

Basic Structure of C# Program

A C# program has the following form:

- 1) Using library
- 2) Namespace declaration
- 3) A class
- 4) De-limiters
- 5) Class attributes and their methods
- 6) A Main method
- 7) Statements and Expressions
- 8) Comments

Let us look at the various parts of the given program –

```
using System;

namespace HelloWorldApplication {
    class HelloWorld {
        static void Main(string[] args) {
            /* my first program in C# using built-in- class.
            WriteLine() is a method of the Console class defined in the System
            namespace.*/
            Console.WriteLine("Hello World"); // display Hello World on the screen
        }
    }
}
```

1. Using namespace/library

The first line of the program using System; - the using keyword is used to include the System namespace in the program. A program generally has multiple using statements.

2. Namespace declaration

The second line has the namespace declaration. A namespace is a collection of classes. The HelloWorldApplication namespace contains the class HelloWorld.

3. A Class

The third line has a class declaration, the class HelloWorld contains the data and method definitions that your program uses. Classes generally contain multiple methods. Methods define the behavior of the class. However, the HelloWorld class has only one method Main.

4. De-limiter

A delimiter in c is a unique character or string that marks the start or stop of a particular statement, string, or function body set. Commas (,), semicolons (;), quotes (" "), braces ({}), pipes (|), and slashes (/) are common delimiters in c.

5. Class Attributes and Methods.

Any variable that is bound in a class is a class attributes. Any function defined within a class is a method. Methods receive an instance of the class, conventionally called self , as the first argument.

6. A main Method

The fourth line defines the Main method, which is the entry point for all C# programs. The Main method states what the class does when executed. Every C# stand-alone program requires the main method as the starting point of the program. This is an essential part of a C# program. There may be many classes in a C# program, and only one class defines the main method. Methods contain data type declaration and executable statements. The Main method specifies its behavior with the statement **Console.WriteLine**("Hello World"); . **WriteLine()** is a method of the Console class defined in the System namespace. This statement causes the message "Hello, World!" to be displayed on the screen

7. Statements & Expressions

- The individual lines of code that make up a C# line is called statement.
- Expressions, combine variables and constants to create new values.
- Statements are expressions, assignments, function calls, or control flow statements, which make up the C# programs.

8. Comments

The fifth line `/*...*/` is ignored by the compiler and it is put to add comments in the program. You can write a comment in this section. Comments are beneficial for the programmer because they help them understand the code. These are optional, but we suggest you use them because they are useful to understand the operation of the program, so you must write comments within the program.

- Used to give additional useful information inside a C# Program.
- For Single line comment : `//`
- For Multiple Lines Comment: `/*.....*/`

Example1 `// this is a single line comment`

Example2 `/* this is a multiline comments
Here start main function */`

Some basic syntax rule for C# program

- C# is a case sensitive language so all C# keywords must be written in lower case letter and built-in classes names or method according to given case.
- All C# statement must end with a semicolon.
- Variables name declare with data types by requirement (`int` salary or `string` name etc)

What is Syntax?

The rules for using a programming language are called syntax. Syntax refers to the spelling and grammar of a programming language. Computers are inflexible machines that understand what you type only if you type it in the exact form that the computer expects. The expected form is called the syntax.

Console.Write(string) and Console.WriteLine(string) method differences:

This **Write(string)** method prints the text on the console and the cursor remains at the end of the text at the console. The next printing takes place from just here.

This **WriteLine(string)** method prints the text on the console and the cursor remains at the start of the next line at the console. The next printing takes place from next line.

Exercise

Theory Questions

1. What do you understand by the term “**C# Case Sensitive Language**”?
2. What difference between the class and object?
3. Identify the two types of comments available in C# and indicate when each would be used.
4. What is difference between the **Write(string)** and **WriteLine(string)** methods in C#?
5. Write names list of C# program structure.

Practical Questions

1. Write a simple C# program to print your BIODATA on the console screen.
2. Elaborate the Error(s) in the code below (if any).

```
Class abc
{
    /* here start main function
       This is multiple line comments
    Public void main(String[] args)
    {
        / This is a comment for C# program
        Console.Write("Display first line ");
        Console.WriteLine("Display second line ")
        Console.WriteLine("Display third line ");
    }
}
```

3. Write down the correct code of above program.
4. Mention the output of above code.
5. Write structure of C# program (sample structure).

Objective and MCQ's

1. What do you use to separate multiple argument that are passed to a function
 - a) A period (.)
 - b) A comma (,)
 - c) A forward slash (/)
 - d) A backward slash (\)
2. You create line comments in C# code by adding ____ to a line you want to use a comment.
 - a) ||
 - b) **
 - c) /*
 - d) //

©Copy Right

<http://www.sirmasood.com>

3. Block comments begin with /* and end with ____
 - a) */
 - b) /*
 - c) //
 - d) **
4. Write(string) method print String data on the console screen.
 - a) The next printing takes place from just here.
 - b) The next printing takes place from next line.
 - c) Print nothing
 - d) Not all above
5. The rules for using a programming language are called _____.
 - a) Protocols
 - b) Syntax
 - c) Variables
 - d) Comments
6. The group of classes that are defined by a name is.
 - a) import
 - b) package
 - c) using
 - d) include
7. All C# statements must be end with a _____.
 - a) Semicolon
 - b) Commas
 - c) Double quotations
 - d) Single quotations
8. Combine variables and constants to create new values that are called _____.
 - a) Variables
 - b) Statement
 - c) Expression
 - d) Concatenation
9. C# keywords and methods of classes must be used in _____ case
 - a) Upper case
 - b) Camel case
 - c) Sentence case
 - d) Lower case and Capitalize each word (camel case)
- 10 The name of memory addresses, where store data that is called
 - a) function
 - b) commands
 - c) variables
 - d) constant