split into list: ['python', 'programming']
Join list with '-': python-programming
```

6. Summary

- Operators +, *, in, not in, ==, <, > are commonly used with strings.
- Python provides **many string methods** for changing case, trimming, replacing, splitting, joining, and searching.
- Strings are **immutable**, so methods always return a **new string**.