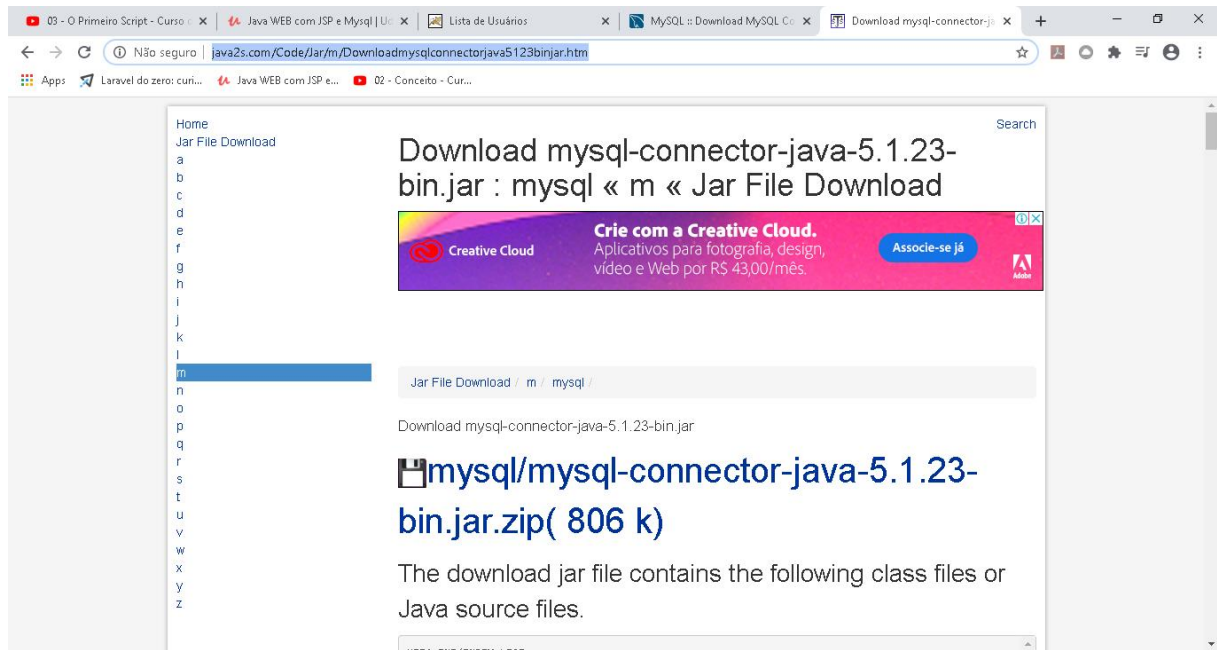


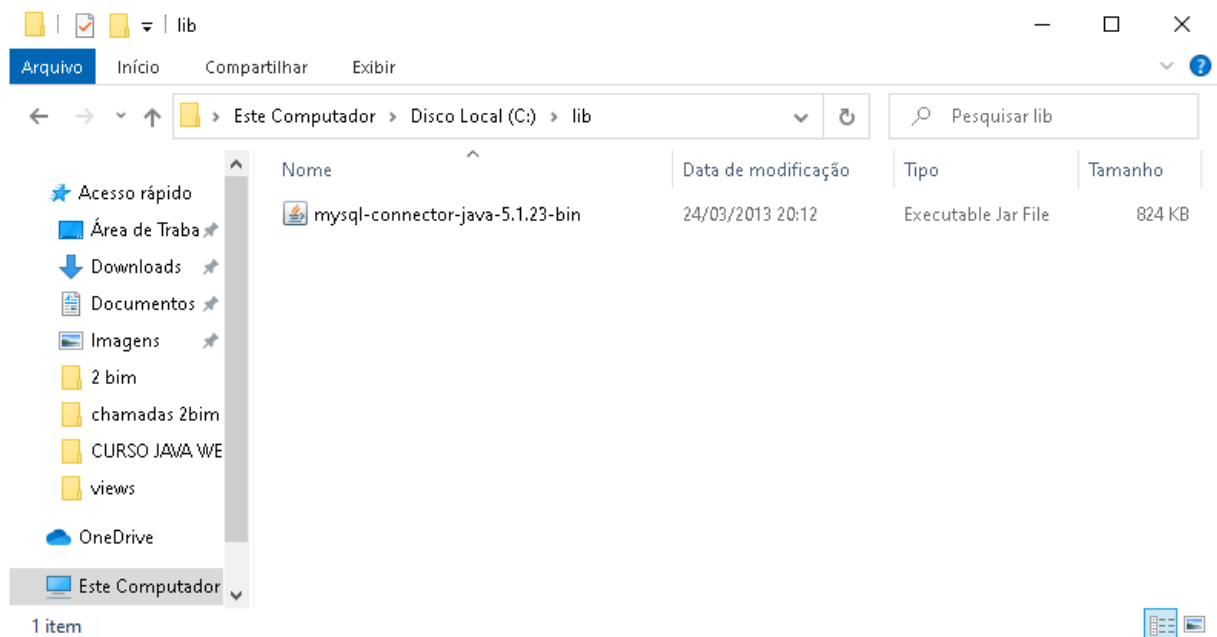
1. CONEXÃO COM BANCO DE DADOS

Primeiro passo é fazer o download de um conector mysql para java. Acesse:

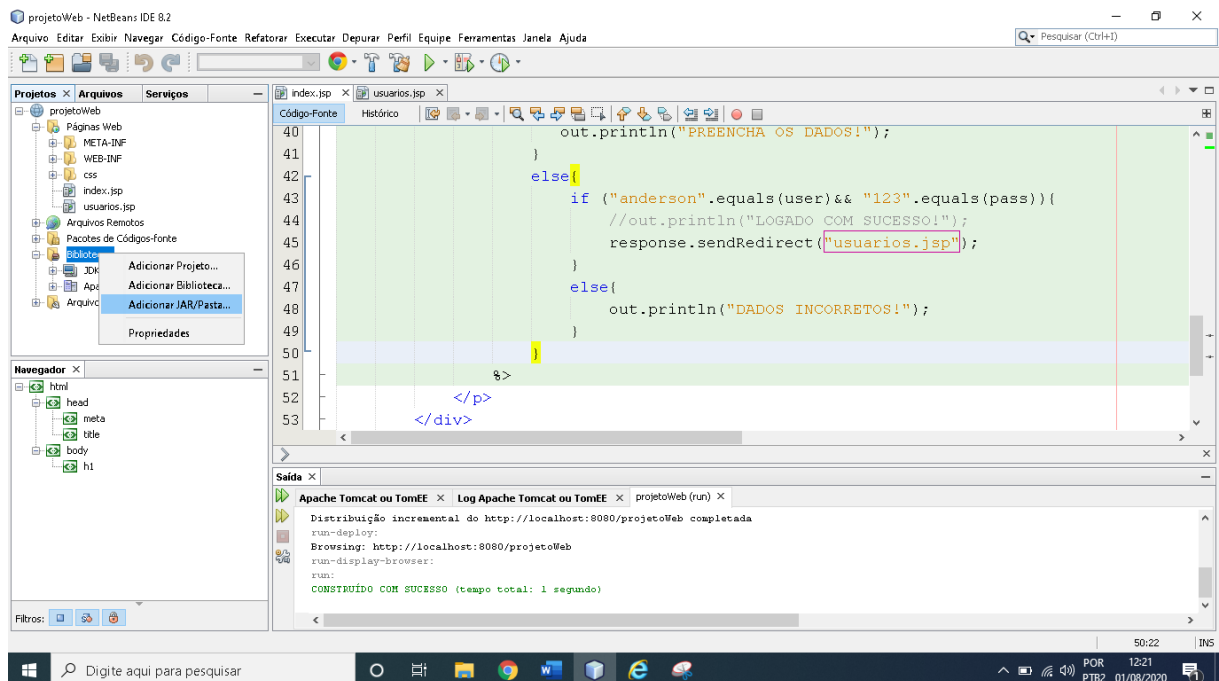
<http://www.java2s.com/Code/Jar/m/Downloadmysqlconnectorjava5123binjar.htm>

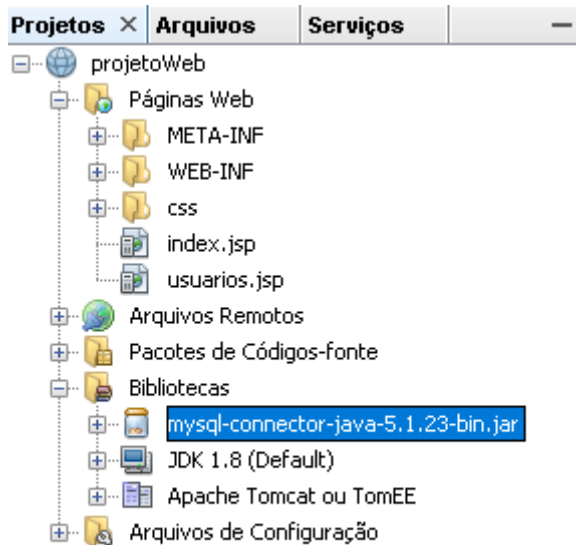


Após baixar o arquivo zip, crie uma pasta em C:\ chamada lib e salve o arquivo nela.



