

Tiago Ramos

ramos.tiago@gmail.com

<http://tramos.railsplayground.net>

MIRANTE TECNOLOGIA

CURSO STRUTS2

Download e Upload

- Trabalhando com resultados do tipo Stream
 - Pode-se criar um retorno global
 - Parâmetros definidos no resultado
 - contentType - o tipo de arquivo que será enviado para o browser (padrão = text / plain).
 - contentLength – tamanho em bytes do fluxo (navegador mostra uma barra de progresso).
 - contentDisposition - especifica o nome do arquivo (padrão filename = "document.pdf")
 - InputName - o nome da propriedade InputStream na ação encadeada (padrão = inputStream).
 - BufferSize - o tamanho do buffer (padrão = 1024).

Download e Upload

- Trabalhando com resultados do tipo Stream
 - Criando o resultado global

```
<global-results>
  <result name="download" type="stream">
    <param name="contentType">${downloadContentType}</param>
    <param name="inputName">downloadInputStream</param>
    <param name="contentDisposition">
      filename=${downloadFileName}
    </param>
    <param name="bufferSize">2048</param>
  </result>
</global-results>
```

Download e Upload

- Trabalhando com resultados do tipo Stream
 - Criando classe de ação

```
public class ExemploDownload {  
  
    private InputStream downloadInputStream;  
  
    public String input() {}  
  
    public String download() throws Exception {  
        downloadInputStream = new FileInputStream("C:\\Projetos\\classeweb\\src\\struts2.pdf");  
        return "download";  
    }  
  
    public InputStream getDownloadInputStream() {  
        return downloadInputStream;  
    }  
  
    public String getDownloadContentType() {  
        return "application/pdf";  
    }  
  
    public String getDownloadFileName() {  
        return "NomeArquivo.pdf";  
    }  
  
}
```

Download e Upload

- Trabalhando com entradas do tipo Stream
 - Struts2 possui um interceptador para facilitar o trabalho de upload (File Upload Interceptor)
 - Precisamos de duas bibliotecas a mais
 - commons-io-1.3.jar
 - commons-fileupload.jar
 - Podemos definir algumas propriedade para os arquivos
 - struts.multipart.saveDir=
 - struts.multipart.maxSize=2097152

Download e Upload

- Trabalhando com entradas do tipo Stream
 - Definindo o formulário

```
<s:form action="upload.action" enctype="multipart/form-data" method="POST">  
  <p>Arquivo: <s:file name="upload"/></p>  
  <p><s:submit value="Salvar"/></p>  
</s:form>
```

Download e Upload

- Trabalhando com entradas do tipo Stream
 - Definindo a classe de ação

```
public class ExemploUpLoad {  
  
    private File upload;  
  
    private String uploadContentType;  
  
    private String uploadFileName;  
  
    public String input() {  
  
    public String upload() throws Exception {  
        File file = new File("ArquivoUpload.jpg");  
        FileUtils.copyFile(upload, file);  
        return "upload";  
    }  
}
```

SiteMesh

- Auxilia na Criação de layout para a aplicação
 - Decoradores podem ser criados em jsp, html ou freemarker
 - Bibliotecas necessárias
 - struts2-sitemesh-plugin-2.0.11.1.jar
 - sitemesh-2.3.jar
 - Arquivos de configuração
 - /web/WEB-INF/decorators.xml

SiteMesh

- Configurando
 - Altere o /web/WEB-INF/web.xml

```
<filter>
  <filter-name>struts-cleanup</filter-name>
  <filter-class>
    org.apache.struts2.dispatcher.ActionContextCleanUp
  </filter-class>
</filter>

<filter>
  <filter-name>sitemesh</filter-name>
  <filter-class>
    com.opensymphony.module.sitemesh.filter.PageFilter
  </filter-class>
</filter>

<filter-mapping>
  <filter-name>struts-cleanup</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>

<filter-mapping>
  <filter-name>sitemesh</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
```

SiteMesh

- Configurando
 - Crie o arquivo /web/WEB-INF/decorators.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<decorators defaultdir="/decorators">

    <!-- Any urls that are excluded will never be decorated by Sitemesh -->
    <excludes>
        <pattern>/decorators/*</pattern>
    </excludes>

    <decorator name="main" page="main.jsp">
        <pattern>/*</pattern>
    </decorator>

</decorators>
```

SiteMesh

- Configurando
 - Crie o decorador /web/decorators/main.jsp

```
<html>
<body>
<table cellpadding="0" cellspacing="0" width="500px">
  <tr bgcolor="blue" height="45px">
    <td width="100%">Minha Aplicação :: <d:title default="" /></td>
  </tr>
  <tr height="400px">
    <td><d:body /></td>
  </tr>
  <tr bgcolor="blue" height="20px">
    <td>Rodapé da Aplicação</td>
  </tr>
</table>
</body>
</html>
```

||| Agora é sua vez...