

# Strings



# What are Strings ?

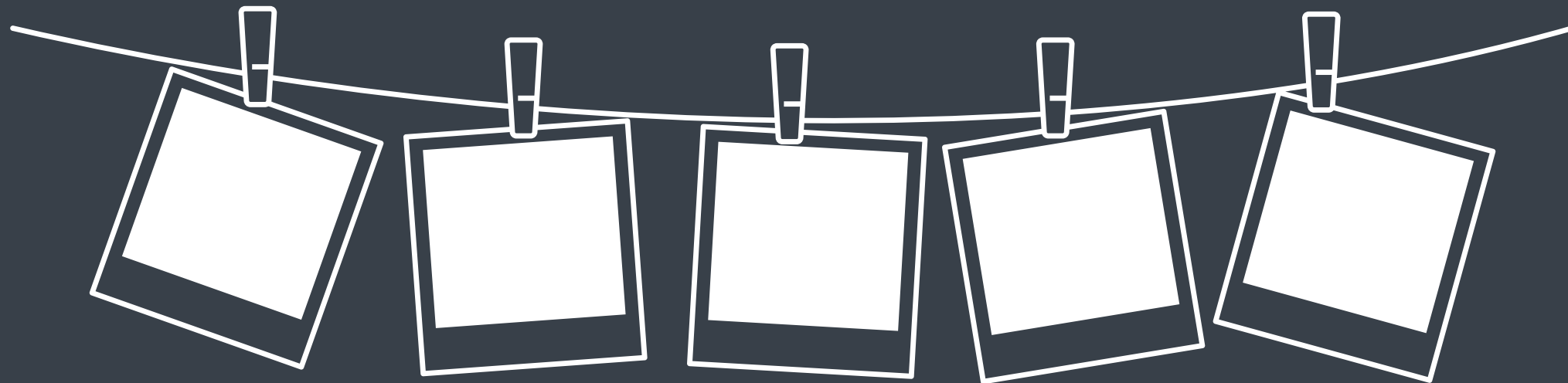
Data type used to represent textual data.

They are sequences of characters and are enclosed in either

**single quotes** (' '),

**double quotes** (" " ),

**triple quotes** (''' '' or '''' ''' )



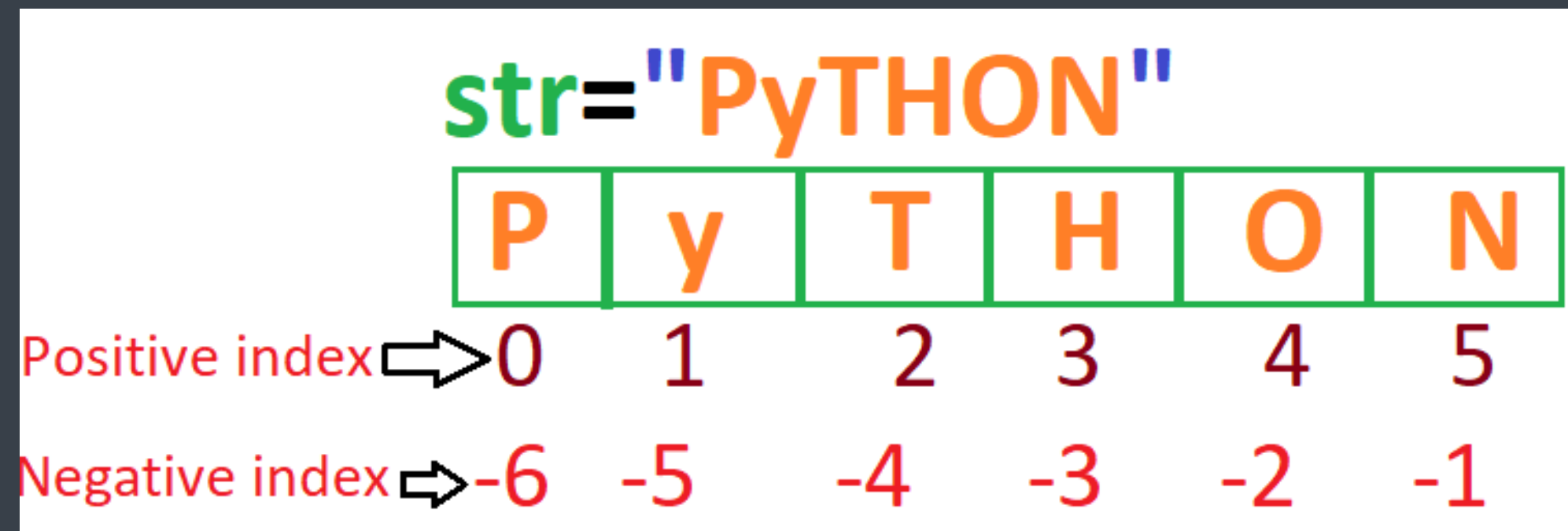
# How to Create String ?

