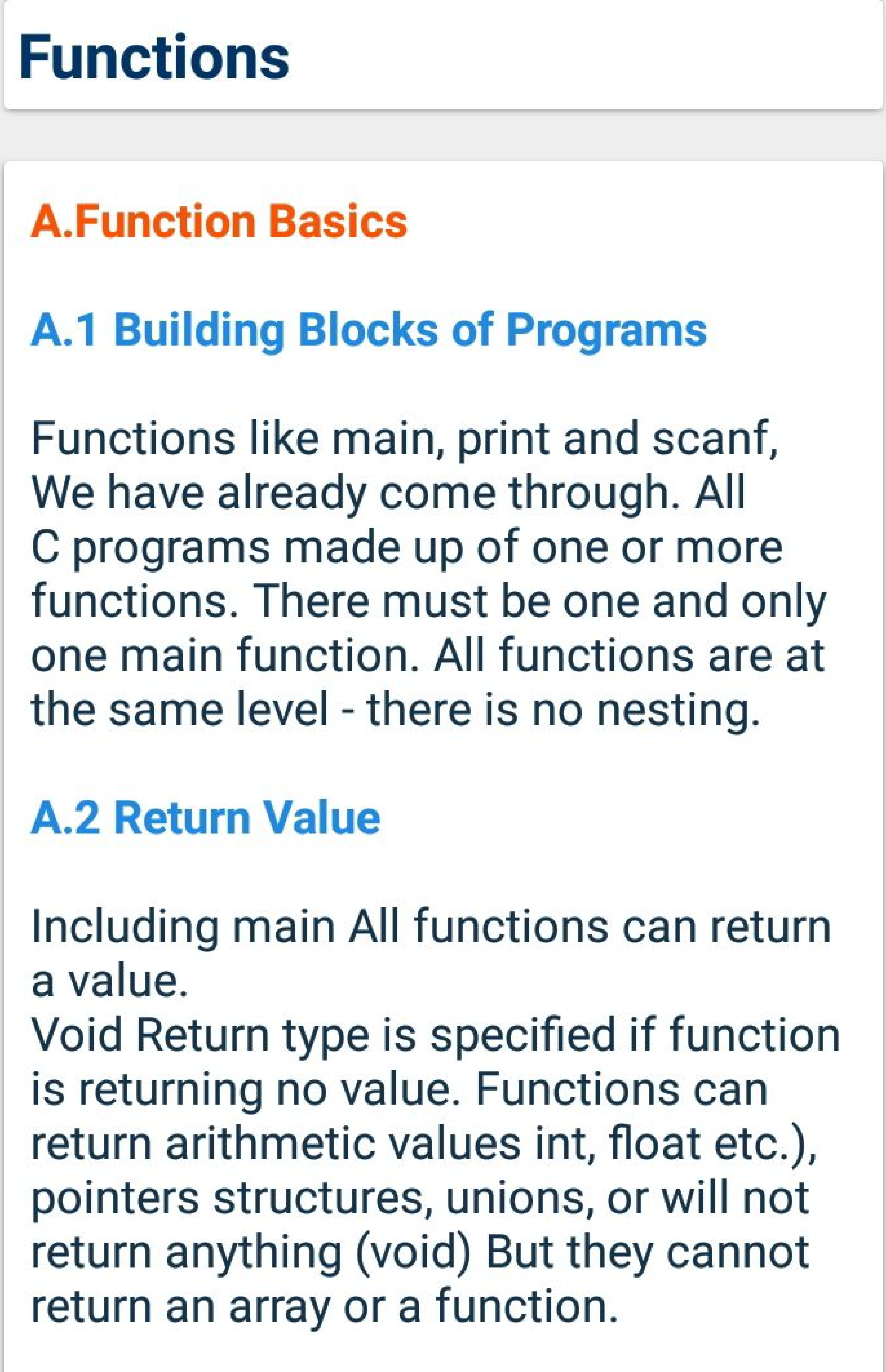


Functions

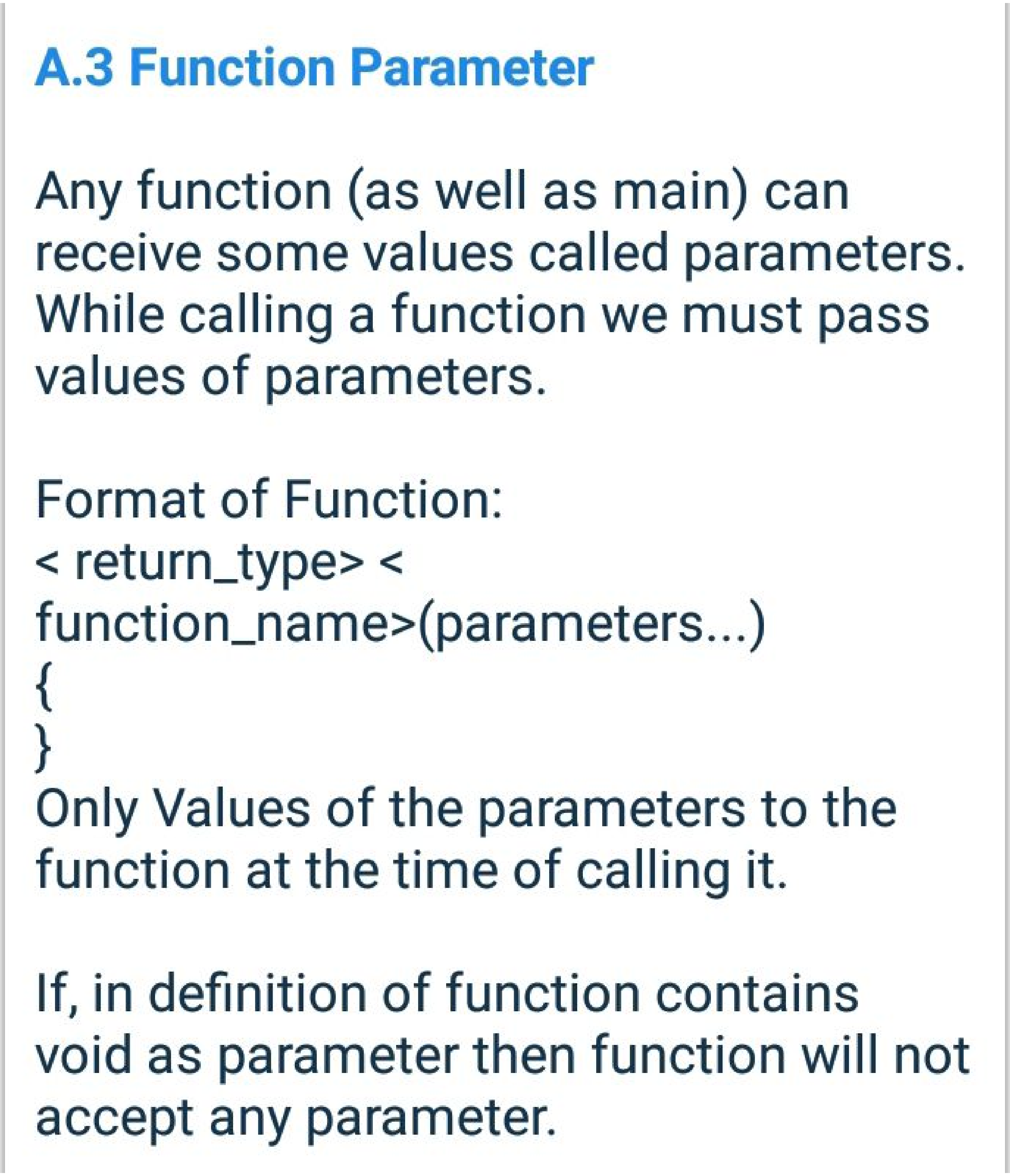
A.Function Basics

A.1 Building Blocks of Programs

Functions like main, print and scanf, We have already come through. All C programs made up of one or more functions. There must be one and only one main function. All functions are at the same level - there is no nesting.

A.2 Return Value

Including main All functions can return a value.
Void Return type is specified if function is returning no value. Functions can return arithmetic values int, float etc.), pointers structures, unions, or will not return anything (void) But they cannot return an array or a function.



A.3 Function Parameter

Any function (as well as main) can receive some values called parameters. While calling a function we must pass values of parameters.

Format of Function:

```
< return_type> <
function_name>(parameters...)
{
}
```

Only Values of the parameters to the function at the time of calling it.

If, in definition of function contains void as parameter then function will not accept any parameter.

