



JAVA & BlueJ

- Estruturas de Controle 2



Sumário

1. Laço controlado
2. Laço for
3. Somando pares
4. Juros compostos
5. Do while
6. Switch
7. Break
8. Continue
9. Operadores lógicos

1. Laço controlado

```
WhileCounter.java
1  public class WhileCounter
2  {
3      public static void main( String args[] )
4      {
5          int counter = 1;
6
7          while (counter <= 10)
8          {
9              System.out.print(counter + " ");
10             ++counter;
11         }
12
13         System.out.println();
14     }
15 }
```

BlueJ: BlueJ: Janela de Terminal - 03_Controla_II

Opções

1 2 3 4 5 6 7 8 9 10

2. Laço for

```
ForCounter.java x
1 public class ForCounter
2 {
3     public static void main( String args[] )
4     {
5         for(int counter = 1; counter <= 10; counter++)
6         {
7             System.out.print(counter + " ");
8         }
9
10        System.out.println();
11    }
12 }
```

BlueJ: BlueJ: Janela de Terminal - 03_Controla_Il

Opções

1 2 3 4 5 6 7 8 9 10

3. Somando pares

```
1  public class Sum
2  {
3      public static void main( String args[] )
4      {
5          int total = 0;
6
7          for(int number = 1; number <= 20; number += 2)
8          {
9              total += number;
10         }
11
12         System.out.printf("A soma e %d\n", total);
13     }
14 }
```

BlueJ: BlueJ: Janela de Terminal - 03_Controla_II

Opções

A soma e 100

4. Juros compostos

```
Interest.java
1 public class Interest
2 {
3     public static void main( String args[] )
4     {
5         double amount; // quantia em deposito ao fim de cada ano
6         double principal = 1000.0; // quantidade inicial antes dos juros
7         double rate = 0.05; // taxa de juros
8
9         System.out.printf( "%s%20s \n", "Ano", "Quantia em deposito" );
10
11         for ( int year = 1; year <= 10; year++ )
12         {
13             amount = principal * Math.pow( 1.0 + rate, year );
14
15             System.out.printf( "%4d%,20.2f\n", year, amount );
16         }
17     }
18 }
```

BlueJ: BlueJ: Janela de Terminal - 03_Controla_II

Opções

Ano	Quantia em deposito
1	1.050,00
2	1.102,50
3	1.157,63
4	1.215,51
5	1.276,28
6	1.340,10
7	1.407,10
8	1.477,46
9	1.551,33
10	1.628,89

5. Do while

```
DoWhileTest.java
1  public class DoWhileTest
2  {
3      public static void main( String args[] )
4      {
5          int counter = 1;
6
7          do
8          {
9              System.out.printf( "%d  ", counter );
10             ++counter;
11         } while ( counter <= 10 );
12
13         System.out.println();
14     }
15 }
```

BlueJ: BlueJ: Janela de Terminal - 03_Control_e_I

Opções

1 2 3 4 5 6 7 8 9 10

6. Switch

```
GradeBook.java
1  import java.util.Scanner;
2
3  public class GradeBook
4  {
5      private String courseName;
6      private int total;
7      private int gradeCounter;
8      private int aCount;
9      private int bCount;
10     private int cCount;
11     private int dCount;
12     private int fCount;
13
14     public void inputGrades()
15     {
16         Scanner input = new Scanner( System.in );
17
18         int grade;
19
20         System.out.printf( "%s\n%s\n  %s\n  %s\n",
21             "Digite notas inteiras entre 0-100.",
22             "Digite o fim-de-linha correspondente para terminar:",
23             "No UNIX/Linux/Mac OS X digite <ctrl> d e depois Enter",
24             "No Windows digite <ctrl> z e depois Enter" );
25
26         while (input.hasNext())
27         {
28             grade = input.nextInt();
29             total += grade;
30             ++gradeCounter;
31
32             incrementLetterGradeCounter( grade );
33         }
34     }
```


6. Switch

```
36     public void incrementLetterGradeCounter( int grade )
37     {
38         switch ( grade / 10 )
39         {
40             case 9: // nota estava entre 90
41             case 10: // e 100
42                 ++aCount; // incrementa aCount
43                 break; // necessario para sair de switch
44
45             case 8: // nota estava entre 80 e 89
46                 ++bCount; // incrementa bCount
47                 break; // sai do switch
48
49             case 7: // nota estava entre 70 e 79
50                 ++cCount; // incrementa cCount
51                 break; // sai do switch
52
53             case 6: // nota estava entre 60 e 69
54                 ++dCount; // incrementa dCount
55                 break; // sai de switch
56
57             default: // nota era menor que 60
58                 ++fCount; // incrementa fCount
59                 break; // opcional; saira de switch de qualquer jeito
60         } // fim do switch
61     }
```

6. Switch

```
63 public void displayGradeReport()
64 {
65     System.out.println( "\nRelatorio de Notas:" );
66
67     if ( gradeCounter != 0 )
68     {
69         double average = (double) total / gradeCounter;
70
71         System.out.printf( "Total das %d notas digitadas e %d\n",
72             gradeCounter, total );
73         System.out.printf( "Media da Classe e %.2f\n", average );
74         System.out.printf( "%s\n%s%d\n%s%d\n%s%d\n%s%d\n%s%d\n",
75             "Numero de estudantes que receberam cada nota:",
76             "A: ", aCount,
77             "B: ", bCount,
78             "C: ", cCount,
79             "D: ", dCount,
80             "F: ", fCount );
81     } // fim do if
82     else // nenhuma nota foi inserida, assim gera a saida da mensagem apropriada
83         System.out.println( "Nenhuma nota foi inserida" );
84 }
85 }
```

