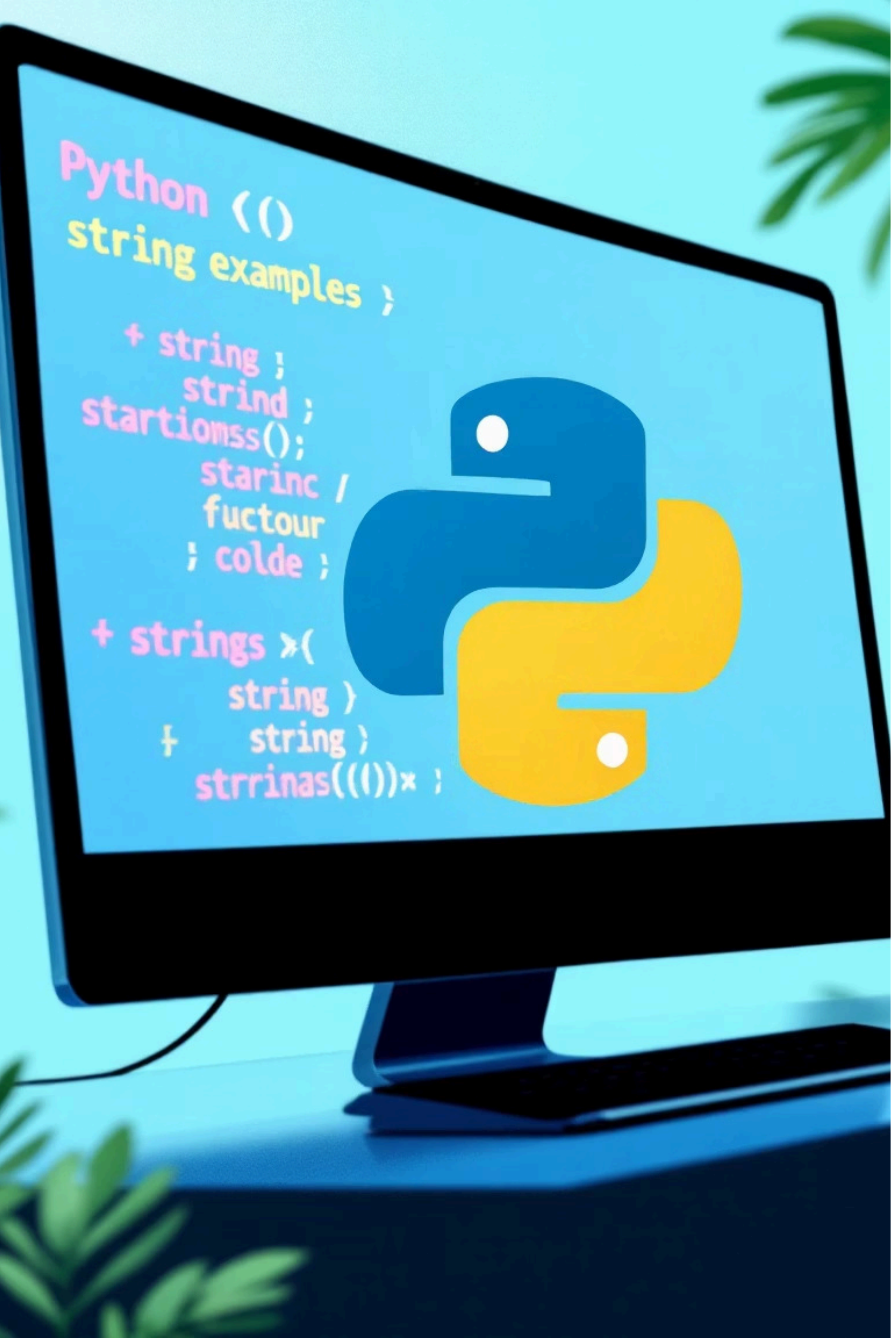


Day 03 | Week 01: Introduction to Python Strings

Today we'll explore strings - one of Python's most versatile and commonly used data types.

What is a String in Python?

A string is a sequence of characters enclosed in quotes. Python **treats everything inside quotes as text** - whether they're letters, numbers, or symbols.

String Creation

Use single quotes: 'hello'

Use double quotes: "world"

Use triple quotes for multi-line: '''python strings'''

Immutability

Once created, strings **cannot be changed directly.** Any "modification" creates a new string.

