

Math: la matematica

La classe `Math` in Java è una classe utilitaria che fornisce metodi statici per operazioni matematiche comuni come il calcolo di radici quadrate, esponenziali, logaritmi, trigonometria, e altro. Non è possibile creare un'istanza di `Math`, poiché tutti i suoi metodi sono statici e non richiedono un oggetto per essere invocati.

Principali Metodi della Classe Math

Ecco alcuni dei metodi più utilizzati della classe `Math`:

1. Funzioni esponenziali e logaritmi

- `Math.pow(double base, double exponent)`: Restituisce il risultato della base elevata all'esponente, cioè $\text{base}^{\text{exponent}}$.

```
double result = Math.pow(2, 3); // 8.0
```

- `Math.exp(double a)`: Restituisce e^a , dove e è la costante di Eulero (circa 2.71828).

```
double result = Math.exp(1); // 2.718281828459045
```

- `Math.log(double a)`: Restituisce il logaritmo naturale (base e) di un numero.

```
double result = Math.log(10); // 2.302585092994046
```

- `Math.log10(double a)`: Restituisce il logaritmo in base 10 di un numero.

```
double result = Math.log10(100); // 2.0
```

2. Funzioni trigonometriche

- `Math.sin(double a)`: Restituisce il seno dell'angolo in radianti.

```
double result = Math.sin(Math.PI / 2); // 1.0
```

- `Math.cos(double a)`: Restituisce il coseno dell'angolo in radianti.

```
double result = Math.cos(Math.PI); // -1.0
```

- `Math.tan(double a)`: Restituisce la tangente dell'angolo in radianti.

```
double result = Math.tan(Math.PI / 4); // 1.0
```

- `Math.toRadians(double angolo)`: Converte un angolo in gradi in radianti.

```
double radians = Math.toRadians(90); // 1.5708 (radianti)
```

- `Math.toDegrees(double angolo)`: Converte un angolo in radianti in gradi.

```
double degrees = Math.toDegrees(Math.PI / 2); // 90.0
```

3. Funzioni per radici e arrotondamenti

- **Math.sqrt(double a):** Restituisce la radice quadrata di un numero positivo.

```
double result = Math.sqrt(16); // 4.0
```

- **Math.cbrt(double a):** Restituisce la radice cubica di un numero.

```
double result = Math.cbrt(27); // 3.0
```

- **Math.round(double a):** Restituisce il valore dell'argomento arrotondato al valore intero più vicino. Restituisce un long.

```
long rounded = Math.round(2.5); // 3
```

- **Math.ceil(double a):** Restituisce il più piccolo intero maggiore o uguale a a.

```
double result = Math.ceil(2.3); // 3.0
```

- **Math.floor(double a):** Restituisce il più grande intero minore o uguale a a.

```
double result = Math.floor(2.7); // 2.0
```

- **Math.abs(double a):** Restituisce il valore assoluto del numero.

```
double result = Math.abs(-5.0); // 5.0
```

4. Funzioni per il calcolo di valori estremi

- **Math.max(double a, double b):** Restituisce il valore massimo tra due numeri.

```
double result = Math.max(10, 20); // 20.0
```

- **Math.min(double a, double b):** Restituisce il valore minimo tra due numeri.

```
double result = Math.min(10, 20); // 10.0
```

5. Costanti

La classe Math contiene alcune costanti predefinite che sono utilizzate frequentemente in calcoli matematici:

- **Math.PI:** Il valore di π (pi), utilizzato in calcoli geometrici.

```
System.out.println(Math.PI); // 3.141592653589793
```

- **Math.E:** La costante e, base del logaritmo naturale.

```
System.out.println(Math.E); // 2.718281828459045
```

Esempi di utilizzo della classe Math

Esempio 1: Calcolo della distanza tra due punti in un piano cartesiano (Teorema di Pitagora)

```
public class MathExample {  
    public static void main(String[] args) {  
        double x1 = 1.0, y1 = 2.0, x2 = 4.0, y2 = 6.0;  
  
        // Calcola la distanza tra i due punti  
        double distance = Math.sqrt(Math.pow(x2 - x1, 2) + Math.pow(y2 - y1, 2));  
  
        System.out.println("La distanza tra i due punti è: " + distance); // 5.0  
    }  
}
```

Esempio 2: Calcolo del seno, coseno e tangente di un angolo in radianti

```
public class TrigonometryExample {  
    public static void main(String[] args) {  
        double angleInRadians = Math.toRadians(45); // 45 gradi in radianti  
  
        double sin = Math.sin(angleInRadians);  
        double cos = Math.cos(angleInRadians);  
        double tan = Math.tan(angleInRadians);  
  
        System.out.println("Seno: " + sin); // 0.7071067811865475  
        System.out.println("Coseno: " + cos); // 0.7071067811865476  
        System.out.println("Tangente: " + tan); // 1.0  
    }  
}
```

}

Conclusione

La classe `Math` fornisce un'ampia gamma di metodi utili per eseguire calcoli matematici in Java. Essendo una classe statica, tutti i metodi sono accessibili senza dover creare un'istanza della classe, il che la rende molto pratica per operazioni matematiche di base e avanzate. Utilizzando correttamente i metodi di `Math`, puoi eseguire operazioni come il calcolo delle radici, dei logaritmi, delle funzioni trigonometriche e molte altre, in modo semplice ed efficiente.

I membri della classe `Math`.

Questa classe serve per fare calcoli matematici e ha due attributi:

Costanti pubbliche statiche

`static double E` ; //E di Eulero

`static double PI`; //Pi greca

Mette a disposizione i metodi per le principali funzioni matematiche

- valore assoluto,
 - tangente,
 - logaritmo,
 - potenza,
 - massimo,
 - minimo,
 - seno,
 - coseno,
 - esponenziale,
 - radice quadrata
 - arrotondamento classico, per eccesso e per difetto
 - generazione di numeri casuali
-

Metodi pubblici statici:

- **double abs(double a)**
 - **float abs(float a)**
 - **int abs(int a)**
 - **long abs(long a)**
 - **double acos(double a)**
 - **double asin(double a)**
 - **double atan(double a)**
 - **double atan2(double y, double x)**
 - **double ceil(double a)**
 - **double cos(double a)**
 - **double exp(double a)**
 - **double floor(double a)**
 - **double log(double a)**
-

Metodi pubblici statici:

- **double max(double a, double b)**
- **float max(float a, float b)**
- **int max(int a, int b)**
- **long max(long a, long b)**
- **double min(double a, double b)**
- **float min(float a, float b)**
- **int min(int a, int b)**
- **long min(long a, long b)**
- **double pow(double a, double b)**
- **double random()**
- **double rint(double a)**
- **long round(double a)**
- **int round(float a)**
- **double sin(double a)**

- **double sqrt(double a)**
- **double tan(double a)**
- **double toDegrees(double angrad)**
- **double toRadians(double angdeg)**