DOUBLE_D

PROGRAMMING HANDBOOK

C# code

```
using System;
namespace name
{
  class class_name
    {
      static void Main(String[] args)
        {
          //code goes here
        }
    }
}
```

Naming conventions

```
Class --> Pascal casing
Method/function --> Pascal casing
Variable & arguments --> Camel casing
Property --> Pascal casing
Namespace --> Pascal casing
Interfaces --> I prefix
```

Output

```
using System;
namespace doubled
{
  class class_name
    {
      static void Main(String[] args)
        {
          Console.WriteLine(“Hello”);
          Console.WriteLine(”World”);         //Hello
        }                                     //World
    }
}
```

```
using System;
namespace doubled
{
  class class_name
    {
      static void Main(String[] args)
        {
          Console.Write(“Hello ”);
          Console.Write(“World”);   //Hello World
        }
    }
}
```

```
using System;
namespace doubled
{
  class class_name
    {
      static void Main(String[] args)
        {
          System.Console.WriteLine(“Hello world”);
        }
    }
}
```

Input

```
using System;
namespace doubled
{
         class a
         {
           static void Main(String[] args)
             {
                   Console.WriteLine("What is your name:");
                   Console.Write("My name is ");
                   string name =  Console.ReadLine();
                   Console.WriteLine("Hi "+ name);
                   Console.Read();
             }
         }
}
```

```
using System;
namespace doubled
{
         class a
         {
           static void Main(String[] args)
             {
                   Console.WriteLine("What is your name:");
                   Console.Write("My name is ");
                   string name =  Console.ReadLine();
                   Console.WriteLine("Hi {0}",name);
                   Console.Read();
             }
         }
}
```

Comments

```
//Single line comment
/*Multi line comment*/
```

Variables

```
int a = 15;
double b = 15.0;
char c = ‘D’;
bool d = true;
string e = “Hi”;
```

Variable scope

Global variable

```
using System;
namespace doubled
{
         class a
         {
            public string x = "xsgsdgsdf";
         }
         class b
         {
            static void Main(String[] args)
                  {
                     a obj = new a();
                     Console.WriteLine(obj.x);
                  }
         }
}
```

Local variable

```
using System;
namespace doubled
{
  class a
    {
      static void Main(String[] args)
        {
          string x = “Hello”;
          Console.WriteLine(x);
        }
    }
}
```

var keyword (up to c#3.0 nedded)

```
using System;
namespace doubled
{
         class a
         {
           static void Main(String[] args)
             {
               var num = 15;
               Console.WriteLine(a);
             }
         }
}
```


DILANGA D AMARASINGHE
PROGRAMMER
COMPUTER SCIENCE
EASTERN UNIVERSITY - SRI LANKA

E-Mail : doubledamarasinghe@gmail.com
GITHUB : https://github.com/DoubleDAmarasinghe
WhatsApp : +960 96 95 658
LinkedIn : @ Dilanga D Amarasinghe

DOUBLE_D

PROGRAMMING HANDBOOK

Constant

```
const variable_type constant_name = constant_value
const double pi =3.14;
```

Data types

```
Integer, Long, Float, Double, Boolean, Char, String
```

Widening(implicit) casting

Smaller to larger type

```
int myint = 9;
double mydouble = myint;
Console.WriteLine(mydouble); //9.0
Console.WriteLine(myint); //9
```

Narrowing(explicit) casting

Larger to smaller

```
double mydouble = 9.7;
int myint = (int) mydouble;
Console.WriteLine(mydouble); //9.7
Console.WriteLine(myint); //9
```

Conversions

```
Convert.ToBoolean();
Convert.ToDouble();
Convert.ToString();
Convert.ToInt32();
Convert.ToInt64
using System;
namespace doubled
{
          class a
          {
            static void Main(string[] args)
              {
                    int a = 15;
                    Console.WriteLine(Convert.ToString(a));
              }
          }
}
```

Function/Method

In same class user defined method

```
using System;
namespace doubled
{
          class a
          {
            static void dd()
              {
                 Console.WriteLine("Hello");
              }
            static void Main(String[] args)
              {
                 dd();
              }
          }
}
```

In different classes user defined method

```
using System;
namespace doubled
{
          class a
          {
            public static void dd()
              {
                 Console.WriteLine("Hello");
              }
          }
          class b
          {
            static void Main(String[] args)
              {
                a.dd();
              }
          }
}
```

Pre-defined main method

```
using System;
namespace doubled
{
          class a
          {
            static void Main(String[] args)
              {
                 Console.WriteLine("Hello");
              }
          }
}
```

