

Ajax

Para exemplificar os conceitos de Ajax vamos fazer algumas modificações no projeto aulaspring.

1. Verifique se sua tabela no banco de dados tem um campo na tabela alunos chamado "formado", do tipo boolean. Obs.: esse novo campo foi adicionado ao banco de dados na aula de Spring. Se seu BD não tem esse campo, adicione-o ou baixe o BD da aula de Spring.
2. Modifique a listagem dos registros (arquivo lista.jsp), acrescentando uma coluna para o campo formado:

```
<table border=1>
  <tr>
    <td><b>Id</b></td>
    <td><b>Nome</b></td>
    <td><b>E-mail</b></td>
    <td><b>Endereço</b></td>
    <td><b>Formado?</b></td>
    <td><b>Alterar?</b></td>
    <td><b>Remover?</b></td>
  </tr>
  <c:forEach items="{alunos}" var="aluno">
    <tr>
      <td>${aluno.id}</td>
      <td>${aluno.nome}</td>
      <td>${aluno.email}</td>
      <td>${aluno.endereco}</td>
      <c:if test="{aluno.formado eq false}">
        <td>Não</td>
      </c:if>
      <c:if test="{aluno.formado eq true}">
        <td>Sim</td>
      </c:if>
      <td><a href="exibeAluno?id=${aluno.id}">Alterar</a></td>
      <td>
        <a href="removeAluno?id=${aluno.id}">Remover</a>
      </td>
    </tr>
  </c:forEach>
</table>
```

3. Modifique o arquivo exibe.jsp acrescentando um campo para armazenar o valor do novo campo:

```
<input type="hidden" id="formado" name="formado" value="{aluno.formado}"/><br />
```

Ajax (Asynchronous Javascript and XML): técnica que possibilita o envio de requisições assíncronas. Em outras palavras, isso significa que podemos manter uma página intacta enquanto aguardamos a resposta da requisição.

Vamos implementar no nosso projeto um link para cada registro que marca o aluno como "formado", sem no entanto sairmos da página que está aberta, ou seja, a página listaAlunos.

- Adicione Ajax no seu projeto: crie uma pasta chamada resources em WebContent. Dentro da pasta resources, crie uma pasta js. Baixe o arquivo com as bibliotecas necessárias no

site da disciplina na aula de Ajax, extraia os arquivos (ou obtenha-os em jquery.com) e coloque dentro da pasta `resources/js`.

- No arquivo `spring-context.xml`, acrescente um comando para mapear um local onde os recursos da aplicação podem ser encontrados (imagens, css, etc). Isso também avisa o controlador Spring que ele não deve atender as requisições que solicitam esse conteúdo:

```
<mvc:resources location="/resources/" mapping="/resources/**"/>
```

- No arquivo `lista.jsp`, acrescente uma coluna na tabela:

```
<table border=1>
  <tr>
    <td><b>Id</b></td>
    <td><b>Nome</b></td>
    <td><b>E-mail</b></td>
    <td><b>Endereço</b></td>
    <td><b>Formado?</b></td>
    <td><b>Alterar status?</b></td>
    <td></td>
    <td></td>
  </tr>
  <c:forEach items="${alunos}" var="aluno">
    <tr>
      <td>${aluno.id}</td>
      <td>${aluno.nome}</td>
      <td>${aluno.email}</td>
      <td>${aluno.endereco}</td>
      <c:if test="${aluno.formado eq true}">
        <td>Sim</td>
      </c:if>
      <c:if test="${aluno.formado eq false}">
        <td id="td1aluno_${aluno.id}">Não</td>
      </c:if>
      <c:if test="${aluno.formado eq true}">
        <td></td>
      </c:if>
      <c:if test="${aluno.formado eq false}">
        <td id="td2aluno_${aluno.id}"><a href="#"
onClick="mudaStatus(${aluno.id})">Mudar</a></td>
      </c:if>
      <td><a href="exibeAluno?id=${aluno.id}">Alterar</a></td>
      <td><a href="remove?id=${aluno.id}">Remover</a></td>
    </tr>
  </c:forEach>
</table>
```

- Ainda no arquivo `lista.jsp`, modifique a tag `head` como segue:

```
<head>
  <script type="text/javascript" src="resources/js/jquery.js"></script>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Listagem dos Alunos</title>
</head>
```

- Ainda no arquivo `lista.jsp`, acrescente o script:

```
<script type="text/javascript">
function mudaStatus(id){
```

```
$.post("mudaStatus",{ 'id' : id},function(){
    $("#td1aluno_"+id).html("Sim");
    $("#td2aluno_"+id).html("");
})
}
</script>
```

- Na classe AlunoController.java, acrescente a ação:

```
@RequestMapping("mudaStatus")
public void mudaStatus(Long id,HttpServletResponse response){
    AlunoDAO dao = new AlunoDAO();
    dao.finaliza(id);
    response.setStatus(200);
}
```