

Java Avançado

Pré-requisito: Para que os alunos possam aproveitar este treinamento ao máximo, é importante que eles tenham participado dos treinamentos de Java Programmer Treinar, ou possuam conhecimentos equivalentes.

Carga Horária: 32 horas

Conteúdo Programático:

Classes

Classes internas;

Instância da classe interna;

Referências na classe interna;

Modificadores de membros;

Classes internas locais;

Classes internas anônimas;

Classes aninhadas;

Métodos Get e Set;

Método get;

Acesso aos atributos da classe via Getters e Setters.

Array

Declaração de arrays;

Acesso aos elementos do vetor;

Acesso aos elementos em um loop;

Atalho para iniciar e construir um array;

Array anônimo;

Array unidimensionais;

Array bidimensionais;

Array multidimensionais;

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 globais;

Variáveis de instância instâncias de referência a objetos;

Variáveis de instância de array;

Tipos primitivos locais;

Para que os alunos possam aproveitar este treinamento ao máximo, é importante que eles tenham participado dos treinamentos de Java Programmer - Módulo 1, Lógica de Programação Visual e UML da os equivalentes.

Acesso aos atributos da classe via Getters e Setters.

Atalho para iniciar e construir um array;

Passando um array como parâmetro;

Criando array com tipos construídos;

arrays unidimensionais;

arrays multidimensionais;

iaáveis de instância e variáveis globais;

Variáveis de instância instâncias de referência a objetos;

Para que os alunos possam aproveitar este treinamento ao máximo, é importante que eles tenham

Módulo 1, Lógica de Programação Visual e UML da

Objetos locais;

Arrays locais;

Array de String.

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;

Implementação de interfaces;

Ordenação e Classificação;

Conjunto Set;

Conjunto Map; Class HashMap;

Class Hashtable;

Conjunto List;

Framework.

Garbage Collector

O que é 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 a Coleta de Lixo 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;
Capturar exceções;
Class Throwable;
Class Exception;
Class ExceptionList;
Class Error;
public class Error;
extends Throwable;
Class SQLException;
Class SQLException;

SQLException;

Exceções personalizadas.

Asserções

Conceito;

Implementação;

Sintaxe das asserções;

Compilando e executando com asserções;

Ativando e desativando asserções; Switches de linha de comando;

Regras para o uso de asserções;

Class AssertionError;

Construtores

JAR

Características de JAR;

Manifest.MF;

Etapas para criar um arquivo JAR;

O atributo Main-class;

Executar um arquivo JAR;

Extrair arquivos de um JAR;

Classpath.

Stream - Fluxo I/O

Class OutputStream;

Class InputStream;

Leitura de arquivos binários;

class FileInputStream;

Leitura de arquivos texto;

Class FileReader;

Class BufferedReader;

Path.

JDBC

Pacote java.sql;

Interface Connection;

Class DriverManager;

Método Close;

Operações Básicas com comandos SQL