You can create strings using single, double, or triple quotes. Triple quotes are used for multiline strings or to include special characters like line breaks.

- `single_quoted` = 'Hello, world!'
- `double_quoted` = "Hello, world!"
- `multiline` = '''Hello,  
world!  
Welcome'''

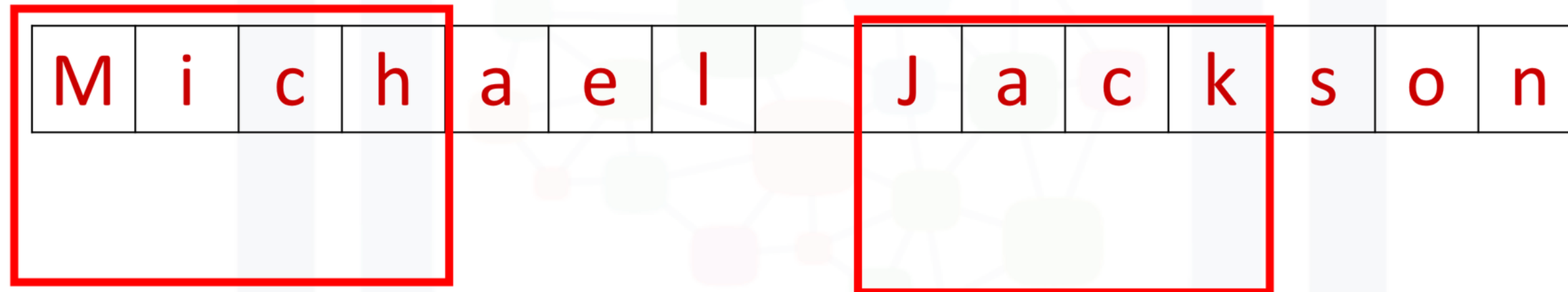
# String Indexing

Strings are ordered sequences, and you can access individual characters using indexing. Python uses zero-based indexing, where the first character has an index of 0.



# String Slicing

Name= "Michael Jackson"