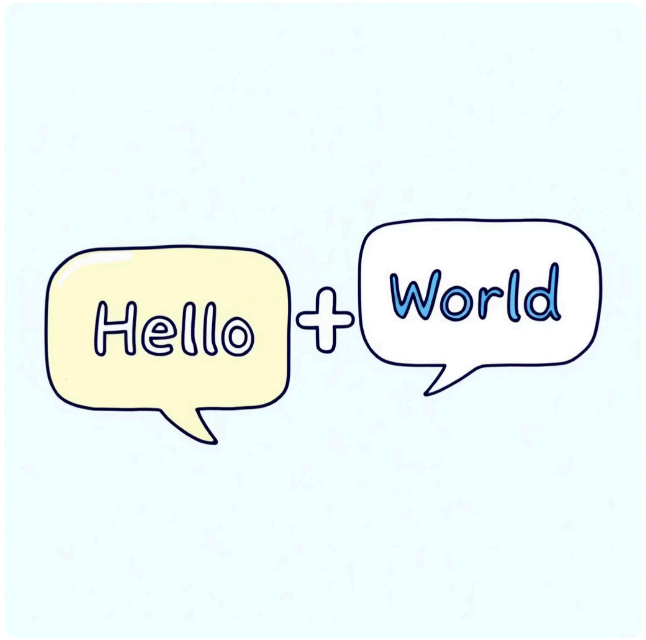
String Concatenation: Joining Strings

Python makes it easy to combine strings using operators:

+ Operator

Joins two or more strings together:

```
first = "Hello, "
second = "World!"
greeting = first + second
# Result: "Hello, World!"
```



Useful String Methods for Manipulation

Case Methods

```
text = "Hello World"
text.upper()  # "HELLO WORLD"
text.lower()  # "hello world"
text.title()  # "Hello World"
text.Capitalize()  # "Hello World"
```

Search Methods

```
text = "Hello World"
text.find("World")  # 6
text.count("l")     # 3
"World" in text     # True
```

Removal Methods

```
text = "  Hello  "
text.strip()    # "Hello"
text.lstrip()   # "Hello  "
text.rstrip()   # "  Hello"
```

Replacement

```
text = "Hello World"
text.replace("World", "Python")
# "Hello Python"

print(text.replace("o", "a"))
# "Hella Warld"
```

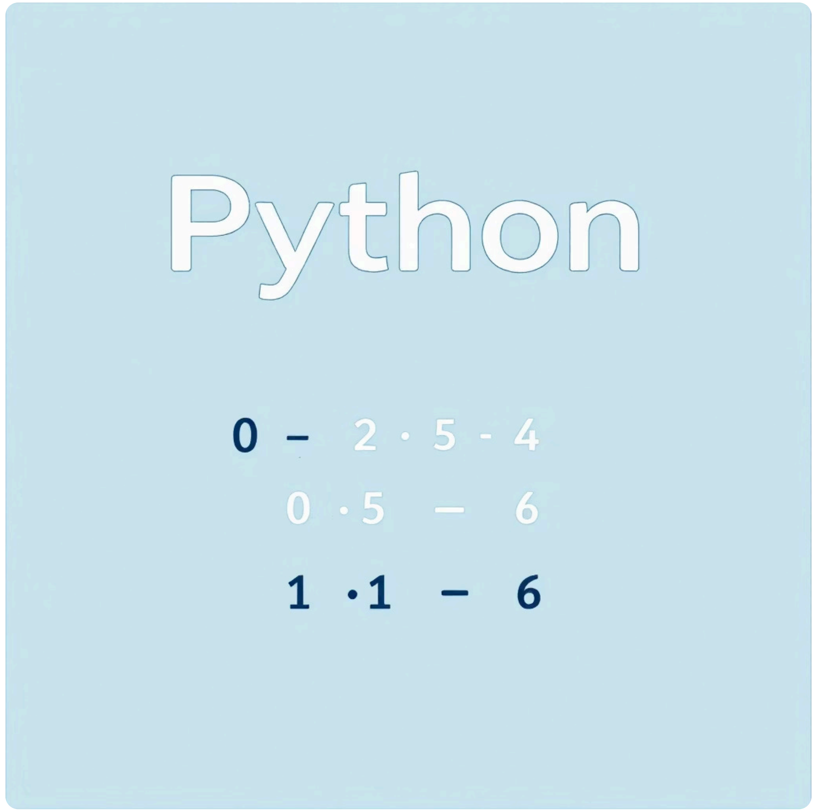
Indexing: Accessing Characters by Position

Python strings are sequences of characters, each with a specific position or index:

- Indexing starts at 0 from the left side
- Negative indexing starts at -1 from the right side

```
word = "Python"
first_char = word[0]    # 'P'
last_char = word[-1]    # 'n'
third_char = word[2]    # 't'
```

Warning: Accessing an index outside the string range causes an IndexError



Made with GAMMA

Slicing: Extracting Substrings

1

Basic Slicing

```
s = "Python"
s[0:3]  # 'Pyt' (characters from index 0, 1, 2)
s[:4]   # 'Pyth' (start omitted = start from beginning)
s[2:]   # 'thon' (end omitted = go until the end)
```

2

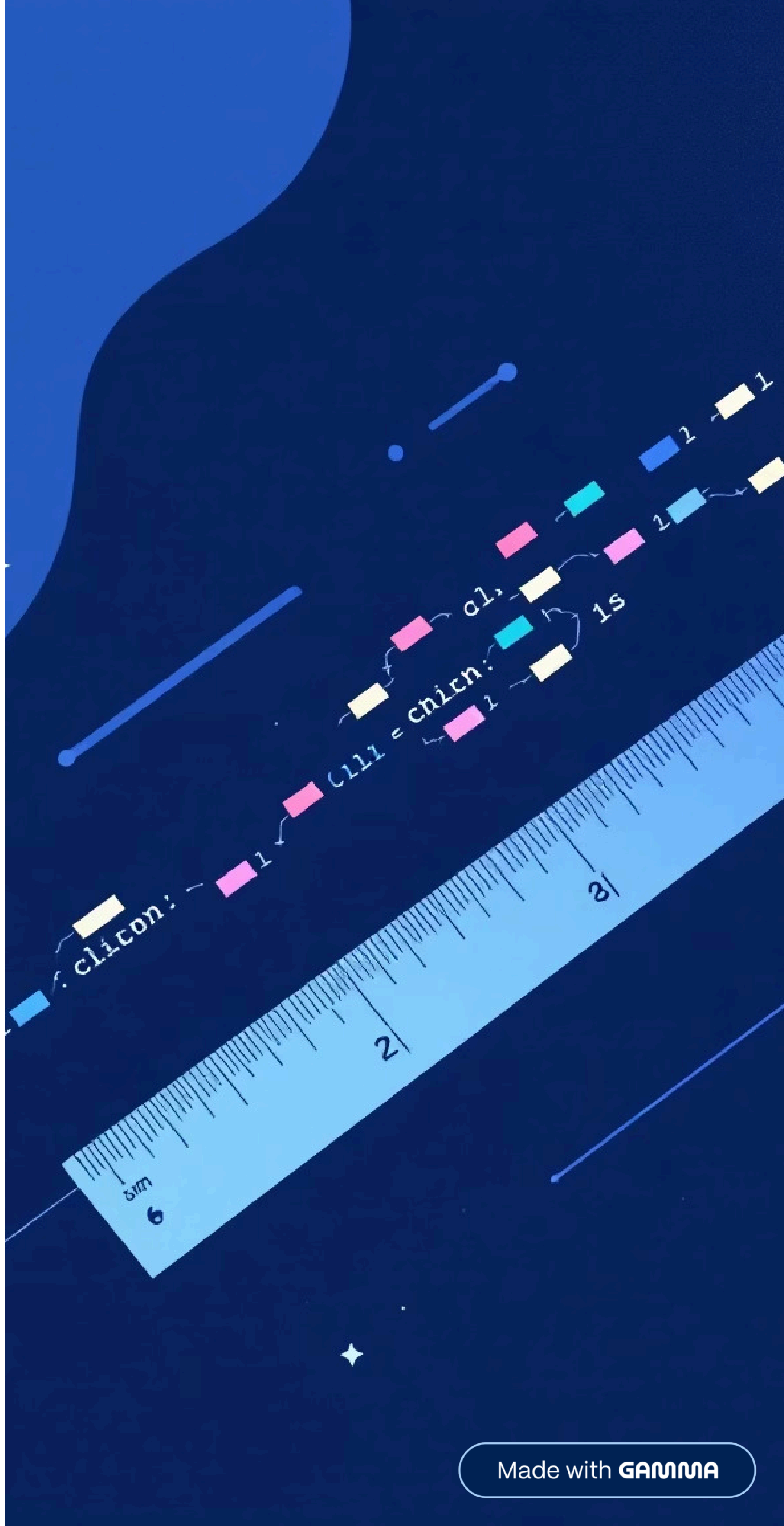
Negative Slicing

```
s = "Python"
s[-3:]  # 'hon' (last 3 characters)
s[:-2]  # 'Pyth' (everything except last 2 characters)
s[-5:-2]  # 'yth' (from 5th-last to 2nd-last)
```

