

Classe Math

A classe Math tem como objetivo dispor aos desenvolvedores funções matemáticas, numeração randômicas e verificação de valores (maiores ou menores).

Auxilia muito, pois não há necessidade do desenvolvedor criar uma classe ou método para desenvolver tais ações matemáticas.

Para utilizar essa classe basta chamar pelo nome dela e em seguida a função, exemplo:

- Math.pow();
- Math.max();
- Math.cos();

Vamos apresentar alguns exemplos e em seguida dispor alguns exercícios para compreender seu funcionamento 😊

MAX & MIN

Esses dois métodos verificam dois valores distintos e retornam o maior e o menor valor. Infelizmente ela apenas compara apenas dois valores, porém futuramente você irá aprender a trabalhar com múltiplos valores utilizando essas funções.

O exemplo abaixo faz a comparação entre a posição 0 do array e a posição 1:

```
package Pacote;

public class Principal {

    public static void main(String[] args) {

        //Array de números
        double numeros[] = {1, 15};

        //Exibir o maior e o menor número
        System.out.println("O maior número é " + Math.max(numeros[0], numeros[1]));
        System.out.println("O menor número é " + Math.min(numeros[0], numeros[1]));

    }
}
```

POTÊNCIA E RAÍZES

Na classe Math podemos trabalhar com potenciação e raízes, existem os seguintes métodos nessa classe:

- pow (base, expoente) - calcula a potência da base elevada ao expoente.
- sqrt (número) - calcula a raiz quadrada de um número
- cbrt (número) - calcula a raiz cúbica de um número
- exp (expoente) - calcula o valor da constante de Euler (E) elevado ao expoente

Vamos exemplificar realizando uma raiz quadrada:

```
package Pacote;  
  
public class Principal {  
  
    public static void main(String[] args) {  
  
        //Potência  
        System.out.println(Math.pow(5, 2));  
  
    }  
}
```

LOGARITMO

Na classe Math encontramos funções para cálculo de logaritmo natural, na base de 10 e a soma do número mais 1. Tais métodos são:

- log (número) - logaritmo natural de um número.
- log10 (número) - logaritmo natural de um número na base 10
- log1p (número) - logaritmo natural de um número somado a 1. Esse método retorna um resultado mais confiável se o número em questão for muito próximo a 0 ou um número fracionado. Ou seja, o resultado não é o mesmo entre os métodos log1p (0.1f) e log (1+0.1f).

```
package Pacote;  
  
public class Principal {  
  
    public static void main(String[] args) {  
  
        //Variável nr contendo um número  
        double nr = 5;  
  
        //Trabalhando com as funções log  
        System.out.println("Resultado: " + Math.log(nr+1));  
  
    }  
}
```

TROGONOMETRIA

A maior parte dos métodos encontrados na classe Math são trigonométricas, o que ajuda muito em cálculos mais complexos que envolvem figuras.

- `sin (graus)` - este método retorna o valor do seno de acordo com o número de graus passado como parâmetro.
- `cos (graus)` - este método retorna o valor do cosseno de acordo com o número de graus passado como parâmetro.
- `tan (graus)` - este método retorna o valor da tangente de acordo com o número de graus passado como parâmetro.
- `asin (graus)` - este método retorna o valor do arco seno de acordo com o número de graus passado como parâmetro.
- `acos (graus)` - este método retorna o valor do arco cosseno de acordo com o número de graus passado como parâmetro.
- `atan (graus)` - este método retorna o valor do arco tangente de acordo com o número de graus passado como parâmetro.
- `sinh (graus)` - este método retorna o valor hiperbólico do seno de acordo com o número de graus passado como parâmetro.
- `cosh (graus)` - este método retorna o valor hiperbólico do coseno de acordo com o número de graus passado como parâmetro.
- `tanh (graus)` - este método retorna o valor hiperbólico da tangente de acordo com o número de graus passado como parâmetro.
- `hypot (x , y)` - retorna o valor da hipotenusa, ou, basicamente, a distância entre dois pontos fundamentada na fórmula $\sqrt{x^2+y^2}$ » `[sqrt (pow(x, 2) + pow(y,2))]`.
- `toRadians (graus)` - retorna um valor aproximado de radianos de acordo com o ângulo medido em graus.
- `toDegrees (raio)` - retorna um valor aproximado de graus de acordo com o ângulo medido em raios.

```
package Pacote;

public class Principal {

    public static void main(String[] args) {

        double ponto1 = 30;
        double ponto2 = 40;

        System.out.println("A distancia entre o " + ponto1 + " e o " + ponto2 + " é " + Math.hypot(ponto1, ponto2));

    }
}
```

