



Faça como eu fiz na aula

Chegou a hora de você pôr em prática o que foi visto na aula. Para isso, siga os passos listados abaixo.

1) No pacote `br.com.alura.gerenciador.acao`, crie a interface `Acao`, com o método `executa`, com retorno de um `String` e que recebe como parâmetro um `HttpServletRequest` e um `HttpServletResponse`.

2) Nas classes `NovaEmpresa`, `AlterarEmpresa` e `RemoverEmpresa`, implemente a interface `Acao`, retorne uma `String` no método `executa` e dentro dele, retorne uma `String` que represente o redirecionamento para a entrada, enviando a ação a ser executada, que é a listagem de empresas:

```
return "redirect:entrada?acao=ListaEmpresas";
```

[COPIAR CÓDIGO](#)

3) Nas classes `ListaEmpresas` e `MostrarEmpresa`, implemente a interface `Acao`, retorne uma `String` no método `executa` e dentro dele, remova o `RequestDispatcher` e retorne uma `String` que represente um `forward` para os seus JSPs, **`listaEmpresas.jsp`** e **`formAlterarEmpresa.jsp`**, respectivamente. Por exemplo, no caso de `MostrarEmpresa`:

```
return "forward:formAlterarEmpresa.jsp";
```

[COPIAR CÓDIGO](#)

4) No pacote `br.com.alura.gerenciador.acao`, crie a classe `NovaEmpresaForm`. Nessa classe, implemente a interface `Acao` e dentro de `executa`, retorne uma `String` que represente um `forward` para **`formNovaEmpresa.jsp`**:

```
public class NovaEmpresaForm implements Acao {  
  
    public String executa(HttpServletRequest request, HttpServletRequest.  
        throws ServletException, IOException {  
        return "forward:formNovaEmpresa.jsp";  
    }  
}
```

[COPIAR CÓDIGO](#)

5) Dentro de **WebContent/WEB-INF**, crie a pasta **view** e **mov**a todos os JSPs para dentro dela.

6) Para melhorar a servlet `UnicaEntradaServlet`, dentro de `service`, monte o nome da classe através do parâmetro `acao` da requisição e dentro de um `try-catch`, carregue e instancie essa classe, chamando o seu método `executa` em seguida e atribuindo o seu retorno à variável `nome`.

7) Ainda na servlet `UnicaEntradaServlet`, dentro de `service`, fora do `try-catch`, ou você utilizará o `RequestDispatcher` para fazer um `forward`, ou fará um `sendRedirect`. Faça isso baseado no tipo e endereço da variável `nome`.

Opinião do instrutor

A classe `UnicaEntradaServlet` ficará assim:

```
@WebServlet("/entrada")  
public class UnicaEntradaServlet extends HttpServlet {  
    private static final long serialVersionUID = 1L;  
  
    protected void service(HttpServletRequest request, HttpS
```

```
throws ServletException, IOException {

String paramAcao = request.getParameter("acao");

String nomeDaClasse = "br.com.alura.gerenciador.acao."
    paramAcao;

String nome;
try {
    Class classe = Class.forName(nomeDaClasse);
    Acao acao = (Acao) classe.newInstance();
    nome = acao.executa(request,response);
} catch (ClassNotFoundException |
    InstantiationException |
    IllegalAccessException e) {
    throw new ServletException(e);
}

String[] tipoEEndereco = nome.split(":");
if(tipoEEndereco[0].equals("forward")) {
    RequestDispatcher rd =
        request.getRequestDispatcher("WEB-INF/view/" +
        rd.forward(request, response);
} else {
    response.sendRedirect(tipoEEndereco[1]);
}
}

}
```

[COPIAR CÓDIGO](#)

Continue com os seus estudos, e se houver dúvidas, não hesite em recorrer ao nosso fórum!