Name[0:4] = Mich

Name[8:12] = Jack

# String Slicing- Exercise

1. `print(s[1])`

2. `print(s[-1])`

3. `print(s[1:3])`

4. `print(s[1:-1])`

5. `print(s[:3])`

6. `print(s[2:])`

7. `print(s[:-1])`

8. `print(s[::2])`

9. `print(s[1::2])`

10. `print(s[::-1])`

`s = 'hello world'`

# String Slicing- Answers

1.e

2.d

3.el

4.ello worl

5.hel

6.llo world

7.hello worl

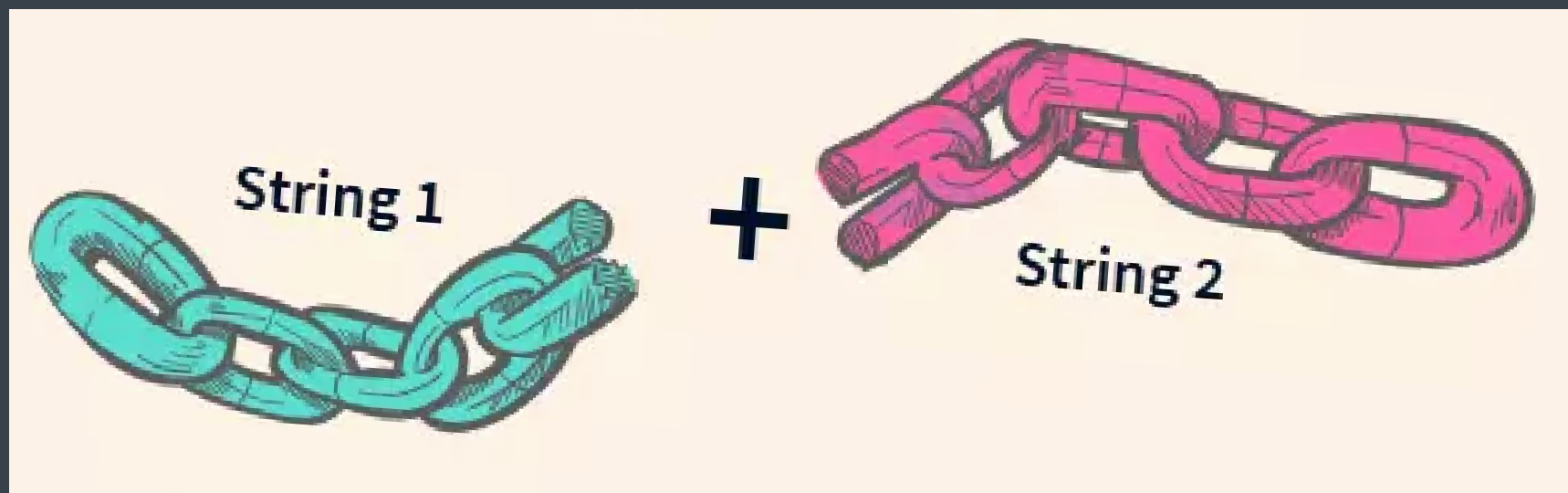
8.hlowrd

9.el ol

10.dlrow olleh

# String Concatenation

You can concatenate strings using the **+** operator:





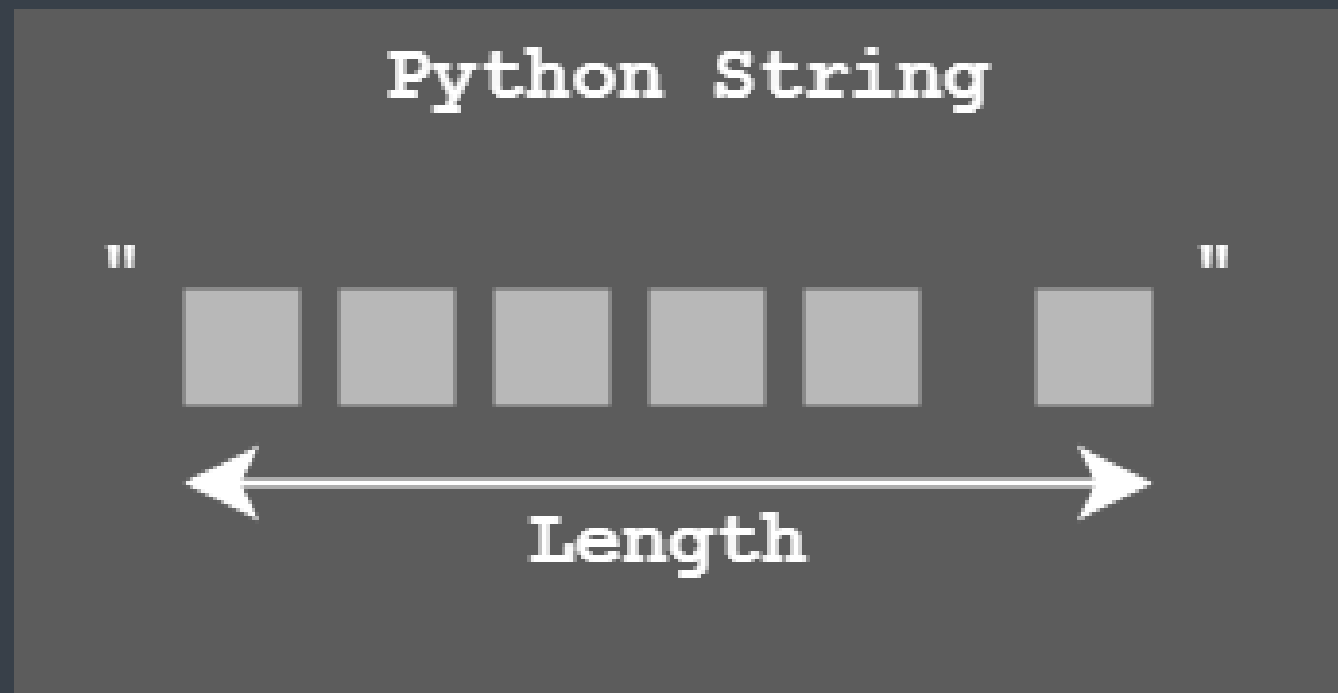
# String Concatenation

python

```
first_name = "John"  
last_name = "Doe"  
full_name = first_name + " " + last_name  
print(full_name) # Output: 'John Doe'
```

# String Length

You can find the length of a string using the `len()` function:



# String Length

python

# Example 1

string1 = "Coding is fun!"

print(len(string1)) # Output: 14

# Example 2

string2 = "Hello, World!"

print(len(string2)) # Output: 13

# Example 3

string3 = "abcdefghijk"

print(len(string3)) # Output: 11

# Example 4

string4 = "The quick brown fox jumps over the lazy dog."

print(len(string4)) # Output: 44

# String Methods

Python provides numerous built-in methods for manipulating strings, such as converting cases, removing whitespaces, replacing characters, splitting, joining, and more.

python

```
# Python String Manipulation Examples
```

```
# Define the original string
```

```
s = "Hello, world!"
```

```
# Convert the string to uppercase
```

```
print(s.upper()) # Output: 'HELLO, WORLD!'
```

```
# Convert the string to lowercase
```

```
print(s.lower()) # Output: 'hello, world!'
```

```
# Remove leading and trailing whitespaces from the string
```

```
print(s.strip()) # Output: 'Hello, world!'
```


```
# Replace all occurrences of 'o' with 'x' in the string
```

```
print(s.replace('o', 'x')) # Output: 'Hellx, wxrld!'
```


```
# Count the number of occurrences of 'a' in the string
```

```
print('Abracadabra'.count('a')) # Output: 4
```

# String Methods

- `str.upper()`
  - `str.lower()`
  - `str.capitalize()`
  - `str.title()`
  - `str.strip()`
  - `str.lstrip()`
  - `str.rstrip()`
  - `str.startswith(prefix).`
  - `str.endswith(suffix)`
  - `str.replace(old, new).`
- 

# String Methods

- `str.split(separator)`
  - `str.join(iterable)`
  - `str.find(substring)`
  - `str.rfind(substring)`
  - `str.index(substring)`
  - `str.rindex(substring)`
  - `str.count(substring)`
  - `str.isalnum()`
  - `str.isalpha()`
- 

# String Methods

- `str.isdigit()`
- `str.islower()`
- `str.isupper()`
- `str.isspace()`
- `str.isnumeric().`
- `str.isdecimal()`
- `str.startswith(prefix, start, end`
- `str.endswith(suffix, start, end)`



# String Formatting

Python supports multiple ways of formatting strings, including old-style % formatting, str.format(), and f-strings (formatted string literals).

```
print ("My name is %s %s and my age is %d" % ("John", "Doe", 45))
```



# String Formatting

```
name = "Alice"
age = 30

# Using the '%' operator for string formatting (old-style)
print("My name is %s and I am %d years old." % (name, age))

# Using the 'format()' method for string formatting
print("My name is {} and I am {} years old.".format(name, age))

# Using f-strings (formatted string literals) for string formatting (Python
print(f"My name is {name} and I am {age} years old.")
```

# Escape sequences

Special character combinations that are used to represent characters that are otherwise difficult or impossible to include directly in a string

- `\b`
- `\f`
- `\n`
- `\r`

- `\'`
- `\t`
- `\"`
- `\\`

# Escape sequences

`\\`: Backslash

`\'`: Single Quote

`\"`: Double Quote

`\n`: Newline (line break)

`\t`: Tab

`\r`: Carriage Return (used for some text file formats)

`\b`: Backspace (moves the cursor back one space)

`\f`: Form Feed (used for some text file formats)

`\v`: Vertical Tab (rarely used)

# Problems On

Strings +  
Numbers +  
Decision Making



# Vowel Counter

Write a program that takes a string input from the user and counts the number of vowels (A, E, I, O, U, and their lowercase equivalents) in the string.

**Sample Input:** "Hello, World!"

**Sample Output:** Number of vowels: 3



# Grade Calculator

Create a program that takes the marks of a student in different subjects as input. Calculate the total marks and average, and then display the corresponding grade based on the average.

**Sample Input:** Marks in Math: 85,  
Marks in Science: 90,  
Marks in English: 78

**Sample Output:** Total Marks: 253,  
Average Marks: 84.33,  
Grade: A



# Palindrome Checker

Write a program that takes a string input from the user and checks if it is a palindrome or not. A palindrome is a word, phrase, number, or sequence of characters that reads the same backward as forward.

**Sample Input:** "radar"

**Sample Output:** It is a palindrome.



# Largest of Three Numbers

Write a program that takes three numbers as input and finds the largest among them using decision-making statements.

**Sample Input:** Enter three numbers: 15, 8, 21

**Sample Output:** The largest number is 21.



# Leap Year Checker

Write a program that takes a year as input and checks if it is a leap year or not.

**Hint:** A leap year is divisible by 4, except for years that are divisible by 100 but not divisible by 400.

**Sample Input:** Enter a year: 2024

**Sample Output:** It is a leap year.

# Temperature Converter:

Build a temperature converter program that allows the user to convert temperatures between Celsius, kelvin and Fahrenheit.

**Sample Input:**      Enter temperature: 32  
                         Enter Units(K or F or C): C

**Sample Output:**  
Temperature in Fahrenheit: 89.6F  
Temperature in Kelvin: 305K