NÚMEROS RANDÔMICOS

Nesse próximo exemplo iremos criar um número aleatório, que chamamos de random.

O método **random()** retorna um valor double em 0.0 e 1.0.

Para desenvolvermos um random precisamos dos seguintes parâmetros:

- Limite Máximo;
- Limite Mínimo;

Além do limite mínimo e máximo precisamos saber que essa função retorna com casas decimais, então precisamos zerar as casas decimais, para isso há um método chamado **round()**.

Em seguida podemos remover a casa decimal zerada pelo **round()** utilizando a opção **(int)** que irá servir como filtro e armazenar apenas o número inteiro.

Vamos ao exemplo:

```
package Pacote;

import javax.swing.JOptionPane;

public class Principal {

    public static void main(String[] args) {

        //Variáveis
        int limiteMinimo = 0;
        int limiteMaximo = 11;

        //Gerar random (Irá gerar um número random quebrado)
        double gerarAleatorio = Math.random();

        //Atribuir um número entre 0 a 10
        int numero = (int) Math.round(limiteMinimo + gerarAleatorio * limiteMaximo);

        //Exibir o número aleatório
        System.out.println(numero);

    }
}
```

Para maiores informações sobre essa classe você poderá acessar o site completo sobre ela:

<https://docs.oracle.com/javase/7/docs/api/java/lang/Math.html>

Fonte para a elaboração deste tutorial:

- <http://www.tiexpert.net/programacao/java/math.php>
- <http://www.javaprogressivo.net/2012/09/classe-math-constantes-principais.html>
- http://www.tutorialspoint.com/java/lang/java_lang_math.htm

Exercícios

1. Utilizando os conceitos vistos desenvolva uma calculadora que irá pedir:
 - a. Uma operação;
 - b. Um número (ou dois dependendo da operação escolhida).

As operações a disposição do usuário serão:

- Soma;
- Subtração;
- Multiplicação;
- Divisão;
- Seno;
- Cosseno;
- Tangente;
- Log;
- Raiz Quadrada;
- Raiz Cúbica.

Após informado o valor deverá ser perguntado se o usuário deseja continuar a utilizar a calculadora ou não, enquanto não informar um termo de sua preferência para sair deverão ser realizadas as perguntas e os cálculos.

2. Desenvolva um sistema onde irá gerar um número entre 0 e 100. O usuário deverá tentar acertar esse número, enquanto não acertar o número gerado deverá ser perguntado um número para realizar o comparativo.

Sempre que o palpite dado pelo usuário for diferente do número gerado deverá ser dado os seguintes auxílios:

- a) Se a diferença entre o número pensado e o chute for de até 10, o computador diz que está perto.
- b) Se a diferença entre o número pensado e o chute for maior que 10 e até 20, o computador diz que está longe.
- c) Se a diferença entre o número pensado e o chute for maior que 20, o computador diz: "Ihhhhh! Muito longe".

Após acertar o número o computador deve informar o número de palpites que foi necessário e imprimir as mensagens abaixo de acordo com o número de tentativas.

- a) Caso acerte de primeira: "Nossa! Acertou na mosca"
- b) Até 5 palpites - "Parabéns! Você é bom de adivinhação"
- c) Até 10 - "Muito bom. Continue assim"
- d) Até 20 - "Mm... Podia ser melhor. Continue tentando"
- e) e) Acima de 20 - "Meio lerdo você, hein?"