TutorialKart

# C# Math.Max() – Syntax & Examples

## C# Math.Max() – Examples

In this tutorial, we will learn about the C# Math.Max() method, and learn how to use this method to find maximum of two numbers/values, with the help of examples.

## Max(Byte, Byte)

Math.Max(val1, val2) returns the larger of two 8-bit unsigned integers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(Byte val1, Byte val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two 8-bit unsigned integers to compare. |
| val2 | The second of two 8-bit unsigned integers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 1 – Max(Byte val1, Byte val2)

In this example, we will find the largest of two 8-bit unsigned integers using Max() method.

**C# Program**

```
using System;
```

```
class Example {
    static void Main(string[] args) {
        byte val1 = 7;
        byte val2 = 25;

        byte result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of 7 and 25 is 25.
```

## Max(Decimal, Decimal)

Math.Max(val1, val2) returns the larger of two decimal numbers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(Decimal val1, Decimal val2)
```

where

| Parameter | Description |
|---|---|
| Decimal | The first of two decimal numbers to compare. |
| Decimal | The second of two decimal numbers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 2 – Max(Decimal val1, Decimal val2)

In this example, we will find the largest of two decimal numbers using Max() method.

**C# Program**

```
using System;

class Example {
    static void Main(string[] args) {
        Decimal val1 = 7.5M;
        Decimal val2 = 7.1M;

        Decimal result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of 7.5 and 7.1 is 7.5.
```

## Max(Double, Double)

Math.Max(val1, val2) returns the larger of two double-precision floating-point numbers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(Double val1, Double val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two double values to compare. |
| val2 | The second of two double values to compare. |

**Return Value**

The method returns value.

### Example 3 – Max(Double val1, Double val2)

In this example, we will find the largest of two double values using Max() method.

**C# Program**

```
using System;

class Example {
    static void Main(string[] args) {
        Double val1 = 7.5785;
        Double val2 = 7.18974;

        Double result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of 7.5785 and 7.18974 is 7.5785.
```

## Max(Int16, Int16)

Math.Max(val1, val2) returns the larger of two 16-bit signed integers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(Int16 val1, Int16 val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two 16-bit signed integers to compare. |
| val2 | The second of two 16-bit signed integers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 4 – Max(Int16 val1, Int16 val2)

In this example, we will find the largest of two 16-bit signed integers using Max() method.

**C# Program**

```
using System;

class Example {
    static void Main(string[] args) {
        Int16 val1 = 7;
        Int16 val2 = 24;

        Int16 result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of 7 and 24 is 24.
```

## Max(Int32, Int32)

Math.Max(val1, val2) returns the larger of two 32-bit signed integers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(Int32 val1, Int32 val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two 32-bit signed integers to compare. |
| val2 | The second of two 32-bit signed integers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 5 – Max(Int32 val1, Int32 val2)

In this example, we will find the largest of two 32-bit signed integers using Max() method.

**C# Program**

```
using System;

class Example {
    static void Main(string[] args) {
        Int32 val1 = 7;
        Int32 val2 = 24;

        Int32 result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of 7 and 24 is 24.
```

## Max(Int64, Int64)

Math.Max(val1, val2) returns the larger of two 64-bit signed integers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(Int64 val1, Int64 val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two 64-bit signed integers to compare. |
| val2 | The second of two 64-bit signed integers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 6 – Max(Int64 val1, Int64 val2)

In this example, we will find the largest of two 64-bit signed integers using Max() method.

**C# Program**

C# Program

```
using System;

class Example {
    static void Main(string[] args) {
        Int64 val1 = 7;
        Int64 val2 = 24;

        Int64 result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of 7 and 24 is 24.
```

## Max(SByte, SByte)

Math.Max(val1, val2) returns the larger of two 8-bit signed integers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(SByte val1, SByte val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two 8-bit signed integers to compare. |
| val2 | The second of two 8-bit signed integers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 7 – Max(SByte val1, SByte val2)

In this example, we will find the largest of two 8-bit signed integers using Max() method.

**C# Program**

```
using System;

class Example {
    static void Main(string[] args) {
        sbyte val1 = -7;
        sbyte val2 = -4;

        sbyte result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of -7 and -4 is -4.
```

## Max(Single, Single)

Math.Max(val1, val2) returns the larger of two single-precision floating-point numbers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(Single val1, Single val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two single-precision floating-point numbers to compare. |
| val2 | The second of two single-precision floating-point numbers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 8 – Max(Single val1, Single val2)

In this example, we will find the largest of two single-precision floating-point numbers using Max() method.

**C# Program**

```
using System;

class Example {
    static void Main(string[] args) {
        Single val1 = 7.1F;
        Single val2 = 7.12F;

        Single result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of 7.1 and 7.12 is 7.12.
```

## Max(UInt16, UInt16)

Math.Max(val1, val2) returns the larger of two 16-bit unsigned integers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(UInt16 val1, UInt16 val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two 16-bit unsigned integers to compare. |
| val2 | The second of two 16-bit unsigned integers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 9 – Max(UInt16 val1, UInt16 val2)

In this example, we will find the largest of two 16-bit unsigned integers using Max() method.

**C# Program**

```
using System;

class Example {
    static void Main(string[] args) {
        UInt16 val1 = 7;
        UInt16 val2 = 8;

        UInt16 result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of 7 and 8 is 8.
```

## Max(UInt32, UInt32)

Math.Max(val1, val2) returns the larger of two 32-bit unsigned integers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(UInt32 val1, UInt32 val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two 32-bit unsigned integers to compare. |
| val2 | The second of two 32-bit unsigned integers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 10 – Max(UInt32 val1, UInt32 val2)

In this example, we will find the largest of two 32-bit unsigned integers using Max() method.

**C# Program**

```
using System;

class Example {
    static void Main(string[] args) {
        UInt32 val1 = 7;
        UInt32 val2 = 8;

        UInt32 result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

**Output**

```
Maximum of 7 and 8 is 8.
```

## Max(UInt64, UInt64)

Math.Max(val1, val2) returns the larger of two 64-bit unsigned integers: `val1` and `val2`.

### Syntax

The syntax of Max() method is

```
Math.Max(UInt64 val1, UInt64 val2)
```

where

| Parameter | Description |
|---|---|
| val1 | The first of two 64-bit unsigned integers to compare. |
| val2 | The second of two 64-bit unsigned integers to compare. |

**Return Value**

The method returns val1 or val2, whichever is maximum.

### Example 11 – Max(UInt64 val1, UInt64 val2)

In this example, we will find the largest of two 64-bit unsigned integers using Max() method.

### C# Program

```
using System;

class Example {
    static void Main(string[] args) {
        UInt64 val1 = 7;
        UInt64 val2 = 8;

        UInt64 result = Math.Max(val1, val2);
        Console.WriteLine($"Maximum of {val1} and {val2} is {result}.");
    }
}
```

### Output

```
Maximum of 7 and 8 is 8.
```

## Conclusion

In this C# Tutorial, we have learnt the syntax of C# Math.Max() method, and also learnt how to use this method, with the help of C# example programs.


**C# Math**

- C# Math.Abs()
- C# Math.Acos()
- C# Math.Acosh()
- C# Math.Asin()
- C# Math.Asinh()
- C# Math.Atan()
- C# Math.Atan2()
- C# Math.Atanh()
- C# Math.BigMul()
- C# Math.BitDecrement()
- C# Math.BitIncrement()
- C# Math.Cbrt()
- C# Math.Ceiling()