

Function fscanf () in C

The int fscanf function (FILE * stream, const char * format, ...) in Library C reads the input formatted from a Stream.

The **int fscanf** function (**FILE * stream, const char * format, .**) in Library C reads the input formatted from a Stream.

Declare the function fscanf () in C

Below is the declaration for fscanf () function in C:

```
int fscanf ( FILE * stream , const char * format , .)
```

Parameters

stream - This is the pointer to a FILE object that identifies the Stream.

format - This is the string that contains one or more of the following: Whitespace characters, Non-whitespace and Format specifier characters. A format specifier will be the form [= % [*] [width] [modifiers] type =] , explained as follows:

ParametersDescription * This is a starting asterisk (optional) to indicate that the data is read from the Stream but ignored, for example, it is not stored in the corresponding parameter width Specifies the number of characters maximum read in current readings modifiers Define a different size from int (in the case of d, I and n), unsigned int (in the case of o, u and x) or float (in case e, f and g) for data pointed by the corresponding additional parameter: h for int short (for d, i and n), or unsigned short int (for o, u and x); l for int long (for d, i and n), or unsigned long int (for o, u and x), or double (for e, f and g); L for long double (for e, f and g) type A character specifying the type of data to be read and how it is expected to be read. You follow the next table

Type specifier for fscanf function

TypeInput Shock type Single character: Read the next character. If a width other than 1 is specified, this function reads the width of characters and stores them in consecutive array positions that have been passed as parameters. No null characters are appended to the end of char * d Decimal integer numbers: This number is arbitrarily preceded by a + sign or - int * e, E, f, g, G floating point numbers: The decimal number contains a decimal point, arbitrarily preceded by a + or - sign and followed arbitrarily by the letter e or E and a decimal number. Two valid examples are -732.103 and 7.12e4 float * o Octal integer int * s Character string. It will read consecutive characters until a whitespace is found (maybe blank, newline) and tab) char * u Decimal integers unsigned int * x, X Hexadecimal integers segment int *

Additional parameters - Depending on the format format string, this function may have an additional parameter array, each containing a value to be inserted instead of each % -tag specified in the format parameter. , if. This number of parameters should be the same number as % -tags that expect a value.

Returns the value

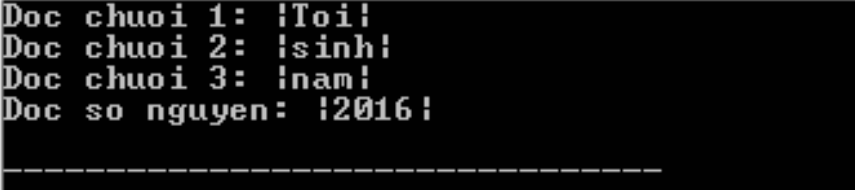
This function returns the number of entries connected and successfully assigned or 0 if the connection failed.

For example

The following program C illustrates the usage of the fscanf () function in C:

```
#include <stdio.h>
int main () { char str1 [ 10 ], str2 [ 10 ], str3 [ 10 ];
```

Compiling and running the above C program will result:



```
Doc chuo i 1: !Toi!
Doc chuo i 2: !sinh!
Doc chuo i 3: !nam!
Doc so nguyen: !2016!
```

According to Tutorialspoint

Previous lesson: The function vsprintf () in C

Next lesson: Function scanf () in C

You finished reading the article "**Function fscanf () in C**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.