

### Number 1 -

- Question -

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

- Answer -

Escape characters represent characters in string values that would otherwise be difficult or impossible to type into code. The escape character allows you to use double quotes when you normally would not be allowed.

```
In [9]:
data = "We have enrolled in \"iNeuron's\" Full Stack Data Science class."
print(data)
We have enrolled in "iNeuron's" Full Stack Data Science class.
-----
-----
```

### Number 2 -

- Question -

What do the escape characters n and t stand for?

- Answer -

The escape characters n and t stand for newline and tab respectively. \n is a newline; \t is a tab

-----

-----

### Number 3 -

- Question -

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

- Answer -

The \ escape character will represent a backslash character.

-----

-----

### Number 4 -

- Question -

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 single quote in Howl's is fine because we have used double quotes to mark the beginning and end of the string.

-----  
-----

## Number 5 -

- Question -

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

- Answer -

Multiline strings allow us to use newlines in strings without the \n escape character.

-----  
-----

## Number 6 -

- Question -

What are the values of the given expressions?

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

- Answer -

	In [10]:
'Hello, world!'[1]	
	Out[10]:
'e'	
	In [12]:
'Hello, world!'[0:5]	
	Out[12]:
'Hello'	
	In [13]:
'Hello, world!':[5]	
	Out[13]:
'Hello'	
	In [14]:
'Hello, world!'[3:]	
	Out[14]:
'lo, world!'	

---

---

### Number 7 -

- Question -

What are the values of the following expressions?

- 'Hello'.upper()
- 'Hello'.upper().isupper()
- 'Hello'.upper().lower()

- Answer -

<code>'Hello'.upper()</code>	In [15]:
<code>'HELLO'</code>	Out[15]:
<code>'Hello'.upper().isupper()</code>	In [16]:
<code>True</code>	Out[16]:
<code>'Hello'.upper().lower()</code>	In [18]:
<code>'hello'</code>	Out[18]:

---

---

### Number 8 -

- Question -

What are the values of the following expressions?

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

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

- Answer -

<code>'Remember, remember, the fifth of July.'.split()</code>	In [19]:
<code>['Remember,', 'remember,', 'the', 'fifth', 'of', 'July.']</code>	Out[19]:
<code>-''.join('There can only one.'.split())</code>	In [20]:

Out[20]:

```
'There-can-only-one.'
```

---

---

### Number 9 -

- Question -

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

- Answer -

The `rjust()`, `ljust()`, and `center()` string methods, respectively are the methods for right-justifying, left-justifying, and centering a string

---

---

### Number 10 -

- Question -

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

- Answer -

The `lstrip()` and `rstrip()` methods remove whitespace from the left and right ends of a string, respectively.

---

---