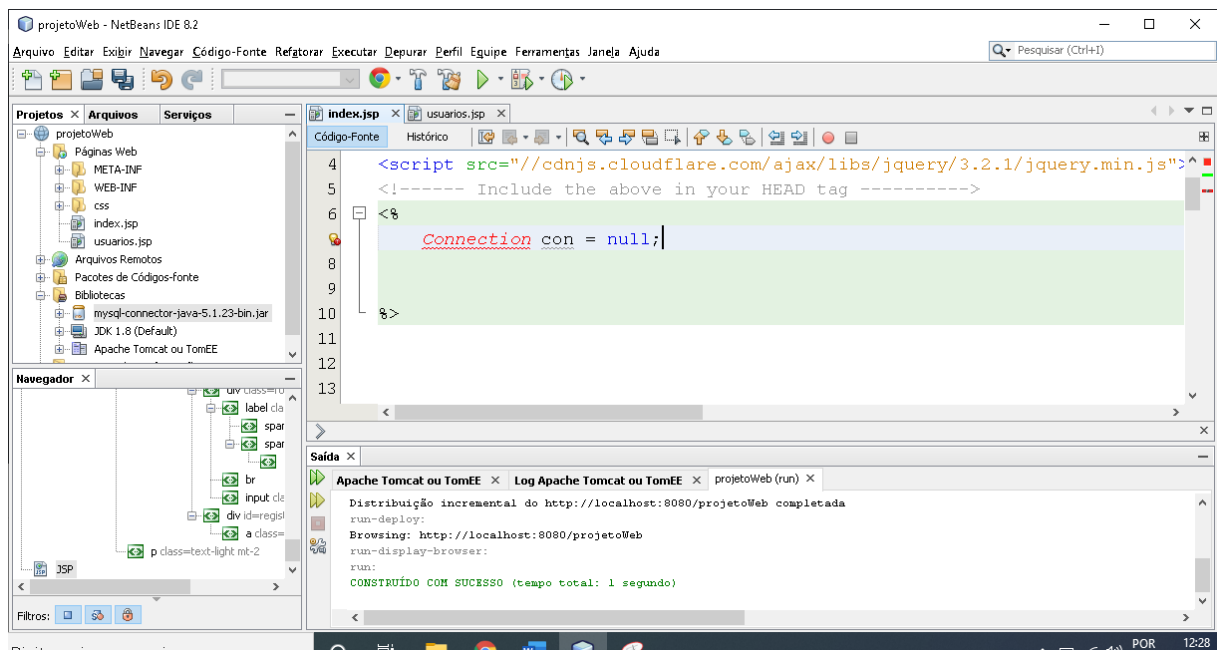
Adicione à biblioteca do projeto o arquivo .jar extraído.



