

Exemplo 5

```
double a;  
int b;  
  
a = 5.0;  
b = a;  
  
System.out.println(b);
```

Entrada de dados em Java

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

Entrada de dados

Usuário → **Programa**
(dentro de variáveis)



Dispositivo de ENTRADA

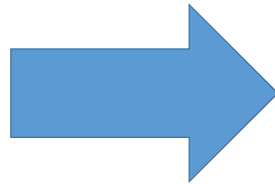


Também chamada de
LEITURA:

"O programa está lendo dados."



32 ENTER



```
int x;
```

Memória RAM

32
x

Scanner

Para fazer entrada de dados, nós vamos criar um objeto do tipo "Scanner" da seguinte forma:

Scanner sc = new Scanner(System.in);

import java.util.Scanner;

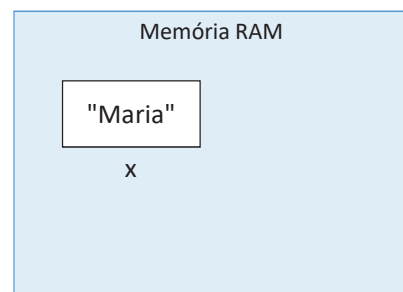
faça sc.close() quando não precisar mais do objeto sc

Para ler uma palavra (texto sem espaços)

Suponha uma variável tipo **String** declarada:

String x;

x = sc.next();

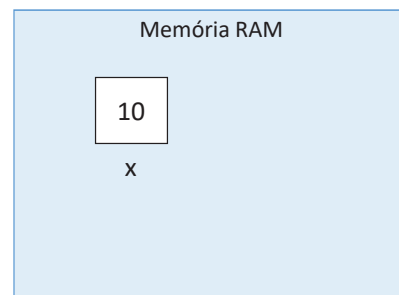


Para ler um número inteiro

Suponha uma variável tipo **int** declarada:

```
int x;
```

```
x = sc.nextInt();
```



Para ler um número com ponto flutuante

Suponha uma variável tipo **double** declarada:

```
double x;
```

```
x = sc.nextDouble();
```

← Localidade do sistema

ATENÇÃO:

Para considerar o separador de decimais como ponto, **ANTES** da declaração do Scanner, faça:

```
Locale.setDefault(Locale.US);
```

Para ler um caractere

Suponha uma variável tipo **char** declarada:

```
char x;
```

```
x = sc.next().charAt(0);
```

Para ler vários dados na mesma linha

```
string x;  
int y;  
double z;
```

```
x = sc.next();  
y = sc.nextInt();  
z = sc.nextDouble();
```

Para ler um texto ATÉ A QUEBRA DE LINHA

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        String s1, s2, s3;

        s1 = sc.nextLine();
        s2 = sc.nextLine();
        s3 = sc.nextLine();

        System.out.println("DADOS DIGITADOS:");
        System.out.println(s1);
        System.out.println(s2);
        System.out.println(s3);

        sc.close();

    }
}
```

ATENÇÃO: quebra de linha pendente

Quando você usa um comando de leitura diferente do `nextLine()` e dá alguma quebra de linha, essa quebra de linha fica "pendente" na entrada padrão.

Se você então fizer um `nextLine()`, aquela quebra de linha pendente será absorvida pelo `nextLine()`.

Solução:

Faça um `nextLine()` extra antes de fazer o `nextLine()` de seu interesse.

```
int x;
String s1, s2, s3;
```

```
x = sc.nextInt();
s1 = sc.nextLine();
s2 = sc.nextLine();
s3 = sc.nextLine();
```

```
System.out.println("DADOS DIGITADOS:");
System.out.println(x);
System.out.println(s1);
System.out.println(s2);
System.out.println(s3);
```

Resumo da aula

- Scanner
 - next()
 - nextInt()
 - nextDouble()
 - next().charAt(0)
- Locale
- Como ler até a quebra de linha
 - nextLine()
 - como limpar o buffer de leitura

Funções matemáticas em Java

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves