

LINGUAGEM DE PROGRAMAÇÃO JAVA

Cefet – Maracanã -RJ BCC/TSI

Prof. Gustavo Guedes E-mail: gustavo.guedes@cefet-rj.br

Métodos úteis

- Object.equals(): boolean
- Object.toString(): String
- String.toCharArray(): char []
- String.split(String): String []
- String.toLowerCase():String
- String.toUpperCase(): String
- String.trim(): String
- List.removeAll(Collection): boolean
- o List.add(Objetct): boolean
- List.indexOf(Object): int
- List.remove(int): Object
- List.remove(Object): boolean
- List.contains(Object): boolean
- Integer.parseInt(String)
- Double.parseDouble(String)

Comparação (==)

- Vamos pensar o seguinte:
- String x = new String("casa");
- String y = new String("casa");
- \circ x == y?
- String g = "carro";
- String g2= "carro";
- \circ g == g2?
- String u = "cama";
- String u2 = new String("cama");
- \circ u==u2?

OBJECT.EQUALS

- Vamos imaginar uma classe Data com um atributo do tipo String: data. Imagine os getters e setters.
- Vamos imaginar que temos uma lista com vários objetos data: "15/10/2010", "10/10/2011", "05/10/1977", etc.
- Queremos saber se na lista já existe um objeto data com data = "10/10/2011". Como fazer isso?
- Imagine o método:
 - boolean existeNaLista(Data x).

EXEMPLO DE STRING

• Exemplo apresentado em sala sobre a razão que devemos comparar String's com equals.

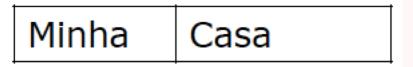
```
public class TestString {
    public static void main(String[] args) {
        String a = "gustavo";
        String b=new String("gustavo");
        String d="gustavo";
        String c="gustavo";
        c = c.trim();
        System.out.println("#"+a+"#");
        System.out.println("#"+c+"#");
        System.out.println(a==c);
    }
}
```

STRING.TOCHARARRAY()

- Ao usar:
 - String x = "Minha casa".
 - char meu_array [] = x.toCharArray();
- o Crie um método para inverter uma String.

STRING.SPLIT(STRING)

- Ao usar:
 - String x = "Minha casa".
 - String [] array = x.split(" ");



- Isso me retorna um array de Strings:
- Dada uma String, conte quantas vezes aparece a palavra X.

STRING.TOLOWERCASE() E TOUPPERCASE()

- Ao usar:
- String x = "Minha casa".
- o String k = x.toLowerCase();
- k= minha casa
- o String k = x.toUpperCase();
- k= MINHA CASA

STRING.TRIM()

- Ao usar:
- String x = "Minha casa".
- o String k= x.trim();
- K="Minha Casa"

INTEGER.PARSEINT(STRING)

- Integer.parseInt("1")
- Retorna "1" como int 1.