

CONGRATULAÇÕES!

**Em instantes começaremos
nossa aula de hoje !!!**



Antonio Andrade .:

Membro ANPPD® | Professor
Mestre TI | Trasmontano Saúde TI...



LinkedIn



Prof. Antonio Andrade
a.a.s@uni9.pro.br

Inovação Tecnológica, Sistemas e Segurança da Informação

Olá! Congratulações!

Teste sua conexão (vídeo e áudio)

Relate no chat. Tudo bem com áudio e vídeo?

Teste de mídia



Em instantes começaremos nossa aula de hoje !!!



ANTONIO ANDRADE

LATTES : <http://lattes.cnpq.br/9400427944771244>

LINKEDIN : <https://www.linkedin.com/in/antonio-andrade-msc/>

FACEBOOK : <https://www.facebook.com/face.aas>

E-MAIL : a.a.s@uni9.pro.br



{ antonio
andrade }



MEIO ACADÊMICO & CORPORATIVO

- » Graduado em Tecnologia da Informação;
- » **Mestre** em Engenharia Biomédica;
- » Atua como **professor** em Instituição de Ensino Superior (IES) desde 2009.
- » Certificações na área de **Privacidade**
- » DPO - Membro do **Comitê de Segurança da APDados (Associação dos Profissionais de Privacidade de Dados)**
- » Sócio das empresas **Viaclient** e **SMIT** com foco em ERP e Inovações Tecnológicas;
- » Coordenador de **desenvolvimento de sistemas** em empresa da área da Saúde;
- » Membro do **Comitê de LGPD** em empresa da área da Saúde e organização do terceiro setor;
- » **Palestrante** em eventos e capacitações relacionados à GDPR e LGPD.





DW – Desenvolvimento Web

OBJETOS IMPLÍCITOS

São nove objetos implícitos:

Application: tem escopo para toda a aplicação e pode ser utilizado para compartilhar os mesmos dados entre páginas JSP e quaisquer outros componentes da mesma aplicação.

Config: tem como escopo a própria página JSP e pode ser utilizado para receber dados de inicialização.

Exception: tem escopo na página. Ele é acessível em páginas definidas como as de erro.

Out: permite enviar dados para os navegadores de forma simples. O escopo desse objeto é a própria página.





DW – Desenvolvimento Web

OBJETOS IMPLÍCITOS



Page: tem escopo de página e é uma instância da classe do Servlet criado a partir do JSP.

pageContext: Acessa informações da página.

Request: representa os dados que foram transmitidos à página corrente. Seu escopo, como seria de se esperar, é de requisição. Seu método mais importante é o `getParameter()`.

Response: tem escopo de página, além de representar os dados enviados para outras páginas. Um método útil desse objeto é o `sendRedirect()`, que redireciona o funcionamento da aplicação para outro endereço.

Session: Seu escopo é de sessão e é um dos objetos mais utilizados.



DW – Desenvolvimento Web

VETORES E COLEÇÕES

Podemos declarar um vetor de Strings da seguinte forma:

```
String[] fones = new String[4];
```

```
<%
```

```
List<Pessoa> pessoas = (ArrayList) request.getAttribute("listaPessoas");
```

```
for (Pessoa p : pessoas) {
```

```
%>
```

```
.....
```

```
<%
```

```
}
```

```
%>
```

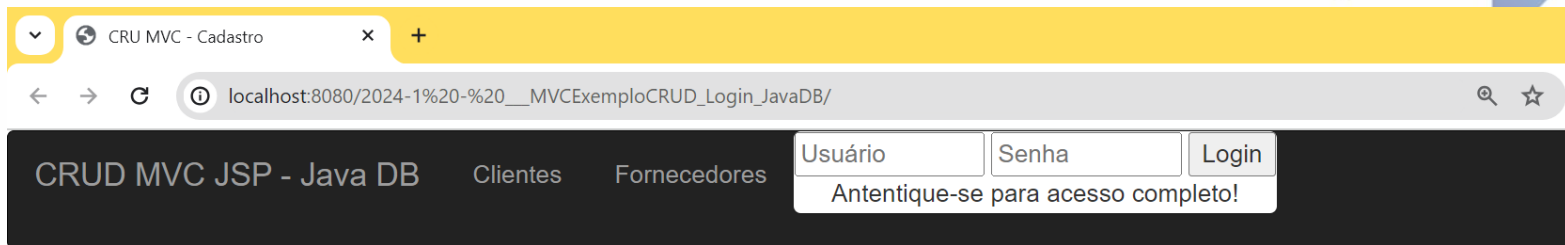



```
public List<Pessoa> listar() throws SQLException {  
    String sql = "SELECT * FROM pessoa ORDER BY nome ASC";  
    List lstPessoas = new ArrayList();  
    try (  
        PreparedStatement ps = conexao.prepareStatement(sql);  
        ResultSet rs = ps.executeQuery()) {  
        while (rs.next()) {  
            Pessoa p = new Pessoa();  
            p.setId(rs.getInt("id"));  
            p.setNome(rs.getString("nome"));  
            p.setTelefone(rs.getString("telefone"));  
            lstPessoas.add(p);  
        }  
    }  
    conexao.close();  
    return lstPessoas;  
}
```



