



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

Criando o projeto:

**File > New > Dynamic Web Project**

New Dynamic Web Project

**Dynamic Web Project**

Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.

**Project name:** aula09projeto

**Project location**

Use default location

Location: C:\Users\Sergio\Desktop\work\aula09projeto

**Target runtime**

Apache Tomcat v8.0

**Dynamic web module version**

3.1

**Configuration**

Default Configuration for Apache Tomcat v8.0

A good starting point for working with Apache Tomcat v8.0 runtime. Additional facets can later be installed to add new functionality to the project.

**EAR membership**

Add project to an EAR

EAR project name: EAR

**Working sets**

Add project to working sets

Working sets:



**Criando a base de dados:  
Modelagem 1 para Muitos**

```
drop database if exists aula09;
create database aula09;
use aula09;

create table fornecedor(
    idfornecedor      integer      auto_increment,
    nome              varchar(50)  not null,
    descricao         varchar(250) not null,
    primary key(idfornecedor));

create table produto(
    idproduto        integer      auto_increment,
    nome              varchar(50)  not null,
    preco             double       not null,
    quantidade       integer      not null,
    datacadastro     date        not null,
    idfornecedor     integer      not null,
    primary key(idproduto),
    foreign key(idfornecedor) references fornecedor(idfornecedor));

--view para obter dados de produtos e fornecedor
create view vwprodutofornecedor
as
select
    p.idproduto,
    p.nome as nomeproduto,
    p.preco,
    p.quantidade,
    f.idfornecedor,
    f.nome as nomefornecedor,
    f.descricao
from produto as p
inner join fornecedor as f
on f.idfornecedor = p.idfornecedor;

desc vwprodutofornecedor;
```

---

**Criando as Classes de entidade:**

```
package br.com.brq.entities;

import java.util.Date;

public class Produto {

    private Integer idProduto;
    private String nome;
    private Double preco;
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
private Integer quantidade;
private Date dataCadastro;

// Relacionamento de Produto com 1 Fornecedor..
private Fornecedor fornecedor; // TEM 1

// construtor default..
public Produto() {
    // TODO Auto-generated constructor stub
}

public Produto(Integer idProduto, String nome, Double preco,
               Integer quantidade, Date dataCadastro) {
    super();
    this.idProduto = idProduto;
    this.nome = nome;
    this.preco = preco;
    this.quantidade = quantidade;
    this.dataCadastro = dataCadastro;
}

public Produto(Integer idProduto, String nome, Double preco,
               Integer quantidade, Date dataCadastro,
               Fornecedor fornecedor) {
    this(idProduto, nome, preco, quantidade, dataCadastro);
    this.fornecedor = fornecedor;
}

public Integer getIdProduto() {
    return idProduto;
}

public void setIdProduto(Integer idProduto) {
    this.idProduto = idProduto;
}

public String getNome() {
    return nome;
}

public void setNome(String nome) {
    this.nome = nome;
}

public Double getPreco() {
    return preco;
}

public void setPreco(Double preco) {
    this.preco = preco;
}

public Integer getQuantidade() {
    return quantidade;
}
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
public void setQuantidade(Integer quantidade) {
    this.quantidade = quantidade;
}

public Date getDataCadastro() {
    return dataCadastro;
}

public void setDataCadastro(Date dataCadastro) {
    this.dataCadastro = dataCadastro;
}

public Fornecedor getFornecedor() {
    return fornecedor;
}

public void setFornecedor(Fornecedor fornecedor) {
    this.fornecedor = fornecedor;
}

@Override
public String toString() {
    return "Produto [idProduto=" + idProduto + ", nome=" + nome
           + ", preco=" + preco + ", quantidade=" + quantidade
           + ", dataCadastro=" + dataCadastro + "]";
}
}

package br.com.brq.entities;

import java.util.Collection;

public class Fornecedor {

    private Integer idFornecedor;
    private String nome;
    private String descricao;

    // Relacionamento (Cada Fornecedor TEM Muitos Produtos)
    private Collection<Produto> produtos;

    public Fornecedor() {
        // TODO Auto-generated constructor stub
    }

    public Fornecedor(Integer idFornecedor, String nome, String descricao)
    {
        super();
        this.idFornecedor = idFornecedor;
        this.nome = nome;
        this.descricao = descricao;
    }
}
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
public Fornecedor(Integer idFornecedor, String nome, String descricao,
Collection<Produto> produtos) {
    this(idFornecedor, nome, descricao);
    this.produtos = produtos;
}

public Integer getIdFornecedor() {
    return idFornecedor;
}

public void setIdFornecedor(Integer idFornecedor) {
    this.idFornecedor = idFornecedor;
}

public String getNome() {
    return nome;
}

public void setNome(String nome) {
    this.nome = nome;
}

public String getDescricao() {
    return descricao;
}

public void setDescricao(String descricao) {
    this.descricao = descricao;
}

public Collection<Produto> getProdutos() {
    return produtos;
}

public void setProdutos(Collection<Produto> produtos) {
    this.produtos = produtos;
}

@Override
public String toString() {
    return "Fornecedor [idFornecedor=" + idFornecedor
           + ", nome=" + nome + ", descricao=" + descricao + "]";
}
}
```

---

### Criando a Classe DAO

Data Access Object

```
package br.com.brq.persistence;

import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
import java.sql.ResultSet;

public class DAO {

    private static final String DRIVER = "com.mysql.jdbc.Driver";
    private static final String URL = "jdbc:mysql://localhost:3306/aula09";
    private static final String USER = "root";
    private static final String PASSWORD = "brqbrq";

    protected Connection con;
    protected PreparedStatement stmt;
    protected CallableStatement call;
    protected ResultSet rs;

    protected void openConnection() throws Exception{
        Class.forName(DRIVER);
        con = DriverManager.getConnection(URL, USER, PASSWORD);
    }

    protected void closeConnection() throws Exception{
        if(con != null){
            con.close();
        }
    }
}
```

---

### Criando a Classe de persistencia para Fornecedor:

DAOFornecedor.java

```
package br.com.brq.persistence;

import java.util.ArrayList;
import java.util.List;

import br.com.brq.entities.Fornecedor;

public class DAOFornecedor extends DAO{

    //método para cadastrar um fornecedor na base de dados..
    public void insert(Fornecedor f) throws Exception{

        String query = "insert into fornecedor(nome, descricao) "
                      + "values(?, ?)";
        openConnection();
        stmt = con.prepareStatement(query);
        stmt.setString(1, f.getNome());
        stmt.setString(2, f.getDescricao());
        stmt.execute();
        stmt.close();

        closeConnection();
    }
}
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
//método para listar os fornecedores da base de dados..
public List<Fornecedor> findAll() throws Exception{

    String query = "select * from fornecedor";

    openConnection();
    stmt = con.prepareStatement(query);
    rs = stmt.executeQuery();

    List<Fornecedor> lista = new ArrayList<Fornecedor>();

    while(rs.next()){ //enquanto houver registros..
        Fornecedor f = new Fornecedor();
        f.setIdFornecedor(rs.getInt("idfornecedor"));
        f.setNome(rs.getString("nome"));
        f.setDescricao(rs.getString("descricao"));

        lista.add(f); //adicionar na lista..
    }

    stmt.close();
    closeConnection();

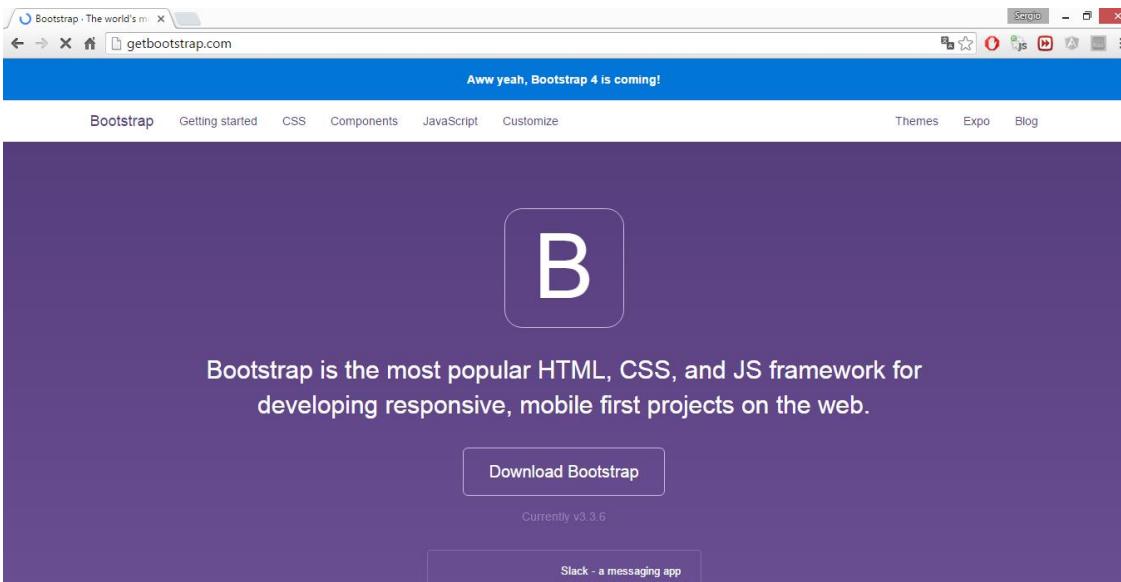
    return lista;
}
}
```

---

## Bootstrap ([www.getbootstrap.com](http://www.getbootstrap.com))

Biblioteca CSS para frontend de aplicações web:

### - Obtendo o bootstrap:



The screenshot shows the official Bootstrap website at [getbootstrap.com](http://getbootstrap.com). The header includes the Bootstrap logo and navigation links for Bootstrap, Getting started, CSS, Components, JavaScript, Customize, Themes, Expo, and Blog. A banner at the top says "Aww yeah, Bootstrap 4 is coming!". The main content features a large purple background with a white square containing a white letter 'B'. Below it, text reads "Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile first projects on the web." A "Download Bootstrap" button is present, along with a note "Currently v3.3.6". At the bottom, there's a small advertisement for Slack.



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

<http://getbootstrap.com/getting-started/#download>

The screenshot shows the 'Getting started' section of the Bootstrap website. It features a purple header with the title 'Getting started'. Below it, a sub-section titled 'Download' provides information on how to get started with Bootstrap, mentioning v3.3.6. It includes three main download options: 'Bootstrap' (compiled CSS, JS, and fonts), 'Source code' (Less, JS, and font files), and 'Sass' (Bootstrap ported from Less to Sass). Each option has a corresponding download button. To the right, there's a sidebar with links like 'Download', 'What's included', 'Compiling CSS and JavaScript', etc.

### Criando a página index.jsp com bootstrap:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
    <title>Treinamento Java BRQ</title>

    <!-- Arquivos CSS do bootstrap -->
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css"/>
    <link rel="stylesheet" type="text/css" href="css/bootstrap-
theme.css"/>

</head>
<body class="container">
    <!-- Cabeçalho da página -->
    <div class="well well-sm">
        <h2>Projeto Controle de Fornecedores e Produtos</h2>
        Treinamento BRQ COTI Informatica - Java Developer
    </div>
    <!-- Conteúdo principal da página -->
    <div class="col-md-12">

        <h3>Bem vindo ao projeto</h3>
        <hr/>
        <p>
            Seleione a operação desejada:
        </p>
        <br/><br/>
        <div class="col-md-3 panel text-center">
            <div class="glyphicon glyphicon-blackboard">
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
        style="font-size: 3em;">></div>
<br/><br/>
<a href="cadastroFornecedor.jsp"
    class="btn btn-default btn-block">
    Cadastrar Fornecedores
</a>
</div>
<div class="col-md-3 panel text-center">
    <div class="glyphicon glyphicon-tasks"
        style="font-size: 3em;">></div>
    <br/><br/>
    <a href="consultaFornecedor.jsp"
        class="btn btn-default btn-block">
        Consultar Fornecedores
    </a>
</div>
<div class="col-md-3 panel text-center">
    <div class="glyphicon glyphicon-erase"
        style="font-size: 3em;">></div>
    <br/><br/>
    <a href="cadastroProduto.jsp"
        class="btn btn-default btn-block">
        Cadastrar Produtos
    </a>
</div>
<div class="col-md-3 panel text-center">
    <div class="glyphicon glyphicon-list-alt"
        style="font-size: 3em;">></div>
    <br/><br/>
    <a href="consultaProduto.jsp"
        class="btn btn-default btn-block">
        Consultar Produtos
    </a>
</div>
</div>
</body>
</html>
```

### Resultado:

Bem vindo ao projeto

Selecione a operação desejada:

Cadastrar Fornecedores    Consultar Fornecedores    Cadastrar Produtos    Consultar Produtos



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

Criando a página de cadastro de fornecedor:  
**cadastroFornecedor.jsp**

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>Treinamento Java BRQ</title>

    <!-- Arquivos CSS do bootstrap -->
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css"/>
    <link rel="stylesheet" type="text/css" href="css/bootstrap-theme.css"/>

</head>
<body class="container">

    <!-- Cabeçalho da página -->
    <div class="well well-sm">
        <h2>Projeto Controle de Fornecedores e Produtos</h2>
        Treinamento BRQ COTI Informatica - Java Developer
    </div>

    <!-- Conteúdo principal da página -->
    <div class="col-md-12">

        <h3>Cadastro de Fornecedores</h3>
        <a href="index.jsp">Voltar</a> para a página inicial.
        <hr/>

        <div class="col-md-4">

            <form name="formulario" method="post"
                action="ControleFornecedor?acao=cadastrar">

                <label>Nome do Fornecedor:</label>
                <input type="text" name="nome" id="nome"
                    class="form-control"
                    placeholder="Digite aqui"/>
                <br/>

                <label>Descrição:</label>
                <textarea name="descricao" id="descricao"
                    class="form-control"></textarea>
                <br/>

                <input type="submit" value="Cadastrar Fornecedor"
                    class="btn btn-success"/>

                <!-- EL (Expression Language) -->
                <h4>${mensagem}</h4>
            </form>
        </div>
    </div>
</body>
</html>
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

### Criando o Servlet para cadastrar o Fornecedor: ControleFornecedor.java

```
package br.com.brq.control;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import br.com.brq.entities.Fornecedor;
import br.com.brq.persistence.DAOFornecedor;

@WebServlet("/ControleFornecedor")
public class ControleFornecedor extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public ControleFornecedor() {
        super();
    }

    //método para centralizar as requisições recebidas pelos métodos doGet e doPost
    protected void execute(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {

        //resgatar o comando enviado pela página..
        String acao = request.getParameter("acao");

        if(acao != null){ //se o valor foi resgatado..
            if(acao.equalsIgnoreCase("cadastrar")){
                try{

                    //resgatando os valores dos campos do formulário..
                    Fornecedor f = new Fornecedor(); //entidade..
                    f.setNome(request.getParameter("nome"));
                    //valor do campo nome..
                    f.setDescricao(request.getParameter("descricao"));
                    //valor do campo descricao..

                    //gravar na base de dados..
                    DAOFornecedor d = new DAOFornecedor();
                    //persistencia..
                    d.insert(f); //gravando o fornecedor..

                    //enviando mensagem para a página..
                }
            }
        }
    }
}
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
        request.setAttribute("mensagem", "Fornecedor "
                + f.getNome() + ", cadastrado com sucesso.");
    }
    catch(Exception e){
        //enviando mensagem de erro.
        request.setAttribute("mensagem",
                "Ocorreu um Erro: " + e.getMessage());
    }
    finally{ //sempre é executado..
        //redirecionar de volta para a página
        //de cadastro do fornecedor...
        request.getRequestDispatcher
                ("cadastroFornecedor.jsp")
        .forward(request, response);
    }
}
}

protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    execute(request, response); //encaminhando para o execute..
}

protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    execute(request, response); //encaminhando para o execute..
}

}
```

---

Criando a Classe de persistencia para Produto  
**DAOProduto.java**

```
package br.com.brq.persistence;

import java.util.ArrayList;
import java.util.List;

import br.com.brq.entities.Fornecedor;
import br.com.brq.entities.Produto;

public class DAOProduto extends DAO{
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
//método para inserir um produto na base de dados..
public void insert(Produto p) throws Exception{

    String query = "insert into produto(nome, preco, quantidade,
                           datacadastro, idfornecedor) "
                  + "values(?, ?, ?, now(), ?);

    openConnection();

    stmt = con.prepareStatement(query);
    stmt.setString(1, p.getNome());
    stmt.setDouble(2, p.getPreco());
    stmt.setInt(3, p.getQuantidade());
    stmt.setInt(4, p.getFornecedor().getIdFornecedor());
    stmt.execute();

    stmt.close();
    closeConnection();
}

//método para atualizar um produto..
public void update(Produto p) throws Exception{

    String query = "update produto set nome = ?, preco = ?,
                           quantidade = ?, idfornecedor = ? "
                  + "where idproduto = ?";

    openConnection();

    stmt = con.prepareStatement(query);
    stmt.setString(1, p.getNome());
    stmt.setDouble(2, p.getPreco());
    stmt.setInt(3, p.getQuantidade());
    stmt.setInt(4, p.getFornecedor().getIdFornecedor());
    stmt.setInt(5, p.getIdProduto());
    stmt.execute();

    stmt.close();
    closeConnection();
}

//método para excluir 1 produto da base de dados..
public void delete(Integer idProduto) throws Exception{

    String query = "delete from produto where idproduto = ?;

    openConnection();
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
stmt = con.prepareStatement(query);
stmt.setInt(1, idProduto);
stmt.execute();

stmt.close();
closeConnection();
}

//método para listar todos os produtos da base de dados..
public List<Produto> findAll() throws Exception{

    String query = "select * from vwprodutofornecedor";

    openConnection();
    stmt = con.prepareStatement(query);
    rs = stmt.executeQuery();

    List<Produto> lista = new ArrayList<Produto>();

    while(rs.next()){ //lendo cada registro obtido da view..
        Produto p = new Produto(); //instanciando..
        p.setFornecedor(new Fornecedor()); //instanciando..

        p.setIdProduto(rs.getInt("idproduto"));
        p.setNome(rs.getString("nomeproduto"));
        p.setPreco(rs.getDouble("preco"));
        p.setQuantidade(rs.getInt("quantidade"));
        p.getFornecedor().setIdFornecedor(rs.getInt("idfornecedor"));
        p.getFornecedor().setNome(rs.getString("nomefornecedor"));
        p.getFornecedor().setDescrição(rs.getString("descrição"));

        lista.add(p); //adicionar dentro da lista..
    }

    stmt.close();
    closeConnection();

    return lista;
}

//método para obter 1 produto pelo id...
public Produto findById(Integer idProduto) throws Exception{

    String query = "select * from vwprodutofornecedor where idproduto = ?";

    openConnection();
    stmt = con.prepareStatement(query);
```



## TREINAMENTO JAVA – BRQ/SP

### Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
stmt.setInt(1, idProduto);
rs = stmt.executeQuery();

Produto p = null; //valor nulo..
if(rs.next()){ //se algum registro foi encontrado..
    p = new Produto(); //instanciando...
    p.setFornecedor(new Fornecedor()); //instanciando..

    p.setIdProduto(rs.getInt("idproduto"));
    p.setNome(rs.getString("nomeproduto"));
    p.setPreco(rs.getDouble("preco"));
    p.setQuantidade(rs.getInt("quantidade"));
    p.getFornecedor().setIdFornecedor(rs.getInt("idfornecedor"));
    p.getFornecedor().setNome(rs.getString("nomefornecedor"));
    p.getFornecedor().setDescricao(rs.getString("descricao"));
}

stmt.close();
closeConnection();

return p; //retornando o produto..
}
}
```

---

## ManagedBeans

Classes para obtenção de dados através dos JavaBeans. Exemplo: Criação de um ManagedBean para obter a listagem de fornecedores do banco de dados e exibir na página de consulta de fornecedores:

```
package br.com.brq.managebeans;

import java.util.List;

import br.com.brq.entities.Fornecedor;
import br.com.brq.persistence.DAOFornecedor;

//classe para entregar dados de fornecedores para as páginas JSP
public class ManagedBeanFornecedor {

    //lista de fornecedores..
    private List<Fornecedor> listagemFornecedores;

    //método para retornar uma lista com os fornecedores..
    public List<Fornecedor> getListagemFornecedores() {

        try{
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
        DAOFornecedor d = new DAOFornecedor();
        listagemFornecedores = d.findAll();
    }
    catch(Exception e){
        //pilha (descrição) do erro no log do tomcat (servidor)
        e.printStackTrace();
    }

    return listagemFornecedores;
}
}
```

---

### Criando a página de consulta de fornecedores:

consultaFornecedor.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>

<!-- Referenciar o ManagedBean -->
<!-- Este comando não é um JSTL, é um comando nativo do JSP -->
<jsp:useBean class="br.com.brq.managebeans.ManagedBeanFornecedor"
id="mb"></jsp:useBean>

<!-- Declarar as bibliotecas JSTL -->
<!-- JSP Standard Tag Libraries -->
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fnc" %>

<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>Treinamento Java BRQ</title>

    <!-- Arquivos CSS do bootstrap -->
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css"/>
    <link rel="stylesheet" type="text/css" href="css/bootstrap-theme.css"/>

</head>
<body class="container">

    <!-- Cabeçalho da página -->
    <div class="well well-sm">
        <h2>Projeto Controle de Fornecedores e Produtos</h2>
        Treinamento BRQ COTI Informatica - Java Developer
    </div>

    <!-- Conteúdo principal da página -->
    <div class="col-md-12">

        <h3>Consulta de Fornecedores</h3>
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
<a href="index.jsp">Voltar</a> para a página inicial.  
<hr/>  
<table class="table table-hover">  
    <thead>  
        <tr>  
            <th>Código</th>  
            <th>Nome do Fornecedor</th>  
            <th>Descrição</th>  
        </tr>  
    </thead>  
    <tbody>  
        <!-- tag JSTL para varrer coleções de objetos -->  
        <c:forEach items="${mb.listagemFornecedores}"  
                  var="f">  
            <tr>  
                <td> ${f.idFornecedor} </td>  
                <td> ${f.nome} </td>  
                <td> ${f.descricao} </td>  
            </tr>  
        </c:forEach>  
    </tbody>  
    <tfoot>  
        <tr>  
            <td colspan="3">  
                Quantidade de registros:  
                ${fn:length(mb.listagemFornecedores)}  
            </td>  
        </tr>  
    </tfoot>  
  </table>  
</div>  
</body>  
</html>
```

### Estrutura das páginas:

- WebContent
  - ▷ css
  - ▷ fonts
  - ▷ js
  - ▷ META-INF
  - ▷ WEB-INF
    - cadastroFornecedor.jsp
    - cadastroProduto.jsp
    - consultaFornecedor.jsp
    - consultaProduto.jsp
    - detalhesProduto.jsp
    - edicaoProduto.jsp
    - index.jsp



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

### Criando um ManagedBean para obter a listagem de produtos:

ManagedBeanProduto.java

```
package br.com.brq.managebeans;

import java.util.List;

import br.com.brq.entities.Produto;
import br.com.brq.persistence.DAOProduto;

public class ManagedBeanProduto {

    //atributo..
    private List<Produto> listagemProdutos;

    //método get para retornar o valor do atributo preenchido
    //com a consulta do banco de dados..
    public List<Produto> getListagemProdutos() {

        try{
            DAOProduto d = new DAOProduto();
            listagemProdutos = d.findAll();
        }
        catch(Exception e){
            //imprimir erro no log do servidor tomcat..
            e.printStackTrace();
        }

        return listagemProdutos;
    }

}
```

---

### Criando o Servlet para controle de produtos:

ControleProduto.java

```
package br.com.brq.control;

import java.io.IOException;

import javax.management.openmbean.OpenDataException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import br.com.brq.entities.Fornecedor;
import br.com.brq.entities.Produto;
import br.com.brq.persistence.DAOProduto;
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
@WebServlet("/ControleProduto")
public class ControleProduto extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public ControleProduto() {
        super();
    }

    protected void execute(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {

        //resgatar uma variavel acao..
        String acao = request.getParameter("acao");

        if(acao != null){
            if(acao.equalsIgnoreCase("cadastrar")){
                try{
                    //resgatar os dados do produto..
                    Produto p = new Produto(); //instanciando..
                    p.setFornecedor(new Fornecedor()); //instanciando..

                    p.setNome(request.getParameter("nome"));
                    p.setPreco(Double.parseDouble(request.getParameter("preco")));
                    p.setQuantidade(Integer.parseInt
                        (request.getParameter("quantidade")));
                    p.getFornecedor().setIdFornecedor(
                        Integer.parseInt
                        (request.getParameter("fornecedor")));

                    DAOProduto d = new DAOProduto(); //persistencia..
                    d.insert(p); //gravando na base de dados..

                    //gerando mensagem de sucesso..
                    request.setAttribute("mensagem", "Produto " + p.getNome()
                        + ", cadastrado com sucesso.");
                }
                catch(Exception e){
                    //gerando mensagem de erro..
                    request.setAttribute("mensagem", "Erro: " + e.getMessage());
                }
                finally{
                    //redirecionar de volta para a página de cadastro de produto..
                    request.getRequestDispatcher("cadastroProduto.jsp")
                        .forward(request, response);
                }
            }
            else if(acao.equalsIgnoreCase("visualizar")){
                try{
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
//resgatar o id enviado pelo link..
Integer idProduto = Integer.parseInt(request.getParameter("id"));

//buscar o produto pelo id..
DAOProduto d = new DAOProduto();
Produto p = d.findById(idProduto);

//enviar um objeto no request.setAttribute...
request.setAttribute("produto", p);
}

catch(Exception e){
    //mensagem de erro..
    request.setAttribute("mensagem", "Erro: " + e.getMessage());
}
finally{
    //redirecionar para a página de detalhes do produto..
    request.getRequestDispatcher("detalhesProduto.jsp")
        .forward(request, response);
}
}

else if(acao.equalsIgnoreCase("excluir")){
try{

    //resgatando o id do produto enviado pela URL (barra de endereço)
    Integer idProduto = Integer.parseInt(request.getParameter("id"));

    DAOProduto d = new DAOProduto();
    d.delete(idProduto); //excluindo..

    //enviando mensagem de sucesso..
    request.setAttribute("mensagem", "Produto excluído com sucesso.");
}
catch(Exception e){
    //enviando mensagem de erro..
    request.setAttribute("mensagem", "Erro: " + e.getMessage());
}
finally{
    //redirecionar para a página de consulta...
    request.getRequestDispatcher("consultaProduto.jsp")
        .forward(request, response);
}
}

else if(acao.equalsIgnoreCase("editar")){
try{

    Integer idProduto = Integer.parseInt(request.getParameter("id"));

    //buscar 1 produto pelo id..
    DAOProduto d = new DAOProduto();
    Produto p = d.findById(idProduto);
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
//enviar o objeto Produto para a página de edição..
request.setAttribute("produto", p);
}
catch(Exception e){
    //enviar mensagem de erro..
    request.setAttribute("mensagem", "Erro: " + e.getMessage());
}
finally{
    //redirecionar para a página de edição..
    request.getRequestDispatcher("edicaoProduto.jsp")
        .forward(request, response);
}
}
else if(acao.equalsIgnoreCase("atualizar")){
try{

    Produto p = new Produto(); //instanciando..
    p.setFornecedor(new Fornecedor()); //instanciando..

    //resgatar os campos enviados pelo formulario..
    p.setIdProduto(Integer.parseInt
        (request.getParameter("idproduto")));
    p.setNome(request.getParameter("nome"));
    p.setPreco(Double.parseDouble(request.getParameter("preco")));
    p.setQuantidade(Integer.parseInt
        (request.getParameter("quantidade")));
    p.getFornecedor().setIdFornecedor(
        Integer.parseInt
        (request.getParameter("fornecedor")));

    DAOProduto d = new DAOProduto();
    d.update(p); //atualizando o produto..

    //enviar mensagem de sucesso para a página..
    request.setAttribute("mensagem", "Produto " + p.getNome()
        + ", atualizado com sucesso.");
}
catch(Exception e){
    //enviar mensagem de erro..
    request.setAttribute("mensagem", "Erro: " + e.getMessage());
}
finally{
    //redirecionar para a página de consulta..
    request.getRequestDispatcher("consultaProduto.jsp")
        .forward(request, response);
}
}
}
}
```

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
        ServletException, IOException {
    execute(request, response);
}

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
    ServletException, IOException {
    execute(request, response);
}
}
```

---

### **cadastroProduto.jsp**

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>

<!-- Referenciar o ManagedBean -->
<jsp:useBean class="br.com.brq.managebeans.ManagedBeanFornecedor"
id="mb"></jsp:useBean>

<!-- JSTL (TagLibraries) -->
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fnc" %>

<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>Treinamento Java BRQ</title>

    <!-- Arquivos CSS do bootstrap -->
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css"/>
    <link rel="stylesheet" type="text/css" href="css/bootstrap-theme.css"/>

</head>
<body class="container">

    <!-- Cabeçalho da página -->
    <div class="well well-sm">
        <h2>Projeto Controle de Fornecedores e Produtos</h2>
        Treinamento BRQ COTI Informatica - Java Developer
    </div>

    <!-- Conteúdo principal da página -->
    <div class="col-md-12">

        <h3>Cadastro de Produtos</h3>
        <a href="index.jsp">Voltar</a> para a página inicial.
        <hr/>

        <div class="col-md-4">

            <form name="formulario" method="post"
                action="ControleProduto?acao=cadastrar">
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
<label>Nome do Produto:</label>
<input type="text" name="nome" id="nome"
       class="form-control"
       placeholder="Digite aqui"/>
<br/>

<label>Preço:</label>
<input type="text" name="preco" id="preco"
       class="form-control"
       placeholder="Digite aqui"/>
<br/>

<label>Quantidade:</label>
<input type="number" min="1" max="1000"
       name="quantidade"
       id="quantidade" class="form-control" />
<br/>

<label>Selecione o Fornecedor:</label>
<select name="fornecedor" id="fornecedor"
        class="form-control">

    <option value="">
        - Escolha uma Opção -
    </option>

    <!-- Varrer os fornecedores e gerar
        opções no campo -->

    <c:forEach items="${mb.listagemFornecedores}"
               var="f">

        <option value="${f.idFornecedor}">
            ${f.nome}
        </option>
    </c:forEach>

</select>
<br/>
<input type="submit" value="Cadastrar Produto"
       class="btn btn-success"/>
<br/>

<!-- EL Expression Language -->
<h4>${mensagem}</h4>
</form>
</div>
</div>
</body>
</html>
consultaProduto.jsp

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
pageEncoding="ISO-8859-1"%>

<!-- Referenciar o ManagedBean -->
<jsp:useBean class="br.com.brq.managebeans.ManagedBeanProduto"
id="mb"></jsp:useBean>

<!-- declarar as TagLibraries (JSTL) -->
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fnc" %>

<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>Treinamento Java BRQ</title>

    <!-- Arquivos CSS do bootstrap -->
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css"/>
    <link rel="stylesheet" type="text/css" href="css/bootstrap-theme.css"/>

</head>
<body class="container">

    <!-- Cabeçalho da página -->
    <div class="well well-sm">
        <h2>Projeto Controle de Fornecedores e Produtos</h2>
        Treinamento BRQ COTI Informatica - Java Developer
    </div>

    <!-- Conteúdo principal da página -->
    <div class="col-md-12">

        <h3>Consulta de Produtos</h3>
        <a href="index.jsp">Voltar</a> para a página inicial.
        <hr/>

        <h4>${mensagem}</h4>

        <table class="table table-hover">
            <thead>
                <tr>
                    <th>Código</th>
                    <th>Nome do Produto</th>
                    <th>Preço</th>
                    <th>Quantidade</th>
                    <th>Total</th>
                    <th>Fornecedor</th>
                    <th>Operações</th>
                </tr>
            </thead>

            <tbody>
                <!-- varrer a lista de produtos com JSTL -->
            </tbody>
        </table>
    </div>
</body>
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
<c:forEach items="#{mb.listagemProdutos}" var="p">

    <tr>
        <td>${p.idProduto}</td>
        <td>${p.nome}</td>
        <td>
            <fmt:formatNumber value="#{p.preco}" type="currency">
                </fmt:formatNumber>
            </td>
        <td>${p.quantidade}</td>
        <td>
            <fmt:formatNumber value="#{p.preco * p.quantidade}"
                type="currency">
                </fmt:formatNumber>
            </td>
        <td>${p.fornecedor.nome}</td>
        <td>
            <a href="ControleProduto?acao=visualizar
                &id=${p.idProduto}"
                class="btn btn-warning btn-sm">
                <span class="glyphicon glyphicon-search"></span>
                Visualizar
            </a>

            <a href="ControleProduto?acao=excluir&id=${p.idProduto}"
                class="btn btn-danger btn-sm"
                onclick="return confirm('Deseja excluir este
                    Produto?');">
                <span class="glyphicon glyphicon-trash"></span>
                Excluir
            </a>

            <a href="ControleProduto?acao=editar&id=${p.idProduto}"
                class="btn btn-primary btn-sm">
                <span class="glyphicon glyphicon-pencil"></span>
                Editar
            </a>
        </td>
    </tr>

</c:forEach>

</tbody>
<tfoot>
    <tr>
        <td colspan="7">
            Quantidade de registros:
            ${fn:length(mb.listagemProdutos)}
        </td>
    </tr>
</tfoot>
</table>

</div>
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
</body>
</html>
```

---

### **detalhesProduto.jsp**

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>Treinamento Java BRQ</title>

    <!-- Arquivos CSS do bootstrap -->
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css"/>
    <link rel="stylesheet" type="text/css" href="css/bootstrap-theme.css"/>

</head>
<body class="container">

    <!-- Cabeçalho da página -->
    <div class="well well-sm">
        <h2>Projeto Controle de Fornecedores e Produtos</h2>
        Treinamento BRQ COTI Informatica - Java Developer
    </div>

    <!-- Conteúdo principal da página -->
    <div class="col-md-12">

        <h3>Detalhes do Produto</h3>
        <a href="consultaProduto.jsp">Voltar</a>
            para a página de consulta.
        <hr/>

        <h4>${mensagem}</h4>

        <h4>Dados do Produto</h4>

        <label>Id do Produto: </label> ${produto.idProduto}
        <br/>
        <label>Nome do Produto: </label> ${produto.nome}
        <br/>
        <label>Preço: </label> ${produto.preco}
        <br/>
        <label>Quantidade: </label> ${produto.quantidade}
        <br/>
        <label>Total: </label> ${produto.preco * produto.quantidade}
        <br/>
        <label>Nome do Fornecedor: </label> ${produto.fornecedor.nome}
        <br/>
        <label>Descrição: </label> ${produto.fornecedor.descricao}
        <br/>

    </div>
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
</body>
</html>
```

---

### edicaoProduto.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>

<!-- Referenciar o ManagedBean -->
<jsp:useBean class="br.com.brq.managebeans.ManagedBeanFornecedor"
id="mb"></jsp:useBean>

<!-- TagLibraries (JSTL) -->
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fnc" %>

<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>Treinamento Java BRQ</title>

    <!-- Arquivos CSS do bootstrap -->
    <link rel="stylesheet" type="text/css" href="css/bootstrap.css"/>
    <link rel="stylesheet" type="text/css" href="css/bootstrap-theme.css"/>

</head>
<body class="container">

    <!-- Cabeçalho da página -->
    <div class="well well-sm">
        <h2>Projeto Controle de Fornecedores e Produtos</h2>
        Treinamento BRQ COTI Informatica - Java Developer
    </div>

    <!-- Conteúdo principal da página -->
    <div class="col-md-12">

        <h3>Editar dados do Produto</h3>
        <a href="consultaProduto.jsp">Voltar</a>
        para a página de consulta.
        <hr/>

        <div class="col-md-4">

            <form name="formulario" method="post"
            action="ControleProduto?acao=atualizar">

                <label>Id do Produto:</label>
                <input type="text" name="idproduto" class="form-control"
                    value="${produto.idProduto}" readonly="readonly"/>
                <br/>

                <label>Nome do Produto:</label>
            
```



# TREINAMENTO JAVA – BRQ/SP

## Terça-feira, 24 de Maio de 2016

Desenvolvimento de projetos web com JSP, Servlets e JDBC.  
Introdução ao JSTL (JSP Standard Tag Libraries)

Aula  
**10**

```
<input type="text" name="nome" class="form-control"
       value="${produto.nome}" />
<br/>

<label>Preço:</label>
<input type="text" name="preco" class="form-control"
       value="${produto.preco}" />
<br/>

<label>Quantidade:</label>
<input type="number" name="quantidade"
       class="form-control"
       value="${produto.quantidade}" />
<br/>

<label>Selecione o Fornecedor:</label>
<select name="fornecedor" class="form-control">

    <c:forEach items="${mb.listagemFornecedores}"
               var="f">

        <option value="${f.idFornecedor}"
               ${f.idFornecedor == produto.fornecedor
               .idFornecedor ? 'selected' : ''}>
            ${f.nome} </option>
    </c:forEach>

</select>
<br/>

<input type="submit" value="Atualizar Dados"
       class="btn btn-success"/>
<br/>

<h4>${mensagem}</h4>

</form>
</div>

</div>

</body>
</html>
```