String

Strings in Python are sequences of characters and are defined using either single (') or double (") quotes.

Here are some common string operations in Python:

1. Creation of string
2. Accessing string elements
3. String concatenation
4. String repetition
5. To find string length
6. Iterating over string
7. Extracting substrings using slicing
8. Usage of different string methods
9. Formatting of a string
10. String comparision
11. Conversion of other data types to string
12. Extract raw string

1. Creation of string

In [1]:
```
my_string = "Hello, Python!"
```

2. Accessing string

In [2]:
```
my_string = "Hello, Python!"
print(my_string[0])      # Output: H
print(my_string[7:13])   # Output: Python
```

```
H
Python
```

3. String concatenation

In [4]:
```
# Concatenating strings using the + operator
str1 = "Hello"
str2 = "World"
result_str = str1 + " " + str2
result_str
```

Out[4]: 'Hello World'

4. String repetition

In [6]:
```
# Repeating a string using the * operator.
my_string = "Python"
repeated_string = my_string * 3
repeated_string
```

Out[6]: 'PythonPythonPython'

5. To find string length

In [7]:
```
my_string = "Python"
length = len(my_string)
length
```

Out[7]: 6

6. Iterating over string

In [8]:
```
my_string = "Python"
for char in my_string:
    print(char)
```

```
P
y
t
h
o
n
```

7. Extracting substrings using slicing

In [9]:
```
my_string = "Python"
substring = my_string[1:4]  # Output: yth
substring
```

Out[9]: 'yth'

8. Usage of different string methods

In [10]:
```
# Python provides a variety of built-in string methods for common operations, such
my_string = "   Hello, World!   "
print(my_string.strip())         # Output: "Hello, World!"
```

```
print(my_string.lower())         # Output: "   hello, world!   "
print(my_string.replace('o', '*'))  # Output: "   Hell*, W*rld!   "
```

```
Hello, World!
   hello, world!
   Hell*, W*rld!
```

9. Formatting of a string

In [12]:
```
# Formatting strings using the % operator or the format() method.
name = "Alice"
age = 30
formatted_string = "My name is %s and I am %d years old." % (name, age)
```

In [13]:
```
# Formatting using format() method
name = "Bob"
age = 25
formatted_string = "My name is {} and I am {} years old.".format(name, age)
```

10. String comparision

In [14]:
```
# Comparing strings using comparison operators (`==`, `!=`, `<`, `>`, `<=`, `>=`).
str1 = "apple"
str2 = "orange"
result = str1 == str2  # Output: False
result
```

Out[14]:
```
False
```

11. Conversion of other data types to string

In [15]:
```
# Converting other data types to strings using `str()`.

number = 42
str_number = str(number)
str_number
```

Out[15]:
```
'42'
```

12. Extract raw string

In [16]:
```
# Using raw strings by prefixing the string literal with `r` to treat backslashes a
raw_string = r"C:\Users\Username\Documents"
raw_string
```

Out[16]:
```
'C:\\Users\\Username\\Documents'
```

In [16]: