

C Library - puts() function

The C library **puts()** function is used to write a string to the standard output (usually the console) followed by a newline character. This function simplifies the process of outputting strings, as it automatically appends a newline at the end, unlike `printf()` which requires explicit inclusion of the newline character.

Syntax

Following is the C library syntax of the **puts()** function –

```
int puts(const char *str);
```

Parameters

This function accepts only a single parameter –

- **str:** A pointer to a null-terminated string that you want to print.

Explore our **latest online courses** and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

Return Value

On success, the `puts()` function returns a non-negative integer and on failure, it returns EOF (End Of File).

Example 1: Basic Usage

This example shows the basic usage of `puts()` to print the string "Hello, World!" followed by a newline.

Below is the illustration of the C library **`puts()`** function.

[Open Compiler](#)

```
#include <stdio.h>

int main() {
    puts("Hello, World!");
    return 0;
}
```

Output

The above code produces the following result—

Hello, World!

Example 2: Handling Empty Strings

This example shows that `puts()` can handle empty strings and will output just a newline when given an empty string.

[Open Compiler](#)

```
#include <stdio.h>

int main() {
    // Prints a newline
    puts("");
    puts("Non-empty string after an empty line");
    return 0;
}
```

Output

After execution of above code, we get the following result —

Non-empty string after an empty line