**The Class Method’s Definition**

Methods are **member functions** that belongs to the class.

There are two ways to define functions that belongs to a class:

* Inside the class definition
* Outside the class definition

**Inside the class definition**

When a member function is defined inside a class, the function definition is just similar to the function definitions we are familiar with.

In the following example, we define a function inside the class, and we name it "myMethod".

**Example:**

class MyClass

{         // The class  
  public:               // Access specifier  
    void myMethod()

{

   // Method/function defined inside the class  
      cout << "Hello World!";  
    }  
};  
  
int main() {  
  MyClass myObj;      // Create an object of MyClass  
  myObj.myMethod();   // Call the method  
  return 0;  
}

**Outside the class definition:**

To define a function outside the class definition, you have to declare it inside the class and then define it outside of the class. This is done by specifying the name of the class, followed the scope resolution :: operator, followed by the name of the function.

The general form of a member function definition outside the class definition is:

Return-type Class-name : : function-name (parameter list)

{

Function body

}

**Example**

class MyClass

{         // The class  
  public               // Access specifier  
    void myMethod();    // Method/function declaration  
};  
  
// Method/function definition outside the class  
void **MyClass::myMethod()**

 {  
  cout << "Hello World!";  
}  
  
int main()

{  
  MyClass myObj;      // Create an object of MyClass  
  myObj.myMethod();   // Call the method  
  return 0;  
}