

1 System.ArithmeticException Class

```
2 [ILAsm]  
3 .class public serializable ArithmeticException extends  
4 System.SystemException  
  
5 [C#]  
6 public class ArithmeticException: SystemException
```

7 Assembly Info:

- 8 • *Name:* mscorlib
- 9 • *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- 10 • *Version:* 2.0.x.x
- 11 • *Attributes:*
 - 12 ○ CLSCompliantAttribute(true)

13 Summary

14 Represents an error caused by an arithmetic operation.

15 Inherits From: System.SystemException

16

17 **Library:** BCL

18

19 **Thread Safety:** All public static members of this type are safe for multithreaded operations.
20 No instance members are guaranteed to be thread safe.

21

22 Description

23 [Note: The Base Class Library includes two types derived from
24 System.ArithmeticException:

- 25 • System.DivideByZeroException
- 26 • System.OverflowException

27 When appropriate, use these types instead of System.ArithmeticException.

28

29]

30

31 [Note: The following CIL instructions throw System.ArithmeticException:

- 32 • ckfinite
- 33 • div

34]

1 **Example**

2 The following example demonstrates an error that causes the system to throw a
3 `System.ArithmeticException` error.

```
4  
5 [C#]  
  
6 using System;  
7 class testNan {  
8     public static void Main() {  
9         double myNan = Double.NaN;  
10        try {  
11            Math.Sign(myNan);  
12        }  
13        catch (ArithmeticException e) {  
14            Console.WriteLine("Error: {0}",e);  
15        }  
16    }  
17 }  
18
```

19 The output is

```
20  
21 Error: System.ArithmeticException: Function does not accept floating point  
22 Not-a-Number values.  
23
```

```
24  
25 at System.Math.Sign(Double value)  
26
```

```
27  
28 at testNan.Main()  
29
```

```
30
```

1 ArithmeticException() Constructor

```
2 [ILAsm]  
3 public rtspecialname specialname instance void .ctor()  
4 [C#]  
5 public ArithmeticException()
```

6 Summary

7 Constructs and initializes a new instance of the `System.ArithmeticException` class.

8 Description

9 This constructor initializes the `System.ArithmeticException.Message` property of the
10 new instance to a system-supplied message that describes the error, such as "The
11 arithmetic operation is not allowed." This message takes into account the current
12 system culture.

13
14 The `System.ArithmeticException.InnerException` property is initialized to `null`.

15

1 ArithmeticException(System.String)

2 Constructor

```
3 [ILAsm]  
4 public rtspecialname specialname instance void .ctor(string message)  
5 [C#]  
6 public ArithmeticException(string message)
```

7 Summary

8 Constructs and initializes a new instance of the `System.ArithmeticException` class.

9 Parameters

Parameter	Description
<i>message</i>	A <code>System.String</code> that describes the error. The content of <i>message</i> is intended to be understood by humans. The caller of this constructor is required to ensure that this string has been localized for the current system culture.

10

11 Description

12 This constructor initializes the `System.ArithmeticException.Message` property of the
13 new instance using *message*. If *message* is null, the
14 `System.ArithmeticException.Message` property is initialized to a system-supplied
15 message. The `System.ArithmeticException.InnerException` property is initialized to
16 null.

17

1 ArithmeticException(System.String, 2 System.Exception) Constructor

```
3 [ILAsm]  
4 public rtspecialname specialname instance void .ctor(string message, class  
5 System.Exception innerException)  
  
6 [C#]  
7 public ArithmeticException(string message, Exception innerException)
```

8 Summary

9 Constructs and initializes a new instance of the System.ArithmeticException class.

10 Parameters

Parameter	Description
<i>message</i>	A System.String that describes the error. The content of <i>message</i> is intended to be understood by humans. The caller of this constructor is required to ensure that this string has been localized for the current system culture.
<i>innerException</i>	An instance of System.Exception that is the cause of the current exception. If <i>innerException</i> is non-null, then the current exception was raised in a catch block handling <i>innerException</i> .

11 12 Description

13 This constructor initializes the System.ArithmeticException.Message property of the
14 new instance using *message*, and the System.Exception.InnerException property
15 using *innerException*. If *message* is null, the System.ArithmeticException.Message
16 property is initialized to a system-supplied message.

17
18 [Note: For information on inner exceptions, see System.Exception.InnerException.]
19
20

21