Character strings overview:

- We have learned all about the char data type
    - Contains a single character.
- To assign a single character to a char variable, the character is enclosed within a pair of single quotation marks,
  plusSign = ‘+’;

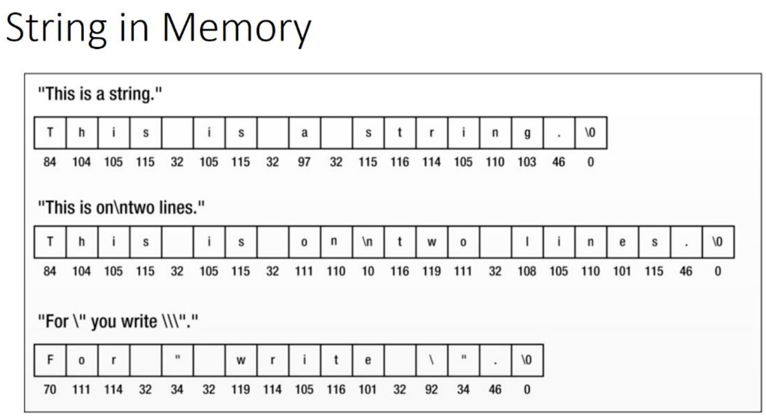
- You have also learned that there is a distinction made between the single quotation and double quotation marks
    - plus Sign = “+”; //incorrect if plusSign is a char

- A string constant or string literal is a sequence of characters or symbols between a pair of double-quote characters.
    - Anything between a pair of double quotes is interpreted by the compiler as a string.
    - Includes any special characters and embedded spaces.
- Everytime you have displayed a message using the printf() function, you have defined the message as a string constant.

```
printf(“This is a string.”);
printf(“This is on\ntwo lines!”);
printf(“for \” you write\\\”.”);
```

- Understand the difference between single quotation and double quotation marks.
    - Both are used to create two different types of constant in C
- For the third example above, you must specify a double quote within a string as the escape sequence \”
    - The compiler will interpret an explicit double quote without a preceding backslash as a string delimiter.

- also , you must also use the escape sequence \\ when you want to include a backslash in a string .
    - A backslash in a string always signal the start of an escape sequence to the compiler.



String in Memory

"This is a string."

T	h	i	s		i	s		a		s	t	r	i	n	g	.	\0
84	104	105	115	32	105	115	32	97	32	115	116	114	105	110	103	46	0

"This is on\ntwo lines."

T	h	i	s		i	s		o	n	\n	t	w	o		l	i	n	e	s	.	\0
84	104	105	115	32	105	115	32	111	110	10	116	119	111	32	108	105	110	101	115	46	0

"For \" you write \\\"."

F	o	r		"		w	r	i	t	e		\	"	.	\0
70	111	114	32	34	32	119	114	105	116	101	32	92	34	46	0

Null character :

- A SPECIAL CHARACTER WITH THE CODE value 0 is added to the end of each string to mark where it ends
    - ➔ This character is known as the null character and you write it as \0
- A string is always terminated by a null character, so the length of a string is always one greater than the number of characters in the string.
- Dont confuse the null character with NULL
    - ➔ Null character is a string terminator.
    - ➔ NULL is a symbol that represents a memory address that doesn’t reference anything.
- You can add a \0 character to the end of a string explictly.
    - ➔ This will create two strings

```
#include <stdio.h>

int main()
{
    printf("The Character \0 is used to terminate a string.");
    return(0);
}
```

- If you compile and run this program, you’ll get this output:
    - ➔ The character
    - ➔ Only the first part of the string has been displayed
    - ➔ Output ends after the first two words because the function stops outputting the string when it reaches the first null character
    - ➔ The second \0 at the end of the string will never be reached
- The first \0 that's found in a character sequence always marks the end of the string.