**JAVA**

.Inicio de um programa

public class exemplo {

public static void main(String[] args) {

.Fim do programa

} -> fechar o main

} -> fechar a class

----------------------------------------------------

.Tipos de variáveis

Inteiros

Double

String

Char

Float

Long

Short

Byte

Boolean

.Definição de variáveis

String x ; - nota tem que ser definido com letra grande

x="ola mundo";

boolean flag;

flag=false;

char character; nota tem que ser definido com a primeira letra pequena

character='x';

Byte b;

b=10;

short pequeno;

pequeno=100;

Long varivallonga;

varivallonga=1000000L;

Float factura;

factura=100.9F;

Double media;

media=5.3;

int contador; nota:tem que ser definido com a primeira letra pequena

contador=0;

Outra forma, definir variável e atribuir valor ao mesmo tempo

String x ="ola mundo"; - nota tem que ser definido com a primeira letra grande

boolean flag =false;

char character =’X’; nota tem que ser definido com a primeira letra pequena

Byte b =10;

Short pequeno=100;

int contador=0; nota: tem que ser definido com a primeira letra pequena

Long varivallonga=1000000L

Nota: tipo de dados long termina com L

Float factura=100.9F

Nota: tipo de dados float termina com F

Double media =5.3;

**NOTA : a definição das variáveis tem que ser feita com letra pequena, sendo que algumas podem ter a primeira letra grande,mas apenas a primeira letra.**

**Alguma tem que ser especificamente começadas por letra grande , e outras por letra pequena, mas apenas a primeira letra, as restantes letras são todas elas minúsculas.**

-----------------------------------------------------

.Escrever para o ecrã e receber valores do teclado

.Escrever para o ecra

System.out.println("Inserir texto aqui");

.Escrevre para o ecra a mensagem, juntamente com o valor de uma variavel

System.out.println("Inserir texto aqui"+variavel); <- escreve e troca de linha

Ou

System.out.print("Inserir texto aqui"+variavel); <- escreve sem trocar de linha

.Ler do teclado

->temos que criar um scanner de forma a ler do teclado

Scanner scanner = new Scanner(System.in);

-> para ler entao fazemos:

.receber uma string

variavelx=scanner.nextLine();

.receber um inteiro

variaveli=scanner.nextInt();

.receber um double

Variávelx= scanner.nextDouble();

NOTA: desta forma podemos apenas inserir por o valor separado por

Virgula (,)

exemplo : 5,3

Outra forma:

Variávelx= Double.parseDouble(scanner.nextLine());

NOTA: assim já aceita virugla (5,3) e ponto(5.3)

DIFERENÇA ENTRE scanner.nextDouble(); e Double.parseDouble(scanner.nextLine());

Double.parseDouble(scanner.nextLine()); -> mais rápido e aceita virgula e ponto

--------------------------------------------------------

.Operadores

==

<

<=

>

>=

!=

&& (e)

|| (ou)

! (não)

Potenciação :

media = Math.pow(a,b);

exemplo: 5^2 , o a sera 5 e o b sera 2

XOR

A operação XOR em java é definida pelo operador ^

Exemplo :

int num1 = 34; // em binário = 100010

int num2 = 50; // em binário = 110010

int resultado = num1 ^ num2; // em binário = 010000

----------------------------------------------------------

.Condições

**IF**

If (condição) {

...

...

}

--------------------

**IF ELSE**

If (condição) {

…..

}else {

….

}

-----------

**FOR**

for (int j = 0; j < 10; j++) {

.......

}

--------------------

**WHILE**

while(condição){

...

...

}

--------------------

**DO WHILE**

do{

………

}while(condição);

------------------------

**SWITCH CASE**

switch(opcao){

case 1:

...

break;

case 2:

...

break;

default:

...

}

**Definir funções em Java**

Exemplo de uma função em java:

public static void Exemplo() <- Definição da função

{

System.out.println("Ola Mundo");

} <- fecho da função

Como Chamar uma função:

Exemplo();

As funções podem também estas serem do tipo inteiro por exemplo, e receber valores.

public static int fact(int k){ <- definição da função, passando por parâmetro as variáveis

necessárias ao seu funcionamento

if (k>2) {

return k\*fact(k-1);

}else{

return k;

}

} <- fecho da função

Notas De JAVA:

* O java usa o ponto e virgula (;) para terminar uma linha de código
* As funções podem ser definidas antes ou depois no main.