

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

**Solution 1:** Escape characters represent characters in string values that would otherwise be difficult or impossible to type into code.

- To insert characters that are illegal in a string, use an escape character.
- An escape character is a **backslash \ followed by the character we want to insert.**
- For example the below string will give an error as we cannot use "" inside string's ""

```
txt = "The movie "Spirited away" tells the story of Chihiro Ogino who, enters the world of spirits"
```

*to remove this error we use escape character-*

```
txt = "The movie \"Spirited away\" tells the story of Chihiro Ogino who, enters the world of spirits"
```

**o/p:** The movie \"Spirited away\" tells the story of Chihiro Ogino who, enters the world of spirits"

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

**Solution 2:** \n is used for newline and \t is a tab. For eg:

- ```
txt = "Hello\nWorld!"
print(txt)
o/p : Hello
      World!
```
- ```
txt2 = "Hi\tPython"
print(txt2)
o/p: Hi Python
```

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

**Solution 3:** To use backslash characters in a string we use double \\.

```
txt = "This will insert one \\ (backslash)."
```

```
print(txt)
```

**o/p:** This will insert one \ (backslash).

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?

**Solution 4:** The single quote in Howl's is fine because we have used one ' and used double quotes to mark the beginning and end of the string.

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

**Solution 5:** Multiline strings allow you to use newlines in strings without the \n escape character.

```
txt="""Python is fun.
```

```
It is used by data scientist widely.
```

```
I like programming in Python!"""
```

```
print(txt)
```

**o/p:**

```
Python is fun.
```

```
It is used by data scientist widely.
```

```
I like programming in Python!
```

6. What are the values of the given expressions?

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

7. What are the values of the following expressions?

```
'Hello'.upper():          HELLO
'Hello'.upper().isupper(): True
'Hello'.upper().lower():  hello
```

8. What are the values of the following expressions?

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

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

**Solution 9:** We use string methods:

- right-justifying : `rjust()`
- left-justifying : `ljust()`
- centering : `center()`

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

**Solution 10:** The `lstrip()` and `rstrip()` methods can be used remove whitespace from the left and right ends of a string.

```
x = " Hello! "
x1 = x.strip()           #strip() to remove whitespace from both right and left
print(x1,"How are you?" )
o/p: Hello! How are you?
```

```
x = " Hello! "
x1 = x.lstrip()          #lstrip() to remove whitespace from left
print(x1,"How are you?" )
o/p: Hello! How are you?
```

```
x = " Hello! "
x1 = x.rstrip()          #rstrip() to remove whitespace from both right
print(x1,"How are you?" )
o/p: Hello! How are you?
```