Strings in Python

Introduction to Strings

- A **string** is a sequence of characters enclosed in **single**, **double**, or **triple quotes**.
- Strings are **immutable**, meaning you cannot change characters once defined.

```
s1 = 'Hello'
s2 = "World"
s3 = '''This is
a multiline
string'''
```

String Operations

1. Concatenation (+)

Used to combine two or more strings.

```
a = "Good"
b = "Morning"
print(a + " " + b)  # Good Morning
```

2. Repetition (*)

Repeats the string specified number of times.

```
print("Hi " * 3)  # Hi Hi Hi
```

3. Membership (in, not in)

Checks if a substring is present in a string.

```
print('a' in 'banana')     # True
print('z' not in 'apple')  # True
```

4. Slicing

Extracts part of a string.

```
s = "Python"
print(s[0:3])   # Pyt
print(s[-1])    # n
```

Traversing a String Using Loops

Using for loop:

```
s = "Hello"
for char in s:
    print(char)
```

Using while loop:

```
i = 0
while i < len(s):
    print(s[i])
    i += 1
```

Built-in String Functions / Methods

Function	Description	Example
len()	Returns the length of the string	len("apple") → 5
capitalize()	Capitalizes the first character	"hello".capitalize() → Hello
title()	Capitalizes the first letter of each word	"my name".title() → My Name
lower()	Converts to lowercase	"HeLLo".lower() → hello
upper()	Converts to uppercase	"hi".upper() → HI

count()	Counts occurrences of a substring	"banana".count('a') → 3
find()	Returns index of first occurrence	"banana".find('n') → 2
index()	Same as find(), but error if not found	"banana".index('b') → 0
endswith()	Checks if string ends with specified value	"file.txt".endswith('.txt') → True
startswith()	Checks if string starts with specified value	"apple".startswith('a') → True

Character Testing Methods

Method	Description	Example
isalnum()	True if all characters are alphanumeric	"abc123".isalnum() → True
isalpha()	True if all characters are alphabetic	"abc".isalpha() → True
isdigit()	True if all characters are digits	"123".isdigit() → True
islower()	True if all characters are lowercase	"abc".islower() → True
isupper()	True if all characters are uppercase	"ABC".isupper() → True
isspace()	True if all characters are whitespace	" ".isspace() → True

Whitespace Handling Methods

Method	Description	Example
lstrip()	Removes whitespace from the left	" hello".lstrip() → "hello"
rstrip()	Removes whitespace from the right	"hello ".rstrip() → "hello"
strip()	Removes whitespace from both sides	" hello ".strip() → "hello"

String Modification Methods

Method	Description	Example
replace()	Replaces part of the string	"hello world".replace(\"world\", \"Python\") → "hello Python"
join()	Joins elements of a sequence with a string	"*".join(['a', 'b']) → "a*b"
partition()	Splits string at first occurrence of sep	"apple:banana".partition(\":\") → ('apple', ':', 'banana')
split()	Splits string into a list by delimiter	"a,b,c".split(',') → ['a', 'b', 'c']

Example Program: All in One

```
s = " Hello Python Programmer "
print("Original:", s)
print("Upper:", s.upper())
print("Lower:", s.lower())
print("Capitalize:", s.capitalize())
print("Title:", s.title())
print("Length:", len(s))
print("Strip:", s.strip())
print("Replace:", s.replace("Python", "Java"))
print("Split:", s.split())
print("Count of o:", s.count('o'))
print("Ends with 'r':", s.endswith('r'))
```