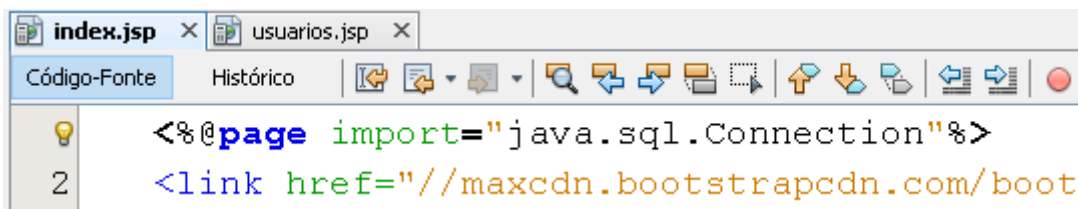
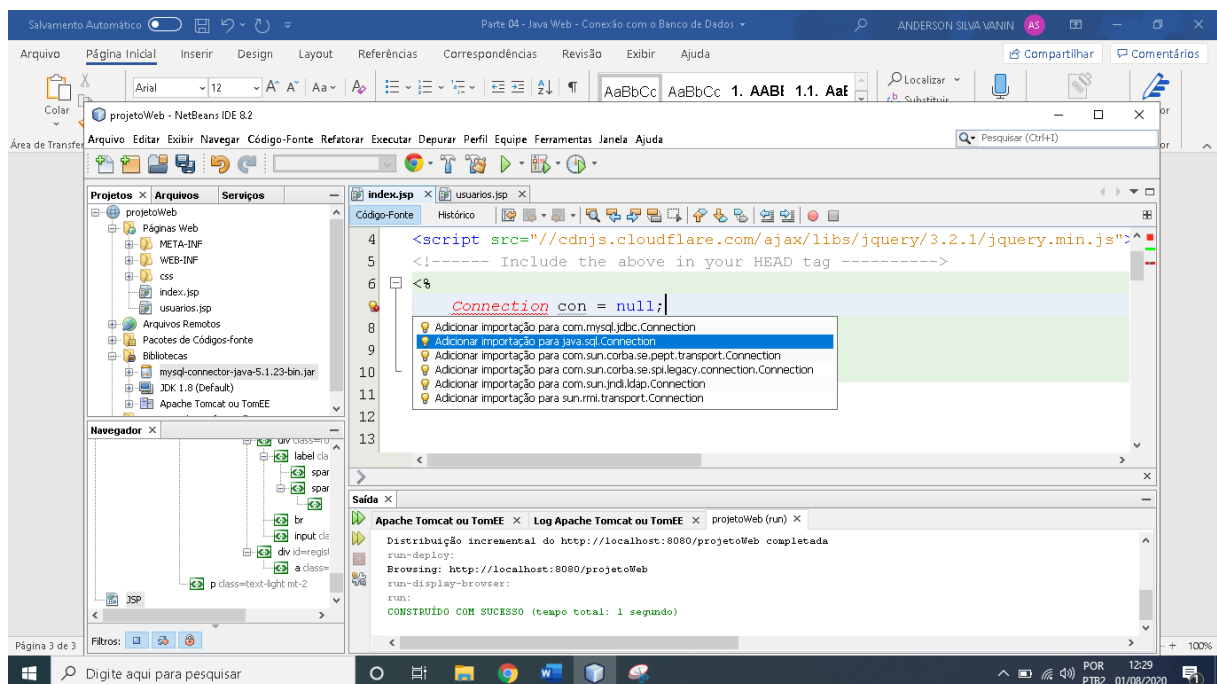


Feito isso, para começarmos a utilizar funções de conexão, precisaremos declarar algumas variáveis que serão utilizadas para esta operação.

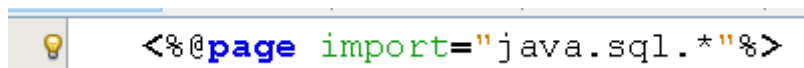
No arquivo index.jsp insira o seguinte código:



Veja que é necessário importar as classes para conexão. Clique no aviso e escolha `java.sql.connection`



No pacote importado, troque a palavra Connection por * para que a biblioteca possa ser executada pelas outras variáveis que iremos criar.