Operators

```
Arithmetic    --> "+ - * / % ++ --"
Relational    --> "== != > < >= <="
Logical       --> "&& || !"
Bitwise       --> "& | ^(xor) ~(not) <<(shift left) >>"
Assignment    --> "= += -= *= /= %= <<= >>= &= ^= !="
Miscellaneous --> "sizeof()(returns size of data type) typeof()(ret
                   class) &(returns the address of variable) *(poin
                   ?:(conditional expression) is(determines whether
                   certain type) as(cast without raising an excepti
                   fails)"
```

Class

Single class

```
using System;
namespace doubled
{
   class a
      {
        //code goes here
      }
}
```

Multi classes

```
using System;
namespace doubled
{
   class a
      {
        //code goes here
      }
   class b
      {
        static void Main(String[] args)
          {
            //code goes here
          }
      }
}
```

Object

```
My_Class obj = new My_Class();
obj.my_fun1();
```

Interface

```
using System;
interface My_Interface
{
   void my_fun1();
}
class My_Class
{
   public void my_fun1()
    {
          Console.WriteLine("Hello World");
    }
   public static void Main()
    {
          My_Class obj = new My_Class();
          obj.my_fun1();
    }
}
```


DILANGA D AMARASINGHE
PROGRAMMER
COMPUTER SCIENCE
EASTERN UNIVERSITY - SRI LANKA

E-Mail : doubledamarasinghe@gmail.com
GITHUB : https://github.com/DoubleDAmarasinghe
WhatsApp : +960 96 95 658
LinkedIn : @ Dilanga D Amarasinghe

DOUBLE_D

PROGRAMMING HANDBOOK

Inheritance

Single-level inheritance

```
using System;
class My_Class1
{
    public void my_fun1()
    {
         Console.WriteLine("Hello World1");
    }
}
class My_Class2 : My_Class1
{
    void my_fun2()
    {
         Console.WriteLine("Hello World2");
    }
    public static void Main()
    {
         My_Class2 obj = new My_Class2();
         obj.my_fun1();
         obj.my_fun2();
    }
}
```

Multi-level inheritance

```
using System;
class My_Class1
 {
    public void my_fun1()
     {
         Console.WriteLine("Hello World1");
     }
}
class My_Class2 : My_Class1
 {
    public void my_fun2()
     {
         Console.WriteLine("Hello World2");
     }
 }
class My_Class3 : My_Class2
 {
    void my_fun3()
     {
         Console.WriteLine("Hello World3");
     }
     public static void Main()
     {
         My_Class3 obj = new My_Class3();
         obj.my_fun1();
         obj.my_fun2();
         obj.my_fun3();
     }
}
```

Multiple inheritance

```
using System;
interface My_Class1
{
    void my_fun1();
}
interface My_Class2
{
    void my_fun2();
}
class My_Class3 : My_Class1, My_Class2
{
    public void my_fun1()
     {
         Console.WriteLine("Hello World1");
     }
    public void my_fun2()
     {
         Console.WriteLine("Hello World2");
     }
    void my_fun3()
     {
         Console.WriteLine("Hello World3");
     }
    public static void Main()
     {
         My_Class3 obj = new My_Class3();
         obj.my_fun1();
         obj.my_fun2();
         obj.my_fun3();
     }
}
```

Hierarchy inheritance

```
using System;
class My_Class1
{
   public void my_fun1()
    {
         Console.WriteLine("Hello World1");
    }
}
class My_Class2 : My_Class1
{
   public void my_fun2()
    {
         Console.WriteLine("Hello World2");
    }
}
class My_Class3 : My_Class1
{
   public void my_fun3()
    {
         Console.WriteLine("Hello World3");
    }
   public static void Main()
    {
         My_Class2 obj1 = new My_Class2();
         My_Class3 obj2 = new My_Class3();
         obj1.my_fun1();
         obj1.my_fun2();
         obj2.my_fun1();
         obj2.my_fun3();
    }
}
```


DILANGA D AMARASINGHE
PROGRAMMER
COMPUTER SCIENCE
EASTERN UNIVERSITY - SRI LANKA

E-Mail : doubledamarasinghe@gmail.com
GITHUB : https://github.com/DoubleDAmarasinghe
WhatsApp : +960 96 95 658
LinkedIn : @ Dilanga D Amarasinghe

DOUBLE_D

PROGRAMMING HANDBOOK

Hybrid inheritance

For loop

```
using System;
namespace doubled
{
          class a
          {
            static void Main(String[] args)
              {
                    for(int i=0;i<10;i++)
                      {
                         Console.WriteLine("Hello "+ i);
                      }
              }
          }
}
```

