String Operators

You can use certain operators for string processing.

+ Concatenation

[+] Appends the second string to the first.

Example

```
>>> a='"Hello, ' #String a
>>> b='Nick",'      #String b
>>> print(a+b+ ' said Harry to Nearly Headless Nick.') #The + here appends the strings a and b to the string in quotes.
"Hello, Nick", said Harry to Nearly Headless Nick.
>>>
```

+ Repetition

[*] Concatenates multiple copies of the same string.

Example

```
>>> a='I must not tell lies…' # String a
>>> print('Again and again Harry wrote the words on the parchment in his own blood – '+a*5) # Here, the * joins or concatenates the string a repeatedly for 5 times.Again and again Harry wrote the words on the parchment in his own blood – I must not tell lies…I must not tell lies…I must not tell lies…I must not tell lies…I must not tell lies…
>>>
```

[] Slice

[] Gives the character at given index.

Example

T	h	e		B	u	r	r	o	w
0	1	2	3	4	5	6	7	8	9

```
>>> a='The Burrow' #String a
>>> a[8] #Here the index number in square brackets [ ] slices out the character at that
```

index, which is o in this example.'o'

>>>

[:] Range Slice

[:] Fetches characters in the range specified by two index operands separated by a colon.

If the first operand is omitted, the range starts at zero index.

If the second operand is omitted, the range goes up to the end of the string.

Note: The slice starts at the first index. The slice ends one index before the second index, that is at the value of the index - 1.

Example

T	h	e		B	u	r	r	o	w
0	1	2	3	4	5	6	7	8	9

>>> a='The Burrow' #String a

>>> a[2:7] #Starting index = 2 = e, Ending index = 7-1 = r'e Bur'

>>> a[:6] #Starting index = 0 = T, Ending index = 6-1 = u'The Bu

'>>> a[5:] #Starting index = 5 = u, Ending index = end of string = w'urrow'

>>>

---

in Membership

[in] Returns true if a character exists in the given string.

Example

>>> a='Harry watched Dumbledore striding up and down in front of him, and thought. He thought of his mother, his father and Sirius. He thought of Cedric Diggory.' #String a

>>> 'v' in a #Checks if the character 'v' is present in the string, aFalse

>>> 'dig' in a #Checks if the characters 'dig' are present in the string, aFalse

>>> 'Dig' in a #Note that this is case-sensitive. As 'Dig' is present in 'Diggory', this returns True.True

>>>

---

not in Membership

[not in] Returns true if a character does not exist in the given string.

Example

>>> a="' “For HIM?” shouted Snape. “Expecto Patronum!"

From the tip of his wand burst the silver doe: She landed on the office floor, bounded once across the office, and soared out of the window. Dumbledore watched her fly away, and as her silvery glow faded he turned back to Snape, and his eyes were full of tears.

“After all this time?”

“Always,” said Snape."' #Multi-line string, a

>>> 'v' not in a #Checks that the character 'v' is not present in the string, a False

>>> 'Red' not in a #Checks that the characters 'Red' iare not present in the string, a True

>>> 'red' notin a #This is case sensitive. Since 'red' is present in 'soared', this returns False. False

>>>