Python Raw String

Python raw string is created by prefixing a string literal with ‘r’ or ‘R’. Python raw string treats backslash (\) as a literal character. This is useful when we want to have a string that contains backslash and don’t want it to be treated as an escape character.

Example:

```
PyStr='Hi\nHello'
print(PyStr)
```

How raw string helps us in treating backslash as a normal character?

Example:

```
PyStr=r'Hi\nHello'
print(PyStr)
```

Example:

```
PyStr='Hi\xHello'
print(PyStr)#SyntaxError: (unicode error)
```

NOTE: Python doesn’t know how to decode ‘\x’ as it doesn’t have any special meaning

Example:

```
PyStr=r'Hi\xHello'
print(PyStr)
```

Example: On-Console

NOTE:
The output having two backslashes. It’s just to show it as a normal python string where backslash is being escaped.

```
>>> PyStr=r'Hi\xHello'
>>> PyStr
```

Unicode String

Normal strings in Python are stored internally as 16-bit Storing ASCII characters in an 8-bit byte. while Unicode strings are stored as 16-bit Unicode.

Example:

```
str1=r"Normal String"
print("PYTHON Normal String is: ",str1)
str2=u"Unicode String"
print("PYTHON Unicode String is: ",str2)
```

Example: Indexing with Unicode Chracters

```
PyStr=u"हिन्दीहिन्दीहिन्दीहिन्दीहिन्दी"

print(PyStr[0])#ह

print(PyStr[-1])#ी

print(PyStr[4])#द

print(PyStr[-4])#न
```

Example:Slicing with Unicode Characters

```
PyStr=u"हिन्दीहिन्दीहिन्दीहिन्दीहिन्दी"
print(PyStr[0:1])#ह
print(PyStr[1:3])#िन
print(PyStr[4:7])#दीह
```

Example:

```
PyUString=u"नमस्ते आप कैसे हैं?"
print(PyUString)
print(PyUString[0])
print(PyUString[4])
print(PyUString[-4])
print(PyUString[-2])
```

NOTE:
As you can see, Unicode strings use the prefix u, just as raw strings use the prefix r.

PYTHON-String built-in Methods:

capitalize() Method
It returns a copy of the string with only its first character capitalized.

Syntax

```
str.capitalize()
```

Example:

```
str1="naresh i Technologies"
print(str1.capitalize())
str2="HELLO HOW ARE U"
print(str2.capitalize())
```

Python String isdigit() Method
It checks whether the string consists of digits only.

Syntax

```
str.isdigit()
```

Example:

```
str1="PYTHON"
print(str1.isdigit())
str2="12345"
print(str2.isdigit())
str3="Hello123"
print(str3.isdigit())
```

islower() Method
It checks whether all the case-based characters of the string are lowercase.

Syntax

```
str.islower()
```

Example:

```
PyStr="PYTHON"
print(PyStr.islower())
PyStr="python"
print(PyStr.islower())
PyStr="pYthOn"
print(PyStr.islower())
```

len() function:
It returns the length of the string.

Syntax
```
len( str )
```

Example:
```
PyStr="PYTHON"
print(len(PyStr))
PyStr="DataScience is ROCKING"
print(len(PyStr))
```

Joining of Strings:
join() Method
It is a string method and returns a string in which the elements of sequence have been joined by str separator.

Syntax
```
Str.join(iterable)
```

Example:
```
PyList=['1','2','3','4']
Sep="-"
print(Sep.join(PyList))
```

Example:
```
PyList=["H","E","L","L","O"]
Sep="__"
print(Sep.join(PyList))
```