

Use `fgets` to Read a String from a Stream

To read a string from a stream, use

```
char* fgets (char* s, int size,
            FILE* stream);
```

- `s` is an array of characters into which the string is stored
- `size` is the size of the array
- `stream` is the stream from which to read
- returns `s` or `NULL` on failure

`fgets` also Stops Reading at End of Line

When does `fgets` stop reading?

At the **first of** the following three

- end of the input (such as a file)
- end of a line (ASCII `0x0A` or `0x0D`)
- end of array `s` (leaving room for a `NUL`).

End of line characters are stored in the array.

`fgets` is the best way to process line-oriented inputs.

Use `fputs` to Write a String to a Stream

To write a string to a stream, use

```
int fputs (const char* s,
          FILE *stream);
```

- `s` is a string
- `stream` is the stream to which to write
- returns non-negative number or `EOF` on failure

`puts` Writes to `stdout` with an End of Line Sequence

There is a shortcut for writing a string to `stdout`:

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

`puts` adds an end of line sequence

- (linefeed, ASCII `0x0A`, on Unix)
- to the end of the string (`fputs` does not).

Do not EVER use shortcut for reading a string from `stdin`. It is a security hazard.