

## **JAVA - AVANÇADO**

### **Conteúdo Programático:**

#### **Classes**

Classes internas; Instância da classe interna; Referências na classe interna; Modificadores de membros; Classes internas locais; Classes internas anônimas; Classes aninhadas; Tipo Enumerado; Scanner; printf; Classe Formatter; Static Import; Retorno de tipo covariant; Métodos Get e Set; Acesso aos atributos da classe via Getters e Setters.

#### **Array**

Declaração de arrays; Arrays unidimensionais; Arrays bidimensionais; Arrays multidimensionais; Acesso aos elementos do vetor; Acesso aos elementos de um loop; Atalho para iniciar e construir um array; Array anônimo; Passando um array como parâmetro; Criando array com tipos construídos; Atribuição aos elementos do array; Variáveis de referência - arrays unidimensionais; Variáveis de referência - arrays multidimensionais; Variáveis de instância e variáveis locais; Variáveis de instância de referência a objetos; Variáveis de instância de array; Tipos primitivos locais; Objetos locais; Arrays locais; Array de String; Generics; Imprimindo elementos do array; Varargs.

#### **Os métodos equals() e hashCode()**

Equivalência de variáveis; Equivalência de objetos; As regras de equals(); Hashing; As regras de hashCode().

#### **Conjuntos**

Principais operações de conjuntos; Principais interfaces de conjuntos; Implementações de interfaces; Ordenação e classificação; Conjunto Set; Class HashSet; Class Iterator; Interface Set; Interface Comparable; Interface Comparator; Conjunto Map; Class HashMap; Class Hashtable; Conjunto List; Java.util.List; Java.util.ArrayList; Framework.

#### **Garbage Collector**

O que é o garbage collector; O funcionamento do garbage collector; A execução do garbage collector; O método finalize(); Preparando o objeto para a coleta; Remover uma referência; Alterar uma referência; Isolar uma referência; A classe Runtime; Desabilitando o garbage collector via interpretador Java; Ciclo de vida dos objetos; Created; In use; Invisible; Unreachable; Collected; Finalized; Deallocated; Objetos de referência; WeakReference, SoftReference e PhantomReference.

#### **Exceções**

Exceções com try e catch; Throws; Finally; Exceções transferidas; Hierarquia e correspondência; Manipulador de exceção; Interface pública; Classe Throwable; Construtores; Métodos; Classe Exception; Construtores; Classe

ExceptionList; Métodos; Classe Error; Construtores; Classe SQLException; Construtores; Métodos; Exceções personalizadas.

### **Assertões**

Implementação; Sintaxe das assertões; Compilando e executando com assertões; Ativando e desativando assertões; Switches de linha de comando; Regras para o uso de assertões; Classe AssertionError; Construtores.

### **Threads**

Multithreaded; Implementação; java.lang.Thread; java.lang.Runnable; Construtores; Estados da thread; Scheduler; Prioridades; Método yield(); Método join(); Método isAlive(); Método sleep(); Sincronização; Synchronized; Race condition; Bloco sincronizado; Bloqueios; Deadlock; Interação entre threads.

### **JAR**

Características de JAR; Manifest.MF; Como criar um arquivo JAR; O atributo Main-Class; Executar um arquivo JAR; Extrair arquivos de um JAR; Classpath.

### **Stream - Fluxo I/O**

Classe OutputStream; Métodos; Classe InputStream; Métodos; Leitura de arquivos binários; Classe FileInputStream; Leitura de arquivos de texto; Classe FileReader; Classe BufferedReader; RandomAccessFile; Path.

### **JDBC**

Pacote java.sql; Carregando drivers; Interface Connection; Classe DriverManager; Executando uma conexão; Método Close; Operações básicas com comandos SQL.