****

**Introdução ao Ecossistema**

**e Documentação Java**

**- Introdução ao Ecossistema Java**

- Características da Linguagem Java

\* É uma linguagem compilada.

\* É uma linguagem interpretada.

\* Fortemente tipada.

\* É uma linguagem de alto nível e orientada a objeto.

\* É executada através de uma máquina virtual – JVM(java virtual machine).

- Algumas funções da JVM

\* Faz a execução de pilhas.

\* Responsável pelo gerenciamento de memória.

\* Gerenciamento de Threads.

\* Otimização de código (compilação JIT – Just in Time).

\* Garbage Collector (GC) (limpeza de memória – ponteiros).

- Instalando o Java no Linux

\* É necessário a instalação do curl que é uma ferramenta de conectividade e para transferência de arquivos.

**- Palavras Reservadas**

São palavras de uso interno de compilador que não podem ser utilizadas para nomear variáveis, constantes, classes ou qualquer outra declaração feita pelo desenvolvedor.

**- Declaração de Classes**

A estrutura básica na declaração de uma classe é dada por:

Modificador de acesso - palavra reservada class - nome da classe

public class NomeClasse{ atributos e métodos }

Obs: O nome da classe começa com a primeira letra em maiúsculo

Atributos:

Modificador de acesso - tipo de valor - nomeAtributo

private String valorString;

private final Double taxa;

**- Modificadores de Acesso**

public - qualquer classe de qualquer pacote pode acessar o atributo ou método.

protected – qualquer classe definida no mesmo pacote ou subclasse.

private – apenas a própria classe.

sem modificador – apenas classes definidas no mesmo pacote.

- **Métodos**

São funções que definem o comportamento de uma classe.

- Métodos Construtores

Define as regras de como uma classe deve ser instanciada.

- Métodos Comuns

Define comportamentos que podem ou não estar atribuídos as regras de negócio.

**- Estruturas de Condições**

São responsáveis por fazer o desvio do fluxo do código.

If( condição){

Comandos;

}else{

comandos;

}

Switch(variavelComValor){

case valor:

return comando;

case valor2:

return comando;

default:

return comando;

}

**- Estruturas de Repetições**

São responsáveis de realizar uma instrução ou estruturas de repetições até que uma condição seja satisfeita.

while(condição){

comandos;

}

do {

comandos;

}while(condição);

for(int i = 0; i <=x ; i++){

comandos;

}

**- Documentações Java**

- Comentários em linha:

Um comentário de linha inicia com “//”

- Comentários em bloco:

Para comentários que exigem mais de uma linha usa-se o comentário em bloco.

Se inicia com /\*\* e termina com \*\*/

- JavaDoc:

É uma ferramenta da documentação em html que se baseia nos comentários do código fonte.

Os comentários precisam conter tags para que a documentação fique legível.

Tags:

- @author – especifica o autor da classe ou método.

- @deprecated – indica classes ou métodos obsoletos.

- @link – possibilita a definição de um link para um outro documento local ou remoto através de um URL.

- @param – descreve um parâmetro que irá se passado a um método.

- @return – descreve qual o tipo de retorno de um método.

- @see – associa a outras classes ou métodos.

- @since – descreve desde quanto uma classe ou método foi adicionado.

- @throws – descreve os tipos de exceções podem ser lançados por um método.

- @version – descreve a versão da classe ou método.

Para gerar o documento HTML da documentação usa-se o comando:

javadoc -d diretórioDeSaida -link scr/