

Frameworks Java

Parte II

Prof. Dr. Edson A. Oliveira Junior

edson@din.uem.br

www.din.uem.br/~edson

Agenda

- Acesso aos dados da requisição
- Organizando o projeto com novos controladores

Recuperando Dados do Formulário

```
<form action="persistePessoa" method="post">
```

- Adicione “persistePessoa” na propriedade “action” do seu formulário JSP
- Criar a classe Pessoa.java em Pacotes de Código Fonte /
br.uem.din.grsse.springmvc.model

Recuperando Dados do Formulário

```
package br.uem.din.grsse.springmvc.model;

public class Pessoa {

    private Long id;
    private String nome;
    private String sobrenome;
    private String nascimento;
    private String rg;
    private String cpf;
    private String login;
    private String senha;

    //Endereco
    private String rua;
    private String numero;
    private String bairro;
    private String estado;
    private String cidade;
    private String cep;

    public Pessoa() {
        super();
    }
}
```

Recuperando Dados do Formulário

- Criando um novo método no controlador:

```
@RequestMapping("/persistePessoa")  
public String persistePessoa(Model m, Pessoa p) {  
    //faz a persistencia...  
    m.addAttribute("pessoa",p);  
    return "mensagemCadastro";  
}
```

Recuperando Dados do Formulário

- Crie uma página JSP chamada
 - mensagemCadastro.jsp
- Adicione à página:

```
<h1>Olá ${pessoa.nome} ${pessoa.sobrenome},  
obrigado por efetuar o cadastro!</h1>
```

Recuperando Dados do Formulário

Dados Pessoais

Nome:

Edson

Sobrenome:

Junior

Nascimento:

dd

RG:

CPF:

Olá Edson Junior, obrigado por efetuar o cadastro!

Criando Novos Controladores

- Para criar um novo controlador
 - Crie uma classe Java e anote-a como @Controller
 - Crie os métodos com suas rotas
- Note que não é necessário fazer nada além disso, nem informar essas alterações
- Pergunta:
 - Como o Spring MVC sabe que agora quem responde por novas rotas específicas é o Novo Controlador???

Criando Novos Controladores

```
1 package br.uem.din.grsse.springmvc.controller;
2
3 import br.uem.din.grsse.springmvc.model.Pessoa;
4 import org.springframework.stereotype.Controller;
5 import org.springframework.ui.Model;
6 import org.springframework.web.bind.annotation.RequestMapping;
7
8 @Controller
9 public class ControllerPessoa {
10
11
12
13     @RequestMapping("/persistePessoa")
14     public String adicionaPessoa(Model m, Pessoa p) {
15
16         //faz a persistencia...
17
18         System.out.println("ControllerPessoa respondendo..");
19
20         m.addAttribute("pessoa", p);
21
22         return "mensagemCadastro";
23     }
24
25
26 }
```

Tarefa

- Com base na atividade da aula passada:
 - Recupere os dados dos formulários e faça a “persistência” usando coleções

Tarefa

- Em vez de guardar os dados sobre contas bancárias, clientes, etc no Model:
 - Armazene tudo na sessão do usuário para que todos os recursos da aplicação possam ter acesso
 - Para isso use o objeto HttpSession session

Tarefa

- Crie uma tela de login para a aplicação e só permita o acesso aos demais recursos da aplicação se um usuário estiver compartilhado na sessão
 - Crie uma classe Usuário com os atributos login e senha
 - Crie um formulário JSP para login
 - Crie um controlador específico para Login
- Crie um método no controlador Login para efetuar logoff e use a instrução: `session.invalidate()` para encerrar a sessão e remover o usuário da sessão