3

Advanced Slicing

```
s = "Python"
s[::2] # 'Pto' (every 2nd character)
s[::-1] # 'nohtyP' (reverse the string)
```



Made with GAMMA

String Immutability & Creating New Strings

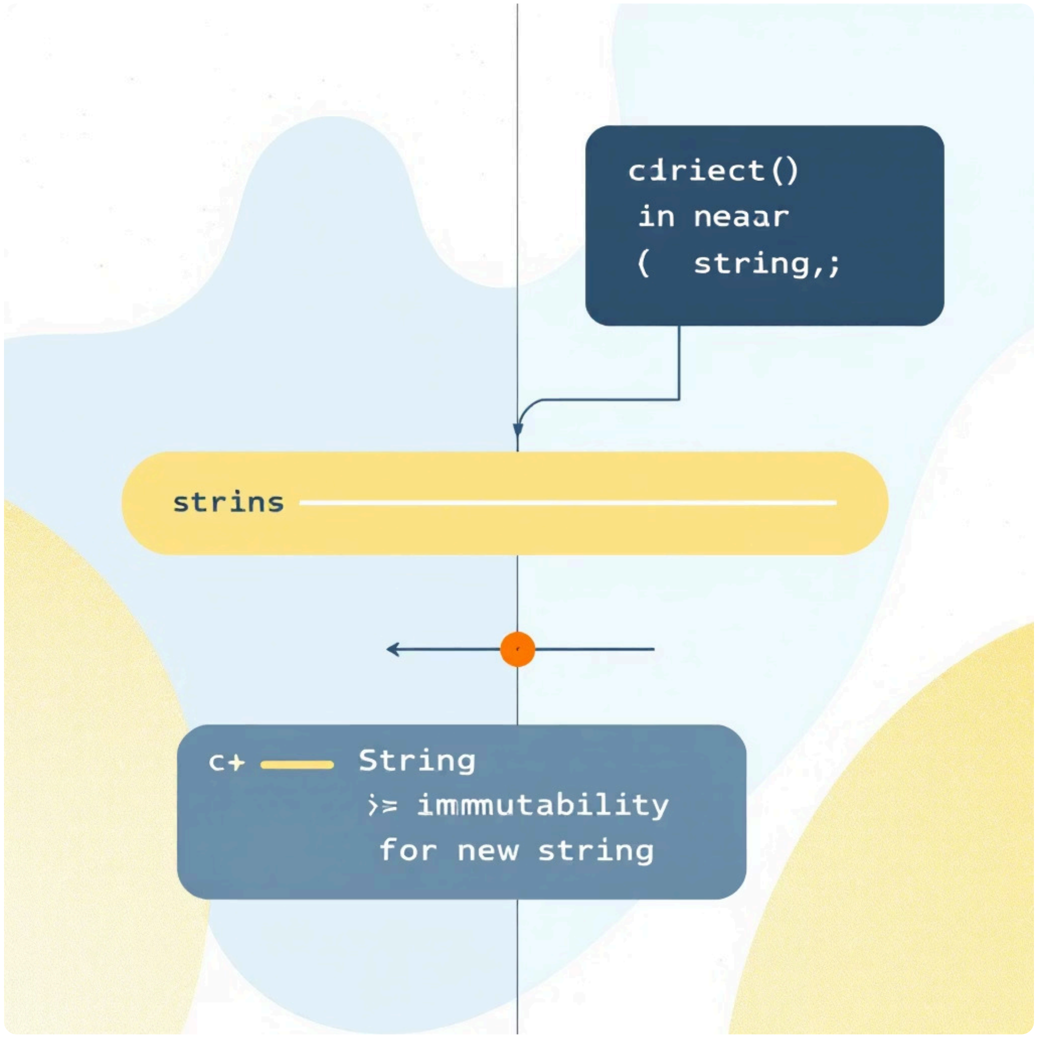
Strings in Python are **immutable** - their content cannot be changed after creation.

```
# This will cause an error
s = "Python"
s[0] = "J"  # TypeError: 'str' object does not support item assignment
```

Instead, create new strings using slicing and concatenation:

```
s = "Python"
new_s = "J" + s[1:]  # "Jython"

name = "John Smith"
first_name = name[:4]  # "John"
last_name = name[5:]   # "Smith"
```



Interacting with User Input

The input() function allows your program to receive text from users:

```
# Basic input
input()
```

