1. **Escape characters** are special characters that are used to represent characters that are difficult or impossible to type directly. They are preceded by a backslash (). For example, to include a double quote inside a string, you can use the escape character like this: **\"**.
2. The escape characters **\n** and **\t** stand for:
   * **\n**: Newline character
   * **\t**: Tab character
3. To include a backslash character in a string, you can use a double backslash (**\\**). For example: **"This is a backslash: \\"**.
4. In the string "Howl's Moving Castle," the single quote character in the word "Howl's" is not a problem because the string itself is enclosed in double quotes. If you were using single quotes to define the string, then you would need to escape the single quote inside the string.
5. If you don't want to use the **\n** character for newlines, you can use triple quotes (**'''** or **"""**) to create multiline strings. For example:

pythonCopy code

multiline\_string = '''This is a string with multiple lines.'''

1. Values of the given expressions:
   * **'Hello, world!'[1]**: **'e'**
   * **'Hello, world!'[0:5]**: **'Hello'**
   * **'Hello, world!'[:5]**: **'Hello'**
   * **'Hello, world!'[3:]**: **'lo, world!'**
2. Values of the following expressions:
   * **'Hello'.upper()**: **'HELLO'**
   * **'Hello'.upper().isupper()**: **True**
   * **'Hello'.upper().lower()**: **'hello'**
3. Values of the following expressions:
   * **'Remember, remember, the fifth of July.'.split()**: **['Remember,', 'remember,', 'the', 'fifth', 'of', 'July.']**
   * **'-'.join('There can only be one.'.split())**: **'There-can-only-be-one.'**
4. Methods for justifying a string:
   * Right-justifying: **str.rjust(width)**
   * Left-justifying: **str.ljust(width)**
   * Centering: **str.center(width)**

Example:

pythonCopy code

text = "centered" centered\_text = text.center(20)

1. The best way to remove whitespace characters from the start or end of a string is to use the **strip()** method. For example:

pythonCopy code

text = " Remove whitespaces " trimmed\_text = text.strip()

This will give you the string **"Remove whitespaces"**, removing leading and trailing whitespaces.