

puts

Defined in header <stdio.h>

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

Writes every character from the null-terminated string `str` and one additional newline character `'\n'` to the output stream `stdout`, as if by repeatedly executing `fputc`.

The terminating null character from `str` is not written.

Parameters

str - character string to be written

Return value

On success, returns a non-negative value

On failure, returns EOF and sets the *error* indicator (see `ferror()`) on stream.

Notes

The `puts` function appends the newline character to the output, while `fputs` function does not.

Different implementations return different non-negative numbers: some return the last character written, some return the number of characters written (or INT_MAX if the string was longer than that), some simply return a non-negative constant.

A typical cause of failure for `puts` is running out of space on the file system, when `stdout` is redirected to a file.

Example

Run this code

```
#include <stdio.h>

int main(void)
{
    int rc = puts("Hello World");

    if (rc == EOF)
        perror("puts()"); // POSIX requires that errno is set
}
```

Output:

Hello World

References

- C11 standard (ISO/IEC 9899:2011):
 - 7.21.7.9 The `puts` function (p: 333)
- C99 standard (ISO/IEC 9899:1999):
 - 7.19.7.10 The `puts` function (p: 299)
- C89/C90 standard (ISO/IEC 9899:1990):
 - 4.9.7.10 The `puts` function

See also

fputs writes a character string to a file stream
(function)

printf
fprintf
sprintf
snprintf (C99) prints formatted output to stdout, a file stream or a buffer
printf_s (C11) (function)
fprintf_s (C11)
sprintf_s (C11)
snprintf_s (C11)

C++ documentation for **puts**

Retrieved from "<https://en.cppreference.com/mwiki/index.php?title=c/io/puts&oldid=85755>"