Programming Fundamentals

Lecture 8 – Python Strings

Iresh Bandara

Learning Outcomes

- This lecture addresses LO1 and LO2 for the module
- On completion of this lecture, students are expected to describe and apply
  - Proper strings usage
  - String slicing
  - Use predefined string methods
  - String comparison

Agenda

- Access characters of a String
- Traversal with a loop
- String slicing
- String is immutable
- Memory address and the value of python variables
- Predefined String methods
- String comparison and ASCII table

Access characters of a String

- Python does not have a character data type
- Bracket operator [] and index to access a character

```
s = “Hello”
print(s[0])# print ‘H’
print(s[2])# print ‘l’

i=3
print(s[i+1]) #print ‘o’

Print(s[-1])  #print ‘o’
```

0	1	2	3	4
H	e	l	l	o
-5	-4	-3	-2	-1

- Similar to the way we access array elements (starting from 0/not 1)

Traversal with a loop

- Can use a while or a for to iterate thorough a String

```
counter =0
word=“Hello”
while counter<len(word):
    print(word[counter])
    counter+=1
```

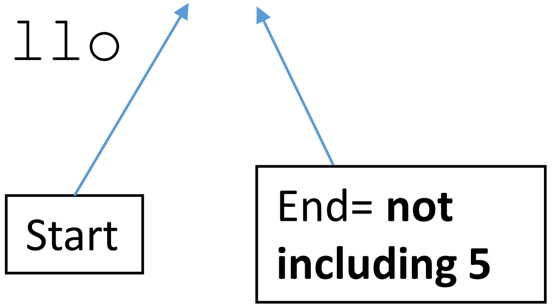
```
word = “Hello”
for i in word:
    print(i)
```

- len() returns the length of the word : in this case length is 5

Slicing a String

- Can use a while or a for to iterate thorough a String

```
word=“HelloWorld”
print(word[2:5])
#print llo
```



Start

End= **not including 5**

```
word=“HelloWorld”
print(word[:3])
#print Hel
```

```
word=“HelloWorld”
print(word[3:])
#print loworld
```

Strings are immutable?

- After a value is assigned to an immutable objects, state cannot be changed
- String, int ,float are immutable types

```
word=“Hello”
print(word) #print Hello
word[0]=“h” # Error – Object does not support
#assignment. Hence immutable

#Solution
newWord=“h”+word[1:]
print(newWord)#print hello
```

Exercise

- id(variable) can be used to check the memory address of the variable

```
 s1=“hello”
s2=“hello”
s3=“world”
print(s1==s2) #True
print(s1==s3) #False
s2=“Hello”
print(s1==s2) #False
```

- Try to understand the relationship between memory address and value

Predefined String methods

- Some of useful String methods

Method	Description
count(“word”)	Returns number of times a certain tag/item was used
replace(“old”,”new”)	Replace all occurrences of “old” with “new” tags
find(“item”)	Returns the index of the item if found, else -1
strip()	Returns the string again after removing leading and trailing whitespaces
capitalize()	Return String first letter capital, rest simple

- Test each String method and see the behaviour

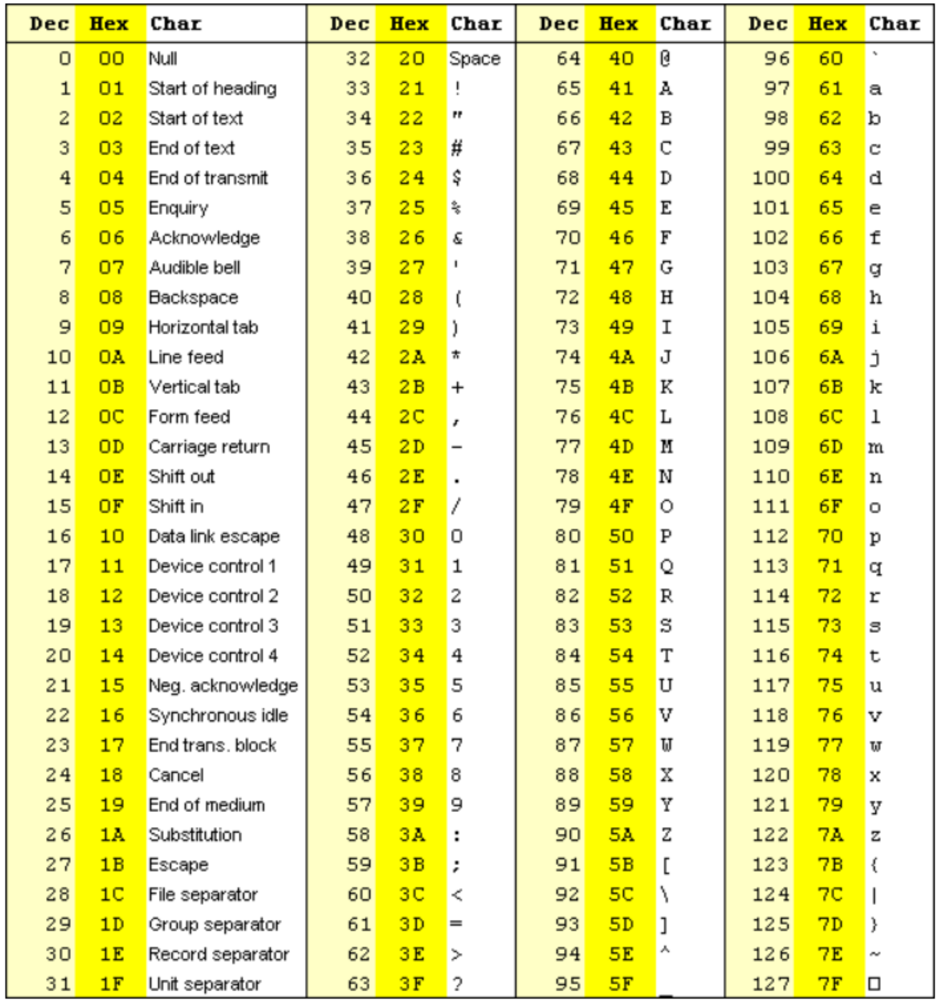
String Comparison

- Check the two strings are equal using “==” and alphabetical order of the Strings using > and <. Capital letters always comes first.

```
greeting='hello‘
greetingNew='Hello‘
if greeting==greetingNew:
    print(“equal")
elif greeting > greetingNew :
      print("greater") #this will execute
elif greeting < greetingNew :
      print("less")
```

ASCII Table

- Simple letter start with 97 (a) and Capital starts with 65 (A)
- Please note that this is part of the ASCII table
  - “Hello” > “hello” = False
  - “hello” < “helloworld” = True
  - “hello” > “hEllo” = True



Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(	72	48	H	104	68	h
9	09	Horizontal tab	41	29	)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

Summary

- [] and index is used to access characters of a String
- WHILE and FOR loops can be used to iterate through a String
- Variations of String slicing was discussed
- Strings, int , float are considered as immutable
- Immutable datatypes cannot change after the initialization
- There is a relationship between variable value and the memory address
- ==,>,< can be used to check the Strings are equal and the alphabetical order
- Alphabetical order is decided using the ASCII table