

Programação Orientada a Objetos
SANTOS, Rafael (PLT)

Capítulo 12 – Strings em Java

Introdução

- São objetos que encapsulam **cadeias de caracteres** delimitadas por **aspas**
- **Caracteres** são do tipo primitivo **char** e delimitados por **apóstrofos**; são números inteiros positivos que **representam** o código de um caractere

Criação de instâncias de String

- Criadas quando:
 - Escrevemos uma cadeia de caracteres
 - Usamos um dos construtores da classe String
 - Como resultado da execução de um método que retorne uma String
- Entre os construtores da classe String existe um que recebe um objeto da classe String e outro recebe um array de char

Métodos básicos

- length()
- charAt()
- Ver classe
 - JogoDaForca
 - DemoJogoDaForca
- toCharArray()

Métodos para comparação

- **Não** use o operador relacional `==` para comparar objetos da classe `String`
- `equals()`
- `equalsIgnoreCase()`
- Ver as classes:
 - `Login`
 - `DemoLogin`
 - `URL`
 - `DemoURL`

Métodos para pesquisa

- `indexOf(String ou char, int)`
- Ver classe
 - `EpocaDeFrutas`
- `lastIndexOf(String ou char, int)`

Métodos que criam instâncias

- Como Strings são **imutáveis**, os métodos a seguir criam uma nova instância de String
 - `toLowerCase()`
 - `toUpperCase()`
 - `trim()`
 - `replace(String ou char, String ou char)`
 - `substring(int, int)`

Métodos de conversão

- **Para String:** método da classe String:
 - `valueOf(argumento)`
- **Para outros tipos:** métodos estáticos das classes empacotadoras (wrappers):
 - `Integer.parseInt(String)`
 - `Long.parseLong(String)`
 - `Double.parseDouble(String)`
 - `Float.parseFloat(String)`
 - `Boolean.parseBoolean(String)`

Prática de laboratório

- Crie objetos da classe String, usando os diferentes procedimentos apresentados, inclusive, a partir de um array de caracteres.
- Indique a forma segura para comparar cadeias de caracteres e explique por quê.
- Crie uma classe executável para exemplificar cada um dos métodos da classe String demonstrados em sala.

Exercício extra

- Modifique o exemplo do jogo de forca para:
 1. Estabelecer um número máximo de tentativas
 2. Indicar quantas consoantes e quantas vogais foram informadas pelo jogador
 3. Usar uma variável de instância para referenciar o objeto da classe Scanner
 4. Recusar a entrada de letras que já tenham sido informadas
 5. Aceitar caracteres maiúsculos

StringBuffer e StringBuilder

```
StringBuilder sb =  
    new StringBuilder("Curso de ");  
System.out.println(sb);
```

```
sb.append("Java ");  
System.out.println(sb);
```

```
sb.append(6);  
System.out.println(sb);
```

```
sb.insert(6, "Intermediario ");  
System.out.println(sb);
```

```
sb.delete(6, 13);  
System.out.println(sb);
```

```
sb.reverse();  
System.out.println(sb);
```

StringTokenizer

```
import java.util.StringTokenizer;

public class TestaStringTokenizer {

    public static void main(String[] args) {

        String nomes = "maria,joao,patricia,afranio";
        StringTokenizer st =
            new StringTokenizer(nomes, ",", false);

        while (st.hasMoreTokens()) {
            System.out.println(st.nextToken());
        }

    }
}
```