



Math

Math est un objet intégré qui possède des propriétés et des méthodes pour les constantes et les fonctions mathématiques. Ce n'est pas un objet fonction.

Math fonctionne avec le [Number](#) type. Cela ne fonctionne pas avec [BigInt](#).

La description

Contrairement à de nombreux autres objets globaux, Math n'est pas un constructeur. Toutes les propriétés et méthodes de Math sont statiques. Vous vous référez à la constante pi en tant que `Math.PI` et vous appelez la fonction sinus en tant que `Math.sin(x)`, où `x` est l'argument de la méthode. Les constantes sont définies avec la précision totale des nombres réels en JavaScript.

Remarque : De nombreuses Math fonctions ont une précision qui *dépend de l'implémentation*.

Cela signifie que différents navigateurs peuvent donner un résultat différent. Même le même moteur JavaScript sur un système d'exploitation ou une architecture différente peut donner des résultats différents !

Propriétés statiques

[Math.E](#)

la constante d'Euler et la base des logarithmes naturels ; environ 2.718 .

[Math.LN2](#)

Logarithme naturel de 2 ; environ 0.693 .

[Math.LN10](#)

Logarithme naturel de 10 ; environ 2.303 .

[Math.LOG2E](#)

Logarithme en base 2 de E ; environ 1.443 .

[Math.LOG10E](#)

Logarithme en base 10 de E ; environ 0.434 .

Math.PI

Rapport de la circonférence d'un cercle à son diamètre; environ 3.14159 .

Math.SQRT1_2

Racine carrée de $\frac{1}{2}$; environ 0.707 .

Math.SQRT2

Racine carrée de 2 ; environ 1.414 .

Méthodes statiques

Math.abs(x)

Renvoie la valeur absolue de x .

Math.acos(x)

Renvoie l'arc cosinus de x .

Math.acosh(x)

Renvoie l'arc cosinus hyperbolique de x .

Math.asin(x)

Renvoie l'arc sinus de x .

Math.asinh(x)

Renvoie l'arc sinus hyperbolique d'un nombre.

Math.atan(x)

Renvoie l'arc tangente de x .

Math.atanh(x)

Renvoie l'arc tangente hyperbolique de x .

Math.atan2(y, x)

Renvoie l'arc tangente du quotient de ses arguments.

Math.cbrt(x)

Renvoie la racine cubique de x .

Math.ceil(x)

Renvoie le plus petit entier supérieur ou égal à x .

Math.clz32(x)

[Math.clz32\(x\)](#)
Renvoie le nombre de bits zéro non significatifs de l'entier 32 bits x .

[Math.cos\(x\)](#)

Renvoie le cosinus de x .

[Math.cosh\(x\)](#)

Renvoie le cosinus hyperbolique de x .

[Math.exp\(x\)](#)

Renvoie e^x , où x est l'argument, et e est la constante d'Euler (2.718..., la base du logarithme népérien).

[Math.expm1\(x\)](#)

Renvoie la soustraction 1 de $\exp(x)$.

[Math.floor\(x\)](#)

Renvoie le plus grand entier inférieur ou égal à x .

[Math.fround\(x\)](#)

Renvoie le plus proche [simple précision](#) représentation flottante de x .

[Math.hypot\(\[x\[, y\[, ...\]\]\]\)](#)

Renvoie la racine carrée de la somme des carrés de ses arguments.

[Math.imul\(x, y\)](#)

Renvoie le résultat de la multiplication d'entiers 32 bits de x et y .

[Math.log\(x\)](#)

Renvoie le logarithme népérien (\log_e ; également, \ln) de x .

[Math.log1p\(x\)](#)

Renvoie le logarithme népérien (\log_e ; également \ln) de $1 + x$ pour le nombre x .

[Math.log10\(x\)](#)

Renvoie le logarithme en base 10 de x .

[Math.log2\(x\)](#)

Renvoie le logarithme en base 2 de x .

[Math.max\(\[x\[, y\[, ...\]\]\]\)](#)

Renvoie le plus grand de zéro ou plusieurs nombres.

[Math.min\(\[x\[, y\[, ...\]\]\]\)](#)

Renvoie le plus petit de zéro ou plusieurs nombres.

[Math.pow\(x, y\)](#)

Renvoie la base x à la puissance de l'exposant y (c'est-à-dire x^y).

[Math.random\(\)](#)

Renvoie un nombre pseudo-aléatoire entre 0 et 1.

[Math.round\(x\)](#)

Renvoie la valeur du nombre x arrondi à l'entier le plus proche.

[Math.sign\(x\)](#)

Renvoie le signe de x , indiquant s'il x est positif, négatif ou nul.

[Math.sin\(x\)](#)

Renvoie le sinus de x .

[Math.sinh\(x\)](#)

Renvoie le sinus hyperbolique de x .

[Math.sqrt\(x\)](#)

Renvoie la racine carrée positive de x .

[Math.tan\(x\)](#)

Renvoie la tangente de x .

[Math.tanh\(x\)](#)

Renvoie la tangente hyperbolique de x .

[Math.trunc\(x\)](#)

Renvoie la partie entière de x , en supprimant tous les chiffres fractionnaires.

Exemples

Conversion entre degrés et radians

Les fonctions trigonométriques `sin()`, `cos()`, `tan()`, `asin()`, `acos()`, `atan()`, et `atan2()` attendent (et renvoient) des angles en *radians*.

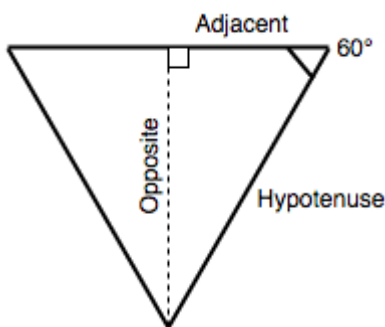
Étant donné que les humains ont tendance à penser en degrés et que certaines fonctions (telles que les transformations CSS) peuvent accepter des degrés, c'est une bonne idée de garder à portée de main les fonctions qui convertissent entre les deux :

```
| function degToRad(degrees) {
```

```
return degrees * (Math.PI / 180);  
};  
  
function radToDeg(rad) {  
    return rad / (Math.PI / 180);  
};
```

Calcul de la hauteur d'un triangle équilatéral

Si nous voulons calculer la hauteur d'un triangle équilatéral et que nous savons que la longueur de ses côtés est de 100, nous pouvons utiliser les formules *longueur de l'adjacent multiplié par la tangente de l'angle est égal à l'opposé*.



En JavaScript, nous pouvons le faire avec ce qui suit :

```
50 * Math.tan(degToRad(60)).
```

Nous utilisons notre `degToRad()` fonction pour convertir 60 degrés en radians, comme [Math.tan\(\)](#) prévu une valeur d'entrée en radians.

Retourner un entier aléatoire entre deux bornes

Ceci peut être réalisé avec une combinaison de [Math.random\(\)](#) et [Math.floor\(\)](#) :

```
function random(min, max) {  
    const num = Math.floor(Math.random() * (max - min + 1)) + min;  
    return num;  
}  
  
random(1, 10);
```

Caractéristiques

spécification