



TREINAMENTO JAVA – BRQ/SP

Segunda-feira, 23 de Maio de 2016

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

Aula
09

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

Segunda-feira, 23 de Maio de 2016

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

Aula
09

```
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

Segunda-feira, 23 de Maio de 2016

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

Aula
09

```
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

Segunda-feira, 23 de Maio de 2016

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

Aula
09

```
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

Segunda-feira, 23 de Maio de 2016

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

Aula
09

```
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

Segunda-feira, 23 de Maio de 2016

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

Aula
09

```
//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)

Biblioteca CSS para frontend de aplicações web:

- Obtendo o bootstrap:

Aww yeah, Bootstrap 4 is coming!

Bootstrap Getting started CSS Components JavaScript Customize Themes Expo Blog

Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile first projects on the web.

Download Bootstrap

Currently v3.3.6

Slack - a messaging app for teams... integration with...



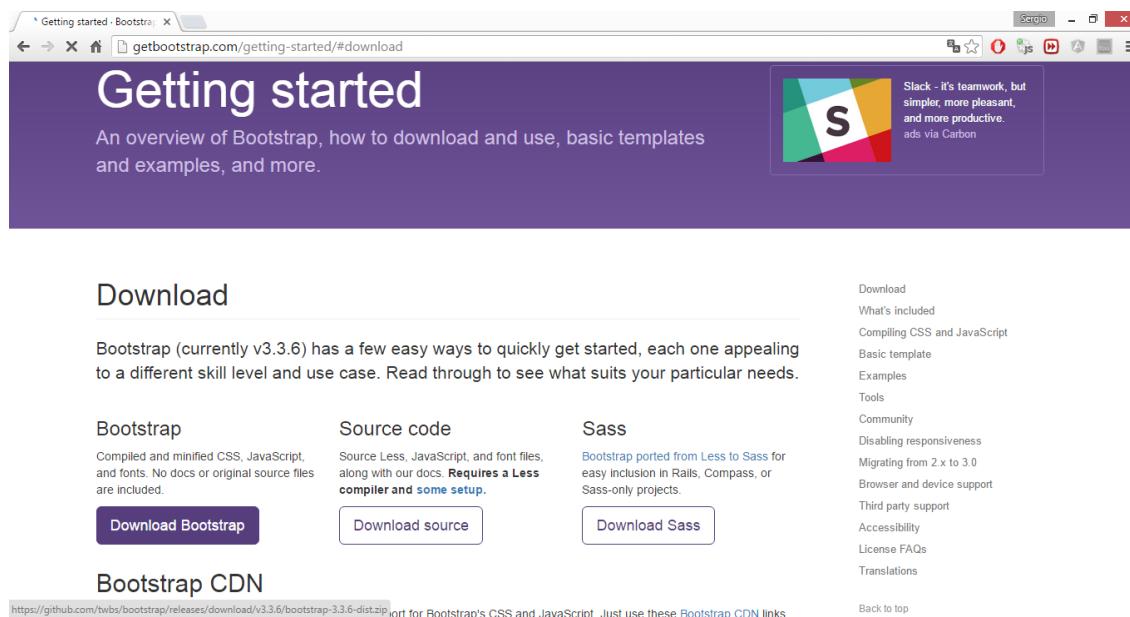
TREINAMENTO JAVA – BRQ/SP

Segunda-feira, 23 de Maio de 2016

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

Aula
09

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



Getting started

An overview of Bootstrap, how to download and use, basic templates and examples, and more.

Download

Bootstrap

Compiled and minified CSS, JavaScript, and fonts. No docs or original source files are included.

Source code

Source Less, JavaScript, and font files, along with our docs. Requires a Less compiler and some setup.

Sass

Bootstrap ported from Less to Sass for easy inclusion in Rails, Compass, or Sass-only projects.

Download Bootstrap

Download source

Download Sass

Slack - it's teamwork, but simpler, more pleasant, and more productive. ads via Carbon

Download

What's included

Compiling CSS and JavaScript

Basic template

Examples

Tools

Community

Disabling responsiveness

Migrating from 2.x to 3.0

Browser and device support

Third party support

Accessibility

License FAQs

Translations

Back to top

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

Segunda-feira, 23 de Maio de 2016

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

Aula
09

```
        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:

The screenshot shows a web browser window with the following details:

- Title Bar:** Treinamento Java BRQ
- Address Bar:** localhost:8082/aula09_projeto/
- Page Title:** Projeto Controle de Fornecedores e Produtos
- Page Content:** Treinamento BRQ COTI Informatica - Java Developer
- Text:** Bem vindo ao projeto
- Text:** Selecione a operação desejada:
- Buttons:** Four buttons with icons and labels:
 - Cadastrar Fornecedores (monitor icon)
 - Consultar Fornecedores (grid icon)
 - Cadastrar Produtos (pencil icon)
 - Consultar Produtos (grid icon)



TREINAMENTO JAVA – BRQ/SP

Segunda-feira, 23 de Maio de 2016

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

Aula
09

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>
```



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

Segunda-feira, 23 de Maio de 2016

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

Aula
09

```
        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..
}

}
```

Continua...