



String Manipulation

सीबीएसई पाठ्यक्रम पर आधारित
कक्षा -11

अध्याय - 5

द्वारा:

संजीव भदौरिया

स्नातकोत्तर शिक्षक (संगणक विज्ञान)

के० वि० बाराबंकी (लखनऊ संभाग)

परिचय

- जैसा की हम सभी जानते है की पाइथन में characters के उस समूह को जो कि (' ' , " " , """ """) single quotes, double quotes या triple quotes से घिरे रहते हैं, उन्हें string कहते हैं।
- पाइथन में strings immutable होती हैं अर्थात इन्हें परिवर्तित नहीं किया जा सकता ।
- String में प्रत्येक अवयव (element) अथवा character की एक अद्वितीय (Unique) स्थिति संख्या (Position number) या इंडेक्स संख्या (index number) होती है जो 0 से शुरू होकर (n-1) तक जाती है ।
- इस अध्याय में हम पाइथन द्वारा प्रदत्त उन तकनीकों के बारे में जानेंगे जिनके द्वारा string manipulation की जाती है ।

String को बनाना

- String को हम निम्न प्रकार बना सकते हैं -

1. Direct variable को value assign करके

```
>>> str="I love my india"  
>>> str  
'I love my india'
```

String Literal

2. Input लेकर

```
>>> str1=input("Enter a string")  
Enter a stringThis is python  
>>> str1  
'This is python'
```

Input() सदैव दिए गए input को string के रूप में ही return करता है |

एक string को travers करना

- किसी भी string के प्रत्येक character को प्राप्त करके उसको प्रयोग करना अथवा उसे प्रदर्शित करना (Display) करना traversing कहलाता है | जैसे

```
name="superb"
```

```
for ch in name:  
    print(ch, "-", end="")
```

Output

s -u -p -e -r -b -

String को उल्टा करके print करने का एक program -

```
str=input("Enter a String")  
print("The string ", str, " in reverse order is: ")  
length=len(str)  
for a in range(-1, (-length-1), -1):  
    print(str[a], end="")
```

Output

Enter a Stringsanjeev

The string sanjeev in reverse order is:

veejnas

String Operators

- String पर 2 operators काम करते हैं + और * | string के केस में इनका काम अलग तरीके से होता है |

» + (इसका प्रयोग दो string को आपस में जोड़ने के लिए लिया जाता है)

- जैसे - "tea" + "pot" हो जायेगा "teapot"
- जैसे - "1" + "2" हो जायेगा "12"
- जैसे - "123" + "abc" हो जायेगा "123abc"

» * (इसका प्रयोग string के कई प्रतिरूप बनाने के लिए करते हैं |)

- जैसे - 5*"@" हो जायेगा "@@@"
- जैसे - "go!" * 3 हो जायेगा "go!go!go!"

नोट : - "5" * "6" ये expression अमान्य है |

String में Membership Operators

- String पर 2 membership operators काम करते हैं in और not in, इन operators को समझने के लिए निम्न उदाहरण देखते हैं -
- in operator का उत्तर True या False में मिलता है - जैसे
 - “a” in “Sanjeev” का उत्तर True मिलेगा |
 - “ap” in “Sanjeev” का उत्तर False मिलेगा |
 - “anj” in “Sanjeev” का उत्तर True मिलेगा |
- not in operator का उत्तर भी True या False में मिलता है |
 - “k” not in “Sanjeev” का उत्तर True मिलेगा |
 - “ap” not in “Sanjeev” का उत्तर True मिलेगा |
 - “anj” not in “Sanjeev” का उत्तर False मिलेगा |

String में Comparison Operators

- निम्न उदाहरण पर ध्यान केन्द्रित करते हैं -
 - "a" == "a" True
 - "abc"=="abc" True
 - "a"!="abc" True
 - "A"=="a" False
 - "abc" == "Abc" False
 - 'a' < 'A' False (क्योंकि lower case की unicode value, upper case से ज्यादा होती है)

Ordinal/Unicode Values कैसे प्राप्त करें?

- निम्न उदाहरण पर ध्यान केन्द्रित करते हैं -

>>>ord('A')	>>>chr(97)
65	a
>>>ord('a')	>>>chr(65)
97	A

String के टुकड़े करना (Slicing)

- निम्न उदाहरण पर ध्यान केन्द्रित करते हैं -

Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Word	R	E	S	P	O	N	S	I	B	I	L	I	T	Y
Reverse index	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

word = "RESPONSIBILITY"

word[0 : 14] का उत्तर आयेगा 'RESPONSIBILITY'

word[0 : 3] का उत्तर आयेगा 'RES'

word[2 : 5] का उत्तर आयेगा 'SPO'

word[-7 : -3] का उत्तर आयेगा 'IBIL'

word[: 14] का उत्तर आयेगा 'RESPONSIBILITY'

word[: 5] का उत्तर आयेगा 'RESPO'

word[3 :] का उत्तर आयेगा 'PONSIBILITY'

String के Functions

<u>String.capitalize()</u>	Converts first character to Capital Letter
<u>String.find()</u>	Returns the Lowest Index of Substring
<u>String.index()</u>	Returns Index of Substring
<u>String.isalnum()</u>	Checks Alphanumeric Character
<u>String.isalpha()</u>	Checks if All Characters are Alphabets
<u>String.isdigit()</u>	Checks Digit Characters
<u>String.islower()</u>	Checks if all Alphabets in a String are Lowercase
<u>String.isupper()</u>	returns if all characters are uppercase characters
<u>String.join()</u>	Returns a Concatenated String
<u>String.lower()</u>	returns lowercased string
<u>String.upper()</u>	returns uppercased string
<u>len()</u>	Returns Length of an Object
<u>ord()</u>	returns Unicode code point for Unicode character
<u>reversed()</u>	returns reversed iterator of a sequence
<u>slice()</u>	creates a slice object specified by range()

String के Program

1. WAP to print the following pattern

1. INDIA	2. I
INDI	IN
IND	IND
IN	INDI
I	INDIA

2. WAP to search a substring from a given line of string.

3. WAP to find the length of a string.

और अधिक पाठ्य-सामग्री हेतु निम्न लिंक पर क्लिक करें -

www.pythontrends.wordpress.com