DW – Desenvolvimento Web

2024-1 - __MVCExemploCRUD_Login_JavaDB



Informações

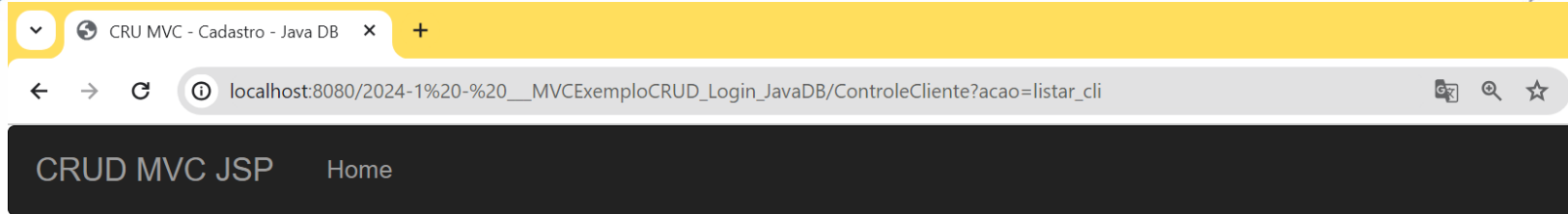
Este é um exemplo de implementação de um sistema de **Cadastro de Pessoas** utilizando o modelo MVC com conexão ao banco de dados MySQL.

No Menu (topo da tela) estão algumas opções do CRUD para manipulação dos dados. As demais (Editar e Excluir) estão disponíveis após uma Pesquisa ou Listagem de Todos os Registros.

Após o estudo completo deste exemplo, é aconselhável que você implemente as diversas validações para as entradas de dados, bem como mensagens de erro, caso ocorram.







DW – Desenvolvimento Web

2024-1-1 - __MVCEmploCRUD_Login_JavaDB



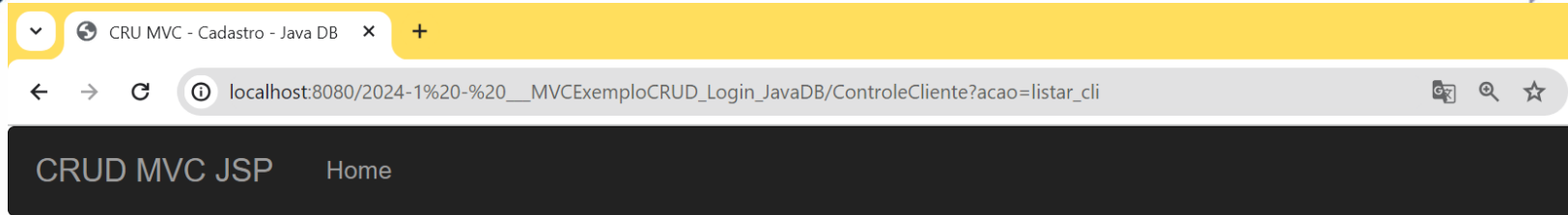
Cientes Cadastrados - Java DB

[Incluir](#) [Pesquisar](#)

ID	Nome	Telefone	Valor Última Venda	Editar	Excluir
3	Mappin Pca Ramos	(11) 343-4011	3011.0		
1	SMIT	(11) 5050-4040	1000.0		
2	Viaclient Tecnologia	(11) 3030-4040	22222.0		

