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

Escape characters are special characters that are used to represent certain whitespace characters or to include characters that are otherwise difficult to type directly in a string. They are preceded by a backslash (\). For example, \n represents a newline, and \t represents a tab.

**What do the escape characters \n and \t stand for?**

\n stands for a newline character, which moves the cursor to the beginning of the next line.

\t stands for a tab character, which inserts a horizontal tab space.

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

To include a backslash character in a string, we need to escape it by using a double backslash (\\). For example:

my\_string = "This is a backslash: \\"

**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?**

The single quote in Howl's is not a problem because the string is enclosed in double quotes ("). In Python, we can use single quotes inside a double-quoted string and vice versa without needing to escape them.

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

write a string of newlines by using triple quotes (''' or """), which allow we to include newlines directly in the string:

my\_string = """This is the first line.

This is the second line.

This is the third line."""

**What are the values of the given expressions?**

'Hello, world!'[1]:

The value is 'e' (the character at index 1).

'Hello, world!'[0:5]:

The value is 'Hello' (characters from index 0 to 4).

'Hello, world!'[:5]:

The value is 'Hello' (same as above, characters from the start to index 4).

'Hello, world!'[3:]:

The value is 'lo, world!' (characters from index 3 to the end).

**What are the values of the following expressions?**

'Hello'.upper():

The value is 'HELLO' (converts the string to uppercase).

'Hello'.upper().isupper():

The value is True (checks if the string is in uppercase).

'Hello'.upper().lower():

The value is 'hello' (converts the string to uppercase and then back to lowercase).

**What are the values of the following expressions?**

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

The value is ['Remember,', 'remember,', 'the', 'fifth', 'of', 'July.'] (splits the string into a list of words).

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

The value is 'There-can-only-one.' (splits the string into words and then joins them with hyphens).

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

str.rjust(width): Right-justifies the string in a field of a given width.

str.ljust(width): Left-justifies the string in a field of a given width.

str.center(width): Centers the string in a field of a given width.

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

The best way to remove whitespace from the start and end of a string is to use the str.strip() method:

my\_string = " Hello, world! "

stripped\_string = my\_string.strip()

To remove whitespace only from the start, use str.lstrip(), and to remove whitespace only from the end, use str.rstrip().