

Forum:[Hack Forums](#)

Thread: [\[C\] What's wrong with my code?](#)

.eXe - June 3, 2014, 4:42 am

Hi, I've been practicing C for a couple of days and I'm trying to get my code to read number from a console then perform a function. Here's my code:

```
Code #include <stdio.h>
#include <windows.h>
#include <unistd.h>
```

```
char dofunction() {
    printf("You entered 5!");
    return 0;
}
```

```
int main(void) {
    printf("enter 5:\n");
    int number;
    number = getchar();

    if (number == 5) {
        dofunction();
    } else if (number != 5) {
        printf("you didn't enter the number...");
    }

}
```

But, it's thinking I didn't enter 5.

_SLQ - June 3, 2014, 4:49 am

Try that.

Code (number == '5')
Read the return value. <http://www.cplusplus.com/reference/cstdio/getchar/>

Your ANSI value for the number 5 is not the same as the digit 5.

However, it's recommended you use scanf();

```
int x = 0;
scanf("%d", &x);
```

Spoiler ([Click to View](#)) On success, the character read is returned (promoted to an int value). The return type is int to accommodate for the special value EOF, which indicates failure: If the standard input was at the end-of-file, the function returns EOF and sets the eof indicator (feof) of stdin. If some other reading error happens, the function also returns EOF, but sets its error indicator (ferror) instead.

.eXe - June 3, 2014, 4:50 am

(Jun 3, 2014 04:49 AM)_SLQ Wrote: [\[CITE\]](#) Try that.

Code (number == '5')

Spoiler ([Click to View](#))Read the return value. <http://www.cplusplus.com/reference/cstdio/getchar/>

Yes! Thank you. By the way, is it best to declare a function like the one I just used as "int dofunction()" or "void dofunction()"?

_SLQ - June 3, 2014, 4:52 am

(Jun 3, 2014 04:50 AM)Evil.eXe Wrote: [\[CITE\]](#) Yes! Thank you. By the way, is it best to declare a function like the one I just used as "int dofunction()" or "void dofunction()"?

Depends on the purpose...

For such a short program, you might as well do the printing inside your if statement rather than spawning a new function.

.eXe - June 3, 2014, 4:55 am

(Jun 3, 2014 04:52 AM)_SLQ Wrote: [\[CITE\]](#) Depends on the purpose...

For such a short program, you might as well do the printing inside your if statement rather than spawning a new function.

I was just practicing functions, I know for something like this they aren't necessary, I'm just trying to get used to it.

MrPip - June 3, 2014, 10:44 am

I looked at how you coded all of that just for a simple task, and I wondered. Dude use the scanf function please.

Goodies - June 3, 2014, 12:06 pm

(Jun 3, 2014 10:44 AM)Gibsons Wrote: [\[CITE\]](#) I looked at how you coded all of that just for a simple task, and I wondered. Dude use the scanf function please.

Or how about not because it's inherently insecure? Use a more secure version.

OT:

This may very well not be the best way, but I don't think it's bad and I know it'll work.

PHP Code

```
<?php
#include <stdio.h>
#include <windows.h>
#include <unistd.h>
```

```
void dofunction(){
    printf("You entered 5!\n");
}
```

```
int main(void){
    printf("Enter 5: ");
    long number;
    char *num = calloc(1, sizeof(long));
    number = strtol(fgets(num, 10, stdin), NULL, 10);
```

```
if (number == 5)
    dofunction();
else
    exit(fprintf(stderr, "You didn't enter the number..\n"));
}
```

tux1 - June 3, 2014, 5:51 pm

(Jun 3, 2014 04:50 AM)Evil.eXe Wrote: [\[CITE\]](#) Yes! Thank you. By the way, is it best to declare a function like the one I just used as "int dofunction()" or "void dofunction()"?

A function always uses this structure:

Quote: return-type function-name(parameter declarations, if any)

```
{
declarations
statements
}
```

So the first part will dictate what the function returns. In this case you could use void because it doesn't return any useful information. You could use int to return 0 because 0 is an integer.