DW – Desenvolvimento Web

2024-1-1 - __MVCEmploCRUD_Login_JavaDB



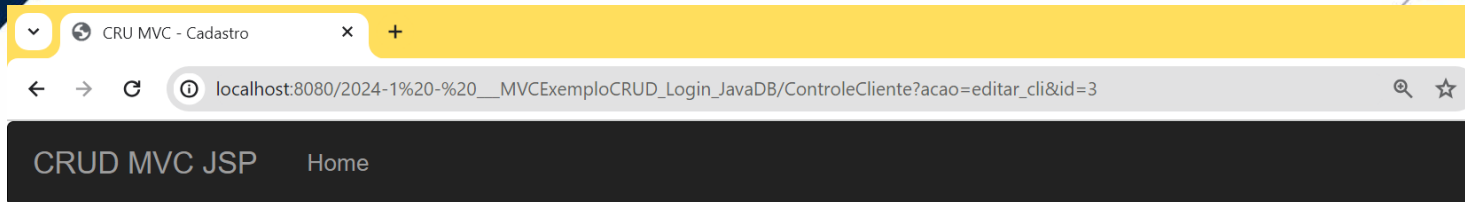
Clientes Cadastrados - Java DB

Incluir Pesquisar

ID	Nome	Telefone	Valor Última Venda	Editar	Excluir
3	Mappin Pca Ramos	(11) 343-4011	3011.0	Editar	Excluir
1	SMIT	(11) 5050-4040	1000.0	Editar	Excluir
2	Viaclient Tecnologia	(11) 3030-4040	22222.0	Editar	Excluir

