



**DECIDA O QUE VOCÊ QUER.
FAÇA UM PLANO.
E TRABALHE NISSO TODO SANTO DIA.**

Primeiro Milhão



Desenvolvimento de Aplicações Desktop

Entrada e Saída de Dados

Professor: Charles Leite

Entrada e Saída de Dados

Lembram dos
comandos **leia** e
escreva de Lógica de
Programação?

Entrada e Saída de Dados

- Para a **saída** de dados, Java utiliza o objeto `System.out`, que representa um dispositivo de saída ou um arquivo
 - Por *default*, esse objeto representa a tela de um monitor
 - Para ser mais preciso, o nome do objeto é `out`, do tipo `PrintStream`, que está armazenado na classe `System`, parte do pacote `lang` da biblioteca de classes padrão do JDK

Entrada e Saída de Dados

- O objeto `System.out` possui vários métodos, dentre eles, os métodos (e suas variações):
 - `print()`
 - `println()`
 - Enquanto o método `println()` imprime uma informação na tela e pula para o início da próxima linha, o método `print()` permanece na mesma linha

Entrada e Saída de Dados

- **Exemplo:**

- A seguinte sequência de instruções

```
System.out.print("10, ");
```

```
System.out.print("9 e ");
```

```
System.out.println("8.");
```

```
System.out.println("Essas serão  
as minhas notas em Java, no  
mínimo.");
```

- Gera a seguinte saída na tela:

```
10, 9 e 8.
```

```
Essas serão as minhas notas em  
Java, no mínimo.
```

Entrada e Saída de Dados

- Para **entrada** de dados, Java utiliza a classe `Scanner`, parte do pacote `util` da biblioteca de classes padrão do JDK
- Ela fornece métodos para leitura de valores de diversos tipos
- A entrada pode vir de várias fontes, como, dados digitados pelo usuário ou armazenados em um arquivo
- A classe `Scanner` pode também ser usada para avaliar uma string de caracteres em partes separadas

Entrada e Saída de Dados

- Por exemplo, se quisermos ler dados digitados pelo usuário, a partir do teclado:

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

Objetos em Java são criados com o operador **new**

A fonte de entrada; neste caso, um objeto que representa uma *stream* de entrada padrão, ou seja, o teclado

Entrada e Saída de Dados

- Em seguida, podemos usar os métodos da classe `Scanner`
- Por exemplo:

```
Scanner scan = new Scanner(System.in);  
String nome = scan.nextLine();  
int idade = scan.nextInt();  
double salario = scan.nextDouble();
```

Os métodos `next...` servem para ler os próximos valores digitados pelo usuário

DÚVIDAS ...

