Functions - Arguments and Parameters

- A parameter is a variable in a function declaration and function definition/implementation.
- When a function is called, the arguments are the data you pass into the functions parameters.
    - ➔ The actual value of a variable that gets passed to the function.
- Function parameters are defined within the function header,
    - ➔ Are placeholders for the arguments that need to be specified when the function is called.
- The parameters for a function are a list of parameter names with their types.
    - ➔ Each parameter is separated by comma.
    - ➔ Entire list of parameters is enclosed between the parentheses that follow the function name .
- A function can have no parameters, in which case you should put void between the parentheses,
- Parameters provide the means to pass data to a function.
    - ➔ Data passed from the calling function to the function that is called,
- The Body of the function should use these parameters in its implementation
- A function body may have additional locally defined variables that are needed by the function's implementation.
- When passing an array as an argument to a function,
    - ➔ You must also pass an additional argument specifying the size of the array.
    - ➔ The function has no means of knowing how many elements there are in the array.
- When the printf() function is called, you always supply one or more values as arguments.
    - ➔ First value being the format string
    - ➔ The remaining values being any variable to displayed.
- Parameters greatly increase the usefulness and flexibility of a function.
    - ➔ The printf() function displays whatever you tell it to display via the parameters and arguments passed.
- It is a good idea to add comments before each of your own functions definitions.
    - ➔ Help explain what the function does and how the arguments are to be used.

Example

```
#include <stdio.h>

void multiplyTwoNumbers (int x, int y)
{
   int retult = x * y;
   printf (“The product of %d multiplied by %d is: %d\n”, x, y, result);
}

int main (void)
{
   multiplyTwoNumbers (10, 20);
   multiplyTwoNumbers (20, 30);
   multiplyTwoNumbers (50, 2);

   return 0;
}
```