Ravikant Diwakar

21051499

CSE-23

**ASSIGNMENT - 1\_HW**

Q1. What are escape characters, and how do you use them?

Ans - Escape characters are special characters that are used to represent certain non-printable or reserved characters within a string. They are preceded by a backslash \.

1. \n: Represents a newline character.

Example :

print("Hello\nWorld")

Output:

Hello

World

1. \t: Represents a tab character.

Example :

print("This is\tindented")

Output:

This is indented

1. \: Represents a literal backslash.

Example :

print("This is a backslash: \\")

Output:

This is a backslash: \

Q2. What do the escape characters n and t stand for?

Ans -\n: Newline character : It is used to insert a new line in a string.

Example :

print("First line\nSecond line")

output:

First line

Second line

\t: Tab character : It is used to insert a horizontal tab in a string. When encountered, it adds space to the text following it.

Example :

print("This is\ttabbed")

Output :

This is tabbed

Q3. What is the way to include backslash characters in a string?

Ans -To include a backslash character (\) in a string, use a double backslash (\\). This is necessary because the backslash is an escape character , and using a single backslash before certain characters is interpreted as a special instruction.

Example :

print("This is a single backslash: \\")

Output :

This is a single backslash: \

Q4. 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?

Ans -If we use single quotes to define a string, we can include double quotes inside the string without any issue. This is because Python allows the use of either type of quote within a string as long as the starting and ending quotes match.

Example :

title = "Howl's Moving Castle"

print(title)

Output :

Howl's Moving Castle

Q5. How do you write a string of newlines if you don't want to use the n character?

Ans - If we want to write a string of newlines without using the \n escape sequence, you can achieve this by using triple-quoted strings. Triple-quoted strings in Python allow you to include multiline text, including newlines, without explicitly using the \n character. You can use triple single quotes (''') or triple double quotes (""") for this purpose.

Example :

multiline\_string = '''This is a string

with multiple lines

without using the \\n character.'''

print(multiline\_string)

Output :

This is a string

with multiple lines

without using the \n character.

Q6. What are the values of the given expressions?

'Hello, world!'[1]

'Hello, world!'[0:5]

'Hello, world!'[:5]

'Hello, world!'[3:]

Ans - print('Hello, world!'[1])

Output: e

print('Hello, world!'[0:5])

Output: Hello

print('Hello, world!'[:5])

Output: Hello

print('Hello, world!'[3:])

Output: lo, world!

Q7. What are the values of the following expressions?

'Hello'.upper()

'Hello'.upper().isupper()

'Hello'.upper().lower()

Ans - print('Hello'.upper())

Output: HELLO

print('Hello'.upper().isupper())

Output: True

print('Hello'.upper().lower())

Output: hello

Q8. What are the values of the following expressions?

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

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

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

Output: ['Remember,', 'remember,', 'the', 'fifth', 'of', 'July.']

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

Output: There-can-only-be-one.

Q9. What are the methods for right-justifying, left-justifying, and centering a string?

Ans - 1. Right-justifying a string : To right-justify strings, use the rjust() function.

1. Left-justifying a string : To left-justify strings, use the ljust() function.
2. Centering a string : To center strings, use the center() function.

Q10. What is the best way to remove whitespace characters from the start or end?

Ans -The best way to remove whitespace characters from the start or end of a string is by using the strip() method. The strip() method removes leading (at the start) and trailing (at the end) whitespace characters, including spaces, tabs, and newline characters.