DW – Desenvolvimento Web

2024-1 - __MVCEmploCRUD_Login_JavaDB



Edição do Registro

Nome

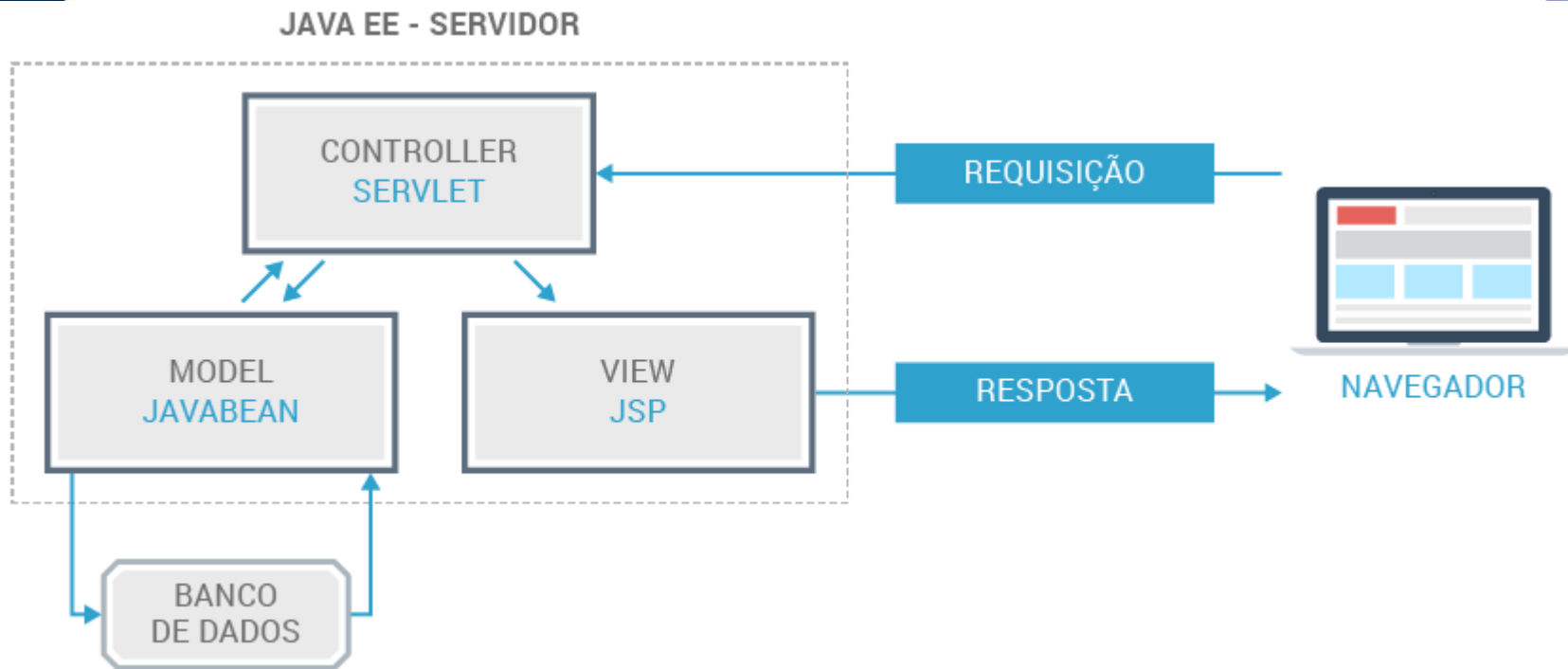
Telefone

Valor Última Venda

Salvar

DW – Desenvolvimento Web

O modelo MVC

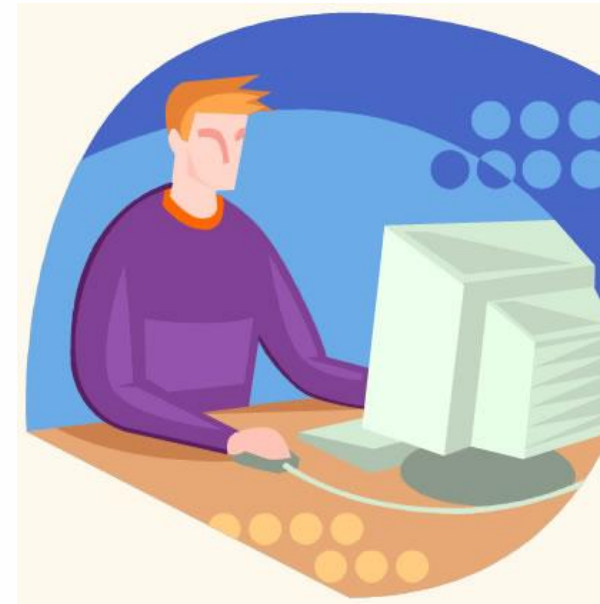


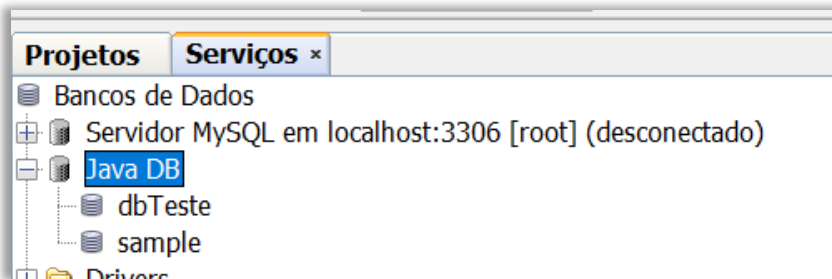
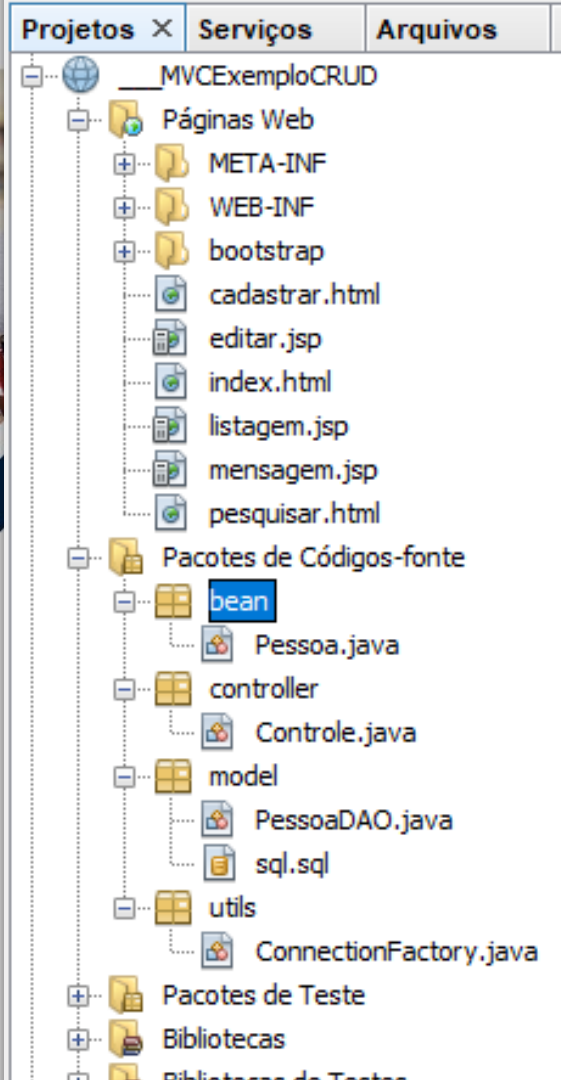
DW – Desenvolvimento Web

