



**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**
RIO GRANDE DO SUL
Câmpus Feliz

Break, continue e métodos da classe Math

Prof. Moser Fagundes

Programação II

Sumário

- `break` e `continue`
- Métodos `Math`
- `Random`
- **Exercícios**

Antes de iniciar a programar ...

- Crie um **projeto** Java no Eclipse

Sugestão: nomear o projeto como **Aula-03**

- Crie um **pacote** dentro do projeto

Sugestão: nomear o pacote como **aula03**

Break

A instrução **break** altera o fluxo de execução de uma estrutura de controle:


- A instrução **break**, quando executada em uma estrutura **while**, **do while**, **for** ou **switch**, causa a **saída imediata** desta estrutura. A execução continua com a primeira instrução após a estrutura de controle.

Exemplo usando break

```
package aula03;

public class TesteBreak {

    public static void main(String[] args) {
        for(int i=0; i < 10; i++) {
            if(i == 5) {
                break;
            }
            System.out.println(i);
        }
        System.out.println("Saiu!");
    }
}
```



O **break** faz com que o laço for termine a sua execução quando i for igual a 5

Continue

A instrução **continue** altera o fluxo de execução de uma estrutura de controle:


- A instrução **continue**, quando executada em uma estrutura **while**, **do while** ou **for**, pula as próximas instruções da estrutura e **prossegue com a próxima iteração do laço**.

Exemplo usando continue

```
package aula03;

public class TesteContinue {

    public static void main(String[] args) {
        for(int i=0; i < 10; i++) {
            if(i == 5) {
                continue;
            }
            System.out.println(i);
        }
        System.out.println("Saiu!");
    }
}
```



O **continue** faz com que o laço for pule para a próxima iteração, **sem** executar as instruções após o **continue**.

Métodos

- Ao longo dos anos, a experiência mostrou que a melhor maneira de desenvolver e manter um programa grande é construí-lo a partir de **pedaços pequenos e simples**, chamados **módulos**.
- Na linguagem Java, estes módulos correspondem aos **métodos e classes**.

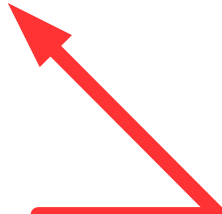
Métodos da classe **Math**

- A classe **Math** nos permite realizar certos cálculos matemáticos comuns.
- Os métodos desta classe são chamados através do nome da classe, seguido de um ponto, do nome do método e dos parâmetros entre parênteses.

Exemplo: **Math.sqrt(900) ;**

Exemplo usando a classe Math

```
package aula03;  
  
public class TesteMath {  
  
    public static void main(String[] args) {  
  
        System.out.println(Math.sqrt(900));  
  
    }  
}
```



Imprime a raiz
quadrada de 900

Métodos da classe Math

// Troque x e y por números

```
Math.abs(x) ;    // Valor absoluto
Math.ceil(x) ;   // Arredonda para cima
Math.floor(x) ;  // Arredonda para baixo
Math.max(x,y) ;  // Retorna o maior
Math.min(x,y) ;  // Retorna o menor
Math.pow(x,y) ;  // Potencia: x elevado a y
Math.sqrt(x) ;   // Raiz quadrada
```

Números aleatórios em Java

```
package exemplos;
```

```
import java.util.Random;
```

```
public class Gerador {
```

```
    public static void main(String[] args) {
```

```
        Random r = new Random();
```

```
        for (int i = 0; i < 30; i++) {
```

```
            int numero = r.nextInt(6);
```

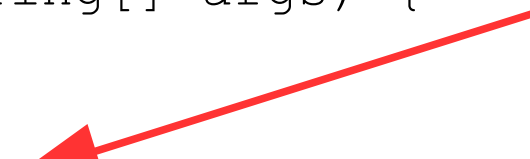
```
            System.out.println(numero);
```

```
        }
```

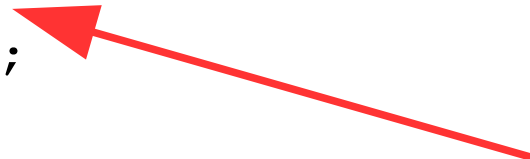
```
    }
```

```
}
```

Cria o gerador
de números
aleatórios



Gera um
número inteiro
entre 0 e 5.



Exercícios

Lista-03 no Moodle!

- Crie um projeto no Eclipse chamado **Lista-03**
- Crie um pacote para cada exercício:
 - ex1
 - ex2
 - ex3
 - ...
- Dentro de cada pacote, crie uma classe para programar o exercício em questão.