All Python String Methods Explained in Layman Terms (with 3 Examples Each)

---

1. `upper()`

Converts all characters in the string to uppercase.

```
print("hello".upper())        # HELLO
print("python123".upper())    # PYTHON123
print("aBcD".upper())         # ABCD
```

2. `lower()`

Converts all characters in the string to lowercase.

```
print("HELLO".lower())        # hello
print("PyTHon".lower())       # python
print("123ABC".lower())       # 123abc
```

3. `capitalize()`

Capitalizes the first letter of the string.

```
print("python".capitalize())      # Python
print("hello world".capitalize()) # Hello world
print("123abc".capitalize())      # 123abc
```

4. `title()`

Capitalizes the first letter of every word in the string.

```
print("hello world".title())     # Hello World
print("my name is".title())      # My Name Is
print("python PROGRAM".title())  # Python Program
```

5. `strip()`

Removes leading and trailing spaces.

```
print("  hello  ".strip())       # hello
print("\thello\n".strip())       # hello
print("   test".strip())         # test
```

6. `lstrip()`

Removes spaces from the left side of the string.

```
print("   hello".lstrip())       # hello
print("\ttext".lstrip())         # text
print("   python  ".lstrip())    # python
```

7. `rstrip()`

Removes spaces from the right side of the string.

```
print("hello   ".rstrip())       # hello
print("python\n".rstrip())       # python
print(" test  ".rstrip())        #  test
```

8. `replace(old, new)`

Replaces a part of the string with another.

```
print("hello world".replace("world", "Python"))  # hello Python
print("a-b-c".replace("-", ":"))                  # a:b:c
print("aaabbb".replace("a", "x"))                # xxxbbb
```

9. `find(substring)`

Finds the first occurrence index of the substring.

```
print("hello".find("l"))     # 2
print("apple".find("p"))     # 1
print("test".find("z"))      # -1
```

10. `count(substring)`

Counts how many times the substring appears.

```
print("banana".count("a"))    # 3
print("apple".count("p"))     # 2
print("test".count("z"))      # 0
```

11. `startswith(substring)`

Checks if the string starts with a specific substring.

```
print("hello".startswith("he"))   # True
print("world".startswith("w"))    # True
print("python".startswith("java")) # False
```

12. `endswith(substring)`

Checks if the string ends with a specific substring.

```
print("hello".endswith("o"))      # True
print("index.py".endswith(".py")) # True
print("data".endswith(".csv"))    # False
```

13. `split(separator)`

Splits the string by the given separator.

```
print("a,b,c".split(","))       # ['a', 'b', 'c']
print("hello world".split())    # ['hello', 'world']
print("1-2-3".split("-"))        # ['1', '2', '3']
```

14. `join(iterable)`

Joins a list of strings using the string as a separator.

```
print("-".join(["a", "b", "c"]))     # a-b-c
print(" ".join(["hello", "world"])) # hello world
print("".join(["1", "2", "3"]))     # 123
```

15. `isalpha()`

Checks if all characters in the string are letters.

```
print("abc".isalpha())     # True
print("abc123".isalpha())  # False
print("ABC".isalpha())     # True
```

16. `isdigit()`

Checks if all characters in the string are digits.

```
print("123".isdigit())     # True
print("abc123".isdigit())  # False
print("007".isdigit())     # True
```

17. `isalnum()`

Checks if all characters are alphanumeric (letters or digits).

```
print("abc123".isalnum())  # True
print("abc!".isalnum())    # False
print("123".isalnum())     # True
```

18. `swapcase()`

Swaps the case of each letter (upper to lower and vice versa).

```
print("Hello".swapcase())       # hELLO
print("PyTHon".swapcase())      # pYthON
print("123ABCabc".swapcase())   # 123abcABC
```

19. `zfill(width)`

Pads the string with zeros on the left to match the width.

```
print("5".zfill(3))        # 005
print("123".zfill(5))      # 00123
print("abc".zfill(6))      # 000abc
```

20. `casefold()`

Lowercases all characters aggressively (better for comparing strings).

```
print("HELLO".casefold())      # hello
print("ß".casefold())          # ss
print("Python123".casefold())  # python123
```

You now have a clear reference of **20 important string methods**, all explained in simple terms with three examples each!