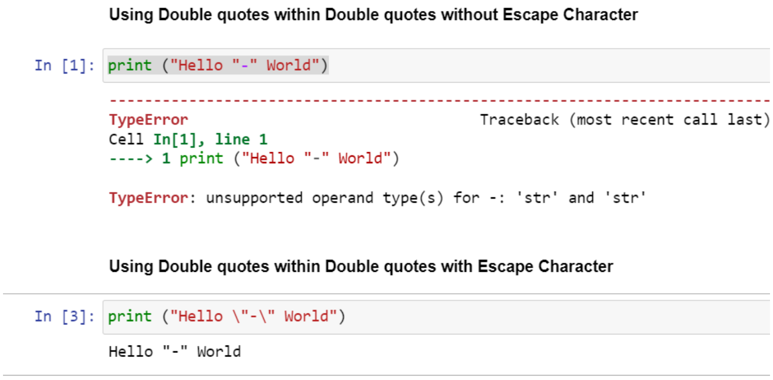
INEURON ASSIGNMENTS

Python Basics : Assignment 6

Question 1: What are escape characters, and how do you use them

Answer:

- Escape characters in python is prefixed with backslash (/) which are used to forcefully insert some characters or line within string.
- Eg., We cannot use double quotes within a string closed with double quotes. Like this: “Hello “-“ world”. But with Escape character, it is possible. Like this: “Hello \” - \” world”. This will also applicable for single quotes.



Using Double quotes within Double quotes without Escape Character

```
In [1]: print ("Hello "-" World")

---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[1], line 1
----> 1 print ("Hello "-" World")

TypeError: unsupported operand type(s) for -: 'str' and 'str'
```

Using Double quotes within Double quotes with Escape Character

```
In [3]: print ("Hello \"-\" World")

Hello "-" World
```

Question 2: What do the escape characters n and t stand for?

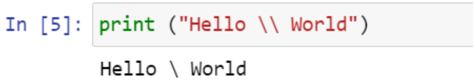
Answer:

- \n – To insert new line
- \t – To insert Tab

Question 3: What is the way to include backslash characters in a string?

Answer:

- Use backslash(\) prefixed with a backslash (\)



```
In [5]: print ("Hello \\ World")

Hello \ World
```

Question 4: The string "Howl's Moving Castle" is a correct value. Why isn't the single quote character in the word Howl's not escaped a problem?

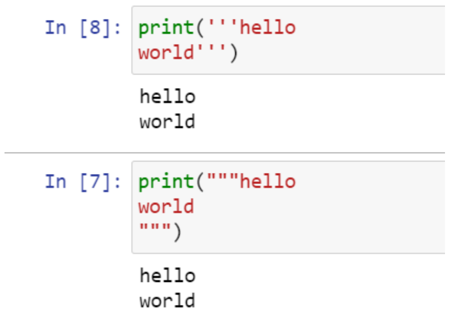
Answer:

- The reason is, python allowed single quotes within the string which covered with double quotes. This is also applicable for using double quotes within a string covered with single quotes

Question 5: How do you write a string of newlines if you don't want to use the n character

Answer:

- Without \n character, String of newlines can be written within ‘’’ ‘’’ triple quotes, either using single or double quotes.



```
In [8]: print('''hello
        world''')

        hello
        world

In [7]: print("""hello
        world
        """)

        hello
        world
```

Question 6: What are the values of the given expressions?

'Hello, world!'[1]

'Hello, world!'[0:5]

'Hello, world!'[:5]

'Hello, world!'[3:]

Answer:

- 'Hello, world!'[1] – 'e'
- 'Hello, world!'[0:5] – 'Hello'
- 'Hello, world!'[:5] - 'Hello'
- 'Hello, world!'[3:] - 'lo, world!'

Question 7: What are the values of the following expressions?

'Hello'.upper()

'Hello'.upper().isupper()

'Hello'.upper().lower()

Answer:

- 'Hello'.upper() → 'HELLO'
- 'Hello'.upper().isupper() → True
- 'Hello'.upper().lower() → 'hello'

Question 8: What are the values of the following expressions

'Remember, remember, the fifth of July.'.split()

'-'.join('There can only one.'.split())

Answer:

Question	Answer
'Remember, remember, the fifth of July.'.split()	['Remember,', 'remember,', 'the', 'fifth', 'of', 'July.']
'-'.join('There can only one.'.split())	'There-can-only-one.'

Question 9: What are the methods for right-justifying, left-justifying, and centering a string?

Answer:

- string.rjust(length, fill-character) - Right justify
- string.ljust(length, fill-character) – Left Justify
- string.center(length, fill-character) - Center

Question 10: What is the best way to remove whitespace characters from the start or end

Answer:

- Using strip() method
  - string.lstrip() -> Remove white space at left side
  - string.rstrip() → To remove white space at right side
  - string.strip() → To remove white space at left side and right side.