Always remember that input() returns a string, even if the user enters numbers:

```
# Getting numeric input
age_str = input("Enter your age: ")
age = int(age_str)  # Convert to integer
```

String Concatenation: Joining Strings

Python makes it easy to combine strings using operators:

+ Operator

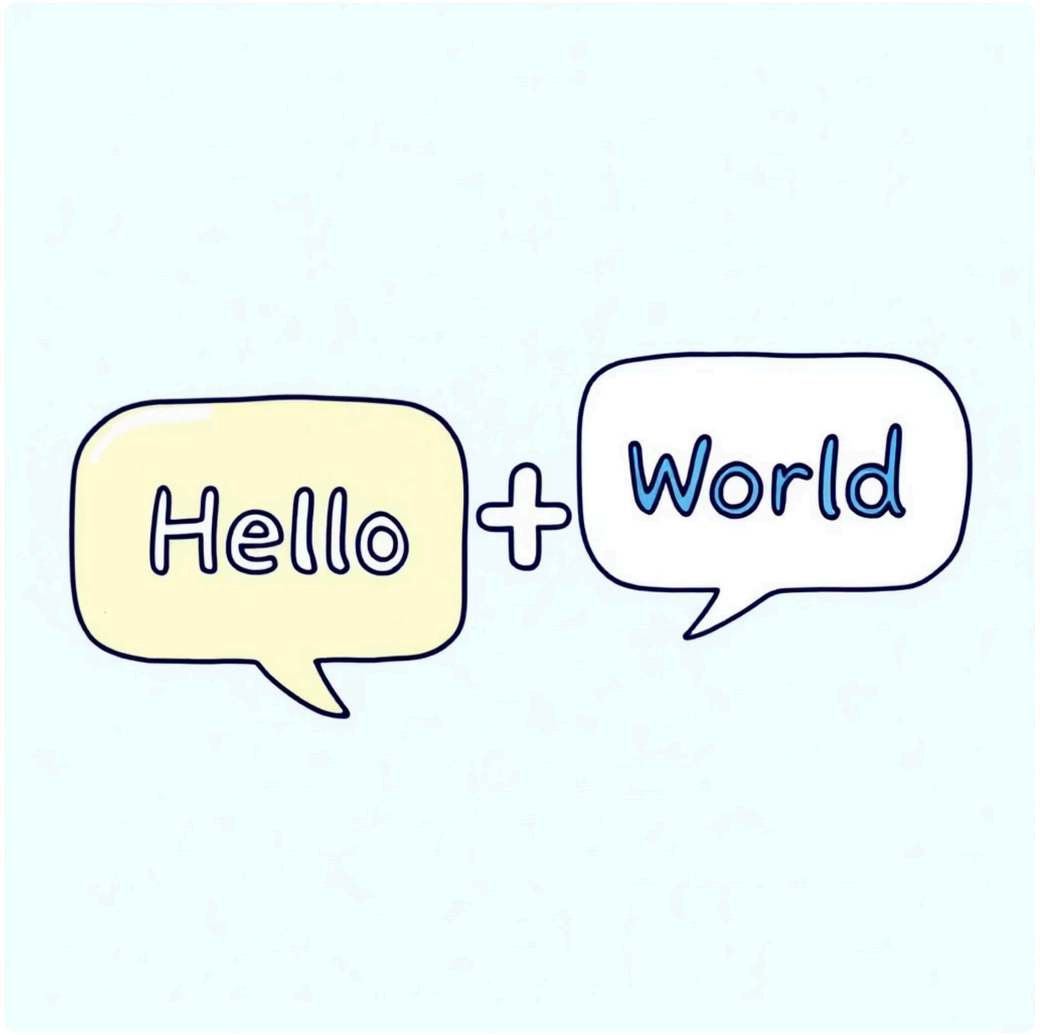
Joins two or more strings together:

```
first = "Hello, "
second = "World!"
greeting = first + second
# Result: "Hello, World!"
```

* Operator

Repeats a string multiple times:

```
cheer = "Hip Hooray! "
crowd_noise = cheer * 3
# Result: "Hip Hooray! Hip Hooray! Hip Hooray! "
```



String Practice Exercises

Basic Exercise

Ask the user for their first and last name, then create a username using the first 3 letters of their first name and the last 2 letters of their last name.

Intermediate Exercise

Create a program that takes a sentence and returns it with all vowels replaced by asterisks (*) using string methods.

Remember to use the concepts we covered today: concatenation, indexing, slicing, and string methods!