

Math

Die Klasse **Math** stellt eine Sammlung mathematischer Funktionen und Konstanten bereit. Sie enthält ausschließlich **statische Methoden und Felder**, weshalb kein Objekt erstellt werden muss.

Mit `Math` lassen sich unter anderem:

- Zahlen runden
- Beträge berechnen
- Potenzen und Wurzeln bestimmen
- Minimal- und Maximalwerte ermitteln
- trigonometrische Funktionen verwenden
- mathematische Konstanten wie π oder e nutzen

Die Klasse gehört zum Namespace **System** und steht in PowerShell standardmäßig zur Verfügung.

Grundlagen

Eigenschaft	Wert
Namespace	<code>System</code>
Assembly	<code>System.Runtime</code>
Instanzierbar	☐ Nein
Statisch	☒ Ja
Vererbt von	<code>Object</code>

Klasse verwenden

Da alle Mitglieder statisch sind, erfolgt der Aufruf direkt über den Klassennamen.

```
[Math]::Sqrt(25)
```

Ausgabe

```
5
```

Keine Instanziierung erforderlich

Im Gegensatz zu vielen anderen .NET-Klassen wird kein Objekt erzeugt.

Nicht erforderlich

```
New-Object Math
```

Stattdessen

```
[Math]::Pow(2,8)
```

Rückgabewerte

Die meisten Methoden geben einen numerischen Wert zurück.

Je nach Methode erfolgt die Rückgabe beispielsweise als:

Datentyp	Beschreibung
Double	Gleitkommazahl
Int32	Ganze Zahl
Decimal	Dezimalzahl (je nach Parameter)

Konstanten

Name	Typ	Beschreibung
PI	Double	Kreiszahl π
E	Double	Eulersche Zahl

Beispiel

```
[Math]::PI  
[Math]::E
```

3,14159265358979

2,71828182845905

Methoden

Übersicht

Methode	Beschreibung
Abs()	Berechnet den Betrag einer Zahl
Ceiling()	Rundet auf die nächstgrößere ganze Zahl
Clamp()	Begrenzt einen Wert auf einen Bereich
Floor()	Rundet auf die nächstkleinere ganze Zahl
Max()	Gibt den größeren Wert zurück
Min()	Gibt den kleineren Wert zurück
Pow()	Berechnet Potenzen
Round()	Rundet Zahlen
Sign()	Bestimmt das Vorzeichen
Sqrt()	Berechnet Quadratwurzeln
Truncate()	Entfernt Nachkommastellen

Abs()

Abs()

Berechnet den Betrag einer Zahl

[Math]::Abs(-15)

Ausgabe

15

Clamp()

Die Methode **Clamp()** begrenzt einen Wert auf einen angegebenen Minimal- und Maximalwert. Liegt der Wert innerhalb des Bereichs, wird er unverändert zurückgegeben.

Diese Methode ist ab **.NET Core 2.0** bzw. **.NET Standard 2.1** verfügbar. Sie steht daher **nicht** in **Windows PowerShell 5.1 (.NET Framework)** zur Verfügung, sondern erst ab **PowerShell Core 6.0**.

Syntax

```
[Math]::Clamp(<Wert>, <Minimum>, <Maximum>)
```

Parameter

Parameter	Typ	Beschreibung
Wert	numerisch	Zu prüfender Wert
Minimum	numerisch	Kleinster erlaubter Wert
Maximum	numerisch	Größter erlaubter Wert

Rückgabewert

Gleicher Datentyp wie die Eingabewerte.

Beispiel

```
[Math]::Clamp(120,0,100)
[Math]::Clamp(-5,0,100)
[Math]::Clamp(50,0,100)
```

Ausgabe

```
100
0
50
```

Ceiling()

Ceiling()

Die Methode **Ceiling()** rundet eine Zahl **immer auf die nächstgrößere ganze Zahl auf**.
Besitzt die Zahl bereits keinen Nachkommateil, bleibt sie unverändert.

Syntax

```
[Math]::Ceiling(<Zahl>)
```

Parameter

Parameter	Typ	Beschreibung
Zahl	Double / Decimal	Die aufzurundende Zahl.

Rückgabewert

Double bzw. **Decimal**

Die aufgerundete ganze Zahl.

