Java

Operadores Unários

++	Incremento	a ++	a = a + 1

O Operador **++** faz um incremento, ou seja, soma uma unidade a variável.

++	Incremento	a ++	a = a + 1
--	Decremento	a --	a = a - 1

O Operador - - é similar ao ++, mas ao contrário do ++ o Operador **- -** decrementa uma unidade a variável.

Cuidado!!!

A posição desse incremento irá influenciar diretamente no resultado.

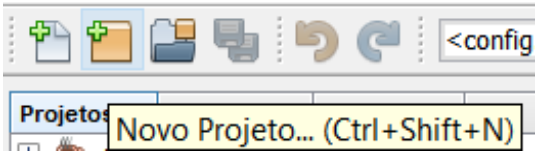
O Java utiliza o conceito de **Pré-Incremento** e **Pós-Incremento** ou **Pré-Decremento** e **Pós-Decremento**.

Exercício

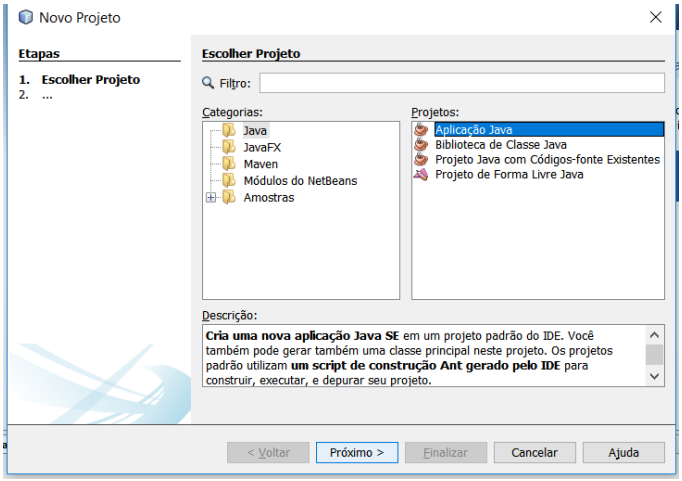
Siga os passos abaixo para realizar tarefas de Incremento e Decremento.

Abra o NetBeans.

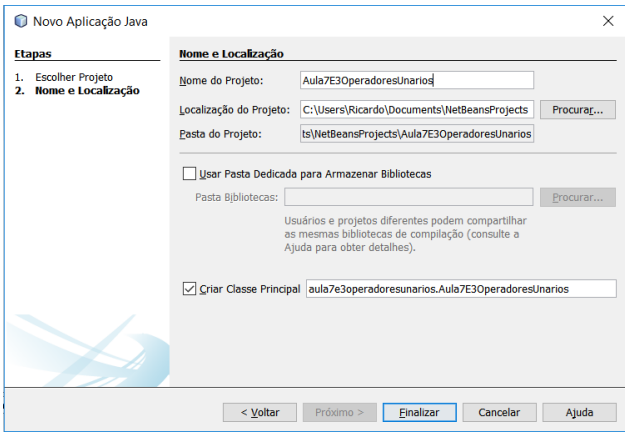
Crie um Novo Projeto.



Em Categoria mantenha Java selecionado e em Projeto, mantenha Aplicação Java e clique no botão Próximo.



Nomeie o Projeto, mantenha “Criar Classe Principal” selecionada e clique em Finalizar.



Pós-Incremento

Crie uma variável “int numero” com o valor 5 e escreva esse o valor de “numero” na tela. Após incremente uma unidade utilizando operadores unários e escreva o conteúdo atual de “numero” na tela.

```
6   package aula7e3operadoresunarios;
7
8   /**
9    *
10   * @author Ricardo
11   */
12  public class Aula7E3OperadoresUnarios {
13
14      /**
15       * @param args the command line arguments
16       */
17      public static void main(String[] args) {
18          // TODO code application logic here
19          int numero = 5;
20          System.out.println(numero);
21
22          numero++;
23          System.out.println(numero);
24      }
25
26  }
```

Execute o programa e verifique o resultado.

```
Saída - Aula7E3OperadoresUnarios (run)
run:
5
6
CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)
```

Pós-Decremento

Altere os caracteres "+" para "-". Desta forma, ficaremos com a seguinte situação "numero - - " que estará decrementando o valor de "numero" em uma unidade.

```
6   package aula7e3operadoresunarios;
7
8   /**
9    *
10   * @author Ricardo
11   */
12  public class Aula7E3OperadoresUnarios {
13
14      /**
15       * @param args the command line arguments
16       */
17      public static void main(String[] args) {
18          // TODO code application logic here
19          int numero = 5;
20          System.out.println(numero);
21
22          numero--;
23          System.out.println(numero);
24      }
25
26  }
```

2 º Exemplo

Observe o código e construa o programa.

```
42          int numero = 5;
43          System.out.println(numero);
44
45          int valor = 5 + numero++;
46
47          System.out.println(valor);
48          System.out.println(numero);
49      }
50
aula7e3operadoresunarios.Aula7E3OperadoresUnarios > main >
Saída - Aula7E3OperadoresUnarios (run)
run:
5
10
6
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
```

Comente o resultado da variável "**valor**"...

