

# JAVA.LANG.MATH.EXPM1 METHOD

[http://www.tutorialspoint.com/java/lang/math\\_exp1.htm](http://www.tutorialspoint.com/java/lang/math_exp1.htm)

Copyright © tutorialspoint.com

## Description

The **java.lang.Math.exp1double** returns  $e^x - 1$ . Note that for values of  $x$  near 0, the exact sum of  $\exp1x + 1$  is much closer to the true result of  $e^x$  than  $\exp x$ . Special cases:

- If the argument is NaN, the result is NaN.
- If the argument is positive infinity, then the result is positive infinity.
- If the argument is negative infinity, then the result is -1.0.
- If the argument is zero, then the result is a zero with the same sign as the argument.

The computed result must be within 1 ulp of the exact result. Results must be semi-monotonic. The result of `expm1` for any finite input must be greater than or equal to -1.0. Note that once the exact result of  $e^x - 1$  is within 1/2 ulp of the limit value -1, -1.0 should be returned.

## Declaration

Following is the declaration for **java.lang.Math.exp** method

```
public static double expm1(double x)
```

## Parameters

- **x** -- the exponent to raise e to in the computation of  $e^x - 1$ .

## Return Value

This method returns the value  $e^x - 1$ .

## Exception

- **NA**

## Example

The following example shows the usage of `lang.Math.exp1` method.

```
package com.tutorialspoint;

import java.lang.*;

public class MathDemo {

    public static void main(String[] args) {

        // get two double numbers
        double x = 5;
        double y = 0.5;

        // call expm1 for both numbers and print the result
        System.out.println("Math.exp1(" + x + ")=" + Math.exp1(x));
        System.out.println("Math.exp1(" + y + ")=" + Math.exp1(y));
    }
}
```

Let us compile and run the above program, this will produce the following result:

`Math.exp(5)=147.4131591025766`

`Math.exp(0.5)=0.6487212707001282`

Loading [MathJax]/jax/output/HTML-CSS/jax.js