

Java

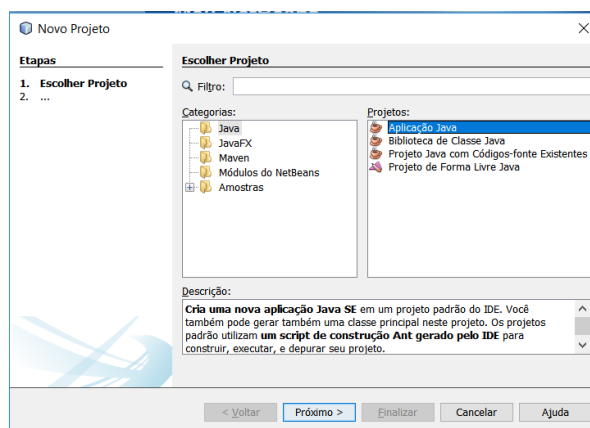
Incompatibilidade entre Tipos

Números x Strings

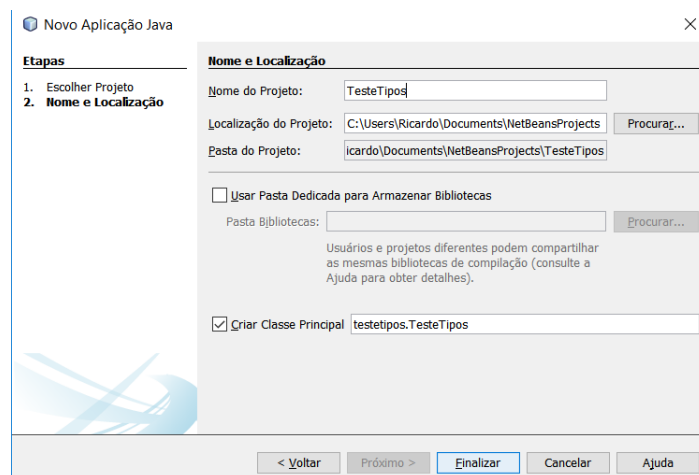
Durante a execução de um software, é muito comum a incompatibilidade entre Números e Strings.

Criaremos programas para testar as incompatibilidade e a compatibilidade entre os Tipos.

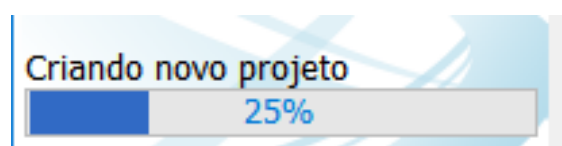
Abra o Netbeans e inicie um Novo Projeto, de Categoria Java e Projetos Aplicação Java, e clique no Botão Próximo.



Nomeie o Projeto como TesteTipos e selecione a caixa Criar Classe Principal e clique no botão Finalizar.



Aguarde o processo.

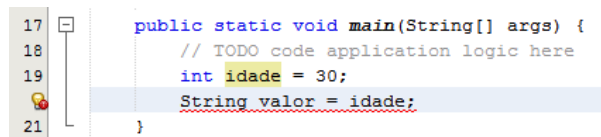


Tarefa 1 – Converter um Valor Inteiro em uma String

1º Erro

Verifique que NÃO é possível atribuir o valor numérico de idade 30 a string idade.

```
Public static void main (String [ ] args ) {  
  
int idade = 30;  
  
String valor = idade;  
  
}
```

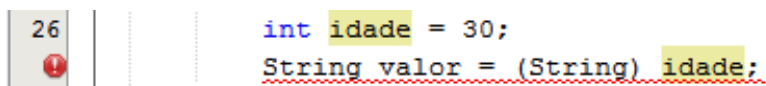


```
17 public static void main(String[] args) {  
18     // TODO code application logic here  
19     int idade = 30;  
20     String valor = idade;  
21 }
```

2º Erro

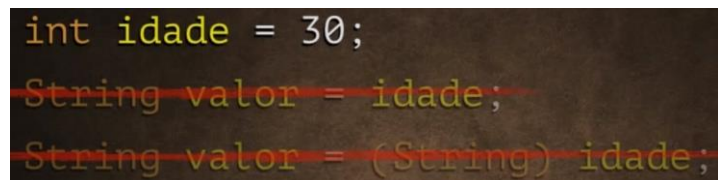
Também NÃO é possível atribuir utilizando Typecast, conforme exemplo.

```
public static void main(String[] args) {  
  
int idade = 30;  
  
String valor = (String) idade;  
  
}
```



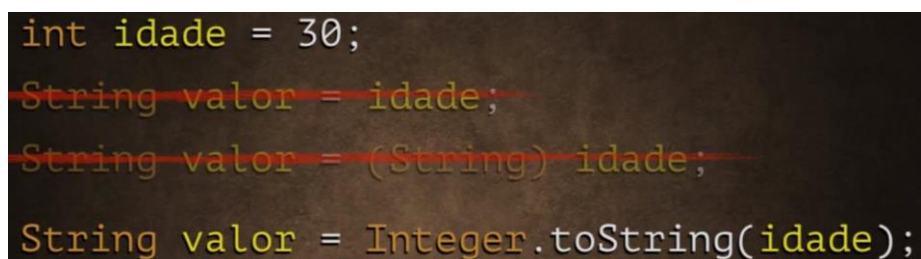
```
26 int idade = 30;  
String valor = (String) idade;
```

As tentativas anteriores NÃO funcionaram pelo fato que String NÃO é um Typecast ; String é uma Classe Invólucra.



```
int idade = 30;  
String valor = idade;  
String valor = (String) idade;
```

Para que essa conversão funcione, será necessário ler a idade que é um número inteiro e converter para String.



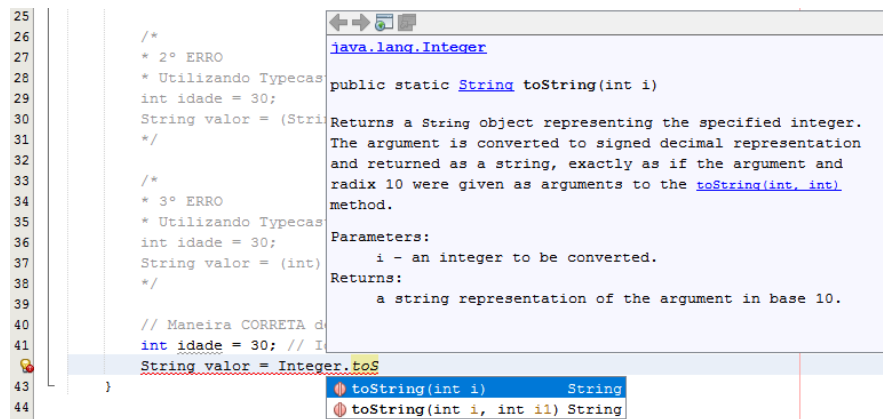
```
int idade = 30;  
String valor = idade;  
String valor = (String) idade;  
String valor = Integer.toString(idade);
```

Solução:

Sendo assim, a maneira correta é fazer com que String valor receba a Wrapper Classe Integer e como toda Classe Invólucra possui Métodos, neste caso, utilizaremos o Método to.String.

O método to.String irá converter um Número Inteiro para String, Integer.toString.

Digite “String valor = Integer.toS”, o menu rápido de ajuda será exibido. Selecione “toString(int i)”.



Desta forma, lendo de trás para frente “idade Integer.toString é igual a String valor”, ou seja, **a idade inteira convertida para String é igual a valor**, através do Método “toString”.

