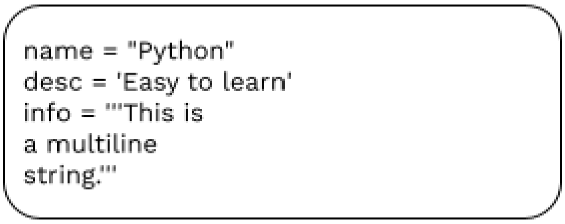
# Chapter : 5 | Strings In Python

## Strings In Python

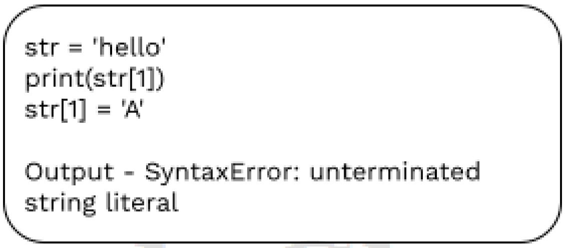
A string is a sequence of characters enclosed in single quotes (' '), double quotes (" "), or triple quotes (''' ''' / """ """).

Example -



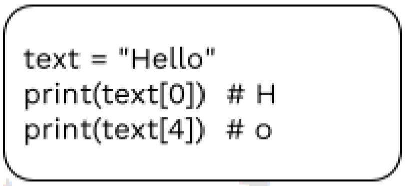
```
name = "Python"
desc = 'Easy to learn'
info = '''This is
a multiline
string.'''
```

- Strings are **immutable**, meaning once created, their characters cannot be changed.



```
str = 'hello'
print(str[1])
str[1] = 'A'

Output - SyntaxError: unterminated
string literal
```



```
text = "Hello"
print(text[0])  # H
print(text[4])  # o
```

FundaClass Academy
Assignment Series

### 1. What are Strings

Strings are indexed collections of characters.They can be :

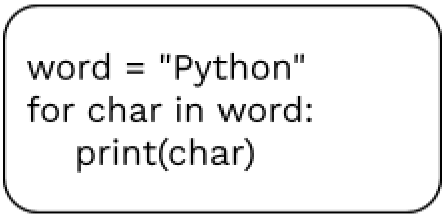
- Created using quotes
- Stored in variables
- Passed to functions

### 2. Creating Strings

You can create strings using quotes or double quotes.

### 3. Traversing a String using Loops

Traversing a string means, accessing each character of a string through for / while loop.
As shown in example.



```
word = "Python"
for char in word:
    print(char)
```

**Practice:**

- Print each character of a user-input string on a new line.
- Count vowels in a given string.

### 4. Special String Operators

| Concatenation (+) | Comparison |
|---|---|
| Repetition (*) | Slicing |
| Membership (in, not in) | |

a. Concatenation (+)

```
Code  -  print("Good" + " " + "Morning")
```

b. Repetition (*)

```
Code  -  print("Hi " * 3)
```

c. Membership (in, not in)

```
print('a' in 'apple')            # True
print('z' not in 'apple')        # True
```

d. Comparison

```
print("Apple" > "Banana")        # False
```

e. Slicing

   i. string[**start : end**]

      1. **start** : The starting index of the slice (inclusive).
      2. **end** : The ending index of the slice (exclusive).

```
my_string = "Hello, World!"
my_string[0:5] would return        "Hello"
my_string[7:] would return         "World!"
my_string[:5] would return         "Hello"
my_string[-6:] would return        "World!"
my_string[::2] would return        "Hlo ol!"
```

FundaClass Academy
Assignment Series

Practice :

- Input a word and print its first 3 and last 3 characters.
- Check whether a given substring is present in a string.