B.Defination & Declaration

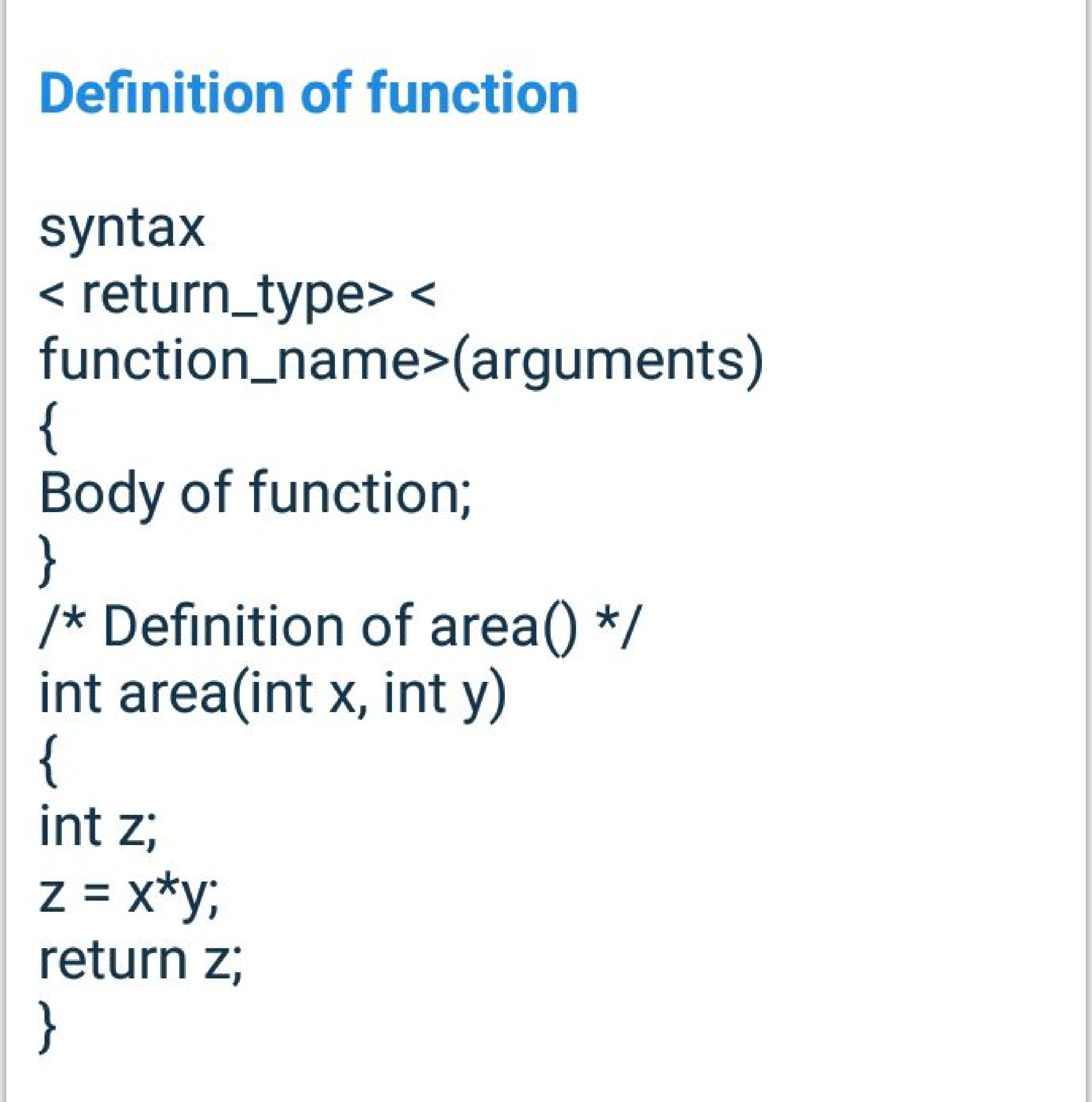
A function definition contains function name, parameters, its code and return type and A function declaration contains only name and return type. User can define a function only once but it can be declared several times.

declaration of function

Syntax :
< return_type> (arguments...);

```
/* Declaration of area() * /

int area(int x, int y);
int main()
{
int x=10, y=25;
printf("%d\n",area(x,y)));
return 0;
}
```



Definition of function

syntax

```
< return_type> <
function_name>(arguments)
{
Body of function;
}
/* Definition of area() */
int area(int x, int y)
{
int z;
z = x*y;
return z;
}
```