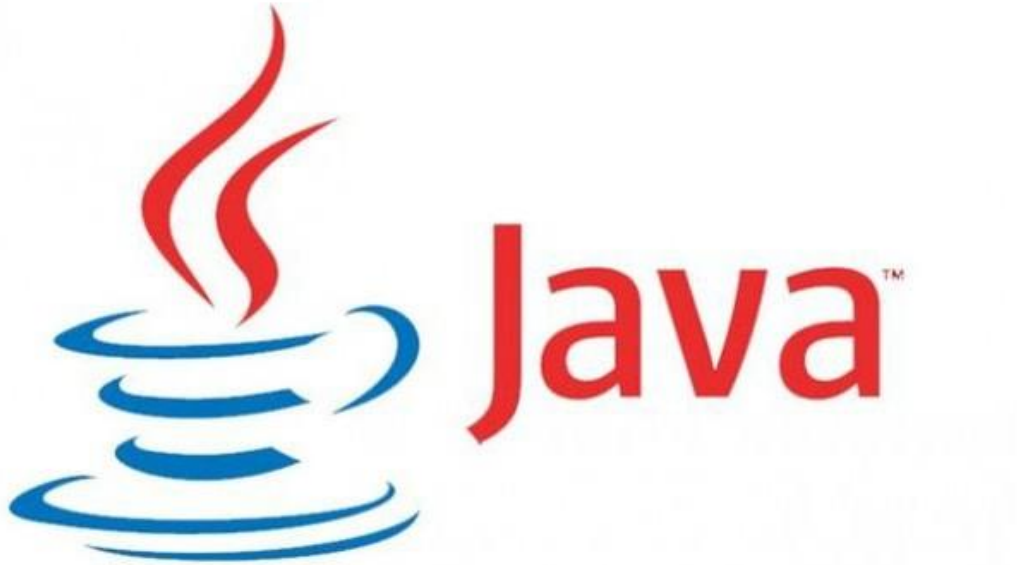
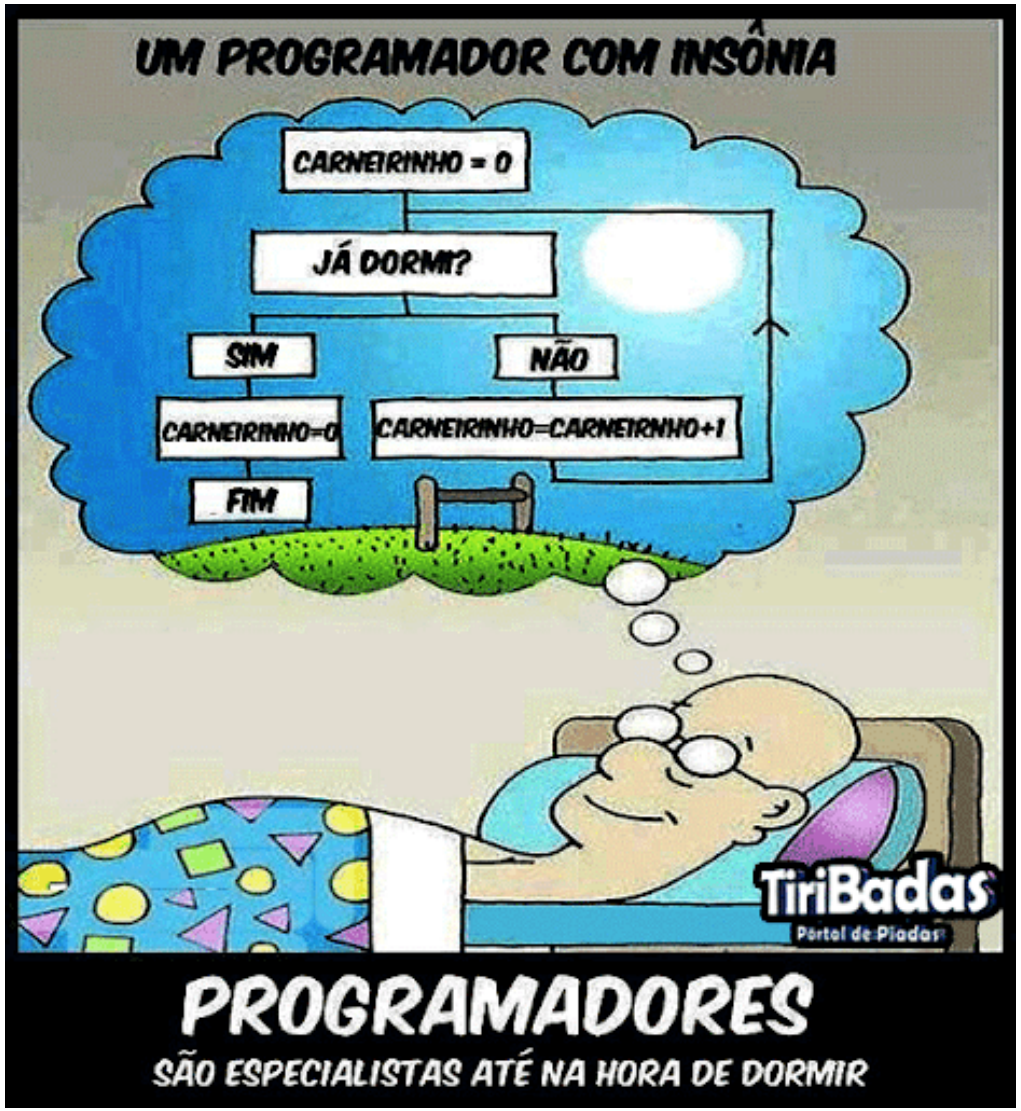


JAVA & BlueJ

- **String e Math**



Sumário

1. Operações com texto
2. Operações básicas com String
3. Comparando strings
4. Math
5. Randômicos

1. Operações com texto

Assinatura	Descrição
char charAt(int index)	Retorna o caractere contido na posição especificada pelo parâmetro index.
int compareTo(String anotherString)	Compara o texto de duas strings e retorna um número inteiro que define a ordem destes textos. Zero significa que os textos são iguais, um número positivo indica que esta string é posterior ao seu argumento e um número negativo indica que esta string é anterior ao argumento.
int compareToIgnoreCase(String anotherString)	Realiza a mesma tarefa que o método compareTo( ). A única diferença é que ignora diferenças entre caracteres minúsculos e maiúsculos.
boolean endsWith(String suffix)	Testa se a string corrente termina com o sufixo especificado pelo parâmetro suffix.
boolean equals(Object anObject)	Compara a string corrente com o objeto especificado pelo parâmetro anObject.
boolean equalsIgnoreCase(String anotherString)	Compara a string corrente com outra string sem levar em conta a distinção entre caracteres maiúsculos e minúsculos.

1. Operações com texto

Assinatura	Descrição
int indexOf(String str)	Retorna a posição inicial, na string corrente, da substring especificada pelo parâmetro str. Se a substring não for encontrada, o retorno é –1.
int lastIndexOf(String str)	Retorna a posição inicial, na string corrente, da última ocorrência da substring especificada pelo parâmetro str. Se a substring não for encontrada, o retorno é –1.
int length( )	Retorna o número de caracteres contidos na string
String replaceAll(String regex, String replacement)	Retorna uma string resultante da substituição, na string corrente, de cada ocorrência da substring especificada no parâmetro regex pela substring especificada no parâmetro replacement.
boolean startsWith(String prefix)	Testa se a string corrente começa com o prefixo especificado pelo parâmetro prefix.
String substring(int beginIndex, int endIndex)	Retorna uma nova string com a seqüência de caracteres que se encontra entre as posições especificadas pelos parâmetros beginIndex e endIndex.
String toLowerCase( )	Retorna uma nova string contendo todos os caracteres da string atual convertidos para minúsculo.
String toUpperCase( )	Retorna uma nova string contendo todos os caracteres da string atual convertidos para maiúsculo.

2. Operações básicas com String

AnalisadorDeTexto.java

```java
import java.io.PrintStream;
import java.util.Scanner;

public class AnalisadorDeTexto
{
  public static void main(String[] args)
  {
    String artista = "";
    PrintStream saida = System.out;

    Scanner scan = new Scanner(System.in);
    saida.print("\nInforme o nome de um artista:\t");
    artista = scan.nextLine();

    saida.println("\nAnálise do nome:\t\t" + artista);

    saida.print("Conversão para maiúsculo:\t");
    saida.println(artista.toUpperCase());

    saida.print("Conversão para minúsculo:\t");
    saida.println(artista.toLowerCase());

```

2. Operações básicas com String

```
23          saida.print("Substituindo o espaco por '_':\t");
24          saida.println(artista.replaceAll(" ","_"));
25
26          saida.print("Quantidade de caracteres:\t");
27          saida.println(artista.length());
28
29          saida.print("A primeira letra:\t\t");
30          saida.println(artista.charAt(0));
31
32          saida.print("Posição da primeira letra 'a':\t");
33          saida.println(artista.indexOf("a"));
34
35          saida.print("Posição da última letra 'a':\t");
36          saida.println(artista.lastIndexOf("a"));
37
38          saida.print("O primeiro nome:\t\t");
39          saida.println(artista.substring(0,artista.indexOf(" ")));
40
41          saida.print("O último sobrenome:\t\t");
42          saida.println(artista.substring(artista.lastIndexOf(" ") + 1,
43            artista.length()));
44      }
45  }
```

```
Informe o nome de um artista:   Wolfang Amadeus Mozart

Análise do nome:                Wolfang Amadeus Mozart
Conversão para maiúsculo:       WOLFANG AMADEUS MOZART
Conversão para minúsculo:       wolfang amadeus mozart
Substituindo o espaco por '_':  Wolfang_Amadeus_Mozart
Quantidade de caracteres:       22
A primeira letra:               W
Posição da primeira letra 'a':  4
Posição da última letra 'a':    19
O primeiro nome:                Wolfang
O último sobrenome:             Mozart
```

3. Comparando strings

ComparaTexto.java

```java
import java.io.PrintStream;
import java.util.Scanner;

public class ComparaTexto
{
  public static void main(String[] args)
  {
    PrintStream saida = System.out;
    Scanner scan = new Scanner(System.in);

    saida.print("\nInforme um texto:\t");
    String t1 = scan.nextLine();

    saida.print("Informe outro texto:\t");
    String t2 = scan.nextLine();

    saida.println("\nComparações:");
    saida.println("Igualdade (C.S.):\t" + (t1.equals(t2)));
    saida.println("Igualdade:\t\t" + (t1.equalsIgnoreCase(t2)));
    saida.println("Ordem (C.S.):\t\t" + (t1.compareTo(t2)));
    saida.println("Ordem:\t\t\t" + (t1.compareToIgnoreCase(t2)));
    saida.println();
  }
}
```

```
Informe um texto:       Gata
Informe outro texto:    gato

Comparações:
Igualdade (C.S.):       false
Igualdade:              false
Ordem (C.S.):           -32
Ordem:                  -14
```

4. Math

Matematica.java

```java
public class Matematica
{
  public static void main(String[] args)
  {
    System.out.println();

    System.out.println("Valor de E:\t"  + Math.E);
    System.out.println("Valor de PI:\t" + Math.PI);

    System.out.println("\nNúmero aleatório:\t" + Math.random());
    System.out.println("Raiz quadrada:\t\t" + Math.sqrt(16));
    System.out.println("Potência:\t\t" + Math.pow(2,4));

    System.out.println("\nPróximo inteiro:\t" + Math.ceil(4.1));
    System.out.println("Inteiro anterior:\t" + Math.floor(4.9));

    System.out.println("\nArredondamento:");
    System.out.println("Com rint( ):\t" + Math.rint(4.6));
    System.out.println("Com round( ):\t" + Math.round(4.6));

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

```
Valor de E:     2.718281828459045
Valor de PI:    3.141592653589793

Número aleatório:       0.7391301471905012
Raiz quadrada:          4.0
Potência:               16.0

Próximo inteiro:        5.0
Inteiro anterior:       4.0

Arredondamento:
Com rint( ):    5.0
Com round( ):   5
```

5. Randômicos

ValorAleatorio.java

```java
import java.io.PrintStream;
import java.util.Random;

public class ValorAleatorio
{
  public static void main(String[] args)
  {
    PrintStream saida = System.out;
    Random acaso = new Random();

    saida.println("\nTipo booleano:\t\t" + acaso.nextBoolean());
    saida.println("Qualquer inteiro:\t"  + acaso.nextInt());
    saida.println("Inteiro de 0 a 99:\t" + acaso.nextInt(100));
    saida.println("Inteiro longo:\t\t"   + acaso.nextLong());
    saida.println("Tipo float:\t\t"      + acaso.nextFloat());
    saida.println("Tipo double:\t\t"     + acaso.nextDouble());
  }
}
```

```
Tipo booleano:          true
Qualquer inteiro:       -2137923408
Inteiro de 0 a 99:      25
Inteiro longo:          -6950392094073074149
Tipo float:             0.3247714
Tipo double:            0.023165131503567138
```