Percebe que dessa vez não foi exibido nenhum erro.

```
int idade = 30;
String valor = Integer.toString(idade);
```

Faça a exibição do resultado incluindo o comando “System.out.println(valor);” e execute o programa.

```
40 // Maneira CORRETA de se programar a conversão dos valores inteiros para string.
41 int idade = 30; // Idade recebe o número inteiro 30
42 String valor = Integer.toString(idade); // conversão de um valor Inteiro para uma String
43 System.out.println(valor); // comando para exibir o valor numério da variável valor
```

Tarefa 2 – Converter uma String em um Valor Inteiro

Faça a atribuição de “30” a variável String valor (String valor = “30”);).

```
51 | String valor = "30"; // neste caso, 30 está em aspas por se tratar de um texto, ou seja, string
```

1º ERRO

Faça com que “int idade” receba “valor” (int idade = valor;) e observe que o Netbeans informará que os Tipos são Incompatíveis.

```
51 | String valor = "30"; // neste caso, 30 está em aspas por se tratar de um texto, ou seja, string
52 |
53 | // 1º ERRO
    | int idade = valor;
```

2º Erro

Utilizando Typecast (int idade = (int) valor;) ainda é informado que os tipos NÃO são compatíveis.

```
52 | incompatible types: String cannot be converted to int
53 |
54 | Circundar com ...
55 | ----
56 | (Alt-Enter mostra dicas)
57 |
    | int idade = (int) valor;
```

Para resolver este problema durante a conversão, assim como na tarefa anterior, será necessário utilizar Classes Invólucro.

Realize a atribuição de idade recebe **Integer.parseInt** de valor, onde “parseInt” significa converter, logo, faremos com que o valor seja convertido, ou seja, parseado para Inteiro.

Parsear

O termo Parsear e seus derivados tem origem no verbo inglês "To Parse". Consiste em um neologismo. Utilizado por programadores e desenvolvedores da área de TI, ainda não foi elegido um termo equivalente que represente uma tradução exata em português. Os possíveis significados são: "decodificar" ; "interpretar" e dependendo do caso "converter".

Ditite , “int idade = Integer.p” e aperte a tecla Enter.

```
44 | int idade = 30; //
45 | String valor = In
46 | System.out.printl
47 | */
48 |
49 | //CONVERSÃO DE UM
50 |
51 | String valor = "3
52 |
53 | // 1º ERRO
54 | // int idade = va
55 |
56 | // 2º ERRO
57 | /* Utilizando Ty
58 | //int idade = (in
59 |
60 | //Maneira CORRETA
    | int idade = Integer.p
```

testetipos.TesteTipos > main > i

Saída - TesteTipos (run) X

run: 30

java.lang.Integer

public static int parseInt(String string) throws NumberFormatException

Parses the string argument as a signed decimal integer. The characters in the string must all be decimal digits, except that the first character may be an ASCII minus sign '-' ('\u002D') to indicate a negative value or an ASCII plus sign '+' ('\u002B') to indicate a positive value. The resulting integer value is returned, exactly as if the argument and the radix 10 were given as arguments to the [parseInt\(java.lang.String, int\)](#) method.

Parameters:

s - a String containing the int representation to be

parseInt(String string) int

parseInt(String string, int i) int

parseUnsignedInt(String string) int

parseUnsignedInt(String string, int i) int

Acrescente o “;” ao final da linha, que deverá ficar da seguinte forma: **int idade = Integer.parseInt(valor);**.

```
51      String valor = "30"; // neste caso, 30 está em aspas por se tratar de um texto, ou seja, string
52
53      // 1º ERRO
54      // int idade = valor;
55
56      // 2º ERRO
57      /* Utilizando Typecast
58      //int idade = (int) valor;
59
60      //Maneira CORRETA de se programar a conversão da string para o valor inteiro
61      int idade = Integer.parseInt(valor);
62
```

Insira a linha “System.out.println(idade);” para exibir o resultado e execute o programa.

```
51      String valor = "30"; // neste caso, 30 está em aspas por se tratar de um texto, ou seja, string
52
53      // 1º ERRO
54      // int idade = valor;
55
56      // 2º ERRO
57      /* Utilizando Typecast
58      //int idade = (int) valor;
59
60      //Maneira CORRETA de se programar a conversão da string para o valor inteiro
61      int idade = Integer.parseInt(valor);
62      System.out.println(idade);
```