**ASSIGNMENT**

## PROGRAMMING IN C

1. STRING:

A string is a sequence of characters that is treated as a single data item. C does not support string as a data type,instead it allows us to represent strings as character arrays. A string can be defined as a one dimensional array of characters terminated by a null character ‘/0’. These are often used to create meaningful and readable programs

**Eg: char[]=”Hello World”; char[]=”sb@100”;**

#### 2)STRING HANDLING FUNTIONS

String handling functions can be used to carry out many of the string manipulations. These functions are packed in the **string.h** library. We have to include **string.h** in programs to use these functions. Mostly used string functions are :

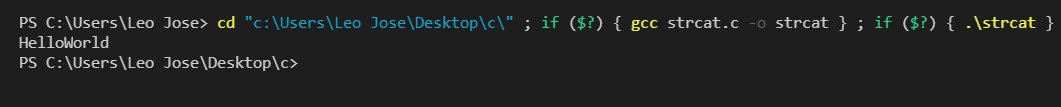
**1.**Strcat**():** It is used to concatenate(combine) two Strings.

Syntax: strcat(str1,str2)

#### CODE



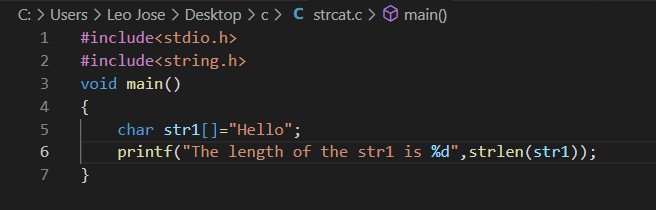
OUTPUT



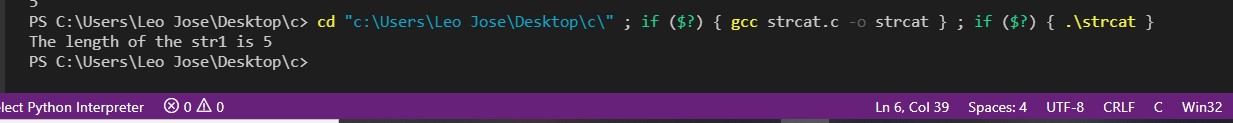
**2**.Strlen**():** It is used to show the length of a string.

Syntax: strlen(str1);

### CODE

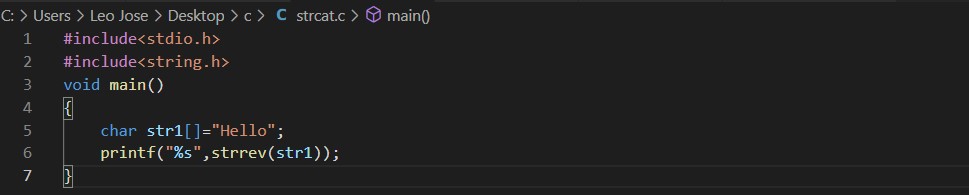


### OUTPUT

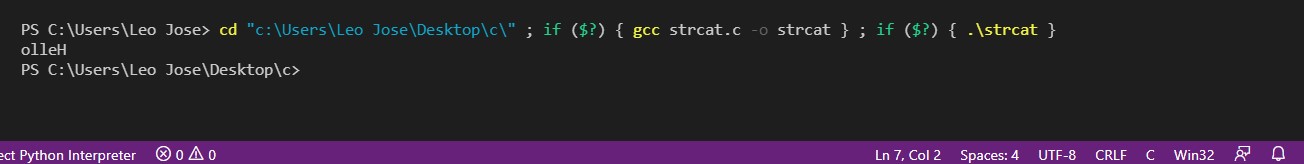


**3.** strrev**() :** It is used to show the reverse of a string. Syntax : strrev(str1)

### CODE

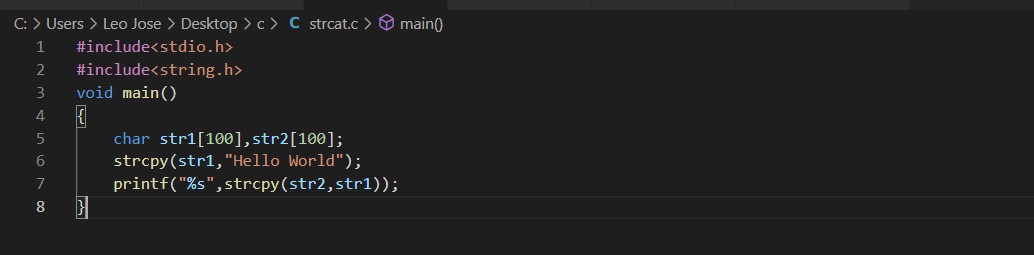


### OUTPUT

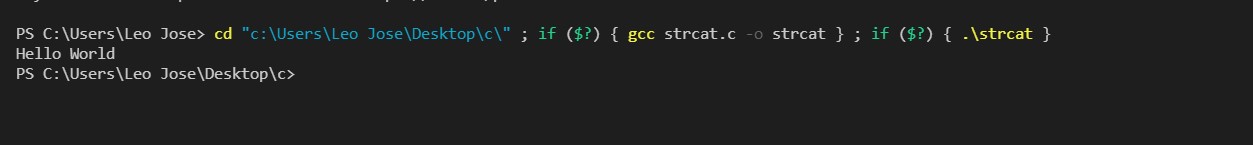


**4.** strcpy**() :** It copies one string into another. Syntax : strcpy(str1,str2)

### CODE



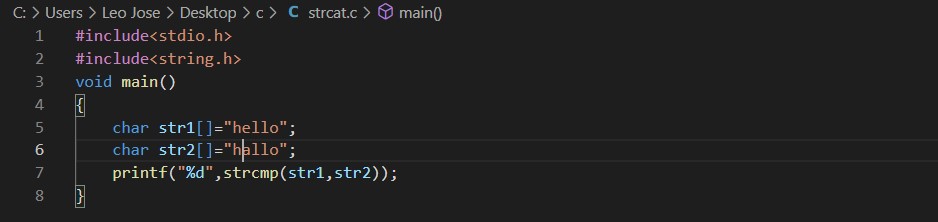
### OUTPUT



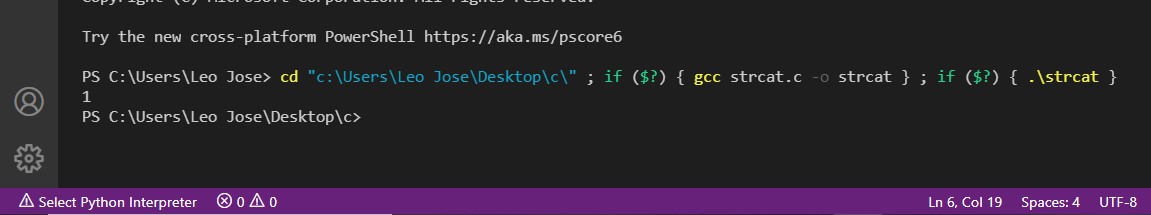
**5.** strcmp**() :** It is used to compare two strings. The strcmp() compares two strings character by character. If the strings are equal, the function returns 0, greater than 0 if the first non-matching character in str1 is greater (in ASCII) than that of str2, less than 0 if the first non-matching character in str1 is lower (in ASCII) than that of str2.

Syntax : strcmp(str1,str2)

