

# Functions



Built-in Functions



User defined Functions

Are function defined by user to accomplish specific tasks that are not done by system and tune them to specific needs



Type  
Conversion  
Functions



Console  
Functions



Array  
Functions



Date  
and  
Time  
Functions



String  
Functions



Math  
Functions



Random  
Functions

Math  
Functions

Math.Abs	Returns the absolute value of a number.	Math.Abs(-10)
Math.Pow	Raises a number to a specified power.	Math.Pow(2, 3);
Math.Max	used to find the highest	Math.Max(5, 10);
Math.Sqrt(x)	ethod returns the square root of x	Math.Sqrt(64);
Math.Min	used to find the lowest value	Math.Min(5, 10);

String  
Functions

string.Length
string.ToUpper and string.ToLower.
string.Trim.
string.Replace
string.Contains
string.Split:

Date  
and Time  
Functions

<b>DateTime.Now</b> Gets the current date and time.
<b>DateTime.Today</b> current date with the time component set to 00:00:00.
<b>DateTime.AddDays</b> Adds a specified number of days to a DateTime object.
<b>DateTime.ToString</b> Converts the date and time to a string in a specified format.
<b>DateTime.Parse and DateTime.TryParse</b> Parse a string representation of a date and time into a DateTime object

## Type Conversion Functions

### **Convert.ToInt32**

Converts a specified value to a 32-bit signed integer.

### **Convert.ToDouble**

Converts a specified value to a double-precision floating-point number.

### **Convert.ToString**

Converts a specified value to a string.

### **int.Parse and int.TryParse**

Convert a string representation of a number to its integer

## Random Functions

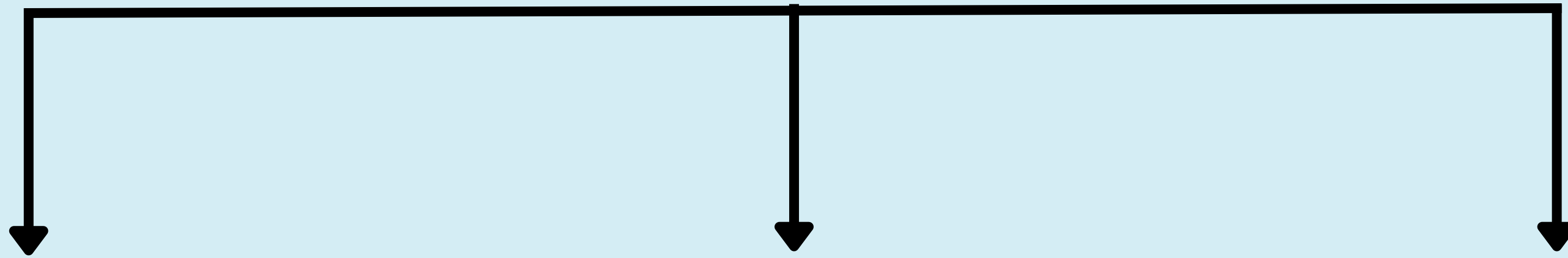
### **Random.Next**

```
Random random = new Random();  
int test = random.Next(0, 1);  
Console.WriteLine(test);  
Console.ReadKey();
```

### **Random.NextDouble**

```
static void Main()  
{  
    Random random = new Random();  
    double randomValue =  
        random.NextDouble();  
    Console.WriteLine("Random  
value between 0.0 and 1.0: " +  
        randomValue);  
}
```

# Errors in Programming



Syntax Errors

in rules of the programming language

Runtime Errors

after running

Logical Errors

output is not what you expected.

Handle Errors:  
using try-catch



```
try
{
    int[] numbers = { 1, 2, 3 };
    Console.WriteLine(numbers
        IndexOutOfRangeException
    )
}
catch (IndexOutOfRangeException
    ex)
{

    Console.WriteLine("An error
        occurred: " + ex.Message);
}
```