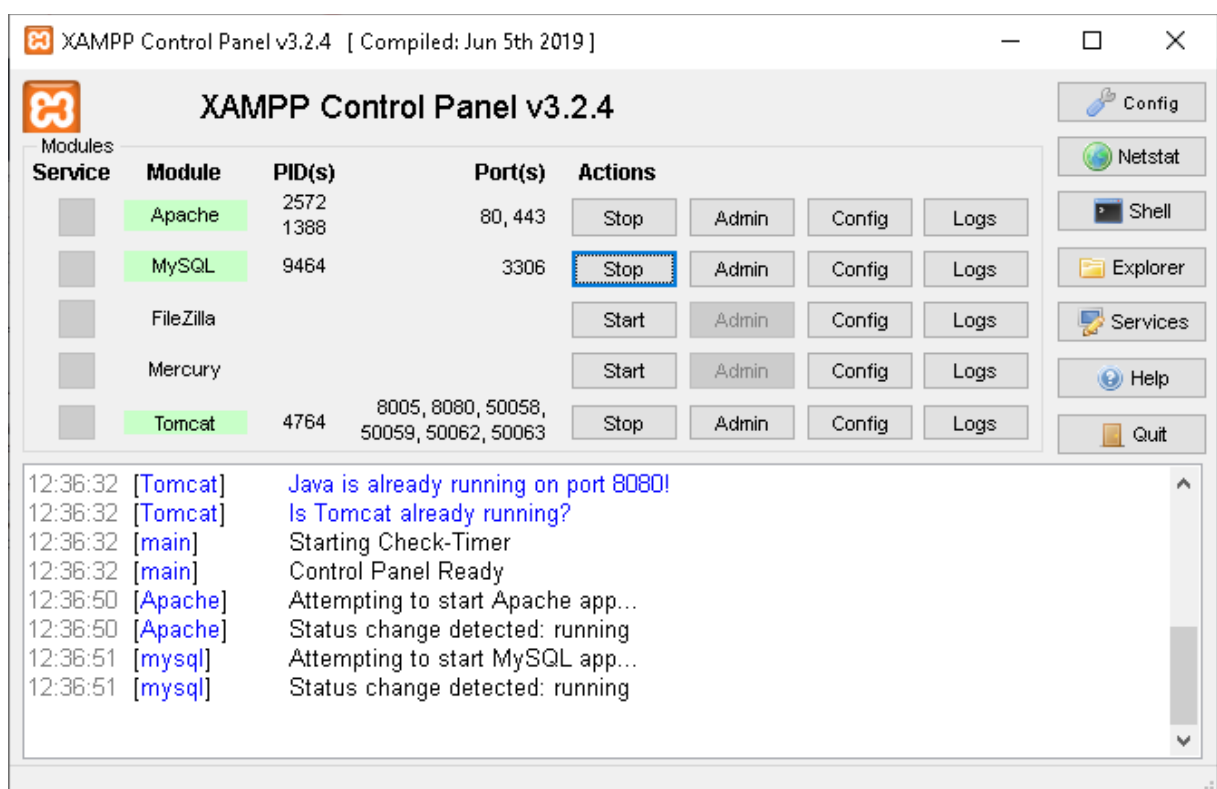
Agora crie as demais variáveis.

```

5      <script src="//cdnjs.cloudflare.com/ajax/lib
6      <!------- Include the above in your HEAD tag
7      <%
8          Connection con = null;
9          Statement st = null;
10         ResultSet rs = null;
11
12
13      %>
14
15      <body>

```

Para conectar ao banco utilizaremos o Xampp. Para isso é necessário que o Apache e o Mysql estejam iniciados.



Vamos agora criar um banco de dados chamado javaweb e uma tabela usuarios com o seguintes campos:

Servidor: 127.0.0.1 > Base de Dados: javaweb > Tabela: usuarios

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Privilegios Operacoes Mais

Estrutura da tabela Visão de relação(ões)

#	Nome	Tipo	Agrupamento (Collation)	Atributos	Nulo	Predefinido	Comentários	Extra	Ações
1	id	int(11)			Não	Nenhum		AUTO_INCREMENT	Muda Elimina Mais
2	nome	varchar(50)	utf8mb4_general_ci		Não	Nenhum			Muda Elimina Mais
3	email	varchar(50)	utf8mb4_general_ci		Não	Nenhum			Muda Elimina Mais
4	senha	varchar(50)	utf8mb4_general_ci		Não	Nenhum			Muda Elimina Mais
5	nivel	varchar(25)	utf8mb4_general_ci		Não	Nenhum			Muda Elimina Mais

Agora vamos inserir alguns dados para que possamos fazer os primeiros testes.

✓ A mostrar registos de 0 - 0 (1 total, A consulta demorou 0,0019 segundos.)

SELECT * FROM `usuarios`

Mostrar tudo | Número de registos: 25 | Filtrar registos: Pesquisar esta tabela

+ Opções

	id	nome	email	senha	nivel
<input type="checkbox"/> Editar Copiar Apagar	1	Zezinho	zezinho@email.com	123	comum

Marcar todos Com os seleccionados: Editar Copiar Apagar Exportar

Voltando ao arquivo index.jsp, vamos agora, utilizando o objeto de conexão con criado com a classe Connection instanciada acima, criar a nossa conexão. Adicione a seguinte linha de código:

```
p align="center" class="text-light mt-2">
<%
    String user = request.getParameter("usuario");
    String pass = request.getParameter("senha");

    con = DriverManager.getConnection("jdbc:mysql://localhost/javaweb?user=root&password=");
    st = con.createStatement();
    rs = st.executeQuery("");
```

