

Lição 5



Capturando entrada de dados através do teclado

Objetivo

Ao final desta lição, o estudante será capaz de:

- Criar códigos para a captura de dados pelo teclado
- Usar a classe `BufferedReader` para captura, através de uma janela de console, de dados digitados no teclado
- Utilizar a classe `Scanner` para captura, através de uma janela de console, de dados digitados no teclado
- Utilizar a classe `JOptionPane` para captura, através de uma interface gráfica, de dados digitados no teclado



API

- Interfaces de Programação de Aplicações, ou Application Programming Interface (API) contêm centenas de classes pré-definidas que se pode utilizar no programas
- Classes são organizadas dentro do que chamamos de **pacotes**
- **Pacotes** contêm classes que se relacionam com um determinado propósito



Capturando entrada de dados através do teclado

- Existem três formas para entrada de dados:
 - Através da Classe `BufferedReader`
 - Através da Classe `Scanner`
 - Através da Classe `JOptionPane`



Usando a Classe BufferedReader

1. Digite as seguintes instruções no início do programa:

```
import java.io.BufferedReader;  
import java.io.InputStreamReader;  
import java.io.IOException;
```

2. Adicione as seguintes instruções no método main:

```
BufferedReader dataIn = new BufferedReader(new  
    InputStreamReader(System.in));
```

3. Declare uma variável temporária do tipo String para receber a entrada de dados e chame o método readLine() para capturar, linha a linha, o que for digitado. Deve ser colocado em um bloco try-catch:

```
try {  
    String temp = dataIn.readLine();  
} catch (IOException e) {  
    System.out.println("Error in getting input");  
}
```



Usando a Classe Scanner

1. Digite a seguinte linha no início do código:

```
import java.util.Scanner;
```

2. Inicialize uma variável da classe Scanner:

```
Scanner sc = new Scanner(System.in);
```



Usando a Classe Scanner

3. Utilize um dos métodos abaixo para capturar os dados:

Método	Finalidade
<code>next()</code>	Aguarda uma entrada em formato String
<code>nextInt()</code>	Aguarda uma entrada em formato Inteiro
<code>nextByte()</code>	Aguarda uma entrada em formato Inteiro
<code>nextLong()</code>	Aguarda uma entrada em formato Inteiro Longo
<code>nextFloat()</code>	Aguarda uma entrada em formato Número Fracionário
<code>nextDouble()</code>	Aguarda uma entrada em formato Número Fracionário

Usando a Classe JOptionPane

- Outro modo de receber a entrada de dados é utilizar a classe JOptionPane, que pertence ao pacote javax.swing
- A JOptionPane possui métodos que permitem mostrar uma caixa de diálogo para que o usuário informe os dados

Obtendo Dados

- A instrução:

```
name=JOptionPane.showInputDialog("Please enter your name");
```

cria uma caixa de entrada, que exibirá um diálogo com uma mensagem, um campo de texto, para receber os dados do usuário, e um botão OK, como mostrado na figura



Mostrando Dados

- A instrução:

```
JOptionPane.showMessageDialog(null, msg);
```

exibirá um diálogo contendo a mensagem e o botão de OK



Sumário

- Foram discutidos três métodos de capturar entrada de dados utilizando-se as Classes:
 - BufferedReader
 - Scanner
 - JOptionPane
- Breve visão de pacotes



Parceiros

- Os seguintes parceiros tornaram JEDI possível em Língua Portuguesa:

