



# Java WebDeveloper - BRQ

Conteúdo: Java Orientado a Objetos  
Data para entrega: 23/05/15

Lista  
**2**

## Teoria:

### 1) Defina o padrão DAO

DAO – Data Access Object, é um padrão de persistência de dados que permite separar regras de negocio das regras de acesso a banco de dados. Em uma aplicação que utilize arquitetura MVC, todas as funcionalidades de bancos de dados, tais como obter conexões, mapear objetos Java para tipos de dados SQL ou executar comandos SQL, devem ser feitas por classes DAO.

### 2) Explique os seguintes componentes da API JDBC:

#### a. Connection

Connection é utilizado para realizar a conexão com o banco de dados.

#### b. PreparedStatement

PreparedStatement é utilizado para executar comandos SQL.

#### c. CallableStatement

CallableStatement é utilizado para executar storedprocedures. Storedprocedures são um conjunto de comandos SQL que podem ser executados de uma só vez, como em uma função. Ele armazena tarefas repetitivas e aceita parâmetros de entrada para a tarefa seja efetuada de acordo com a necessidade individual

#### d. ResultSet

ResultSet é utilizado para ler resultados de consultas realizado no banco de dados.

### 3) O que são Classes abstratas?

As classes abstratas são as que não permitem realizar qualquer tipo de instancia. São classes feitas especialmente para serem modelos para suas classes derivadas. As classes derivadas, via de regra, deverão sobrescrever os métodos para realizar a implementação dos metos. As classes derivadas das classes abstratas são conhecidas como classes concretas.

As classes abstratas tem a função semelhante a Interfaces, com a principal diferença em que do contrario das Interfaces, nas classes abstratas podemos criar métodos com corpo, enquanto nas interfaces deve-se colocar apenas chamadas de métodos.

### 4) Explique a importância do Javadoc para projetos Java.

Javadoc é um gerador de documentação criado pela Sun Microsystems para documentar a API dos programas em Java, a partir do código-fonte. O resultado é expresso em HTML. É constituído, basicamente por algumas marcações muito simples inseridas nos comentários do programa , utilizando a notação "/\* \*/" , A importância para o Javadoc, é descrever classes, métodos, suas funções e o funcionamento para seu Programa Java.



# Java WebDeveloper - BRQ

Conteúdo: Java Orientado a Objetos  
Data para entrega: 23/05/15

Lista  
**2**

## 5) O que são arquivos .jar?

Arquivos JAR ou (Java ARchive) é um arquivo compactado usado para distribuir um conjunto de classes Java, outros itens como imagens, XMLS, entre outros. É usado para armazenar classes compiladas e metadados associados que podem constituir um programa. O arquivo Jar é parecido com arquivos como RAR ou ZIP, mas voltado para arquivos Java.

## Prática:

1. Crie uma tabela em banco de dados denominada: Aluno e contendo os atributos: idaluno, nome e cpf

Arquivo lista02\_exercicios.

src/br.com.hiro.sql

script.sql

2. Crie uma Classe de entidade Java que modele na forma de objeto a entidade Aluno e contendo:
  - a. Atributos privados
  - b. Sobrecarga de Construtores
  - c. Encapsulamento
  - d. Sobrescrita dos métodos de Object:
    - i. toString
    - ii. equals
    - iii. hashCode
  - e. Implementação de Comparable

Arquivo lista02\_exercicios.

Src/br.com.hiro.entities

Aluno.java

3. Crie uma Classe Dao que faça conexão a uma base de dados do MySql

Arquivo lista02\_exercicios.

Src/br.com.hiro.persistence

DAO.java

OBS: no DAO, o password esta "rhtrht" , devido que já possuía o Mysql instalado na maquina.

4. Crie uma Classe AlunoDao com as operações: insert, update, delete, findAll e findById

Arquivo lista02\_exercicios.

Src/br.com.hiro.persistence

DAOAluno.java

5. Crie uma classe de controle contendo um menu que permita ao usuário cadastrar, atualizar, excluir ou listar todos os clientes

Arquivo lista02\_exercicios.

Src/br.com.hiro.input -- InputAluno.java

Src/br.com.hiro.controls – ControleAluno.java

Src/br.com.hiro.controls.types – Menu.java

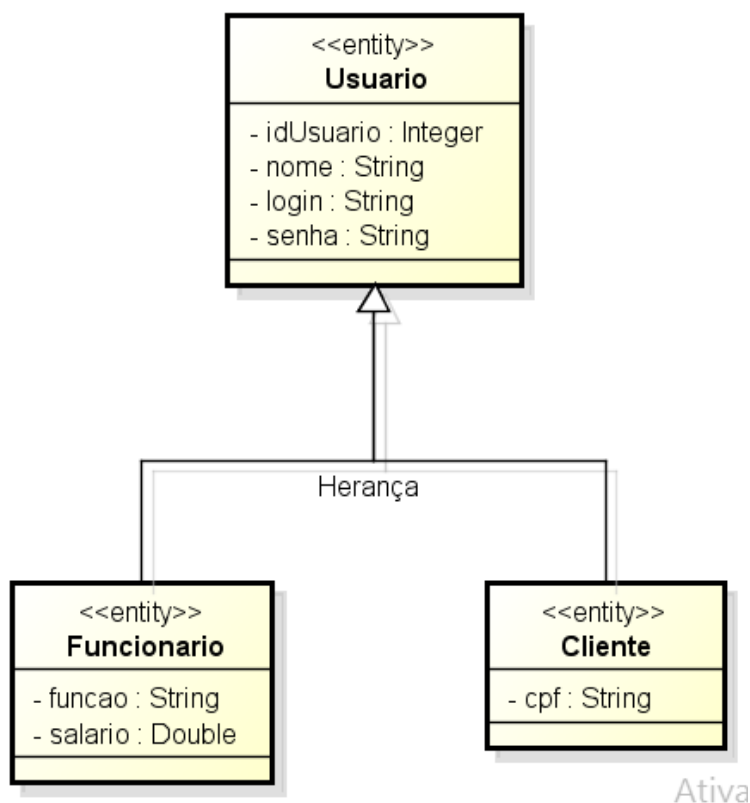
6. Demonstre na Main a execução

Arquivo lista02\_exercicios.

Src/br.com.hiro.principal – Main.java

## Prática:

7. Crie uma modelagem de entidades conforme abaixo:



- Crie uma classe abstrata **ControleUsuario<T extends Usuario>** contendo dois metodos:

### **exportarXml**

void | recebe uma lista de <T>

### **importarXml**

retorna List<T> | não recebe argumentos

- Crie 2 classes: ControleCliente e ControleFuncionario herdando ControleUsuario e implemente os métodos abstratos.
- Demonstre na Main a execução

Arquivo lista02exerciciospt02

Src/br.com.hiro.entities - Cliente.java - Funcionario.java - Usuario.java

Src/br.com.hiro.controls - ControleCliente.java - ControleUsuario.java - ControleFuncionario.java

Src/br.com.hiro.principal - MainCliente.java - MainFuncionario.java