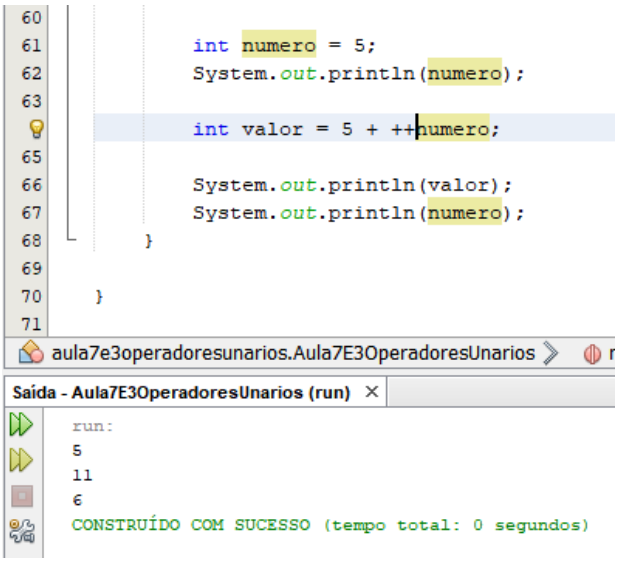
... Neste código é criada a variável "**numero**" igual a **5** e criada a variável "**valor**" igual a **5 + numero** e somente depois do cálculo já realizado que "**numero**" é **incrementada** em mais 1.

Desta forma, o resultado é 10 porque quando o cálculo 5 + numero foi realizado, a variável numero ainda NÃO havia sido incrementada... daí vem o nome de Pós-Incremento, pois a variável "numero" só foi incrementada em 1 somente depois que o cálculo já havia sido realizado.

1. **Observe que primeiramente é impresso o valor 5, que é o valor da variável numero.**
2. **Depois é impresso o valor 10 que é a soma de 5 + numero, ainda não incrementado.**
3. **Em seguida é impresso o valor 6, pois a variável "numero" só é incrementada em 1 após o cálculo realizado.**

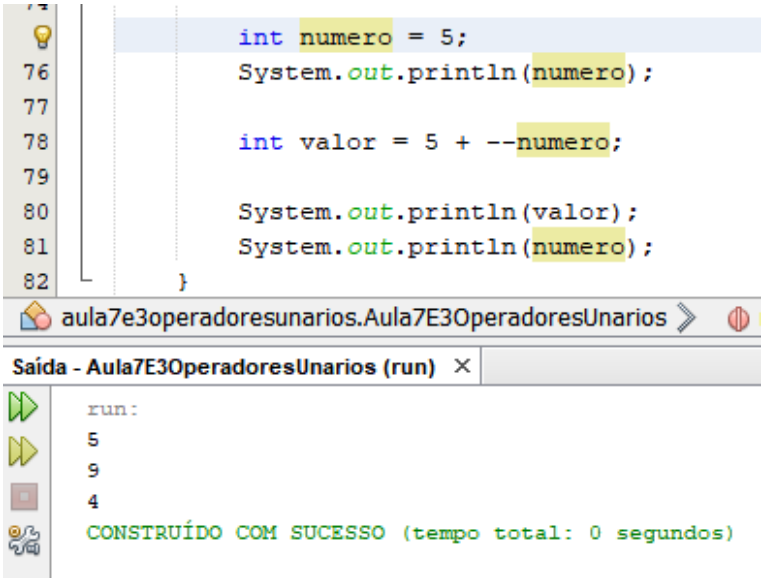
Pré-Incremento

No código anterior, altere a posição de "**numero++**" para "**++numero**". Desta forma, primeiro a variável "numero" será incrementada em 1 e somente depois será realizado o cálculo 5 + numero, retornando 11.



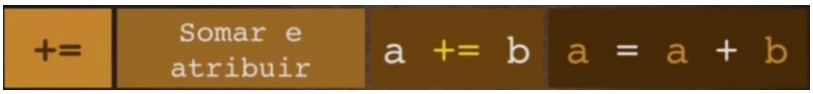
Pré-Decremento

Altere o Operador Unário ++ para --. Desta forma, o valor de numero será decrementado em 1, antes do cálculo ser efetuado.

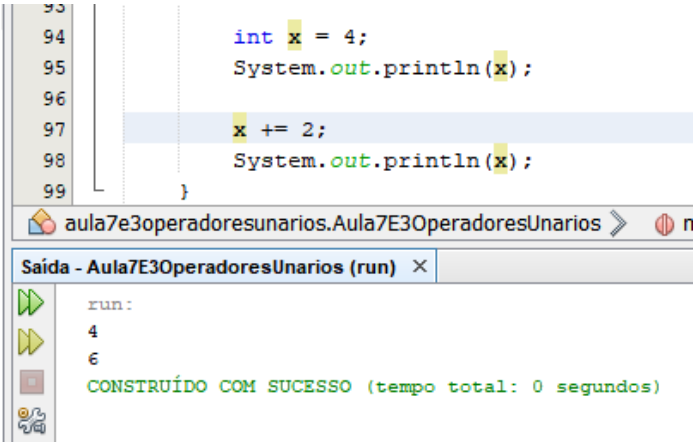


OPERADORES DE ATRIBUIÇÃO

Diferentemente dos Operadores Unários em que o valor era incrementado ou decrementado em 1, com os **Operadores de Atribuição é possível informar qual o valor do incremento**.



Neste caso, par adicionar 2 unidades a variável “x” digite “**x += 2**;”.



Da mesma forma da adição, temos para subtração, multiplicação, divisão e resto da divisão.

+=	Somar e atribuir	a += b	a = a + b
-=	Subrair e atribuir	a -= b	a = a - b
*=	Multiplicar e atribuir	a *= b	a = a * b
/=	Dividir e atribuir	a /= b	a = a / b
%=	Resto e atribuir	a %= b	a = a % b

Então, altere o símbolo de + para * para utilizarmos a multiplicação.

```
94          int x = 4;
95          System.out.println(x);
96
97          x *= 2;
98          System.out.println(x);
99      }
```

aula7e3operadoresunarios.Aula7E3OperadoresUnarios

Saída - Aula7E3OperadoresUnarios (run)

```
run:
4
8
CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
```