ESTRITAMENTE **CONFIDENCIAL**

ELABORATA
INFORMÁTICA

LINGUAGEM JAVA

Strings e Caracteres

ELABORATA INFORMÁTICA

Tópicos

- Strings
- Caracteres

ELABORATA INFORMÁTICA

Strings

Descrição

- Uma string é um conjunto de caracteres que representam um dado ou uma informação. Na Linguagem Java, string não é um tipo primitivo e sim uma classe

Exemplo

```
String texto1 = new String("Linguagem Java");
String texto2 = "String são objetos";
```

Comparando strings

- Pelo fato de strings serem objetos na Linguagem Java, a comparação entre strings não pode ser feita utilizando-se o operador de igualdade ==

O que acontece se utilizar o ==

- Quando se usa o operador == para comparar dois objetos o que esta sendo avaliado é o objeto em si e não o seu conteúdo.

equals

- O equals é o recurso de um objeto do tipo String utilizado para comparar duas strings.

Exemplo equals

```
public class ComparandoStrings
{
  public static void main(String[] args) {
    String texto1 = new String("Linguagem Java");
    if (texto1 == "Linguagem Java") {
      System.out.println("Contem o texto");
    } else {
      System.out.println("Não contem o texto");
    }
    if (texto1.equals("Linguagem Java")) {
      System.out.println("Contem o texto");
    } else {
      System.out.println("Não contem o texto");
    }
  }
}
```

Concatenando strings

- Concatenar strings significa unir duas ou mais strings em uma só. Para concatenar duas strings deve-se utilizar o operador **+**.

Exemplo concatenação de strings

```
public class ContatenandoStrings
{
  public static void main(String[] args) {
    String str1 = new String("Linguagem");
    String str2 = new String("Java");

    String str3 = str1 + " " + str2;

    System.out.println(str3);
  }
}
```

StringBuilder

- Uma string na Linguagem Java é um objeto imutável. Isto significa que quando uma string é criada, seu conteúdo não é alterado. Para concatenações onde o resultado deve ser armazenado na mesma string deve-se utilizar a classe **StringBuilder**.

Exemplo de StringBuilder

```
public class ExemploStringBuilder
{
  public static void main(String[] args) {
    StringBuilder str = new StringBuilder();
    str.append("Linguagem");
    str.append(" ");
    str.append("Java");
    System.out.println(str.toString());
  }
}
```

Extraindo partes de uma string

- Para extrair partes de uma string deve-se utilizar o recurso chamado substring de um objeto do tipo String.

Exemplo substring

```
public class ExtraindoDadosString
{
  public static void main(String[] args) {
    String curso = new String("Curso Linguagem Java");
    String trecho = curso.substring(0, 5);
    System.out.println(trecho);
  }
}
```

Exercício 01

- Criar um programa que solicite um login e verifique se o login é válido. Se o login for inválido, ou seja, for vazio, o programa deve mostrar a mensagem Login inválido, tente novamente e solicitar novamente o login. O programa deve solicitar o login até o usuário informar um login válido. O login será válido quando for diferente de espaço em branco.

Exercício 02

- Criar um programa que solicite o nome completo do usuário e depois mostra na tela o primeiro nome do usuário.

ELABORATA
INFORMÁTICA

Caracteres

Descrição

- Um caractere é um símbolo gráfico utilizado numa linguagem textual. Um conjunto de caracreres forma uma palavra ou uma frase. Na Linguagem Java uma palavra ou uma frase é uma string.

Caracteres unicode

- Diferente de outras linguagens de programação, na Linguagem Java é possível armazenar caracteres unicode numa variável do tipo **char**.

Exemplo caractere unicode

```
char letraAemUnicode = '\u0061';
System.out.println(letraAemUnicode);
```

Identificando caracteres

Recurso	Significado
isAlphabetic	Identifica se o caractere é um dígito ou uma letra
isDigit	Identifica se o caractere é um dígito
isLetter	Identifica se o caractere é uma letra
isLowerCase	Identifica se o caractere é uma letra em minúsculo
isUpperCase	Identifica se o caractere é uma letra em maiúsculo
isWhitespace	Identifica se o caractere é um espaço em branco
toLowerCase	Converte o caractere para minúsculo
toUpperCase	Converte o caractere para maiúsculo

Exemplo

```
public class ExemploCaracteres
{
  public static void main(String[] args) {
    String texto = new String("Curso Linguagem Java");
    boolean temEspacoEmBranco = false;
    char tmp;
    for (int i=0; i < texto.lenght(); i++) {
      tmp = texto.charAt(i);
      if (Character.isWhitespace(tmp)) {
        temEspacoEmBranco = true;
        break;
      }
    }
    if (temEspacoEmBranco) {
      System.out.println("O texto tem um espaço em branco.");
    } eles {
      System.out.println("O texto não tem um espaço em branco.");
    }
  }
}
```

Exercício 01

- Criar um programa que leia um texto e depois mostre na tela a quantidade de caracteres excluindo os espaços em branco.

Exercício 02

- Criar um programa para verificar se um nome de pessoa é um nome válido. Um nome válido para pessoa é aquele que contem apenas letras e espaços em branco.

OBRIGADO!

ESTRITAMENTE **CONFIDENCIAL**

ELABORATA
INFORMÁTICA

www.**elaborata**.com.br

Horário de Atendimento Comercial
Segunda à sexta – das 9:00h às 19:30h e
Sábado - das 8:00h às 15:00h.

Rua Monsenhor Celso, 256 - 1º Andar
Centro - Curitiba - PR

41.3324.0015
41.99828.2468

cursos@elaborata.com.br