Vamos praticar... Ex:

2024-1 - ____MVCExemploCRUD_Login_JavaDB

- Colocar o exemplo para funcionar
- Aguardar instruções de complemento para atividade...



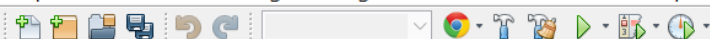


1. Fazer download do exemplo
2. Descompactar em pasta local
3. Entrar no *Netbeans*
4. Abrir o projeto.
5. Entrar no JavaDB (Derby)
6. Criar o BD: **dbTeste**

Como criar o BD?

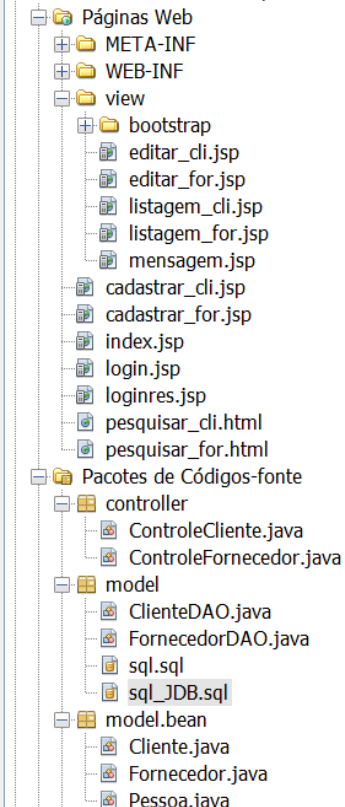
Veja a seguir...





Projetos x Serviços

2024-1 - __MVCEmploCRUD_Login_JavaDB



sql_JDB.sql x

Código-Fonte

Histórico

Conexão:

```
1  -- EXEMPLO:
2  -- https://db.apache.org/derby/docs/10.2/ref/rrefsqj24513.html
3  --
4  CREATE TABLE CliFor (
5      id                INT NOT NULL GENERATED ALWAYS AS IDENTITY
6      CONSTRAINT CliFor_PK PRIMARY KEY,
7      nome              varchar(30) NOT NULL,
8      telefone          varchar(20) NOT NULL,
9      tipo              varchar(10) NOT NULL,
10     valorUltOper       float      NOT NULL
11 );
12
13
```

Estrutura do projeto e a tabela de exemplo para CRUD

Variáveis x

DW – Desenvolvimento Web

- Adicionar os seguintes campos no exemplo aplicado:
- **Nome:**
- **CEP:**
- **Logradouro:**
- **Bairro:**
- **Cidade:**
- **Estado**
- **E-Mail:**
- **Telefone:**

PV – Programação Visual - c#

ATIVIDADE





DICAS PARA ESTUDO:

- Seja “**CURIOSO**”:
 - Procure revisar o que foi estudado.
 - Pesquise as referências bibliográficas.
- Seja “**ANTENADO**”:
 - Leia a próxima aula pelo AVA.
- Seja “**COLABORATIVO**”:
 - Publique assuntos relevantes nos fóruns do AVA.
 - Coloque suas dúvidas relativas às aulas nos fóruns do AVA.
 - Responda à dúvidas dos colegas nos fóruns do AVA.

Prof. Antonio Andrade

a.a.s@uni9.pro.br



Atividade

- Adicionar 1 campo na aplicação:

1. BD - Tabela pessoa (sql)
2. Model
 - I. Pessoa.java (javabean)
 - II. PessoaDAO
 - a. Inserir
 - b. Editar
 - c. Alterar
 - d. Pesquisar
 - e. Listar
3. Cadastrar.html
4. Editar.jsp
5. Listagem.jsp
6. Controller
 - I. Pegar campo do formulário para passar para model.
 - I. (cadastrar e alterar)