6. Switch

```
GradeBookTest.java x
1  public class GradeBookTest
2  {
3      public static void main( String args[] )
4      {
5          GradeBook myGradeBook = new GradeBook();
6
7          myGradeBook.inputGrades();
8          myGradeBook.displayGradeReport();
9      }
10 }
```

BlueJ: BlueJ: Janela de Terminal - 03_Controla_II

Opções

Digite notas inteiras entre 0-100.

Digite o fim-de-linha correspondente para terminar:

No UNIX/Linux/Mac OS X digite <ctrl> d e depois Enter

No Windows digite <ctrl> z e depois Enter

34

56

67

78

89

93

100

77

88

99

Relatorio de Notas:

Total das 10 notas digitadas e 781

Media da Classe e 78,10

Numero de estudantes que receberam cada nota:

A: 3

B: 2

C: 2

D: 1

F: 2

7. Break

```
BreakTest.java
1  public class BreakTest
2  {
3      public static void main( String args[] )
4      {
5          int count;
6
7          for ( count = 1; count <= 10; count++ )
8          {
9              if ( count == 5 )
10             break;
11
12             System.out.printf( "%d ", count );
13         }
14
15         System.out.printf( "\n0 laço termina com count = %d\n", count );
16     }
17 }
```

BlueJ: BlueJ: Janela de Terminal - 03_Controla_II

Opções

1 2 3 4

0 laço termina com count = 5

8. Continue

```
ContinueTest.java
1  public class ContinueTest
2  {
3      public static void main( String args[] )
4      {
5          for ( int count = 1; count <= 10; count++ )
6          {
7              if ( count == 5 )
8                  continue;
9
10             System.out.printf( "%d ", count );
11         }
12
13         System.out.println( "\n0 continue e usado para pular o 5" );
14     }
15 }
```

BlueJ: BlueJ: Janela de Terminal - 03_Controla_II

Opções

1 2 3 4 6 7 8 9 10

0 continue e usado para pular o 5

9. Operadores lógicos

```
LogicalOperators.java x
1 public class LogicalOperators
2 {
3     public static void main( String args[] )
4     {
5         // cria a tabela-verdade para o operador && (E condicional)
6         System.out.printf( "%s\n%s: %b\n%s: %b\n%s: %b\n%s: %b\n\n",
7             "Conditional AND (&&)", "false && false", ( false && false ),
8             "false && true", ( false && true ),
9             "true && false", ( true && false ),
10            "true && true", ( true && true ));
11
12        // cria a tabela-verdade para o operador || (OU condicional)
13        System.out.printf( "%s\n%s: %b\n%s: %b\n%s: %b\n%s: %b\n\n",
14            "Conditional OR (||)", "false || false", ( false || false ),
15            "false || true", ( false || true ),
16            "true || false", ( true || false ),
17            "true || true", ( true || true ));
18
19        // cria a tabela-verdade para o operador & (E lógico booleano)
20        System.out.printf( "%s\n%s: %b\n%s: %b\n%s: %b\n%s: %b\n\n",
21            "Boolean logical AND (&)", "false & false", ( false & false ),
22            "false & true", ( false & true ),
23            "true & false", ( true & false ),
24            "true & true", ( true & true ));
```

9. Operadores lógicos

```
26 // cria a tabela-verdade para | (OU inclusivo lógico booleano)
27 System.out.printf( "%s\n%s: %b\n%s: %b\n%s: %b\n%s: %b\n\n",
28     "Boolean logical inclusive OR (|)",
29     "false | false", ( false | false ),
30     "false | true", ( false | true ),
31     "true | false", ( true | false ),
32     "true | true", ( true | true ));
33
34 // cria a tabela-verdade para ^ (OU exclusivo lógico booleano)
35 System.out.printf( "%s\n%s: %b\n%s: %b\n%s: %b\n%s: %b\n\n",
36     "Boolean logical exclusive OR (^)",
37     "false ^ false", ( false ^ false ),
38     "false ^ true", ( false ^ true ),
39     "true ^ false", ( true ^ false ),
40     "true ^ true", ( true ^ true ));
41
42 // cria a tabela-verdade para ! (negação lógica)
43 System.out.printf( "%s\n%s: %b\n%s: %b\n", "Logical NOT (!)",
44     "!false", ( !false ), "!true", ( !true ) );
45 }
46 }
```

BlueJ: BlueJ: Janela de Terminal - 03_Controla_II

Opções

Conditional AND (&)
false & false: false
false & true: false
true & false: false
true & true: true

Conditional OR (||)
false || false: false
false || true: true
true || false: true
true || true: true

Boolean logical AND (&)
false & false: false
false & true: false
true & false: false
true & true: true

Boolean logical inclusive OR (|)
false | false: false
false | true: true
true | false: true
true | true: true

Boolean logical exclusive OR (^)
false ^ false: false
false ^ true: true
true ^ false: true
true ^ true: false