Literais

- São palavras colocadas literalmente no programa.
  - Números literais
  - Caracteres literais
  - Strings literais
    - string=corda (de instrumentos musicais), corrente (acessório)
    - String, em programação: cadeia de caracteres

Números

- Inteiros
  - 32 bits
    - -2,147,483,648 a 2,147,483,647 (inclusive)
  - 64 bits (inteiros longos)
    - -9,223,372,036,854,775,808 a 9,223,372,036,854,775,807 (inclusive).
- Ponto flutuante
  - 32 bits (float)
    - 1.4012984632481707e-45 to 3.4028234663852860e+38 (positive or negative).
  - 64 bits (double)
    - 4.94065645841246544e-324d to 1.79769313486231570e+308d (positive or negative).

Usando o ola mundo

- System.out.println (“Hello owwww”);
- System.out.println (123456);
- System.out.println (1234.56);
- System.out.println (1234.56e-3);
- System.out.println (1234.56e2);
-

Operações aritiméticas

- + adição (também concatenação de Strings)
- - subtração
- * multiplicação
- / divisão
- % resto da divisão (ou mod)

Tipo numérico default

- Em Java, tem que se saber o tipo de todos os literais.
- Como decidir se 124 é inteiro, long, float ou double?
- É do tipo default a menos de informação em contrário.
- O default é inteiro. Caso tenha ponto (12.33; 12.0) é double
- Ao final do numero pode ter {d,D} para indicar double ou {l,L} para indicar long.
- Esse tipo vale até nas contas. Por isso em algumas contas da lista, o número é truncado.

Operações relacionais

- == igual
- != diferente
- > maior
- >= maior igual
- < menor
- <= menor igual
- && E relacional
- || OU relacional

Resultado de uma operação relacional

- Operações relacionais resultam true (verdadeiro) ou false (falso) informando se a proposição feita é ou não verdadeira.
-
- 5<7
- 5>7
- 57==75
- (99==99)&&(6>9)

Misturando operações aritméticas e relacionais

- (7-5)==(5-7)
- (31*5)==(31*3+31*2)

Precedência entre operadores

- A mesma da matemática.
- Operações relacionais são avaliadas em sequencia.
- Se você quer outra ordem ou não tem certeza que o computador vai fazer do jeito que você quer, USE PARENTESIS!!!
- (31*5)==(31*3+31*2)
- (31*5)==((31*3)+(31*2))

Caracteres e Strings

- 'c'; '5'; '\u0041'
- “c”; “5”; “\u0041”; “Hello”; “Buzz”
- Operações:
  - + em caracteres soma o valor dos caracteres;
  - + em Strings concatena as Strings.

Misturando números e Strings

- Os números são “promovidos” para Strings, ou seja, System.out.println (“Quero” + 1 + “ duzia”) vai escrever na tela “Quero 1 duzia” pois o inteiro é promovido para String e as Strings são concatenadas.
- Testar
  - System.out.println (“Quero ” + 1 +1 + “ duzia”)
  - System.out.println (“Quero ” + (1 +1) + “ duzia”)
- Diga por que não há surpresa no resultado.

Variáveis

- A ideia é a mesma da matemática: “fazer contas com letrinhas”
- X^2+5 com X=15 é igual a 230
- Variáveis tem que ter tipos
- {boolean, byte, short, int, long, float, double, char, String}

Acrescentando variáveis ao Olá mundo

- Programa para resolver a posição em cinemática: movimento uniformemente variável
- S=S0+v0*t+(a*t^2)/2
- Muv.java

Bloco de código

- Grupo de zero ou mais comandos delimitado por chaves.
- Indentação para facilitar leitura.

```
class BlockDemo {
    public static void main(String[] args) {
        boolean condition = true;
        if (condition) { // begin block 1
            System.out.println("Condition is true.");
        } // end block one
        else { // begin block 2
            System.out.println("Condition is false.");
        } // end block 2
    }
}
```

Escopo de variáveis

- Variáveis podem ser declaradas em qualquer parte de um bloco de código.
  - Existem somente dentro deste bloco
  - São visíveis por sub-blocos contidos no bloco
  - São ocultadas quando no sub-bloco se declara uma variável de mesmo nome

Comandos básicos

- Dois deles para poder pensar na lista 2
-
- If ( ).. then { } else { }
-
- While () { }
-
- Apresentar lista2
-