1. What are escape characters, and how do you use them?

Answer:

Escape charters represent charters that are present in the string to perform some specific task that is difficult or impossible to type into code. In other words, to insert characters that are unlawful in a string, escape characters are generally used.

Simply add a backslash (\) before the character that we want to escape. Following are the examples of uses of escape charters:

\’ -> single quote

\r -> carriage return

\\ -> backslash

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

Answer:

\n -> new line

\t -> tab

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

Answer:

The backslash in front of another character changes the meaning of that character.

For example, the t is a literal character. If we use the backslash character in front of the letter t, it’ll become the tab character (\t).

Another example, if we have a string that has a single quote inside a single-quoted string like the following string, we need to use the backslash to escape the single quote character:

s = 'It\'s raining'  
print(s)

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:

Here, double quotes are being used to mark the beginning and ending of the string.

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

Answer:

Multiline strings (declared by triple single quotes or triple double quotes) allow us to use newlines in strings without the \n escape character. Here is an example of multiline string:

multiline\_string = """Hello, World!

Happy New Year 2023!

Learning Python Programming Language"""

print(multiline\_string)

6. What are the values of the given expressions?

'Hello, world!'[1]

'Hello, world!'[0:5]

'Hello, world!'[:5]

'Hello, world!'[3:]

Answer:

-> 'e'

-> 'Hello'

-> 'Hello'

-> 'lo, world!'

7. What are the values of the following expressions?

'Hello'.upper()

'Hello'.upper().isupper()

'Hello'.upper().lower()

Answer:

-> 'HELLO'

-> True

-> 'hello'

8. What are the values of the following expressions?

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

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

Answer:

-> ['Remember,', 'remember,', 'the', 'fifth', 'of', 'July.']

-> 'There-can-only-one.'

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

Answer:

Right-justifying a string -> rjust()

Left-justifying a string -> ljust()

Centering a string -> center()

10. What is the best way to remove whitespace characters from the start or end?

Answer:

lstrip() -> remove whitespace from the left or start of a string

rstrip() -> remove whitespace from the right or end of a string