Beispiel

```
[Math]::Ceiling(4.2)  
[Math]::Ceiling(-4.2)
```

Ausgabe

```
5  
-4
```

Floor()

Floor()

Die Methode **Floor()** rundet eine Zahl **immer auf die nächstkleinere ganze Zahl ab**.

Syntax

```
[Math]::Floor(<Zahl>)
```

Parameter

Parameter	Typ	Beschreibung
Zahl	Double / Decimal	Die abzurundende Zahl.

Rückgabewert

Double bzw. **Decimal**

Die abgerundete ganze Zahl.

Beispiel

```
[Math]::Floor(4.9)
[Math]::Floor(-4.2)
```

Ausgabe

```
4
-5
```

Min()

Min()

Die Methode **Min()** gibt den kleineren von zwei Werten zurück.

Syntax

```
[Math]::Min(<Wert1>, <Wert2>)
```

Parameter

Parameter	Typ	Beschreibung
-----------	-----	--------------

Wert1	numerisch	Erster Vergleichswert
Wert2	numerisch	Zweiter Vergleichswert

Rückgabewert

Gleicher Datentyp wie die Eingabewerte.

Beispiel

```
[Math]::Min(12,8)
```

Ausgabe

```
8
```

Max()

Max()

Den größeren Wert bestimmen

```
[Math]::Max(15,42)
```

Ausgabe

```
42
```

Pow()

Pow()

Berechnet Potenzen

```
[Math]::Pow(5,3)
```

Ausgabe

125

Round()

Round()

Rundet Zahlen

```
[Math]::Round(3.14159,2)
```

Ausgabe

```
3.14
```

Sign()

Sign()

Beschreibung

Die Methode **Sign()** bestimmt das Vorzeichen einer Zahl.

Syntax

```
[Math]::Sign(<Zahl>)
```

Parameter

Parameter	Typ	Beschreibung
Zahl	numerisch	Zu prüfende Zahl

Rückgabewert

Rückgabotyp: `System.Int32`

Rückgabe	Bedeutung
----------	-----------

-1	Zahl ist negativ
0	Zahl ist 0
1	Zahl ist positiv

Beispiel

```
[Math]::Sign(-20)
```

```
[Math]::Sign(0)
```

```
[Math]::Sign(20)
```

Ausgabe

```
-1
```

```
0
```

```
1
```

Sqrt()

Sqrt()

Berechnet Quadratwurzeln

```
[Math]::Sqrt(64)
```

Ausgabe

```
8
```

Truncate()

Truncate()

Die Methode **Truncate()** entfernt den Nachkommateil einer Zahl, ohne zu runden.

Syntax

```
[Math]::Truncate(<Zahl>)
```

Parameter

Parameter	Typ	Beschreibung
Zahl	Double / Decimal	Zu kürzende Zahl

Rückgabewert

Double bzw. **Decimal**

Die Zahl ohne Nachkommastellen.

Beispiel

```
[Math]::Truncate(4.9)  
[Math]::Truncate(-4.9)
```

Ausgabe

```
4  
-4
```

Weitere Details

Häufige Einsatzgebiete

Die Klasse `Math` wird häufig verwendet für:

- mathematische Berechnungen
- Koordinaten- und Geometrieberechnungen
- Diagramme und Visualisierungen
- Spieleentwicklung
- Größen- und Positionsberechnungen
- statistische Auswertungen
- wissenschaftliche Berechnungen

Hinweise

- Alle Methoden sind **statisch**.
 - Die Klasse kann nicht instanziiert werden.
 - Für viele Methoden existieren mehrere Überladungen, die unterschiedliche numerische Datentypen unterstützen (`Int32`, `Double`, `Decimal` usw.).
 - Die meisten Berechnungen arbeiten mit dem Datentyp `Double`, wodurch Rundungsungenauigkeiten bei Gleitkommazahlen auftreten können.
-

Siehe auch

- `System.Random`
 - `System.Convert`
 - `System.Decimal`
 - `System.Numerics`
-

Revision #3

Created 2026-07-24 13:07:54 UTC by John-Andreas Borinas

Updated 2026-08-18 08:07:04 UTC by John-Andreas Borinas