Python

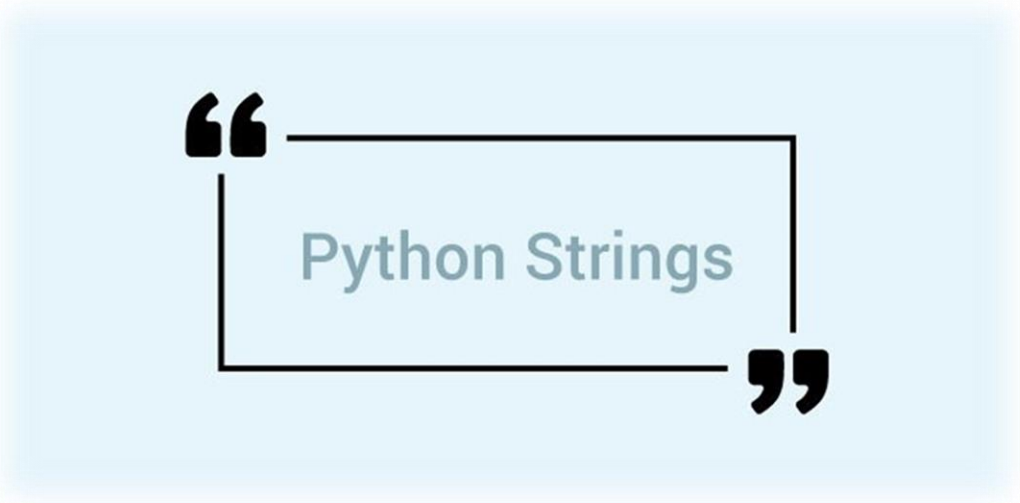
Full stack Skills Bootcamp

Introducing Python Strings

What are Strings?

Strings are sequences of characters enclosed in quotes.

- Used for representing text in Python.
- Syntax: Enclosed in quotes: ‘ ‘, “ “, “”” “””.



Creating and Initializing Strings

- Single Quotes:

```python
single_quote_str = 'Hello, World!'
print("Single Quote String:", single_quote_str)
```

- Double Quotes:

```python
double_quote_str = "Hello, World!"
print("Double Quote String:", double_quote_str)
```

- Triple Quotes (Multi-line Strings):

```python
triple_quote_str = """Hello,
World!
This is a multi-line string."""
print("Triple Quote String:", triple_quote_str)
```

String Concatenation

Combining Strings

- Use the + operator to concatenate strings.

```python
greeting = "Hello"
name = "Alice"
message = greeting + ", " + name + "!"
print("Concatenated String:", message)
```

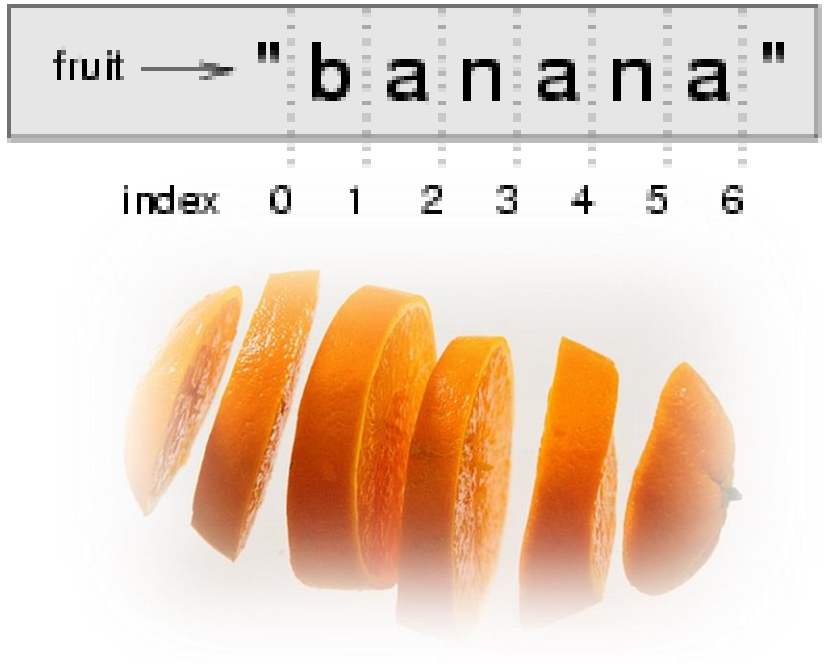
- You can use join() method to concatenate strings from an iterable (like a list).

```python
greeting = "Hello"
name = "Alice"
message = " ".join([greeting, name])
print("Concatenated String using join():", message)
```

String Slicing

- Extracting Substrings
  - Use slicing to access parts of a string.

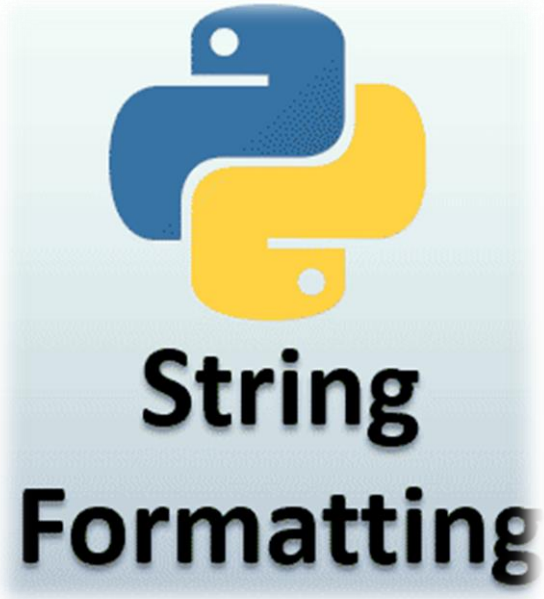
```python
text = "Python Programming"
first_word = text[0:6]  # Extracts "Python"
second_word = text[7:]  # Extracts "Programming"
print("First Word:", first_word)
print("Second Word:", second_word)
```



String Formatting

- Formatting Strings
  - Using f-strings.

```python
age = 30
formatted_string = f"My name is {name} and I am {age} years old."
print("Formatted String (f-string):", formatted_string)
```



  - Using format() Method.

```
formatted_string2 = "My name is {} and I am {} years old.".format(name, age)
```

Common String Methods

- Methods:
  - .upper().
  - .lower().
  - .strip().
  - .replace()

```python
# Common String Methods in Python

# Define a sample text
text = "Python Programming"

# 1. Upper and Lower Case
# Convert to upper case
upper_case_str = text.upper()
print("Upper Case String:", upper_case_str)  # Output: "PYTHON PROGRAMMING"

# Convert to lower case
lower_case_str = text.lower()
print("Lower Case String:", lower_case_str)  # Output: "python programming"

# 2. Stripping Whitespace
# Define a string with extra spaces
whitespace_str = "   Extra spaces   "
# Remove leading and trailing whitespace
stripped_str = whitespace_str.strip()
print("Stripped String:", stripped_str)  # Output: "Extra spaces"

# 3. Replacing Substrings
# Replace "Python" with "JavaScript"
replaced_str = text.replace("Python", "JavaScript")
print("Replaced String:", replaced_str)  # Output: "JavaScript Programming"
```

Conclusion

- Strings in Python:
  - Essential for handling and manipulating text.
  - String operations are crucial for text processing and formatting tasks.