

Oracle 10g: Programação em Java

Duração: 5 Dias

Objetivos do Curso

Este curso ensina os conceitos e os elementos essenciais da linguagem Java. Ao aprender Java, você criará aplicações usando tecnologias Java padrão. O ambiente de desenvolvimento usado para esse curso é o Oracle JDeveloper 10g. Além de codificar em Java, você aprenderá a usar os recursos de desenvolvimento gráficos do JDeveloper para auxiliar o desenvolvimento da aplicação.

Público

Desenvolvedor em Java
J2EE Developer
Java Developer

Pré-requisitos

Pré-requisitos Recomendados

Experience using a structured 3GL, such as C or Pascal
Basic knowledge of Object Oriented principles

Objetivos do Curso

Acessar registros de bancos de dados usando o JDBC
Criar, gerar e testar componentes de aplicações usando o Oracle
Gravar aplicações independentes com a linguagem de programação Java

Tópicos do Curso

Apresentando as Plataformas Java e Oracle

O que é Java?
Benefícios do ambiente de Runtime e Programação Java
Abordagem orientada a objetos
Disponibilização de arquitetura neutra
Computação para Java e Internet
A JVM (Java Virtual Machine)
Oracle10g, OracleAS10g e Java

Definindo Princípios Orientados a Objetos

Visão geral de classes, objetos e métodos
Comunicação entre objetos
Descrever Abstração, Herança, Encapsulamento, Agregação e Polimorfismo
Encapsulamento e palavras-chave públicas e privadas
Comparar e Contrastar membros de Subclasses e de Classe
Examinar o modelo de classe da aplicação do curso

Sintaxe Básica para Java e Convenções de Códigos

Palavras-chave de Java
Estrutura de arquivo Java (package, importação, classe)

Especificando Métodos

Instruções

Blocos de Código

Compilador Java

Explorando Operadores e Tipos de Dados Principais

Tipos de dados principais

Declarando e inicializando variáveis

Regras e convenções para nomeação de variável

Literais de caracteres e seqüências de escape Unicode

Variáveis Booleana e Primitiva

Operadores de atribuição

Operadores de atribuição compostos

Literais do objeto string e o Operador de Concatenação

Controlando o Fluxo do Programa

Decisões: se, alternância operador condicional

Repetição: while, do-while, for

Interromper e continuar

Enumeradores e iteradores

Interromper e continuar identificado

Criando Java Usando o Oracle JDeveloper

Componentes do JDeveloper

Espaços de trabalho e usando o Project Wizard

Criando uma Aplicação

Criando e executando a aplicação

Navegando no painel de Estrutura

Destaques da sintaxe e estabelecimento de vinculação de parâmetros

Modificando as opções de ambiente

Criando Classes e Objetos

Classes

Definindo variáveis e métodos de instância

Criando objetos usando o novo Construtor e o Construtor default

Variáveis primitivas versus referências a objeto

Packages

Encapsulamento usando modificadores de acesso público e privado

Exemplos de métodos de classe na Biblioteca Java

Carga de Classes, Ciclo de Vida de Objeto e Classes Internas

Variáveis e métodos de instância: uma verificação

Sobrecarga de método

Inicializando variáveis de instância usando inicializadores

Inicializando variáveis de instância usando construtores

Construtores sobrecarregados

Definindo Classes Internas e Anônimas

O método finalizar

Usando Strings, StringBuffer, Encapsulador e formatando Classes

A classe String Java

Conversão e comparação de string

Visão geral das classes encapsuladoras

Conversões de e para tipos de primitiva usando as classes encapsuladoras

Classes DataFormat, DecimalFormat e MessageFormat

OutputStreams e PrintStreams Padrão

Streams de Leitura e Gravação

Serialização de Objeto

Reutilizando o Código com Herança e Polimorfismo

Subclasse e herança: os conceitos

Hierarquias de herança

Construtores e herança

Sobrepondo métodos de superclasse

Chamando métodos de superclasse

Usando o operador instanceof

Fusão de classe

Variáveis e métodos protegidos

Usando Arrays e Conjuntos

Construir um array de primitiva

Construir um array de referências a objetos

Processar argumentos da linha de comando

Controlar os parâmetros da linha de comandos em uma aplicação Java

Usando Vetores

HashTables e Propriedades

Lendo linha de comandos e sistema, propriedades

Sincronizando manualmente ArrayLists e HashMaps

Estruturando o Código Usando Classes e Interfaces Abstratas

Classes abstratas: os conceitos e a sintaxe

Métodos abstratos

Definindo e implementando interfaces

Polimorfismo com classes abstratas

Usando interfaces para evitar várias heranças

Polimorfismo com interfaces

Gerando e Detectando Exceções

O que é uma exceção?

Classes açãoáveis

Detectando exceções usando a tentativa e a captura

Garantindo que o código é executado usando um bloco final

Exceções declaradas

Design da Interface do Usuário: Planejando um Layout de Form

Breve Histórico e comparação entre AWT e Swing

Containers Swing

Container Hierarquia para controlar grupos e layout de componente

Adicionando componentes JButton e JTextField em um container

Criando um framework GUI manualmente (exercício sem o JDeveloper)

Usando o JDeveloper para criar o framework GUI

A barra de ferramentas do container Swing

Adicionando Componentes da Interface do Usuário e Tratamento de Eventos

Componentes Swing padrão
Interface Plugável e conceitos básicos do UIManager
O modelo de eventos Java
Adicionando os handlers de eventos usando o Oracle JDeveloper
Princípios do MVC (Model View Controller) usando um componente List
Usando o JOptionPane para obter mensagens de erro e informativas

Acessando o Banco de Dados com o JDBC

Java no Banco de Dados, o OracleJVM
Etapas para usar o JDBC a fim de executar uma instrução SQL
Registrando o driver
Estabelecendo conexão com um banco de dados
Executando uma instrução SQL
Tratando exceções
Gerenciando transações

Disponibilizando Aplicações Usando o Java Web Start

Visão geral da arquitetura
Criar um perfil do Web Start
Integrar componente de negócios
Disponibilizar archive
JNLP (Java Network Launching Protocol)
Java Web Start com JClient
Arquivos adicionados a projetos para Web Start