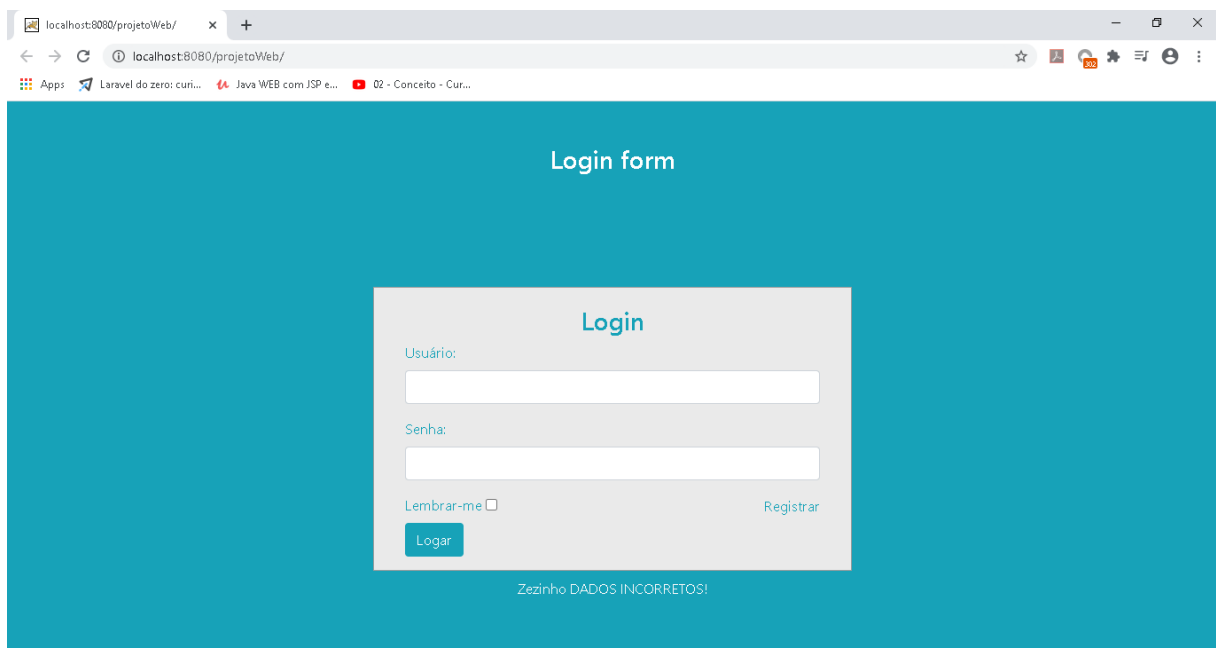
A variável rs deve receber uma consulta que será realizada na tabela usuários. Antes de verificarmos se o usuário e a senha passados no formulário existem, vamos trazer todos os dados da tabela usuários para a nossa página, de modo a testarmos se a conexão e a consulta estão funcionando perfeitamente.

```
align="center" class="text-light mt-2">
<%
String user = request.getParameter("usuario");
String pass = request.getParameter("senha");

try {
    Class.forName("com.mysql.jdbc.Driver");
    con = DriverManager.getConnection("jdbc:mysql://localhost/javaweb?user=root&password=");
    st = con.createStatement();
    rs = st.executeQuery("SELECT * FROM usuarios");
    while (rs.next()) {
        out.println(rs.getString(2));
    }
} catch (Exception e) {
    out.println(e);
}

if ({user == ""} || {pass == ""}) {
    out.println("PREENCHA OS DADOS!");
} else {
```

Teste o arquivo e veja que o nome do usuário já aparece na execução do arquivo.



Agora que já sabemos que deu tudo certo com a conexão, vamos fazer a validação com os dados existentes no banco. Altere o código conforme mostrado abaixo.

```
43 <p align="center" class="text-light mt-2">
44 <%
45     String user = request.getParameter("usuario");
46     String pass = request.getParameter("senha");
47     String usuario = "";
48     String senha = "";
49
50     try {
51         Class.forName("com.mysql.jdbc.Driver");
52         con = DriverManager.getConnection("jdbc:mysql://localhost/javaweb?user=root&password=");
53         st = con.createStatement();
54         rs = st.executeQuery("SELECT * FROM usuarios");
55         while (rs.next()) {
56             //out.println(rs.getString(2));
57             usuario = rs.getString(3);
58             senha = rs.getString(4);
59         }
60     } catch (Exception e) {
61         out.println(e);
62     }
63
64     if ((user == null) || (pass == null)) {
65         out.println("PREENCHA OS DADOS!");
66     } else {
67         if (user.equals(usuario) && pass.equals(senha)) {
68             //out.println("LOGADO COM SUCESSO!");
```

Teste o código com os dados de usuário cadastrados no banco de dados.

O código funciona porém ainda não está sendo filtrado por email e senha. Na próxima aula faremos a filtragem.