Strings

- A String constant is a sequence of characters in single or double quotation marks 'Hello' or "Hello"
- A String can be assigned to a variable that becomes a string variable `str1 = "Hello World"`
- A String is an array of unchangeable characters represented by Unicode values
- Number of characters in the String `len(str1)`
- The characters in a String can be accessed using square brackets as values in a List

```
for i in range(0,len(str1)):
    print(str1[i])
```

```
for char in str1:
    print(char)
```

```
for i in range(-1,-(len(str1)+1),-1):
    print(str1[i])
```

- A sub-string can be obtained by specifying a range of indexes (slice)

`str1[start:end:step]`

`str1[:3]`	`str1[2:]`	`str1[-5:-1]`	`str1[0:10:2]`
"Hel"	"llo World"	"Worl"	"HloWr"

- Check if a sequence of characters exists in a String

```
if "World" in str1 :
    print("World is in the string“)
```

- Strings can be combined using format or the concatenation operator + `str3 = str1 + " " + str(a)`

```
a = 20
print("string1:{}  integer:{}".format(str1,a))
```

```
print(f"string1:{str1}  integer:{a}")
```

Strings

- Some String built-in methods:

string.count(value)	Returns the number of times that value appears in the string
string.find(value)	Searches the string for value and returns its position
string.format()	Formats values in a string
string.isalnum()	Returns True if all characters are alphanumeric
string.isalpha()	Returns True if all characters are in the alphabet
string.isdigit()	Returns True if all characters are digits
string.islower()	Returns True if all characters are lower case
string.isupper()	Returns True if all characters are upper case
string.lower()	Converts a string into lower case
string.replace(str1,str2)	Returns a string where str1 is replaced by str2
string.split(separator)	Splits the string at the separator (default space), and returns a list
string.strip()	Returns the string with the starting and ending spaces removed
string.upper()	Converts a string into upper case

Some Python built-in functions

abs()	Returns the absolute value of the argument
bool()	Returns the boolean value of the argument
chr()	Returns the character corresponding to the Unicode code argument
complex(real, imag)	Returns a complex number with real and imag values
dict()	Returns a dictionary
dir()	Returns a list of the argument's properties and methods
eval()	Evaluates and executes the argument expression
exec()	Executes the argument code
float()	Returns a floating point number of the argument
format()	Formats a specified value
hex()	Converts the argument number into a hexadecimal value
id()	Returns the id of the argument
input()	Reads user input
int()	Returns an integer number of the argument
isinstance(object, type)	Returns True if an object is of type instance
len()	Returns the length of the argument
list()	Returns a list of the argument
max()	Returns the largest item
min()	Returns the smallest item
ord()	Returns the Unicode integer of the character argument
pow(x,y)	Returns the value of x to the power of y
print()	Prints to the output device
range(start,end,inc)	Returns a sequence of numbers, starting from 0 and increments by 1 (by default)
reversed()	Returns the reversed elements of the argument
round(number,decimals)	Rounds a number to the number of decimals (default 0)
set()	Returns a new set
sorted()	Returns a sorted list
str()	Returns a string of the argument
sum()	Sums the elements
tuple()	Returns a tuple
type()	Returns the type of the argument

Examples of string built-in functions

len(*string*)

Retuns the number of characters in the ***string***
Ex: **len**("Good morning") R: **12**

chr(*ascii*)

Retuns the character corresponding to the ***ascii*** code (Unicode)
Ex: **chr**(65) R: **A**

ord(*character*)

Returns the ASCII (Unicode) code of the ***character***
Ex: **ord**("A") R: **65**