### 5. String Methods and Built-in Functions

| Method | Description | Example |
|---|---|---|
| len(s) | Returns number of characters | len("hi") → 2 |
| capitalize() | Capitalizes 1st letter | "hello".capitalize() → Hello |
| title() | Capitalizes 1st letter of each word | "my python".title() → My Python |
| lower() | Converts to lowercase | "HELLO".lower() → hello |
| upper() | Converts to uppercase | "hello".upper() → HELLO |
| count(sub) | Counts occurrences of a substring | "hello".count('l') → 2 |
| find(sub) | Index of first occurrence (-1 if not) | "apple".find('p') → 1 |
| index(sub) | Same as find() but gives error if not found | "apple".index('p') → 1 |
| endswith(suffix) | Checks if string ends with suffix | "code.py".endswith(".py") → True |
| startswith(prefix) | Checks if string starts with prefix | "code.py".startswith("co") → True |
| isalnum() | Returns True if alphanumeric | "abc123".isalnum() → True |
| isalpha() | True if all characters are alphabetic | "hello".isalpha() → True |

| | | |
|---|---|---|
| isdigit() | True if all digits | "1234".isdigit() → True |
| islower() | True if all lowercase | "hello".islower() → True |
| isupper() | True if all uppercase | "HELLO".isupper() → True |
| isspace() | True if only whitespace | " ".isspace() → True |
| lstrip() | Removes spaces from left | " hello".lstrip() → "hello" |
| rstrip() | Removes spaces from right | "hello ".rstrip() → "hello" |
| strip() | Removes spaces from both ends | " hello ".strip() → "hello" |
| replace(old,new) | Replaces all occurrences of old with new | "apple".replace('p','b') → "abble" |
| join(iterable) | Joins elements with a string separator | ".".join(["a","b","c"]) → a.b.c |
| partition(sep) | Splits into 3 parts around sep | "email@domain.com".partition("@") → ('email', '@', 'domain.com') |
| split(sep) | Splits string into list by separator | "a,b,c".split(",") → ['a', 'b', 'c'] |

FundaClass Academy
Assignment Series

Practice:

- Write a program to count how many times a letter appears in a string.
- Check if a string starts and ends with a vowel.
- Replace all vowels in a string with *.
- Input a list of names separated by commas, split and print each on a new line.
- Ask user to enter a sentence and convert it to title case.
- Join characters of a word with hyphens -.

### 6. Other Functions

```
print(ord('A'))     # 65
print(chr(97))      # a
print(max("abcXYZ"))  # c
print(min("abcXYZ"))  # X
```

**Question - String**

1. Print string in reverse using while loop.
2. Give output - str1 = ‘Hello World’
    a. str1[1:5]        str1[0:10:3]
    b. str1[7:10]        str1[0:10:2]
    c. str1[3:20]        str1[-6:-1]
    d. str1[ :5]        str1[: : -1]
    e. str1[6: ]        str1[: :]
3. Write a program to count the number of times a character (ask from user) occurs in the given string.
4. Write a program which replaces all vowels in the string with '*'.
5. Write a program to input a string from the user and print it in the reverse order without creating a new string using loop.
6. Write a program to check if a string is a palindrome or not. (A string is called palindrome if it reads same backwards as forward. For example, **Kanak** is a palindrome.)
    a. Code -

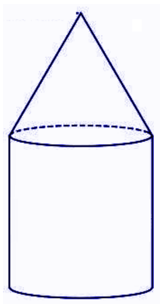
```
str = input('Enter a word : ').lower()

i = 0
j = len(str)-1
status = True
while i<=j:
    if str[i]!=str[j]:
        status = False
        break
    i = i + 1
    j = j - 1

if status:
    print('String : ',str,'is palindrome.')
else:
    print('String : ',str,'is not a palindrome.')
```

FundaClass Academy
Assignment Series

7. Write a program which takes two parameters : one is a string and other is a character. Create a new string after deleting all occurrences of the character from the string and return the new string.
8. Program to calculate the payable amount for the tent.
    a. Accept user requirements for the tent, such as
        i. height
        ii. radius
        iii. slant height of the conical part
    b. Calculate the area of the canvas used
        i. csa_conical : Area of conical part : **π**rl
        ii. csa_cylindrical : Area of cylindrical part : 2**π**rh



iii. canvas_area = csa_conical + csa_cylindrical

c. Calculate the cost of the canvas used for making the tent.

    i. total_cost= unit_price * canvas_area

d. Calculate the net payable amount by the customer that is inclusive of the 18% tax

    i. net_price = total_cost + tax

FundaClass Academy
Assignment Series