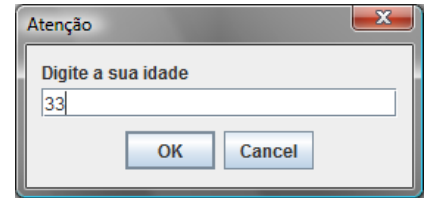


Lista de Exercícios 4 – Strings e Modularização

- 1) Faça o método estático `LerInt()` em Java que recebe como parâmetro um título e uma mensagem. O método deve abrir uma caixa de entrada no estilo `JOptionPane.showInputDialog(...)` com o título e a mensagem recebidos como parâmetro e retornar na forma de um inteiro, o valor digitado pelo usuário. Caso o usuário digite um valor inválido a função deve retornar o valor zero. Faça também versões dessas funções para a leitura de valores reais e strings.

Ex: `int idade=LerInt("Atenção","Digite a sua idade");`



- 2) Desenvolva uma função estática em Java que receba um número de CPF como parâmetro (na forma de String) e retorne um boolean informando sobre a validade do mesmo. Para calcular os dígitos verificadores do CPF existe a seguinte fórmula:

a) Para calcular o 1º dígito do CPF: 111.111.111-11

CPF	1	1	1	.	1	1	1	.	1	1	1	
	x	X	x		x	x	x		x	x	x	
Multiplicar por	10	9	8		7	6	5		4	3	2	
=	=	=	=		=	=	=		=	=	=	
Somar os resultados desta multiplicação	10	9	8		7	6	5		4	3	2	-> Soma = 54

Dividir a soma por 11 e pegar o Resto (parte inteira):
$$\begin{array}{r} 54 \overline{) 11} \\ 10 \quad 4 \end{array}$$

Subtrair este Resto de 11 $\rightarrow 11 - 10 = 1$ (este é o primeiro dígito verificador)

OBS: se o Resultado da subtração for 10 ou 11, o dígito verificador será 0

b) Para calcular o 2º dígito do CPF: 111.111.111-11

CPF	1	1	1	.	1	1	1	.	1	1	1	-	1	
	x	x	x		x	x	x		x	x	x		X	
Multiplicar por	11	10	9		8	7	6		5	4	3		2	
=	=	=	=		=	=	=		=	=	=		=	
Somar os resultados desta multiplicação	11	10	9		8	7	6		5	4	3		2	-> Soma = 65

Dividir a soma por 11 e pegar o Resto (parte inteira):
$$\begin{array}{r} 65 \overline{) 11} \\ 10 \quad 5 \end{array}$$

Subtrair este Resto de 11 $\rightarrow 11 - 10 = 1$ (este é o segundo dígito verificador)

OBS: se o Resultado da subtração for 10 ou 11, o dígito verificador será 0

Utilize a seguinte solução para converter de caracter para inteiro:

```
String cpf="12345678901";
char c=cpf.charAt(0); // retorna o primeiro caracter da string
int i=Character.digit(c,10); // converte de char para int
                        // o parâmetro '10' representa a
                        // base numérica decimal
System.out.println(i); // imprime o valor 1]
```

3) *Desenvolva um método estático com a finalidade de sugerir o significado de uma string recebida como parâmetro. O método deve analisar a string recebida e retornar uma das seguintes sugestões:*

- a. *[NOME], caso a string seja parecida com um nome de pessoa*
- b. *[EMAIL], caso se pareça com um email*
- c. *[VARIÁVEL], para strings que podem ser consideradas como nome de variável em Java*
- d. *[PLACA], se assemelha a uma placa de veículo (formato: AAA-9999)*
- e. *[DESCONHECIDO], caso não se assemelha a nenhum dos casos acima*

4) *Formatação de data - Elaborar uma função estática que recebe como parâmetros o dia, mês e ano como inteiros. A função deverá retornar a data por extenso conforme o exemplo abaixo:*

```
String sdata(int dia, int mês, int ano);
```

Exemplo:

```
String data=sdata(10,04,2010);
```

```
System.out.println(data);    => imprime: 10 de abril de 2010.
```

5) *Palíndromos - Implemente um método que verifica se uma frase é palíndroma (a mesma se lida normalmente ou de trás para a frente: "roma me tem amor", "socorram me subi no onibus em marrocos").*

```
boolean palindromo(String frase);
```

// a função retorna true se a frase for palíndromo e false caso contrário

6) *Peça ao usuário digitar uma frase com no mínimo 45 palavras a qual será lida e atribuída a uma String. Após a digitação, mostre a frase acomodada e justificada em uma área com no máximo 40 colunas. As palavras não poderão ser divididas. Exemplo:*

O usuário digitou:

O mundo é um lugar perigoso de se viver, não por causa daqueles que fazem o mal, mas sim por causa daqueles que observam e deixam o mal acontecer. Albert Einstein

A frase deverá ser apresentada assim:

**O mundo é um lugar perigoso de se
viver, não por causa daqueles que fazem
o mal, mas sim por causa daqueles que
observam e deixam o mal acontecer.
Albert Einstein**

Sugestão: após ler frase e atribuir à String **sfrase**, transfira seqüencialmente todas as palavras para a String **sfraseresult** e adicione o "\n" no final de cada linha, tomando o cuidado para que cada linha não exceda 40 colunas. Imprima depois a String **sfraseresult**.