While loop

```
using System;
namespace doubled
{
          class a
          {
            static void Main(String[] args)
              {
                    int i=0;
                    while(i<10)
                      {
                         Console.WriteLine("Hello "+ i);
                         i++;
                      }
              }
          }
}
```

Foreach loop

Do while loop

```
using System;
namespace doubled
{
          class a
          {
            static void Main(String[] args)
              {
                 int i=0;
                 do
                   {
                       Console.WriteLine("Hello "+ i);
                       i++;
                   }
                 while(i<10);
              }
          }
}
```

Infinite loop

```
using System;
namespace doubled
{
          class a
          {
            static void Main(String[] args)
              {
                 for(; ;)
                   {
                      Console.WriteLine("Hello ");
                   }
              }
          }
}
```

If, elseif, else statements

```
using System;
namespace doubled
{
          class a
          {
            static void Main(String[] args)
              {
                int x = 5;
                if(x == 1)
                   {
                      Console.WriteLine("one");
                   }
                else if(x == 5)
                   {
                       Console.WriteLine("five");
                   }
                else
                   {
                       Console.WriteLine("error");
                   }
              }
          }
}
```

Switch statements

```
using System;
namespace doubled
{
          class a
          {
            static void Main(String[] args)
              {
                 int x = 2;
                 switch(x)
                   {
                     case 1:
                     Console.WriteLine("one");
                     break;

                     case 2:
                     Console.WriteLine("two");
                     break;

                     case 3:
                     Console.WriteLine("three”);
                     break;

                     case 4:
                     Console.WriteLine("four");
                     break;


                     default :
                     Console.WriteLine("one");
                     break;
                   }
              }
          }
}
```


DILANGA D AMARASINGHE
PROGRAMMER
COMPUTER SCIENCE
EASTERN UNIVERSITY - SRI LANKA

E-Mail : doubledamarasinghe@gmail.com
GITHUB : https://github.com/DoubleDAmarasinghe
WhatsApp : +960 96 95 658
LinkedIn : @ Dilanga D Amarasinghe

DOUBLE_D

PROGRAMMING HANDBOOK

Try catch

```
using System;
namespace doubled
{
  class a
    {
      static void Main(string[] args)
        {
          try
            {
              string result = “k”;
              Console.WriteLine(Convert.ToInt32(result+10));
            }
          catch(Exception e)
            {
              Console.WriteLine(ex.Message); //Input string was not
            }
        }
    }
}
```

Command line arguments

```
using System;
namespace doubled
{
         class a
         {
            static void Main(String[] args)
              {
                   Console.WriteLine("First name:"+args[0]);
                   Console.WriteLine("Last name:"+args[1]);
                   Console.Read();
              }
         }
}
```

Simple calculator

```
using System;
namespace doubled
{
         class a
         {
            static void Main(string[] args)
              {
                   Console.WriteLine("Enter First Int:");
                   string snum1 = Console.ReadLine();
                   int num1 = Convert.ToInt32(snum1);
                   Console.WriteLine("Enter Second Int:");
                   string snum2 = Console.ReadLine();
                   int num2 = Convert.ToInt32(snum2);
                   int sum = num1 + num2;
                   Console.WriteLine("Sum is: "+sum);
              }
         }
}
```

Keywords

abstract	else	join	sbyte
add	enum		sealed
as	equals	let	select
ascending	explicit	lock	set
async	extern	long	short
await			sizeof
	false	namespace	stackalloc
base	finally	new	static
bool	fixed	null	string
break	float		struct
by	for	object	switch
byte	foreach	on	
	from	operator	this
case		orderby	throw
catch	get	out	true
char	global	override	try
checked	goto		typeof
class	group	params	
const		partial	unit
continue	if	private	ulong
	implicit	protected	unchecked
decimal	in	public	unsafe
default	int		ushort
delegate	interface	readonly	using
descending	internal	ref	value
do	into	remove	var
double	is	return	virtual
dynamic			void
			volatile

Using in GUI

```
using System;
using System.Drawing;
using System.Windows.Forms;
```

C# inheritance

```
class MyForm : System.Windows.Forms.Form{}
                  or
class MyForm : System.WinForms.Form{}
```

Creating objects

```
System.Windows.Forms.Form frm = new Form();
Panel pnl = new Panel();
Label lb = new Label();
Button btn = new Button();
TextBox txt = new TextBox();
RadioButton rb = new RadioButton();
ComboBox cb = new ComboBox();
CheckBox hb = new CheckBox();
Random rnd = new Random();
MainMenu mainMenu = new MainMenu();
MenuItem menuitem1 = new MenuItem();
```

