



Menu ▼

Log in



HTML

CSS



C Function Declaration and Definition

[< Previous](#)[Next >](#)

Function Declaration and Definition

You just learned from the previous chapters that you can create and call a function in the following way:

Example

```
// Create a function
void myFunction() {
    printf("I just got executed!");
}

int main() {
    myFunction(); // call the function
    return 0;
}
```

[Try it Yourself »](#)

A function consist of two parts:

- **Declaration:** the function's name, return type, and parameters (if any)
- **Definition:** the body of the function (code to be executed)

```
void myFunction() { // declaration
    // the body of the function (definition)
}
```

For code optimization, it is recommended to separate the declaration and the definition of the function.

You will often see C programs that have function declaration above `main()` , and function definition below `main()` . This will make the code better organized and easier to read:

Example

```
// Function declaration
void myFunction();

// The main method
int main() {
    myFunction(); // call the function
    return 0;
}

// Function definition
void myFunction() {
    printf("I just got executed!");
}
```

Try it Yourself »

Another Example

If we use the example from the previous chapter regarding function parameters and return values:

Example

```
int myFunction(int x, int y) {  
    return x + y;  
}  
  
int main() {  
    int result = myFunction(5, 3);  
    printf("Result is = %d", result);  
  
    return 0;  
}  
// Outputs 8 (5 + 3)
```

Try it Yourself »

It is considered good practice to write it like this instead:

Example

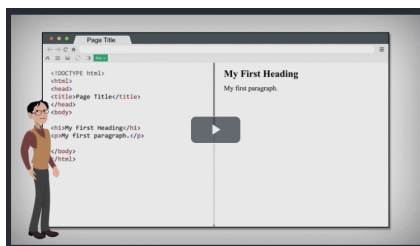
```
// Function declaration  
int myFunction(int, int);  
  
// The main method  
int main() {  
    int result = myFunction(5, 3); // call the function  
    printf("Result is = %d", result);  
  
    return 0;  
}  
  
// Function definition  
int myFunction(int x, int y) {  
    return x + y;  
}
```

[Try it Yourself »](#)[◀ Previous](#)[Next ▶](#)

ADVERTISEMENT

NEW

We just launched
W3Schools videos

[Explore now](#)

COLOR PICKER



Get certified
by completing

a course today!



Get started

CODE GAME



Play Game

ADVERTISEMENT



ADVERTISEMENT

ADVERTISEMENT

[Report Error](#)

[Spaces](#)

[Pro](#)

[Get Certified](#)

Top Tutorials

[HTML Tutorial](#)
[CSS Tutorial](#)
[JavaScript Tutorial](#)
[How To Tutorial](#)
[SQL Tutorial](#)
[Python Tutorial](#)
[W3.CSS Tutorial](#)
[Bootstrap Tutorial](#)
[PHP Tutorial](#)
[Java Tutorial](#)
[C++ Tutorial](#)
[jQuery Tutorial](#)

Top References

[HTML Reference](#)
[CSS Reference](#)
[JavaScript Reference](#)
[SQL Reference](#)
[Python Reference](#)
[W3.CSS Reference](#)
[Bootstrap Reference](#)
[PHP Reference](#)
[HTML Colors](#)
[Java Reference](#)
[Angular Reference](#)
[jQuery Reference](#)

Top Examples

[HTML Examples](#)
[CSS Examples](#)
[JavaScript Examples](#)
[How To Examples](#)
[SQL Examples](#)
[Python Examples](#)
[W3.CSS Examples](#)
[Bootstrap Examples](#)
[PHP Examples](#)
[Java Examples](#)

[XML Examples](#)
[jQuery Examples](#)

Get Certified

[HTML Certificate](#)
[CSS Certificate](#)
[JavaScript Certificate](#)
[Front End Certificate](#)
[SQL Certificate](#)
[Python Certificate](#)
[PHP Certificate](#)
[jQuery Certificate](#)
[Java Certificate](#)
[C++ Certificate](#)
[C# Certificate](#)
[XML Certificate](#)

[FORUM](#) | [ABOUT](#)

W3Schools is optimized for learning and training. Examples might be simplified to improve reading and learning. Tutorials, references, and examples are constantly reviewed to avoid errors, but we cannot warrant full correctness of all content. While using W3Schools, you agree to have read and accepted our terms of use, cookie and privacy policy.

Copyright 1999-2022 by Refsnes Data. All Rights Reserved.
W3Schools is Powered by W3.CSS.

