

Comando	Portugol (Visualg)	Java
Variável tipo inteiro	i: inteiro	short i; int i; long i; byte i;
Variável tipo real	d: real	float d; double d;
Variável tipo lógico	b: logico	boolean b;
Variável tipo texto	s: caractere	char s; String s;
Variável tipo vetor	v: vetor[<i>linha1..linhaN</i>] de <i>tipo</i>	<i>tipo</i> [] v = new <i>tipo</i> [<i>linhas</i>];
Variável tipo matriz	m: vetor[<i>linha1..linhaN, coluna1..colunaN</i>] de <i>tipo</i>	<i>tipo</i> [][] m = new <i>tipo</i> [<i>linhas</i>][<i>colunas</i>];
Exemplos de literais (valores) do tipo inteiro	3	3 // int 3L // long
Exemplos de literais (valores) do tipo real	3.14	3.14F // float 3.14 // double
Exemplos de literais (valores) do tipo lógico	FALSO VERDADEIRO	false true
Exemplos de literais (valores) do tipo texto	"texto entre aspas"	'C' // char "texto entre aspas" // String

Comando	Portugol (Visualg)	Java
Ler valor tipo inteiro	leia(i)	i = Short.parseShort(console.nextLine()); // short i = Integer.parseInt(console.nextLine()); // int i = Long.parseLong(console.nextLine()); // long i = Byte.parseByte(console.nextLine()); // byte
Ler valor tipo real	leia(d)	d = Float.parseFloat(console.nextLine()); // float d = Double.parseDouble(console.nextLine()); // double
Ler valor tipo lógico	leia(b)	b = Boolean.parseBoolean(console.nextLine());
Ler valor tipo texto	leia(s)	s = console.nextLine().charAt(0); // char s = console.nextLine(); // String
Escrever	escreva(...)	System.out.print(...);
Escrever, pulando linha	escreval(...)	System.out.println(...);
Escrever com parâmetros	escreva(..., a, ..., b, ...)	System.out.print("..." + a + "..." + b + "...");
Escrever com parâmetros, pulando linha	escreval(..., a, ..., b, ...)	System.out.println("..." + a + "..." + b + "...");
Limpar tela	limpatela	(não existe)
Operação recebe	a <- b	a = b;
Operação soma	a + b	a + b
Operação subtração	a - b	a - b
Operação divisão	a / b	a / b
Operação divisão inteira	a \ b	a / b (sendo a e b inteiros)
Operação resto da divisão	a % b a mod b	a % b
Operação potenciação	a ^ b	Math.pow(a, b)
Operação raiz quadrada	RaizQ(a)	Math.sqrt(a)
Operação concatenação	"texto" + "texto" "texto" + a a + "texto"	"texto" + "texto" "texto" + a a + "texto"
Operação acessar valor do vetor	v[linha]	v[linha] (índice começa em 0 e termina em tamanho-1)
Operação acessar valor da matriz	m[linha, coluna]	m[linha][coluna] (índices começam em 0 e terminam em tamanho-1)
Operação acessar o tamanho do vetor	(não existe)	v.length

Ver observação no final da tabela

Comando	Portugol (Visualg)	Java
Operação menor	$a < b$	$a < b$
Operação menor ou igual	$a \leq b$	$a \leq b$
Operação maior	$a > b$	$a > b$
Operação maior ou igual	$a \geq b$	$a \geq b$
Operação igual	$a = b$	$a == b$
Operação diferente	$a \neq b$	$a != b$
Operação e	$a \text{ e } b$	$a \&\& b$
Operação ou	$a \text{ ou } b$	$a \ \ b$
Operação ou-exclusivo	$a \text{ xou } b$	(não existe)
Operação negação	não a	!a
Operação bit-a-bit e	(não existe)	$a \& b$
Operação bit-a-bit ou	(não existe)	$a \ \ b$
Operação bit-a-bit ou-exclusivo	(não existe)	$a \wedge b$
Operação bit-a-bit negação	(não existe)	$\sim a$
Operação de incremento	(não existe operador específico, mas pode ser feita com $a \leftarrow a + 1$)	$a++$ $++a$
Operação de decremento	(não existe operador específico, mas pode ser feita com $a \leftarrow a - 1$)	$a--$ $--a$
Operação com atribuição	(não existe)	$a \text{ op } = b$ (é o mesmo que $a = a \text{ op } b$)

Comando	Portugol (Visualg)	Java
Comando se-então	se <i>condição</i> entao ... fimse	if (<i>condição</i>) { ... }
Comando se-então-senão	se <i>condição</i> entao ... senao ... fimse	if (<i>condição</i>) { ... } else { ... }
Comando se-então-senão encadeados	se <i>condição1</i> entao ... senao se <i>condição2</i> entao ... senao ... fimse fimse	if (<i>condição1</i>) { ... } else { if (<i>condição2</i>) { ... } else { ... } }
Comando escolha-caso	escolha <i>variável</i> caso <i>valor1</i> ... caso <i>valor2, valor3</i> ... outrocaso ... fimescolha	switch (<i>variável</i>) { case <i>valor1</i> : ... break; case <i>valor2</i> : case <i>valor3</i> : ... break; default: ... break; }
Comando enquanto-faça	enquanto <i>condição</i> entao ... fimenquanto	while (<i>condição</i>) { ... }
Comando para	para <i>variável</i> de <i>valor1</i> ate <i>valorN</i> passo <i>P</i> faca ... fimpara	for (<i>variável</i> = <i>valor1</i> ; <i>variável</i> <= <i>valorN</i> ; <i>variável</i> += <i>P</i>) { ... }
Comando repita-até	repita ... ate <i>condição</i>	(não existe)
Comando faça-enquanto	(não existe)	do { ... } while (<i>condição</i>);
Comando de interrupção	interrompa	break;
Comando de próxima iteração	(não existe)	continue;

Comando	Portugol (Visualg)	Java
Criar procedimento	procedimento <i>nome(parâmetro: tipo; ...)</i> var ... inicio ... fimprocedimento	static void <i>nome(tipo parâmetro, ...)</i> { ... }
Criar função	funcao <i>nome(parâmetro: tipo; ...): tipo</i> var ... inicio ... retorne <i>valor</i> fimfuncao	static <i>tipo nome(tipo parâmetro, ...)</i> { ... return <i>valor</i> ; }
Chamar procedimento	<i>nome(valor, ...)</i>	<i>nome(valor, ...)</i> ;
Chamar função	<i>a <- nome(valor, ...)</i>	<i>a = nome(valor, ...)</i> ;

Observação: para usar o objeto "console", devemos, antes, criá-lo:

```
import java.util.Scanner;

public class SuaClasse {

    public static void main(String[] args) {
        Scanner console = new Scanner(System.in);
        // a partir daqui, pode-se fazer leitura de dados usando o objeto console.

    }

}
```

Você também pode usar a interface gráfica para escrever e ler dados:

```
import javax.swing.JOptionPane;

public class SuaClasse {

    public static void main(String[] args) {
        // para mostrar algo pela interface gráfica:
        JOptionPane.showMessageDialog(null, "Mensagem a ser mostrada");

        // para pedir algo pela interface gráfica (a mensagem a ser mostrada é usada para indicar para o usuário o que você está pedindo para ele digitar):
        String linha = JOptionPane.showInputDialog("Mensagem a ser mostrada");

    }

}
```