Call a function

```
public MyForm(){}
```

Form attributes (Properties)

```
this.frm.Text = "DoubleD";
frm.Size = new Size(700,700);
frm.StartPosition = FormStartPosition.CenterScreen;
frm.Opacity = 100.0;
frm.MaximizeBox = false;
frm.MinimizeBox = false;
frm.AutoSize = true;
frm.HelpButton = true;
frm.CancelButton = btn1;
frm.AcceptButton = btn2;
frm.FormBorderStyle = FormBorderStyle.FixedDialog; //Sizable
```

Panel attributes (Properties)

```
this.pnl.Text = "DoubleD";
pnl.AutoSize = true;
pnl.Size = new Size(700,700);
pnl.BackColor = Color.Pink;
pnl.Visible = true;
```


DILANGA D AMARASINGHE
PROGRAMMER
COMPUTER SCIENCE
EASTERN UNIVERSITY - SRI LANKA

E-Mail : doubledamarasinghe@gmail.com
GITHUB : https://github.com/DoubleDAmarasinghe
WhatsApp : +960 96 95 658
LinkedIn : @ Dilanga D Amarasinghe

DOUBLE_D

PROGRAMMING HANDBOOK

Label attributes (Properties)

```
this.lb.Name = "lb";
lb.Text = "label";
lb.AutoSize = true;
lb.Location = new Point(100, 100);
lb.Size = new Size(150,60);
lb.BackColor = Color.Green;
lb.ForeColor = Color.Red;
lb.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
lb.Font = new Font("Arial", 36, FontStyle.Bold);
```

Button attributes (Properties)

```
this.btn.Name = "btn";
btn.Text = "button";
btn.Location = new Point(100,200);
btn.Size = new Size(150,60);
btn.BackColor = Color.Yellow;
btn.BackgroundImage
btn.AutoEllipsis = true;
btn.AutoSize = true;
btn.Enabled = true;
btn.Events
btn.Font = new Font("Arial", 36, FontStyle.Bold);
btn.Padding = new Padding(5,5,5,5);
btn.MouseClick += greeting;
btn.Margin = new Thickness(5);
```

Events on button

```
Click, DoubleClick, Enter, KeyPressed, Leave, MouseClick, MouseDoub
MouseHover, MouseLeave
```

Button action

```
void greeting(Object sender,EventArgs e)
        {
           btn.PerformClick();
           lb.Visible = true;
           lb.Text = Convert.ToString(txt.TextLength);
           txt.Text = "";
           txt.Focus();
        }
```

Textbox attributes (Properties)

```
this.txt.Name = "txt";
txt.Location = new Point(100,300);
txt.Size = new Size(150,60);
txt.Multiline = false;
txt.AcceptsReturn = true;
txt.AutoSize = true;
txt.BackColor = Color.Red;
txt.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
txt.CharacterCasing = CharacterCasing.Upper; //Lower
txt.Font = new Font("Arial", 18, FontStyle.Bold);
txt.ForeColor = Color.White;
txt.MaxLength = 1000;
txt.PasswordChar = '*';
txt.TextAlign = HorizontalAlignment.Right;
txt.Visible = true;
txt.UseSystemPasswordChar =  false;
txt.Margin =  new Thickness(10);
txt.AutoEllipsis = true;
textBox1.ReadOnly = true;
txt.KeyDown += new KeyEventHandler(greeting);
```

Textbox action

If

```
void greeting(Object sender,KeyEventArgs e)
        {
                if(e.KeyCode == Keys.Space) //Enter or any key
                   {
                     lb.Visible = true;
                     lb.Text = Convert.ToString(txt.TextLength);
                     txt.Text = "";
                     txt.Focus();
                   }
        }
```

Switch

```
void top(Object sender,KeyEventArgs e)
        {
                        switch(e.KeyCode)
                        {
                                case(Keys.Up):
                                lb.Location = new Point(lb.Location.
                                break;

                                case(Keys.Down):
                                lb.Location = new Point(lb.Location.
                                break;

                                case(Keys.Left):
                                lb.Location = new Point(lb.Location.
                                break;

                                case(Keys.Right):
                                lb.Location = new Point(lb.Location.
                                break;
                        }
        }
```

Random number

Variable

```
private int num;
```

Method

```
num = rnd.Next(1, 9);
```


DILANGA D AMARASINGHE
PROGRAMMER
COMPUTER SCIENCE
EASTERN UNIVERSITY - SRI LANKA

E-Mail : doubledamarasinghe@gmail.com
GITHUB : https://github.com/DoubleDAmarasinghe
WhatsApp : +960 96 95 658
LinkedIn : @ Dilanga D Amarasinghe