### CODE



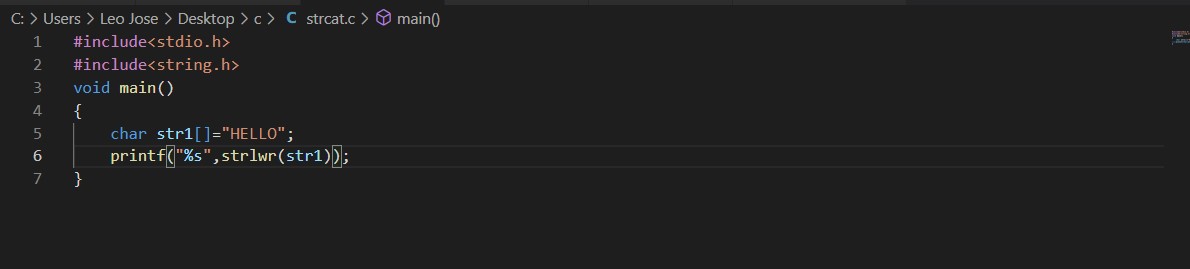
### OUTPUT



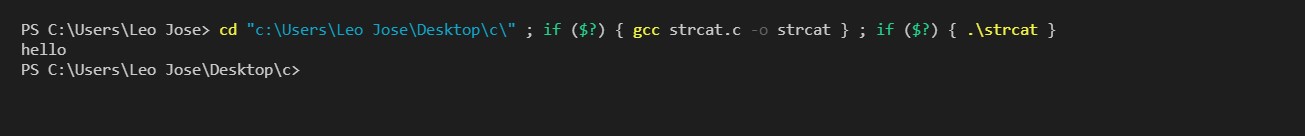
**6.** strlwr**()**

It is used to convert the input to lowercase. Syntax : strlwr(str1)

### CODE



### OUTPUT

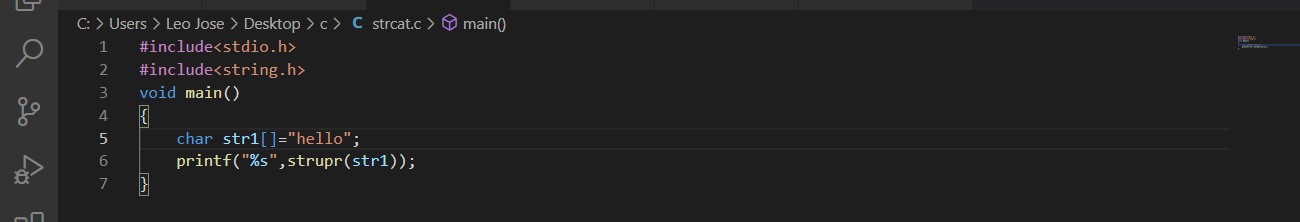


7)strupr**()**

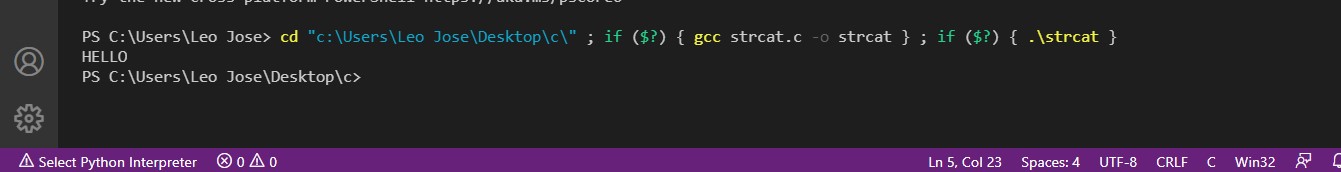
It is used to convert the input to uppercase.

Syntax : strupr(str1)

### CODE



### OUTPUT

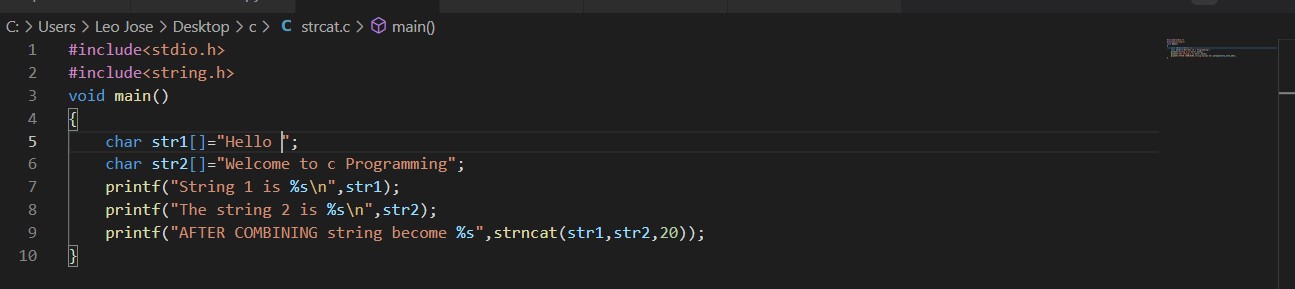


**8)** strncat**()**

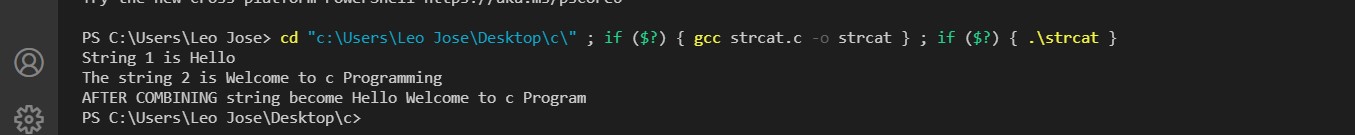
It is used to concatenate n characters of second string to first string.

Syntax : (str1, str2, n)

### CODE



### OUTPUT

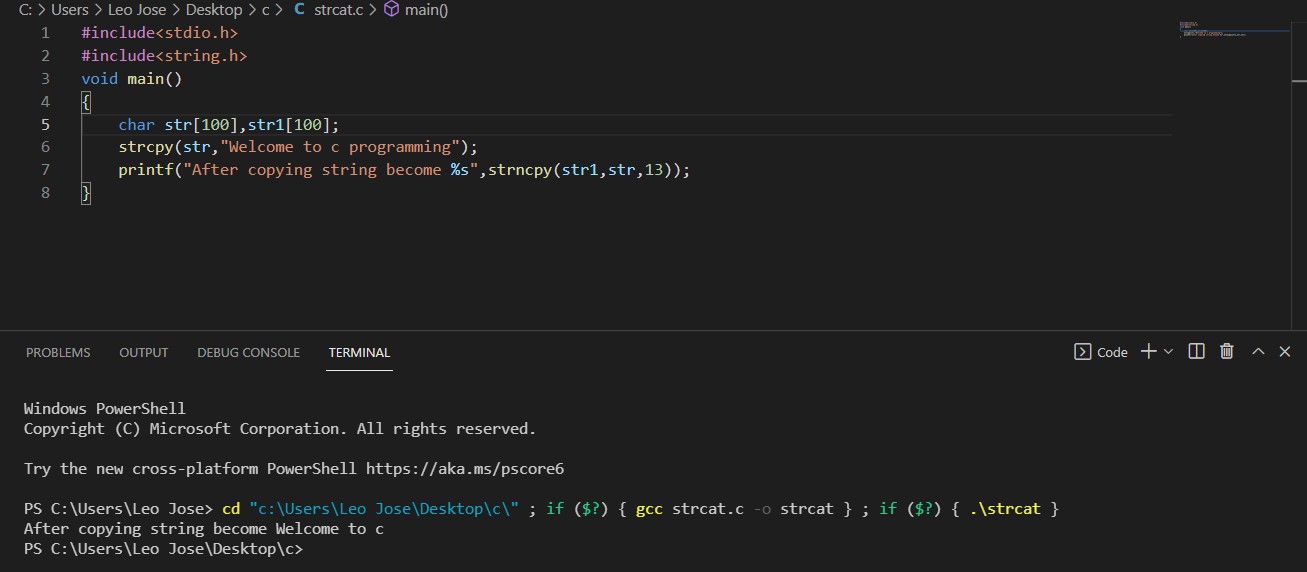


**9**) strncpy**()**

It copies a given number of characters of one string into another.

Syntax : strncpy(str1, str2, n)

### CODE & OUTPUT



**10) strstr()** It returns the pointer of the first occurrence of str2 in str1.

Syntax : strstr(str1,str2)

### CODE & OUTPUT

