Parameters are an essential part of writing code. They are used to pass vital information between functions, or in other words to provide external data to the method the parameter is passed to. A formal parameter is a variable that has been declared in a function definition. An actual parameter is the actual input that is passed as a parameter to the function call at the moment it is initialised. An example of each below:

Formal parameter:

In this example of code extracted below, you can see a variable named formalParameter that is passed as a parameter in the brackets of a method or function call. The same variable is then printed out so that we can see the value of the parameter. This is a formal parameter that is passed to a method.

class Parameter {

public static void methodParameter(int formalParameter) {

System.out.println(formalParameter);

}

Actual parameter:

In this example of code extracted below, you can see a variable named actualParameter that is passed as a parameter in the brackets of the main’s method or function call. The same variable is then printed our so that we can see the value of the parameter. This is an actual parameter that is passed to a method. Once both values have been printed out one would realise they have the same value and that the actual parameter is passed to the method and is then rebranded as a formal parameter albeit a variable that’s value has not changed.

public static void main(String[] args) {

int actualParameter = 1;

methodParameter(actualParameter);

System.out.println(actualParameter);

}

}

Total Words (263)