



Matheus Felipe Basilio de Souza

Matrícula: 202301178835

1813 POLO SANTA INÊZ - BELO HORIZONTE – MG

Objetivo da Prática:

Implementação de sistema cadastral com interface Web, baseado nas tecnologias de Servlets, JPA e JEE.

- Implementar persistência com base em JPA.
- Implementar regras de negócio na plataforma JEE, através de EJBs.
- Implementar sistema cadastral Web com base em Servlets e JSPs.
- Utilizar a biblioteca Bootstrap para melhoria do design.

1º PROCEDIMENTO | CAMADAS DE PERSISTÊNCIA E CONTROLE.

Análise e Conclusão:

Como é estruturado um projeto corporativo no NetBeans?

Um projeto corporativo no NetBeans é estruturado em módulos. Cada módulo representa um componente lógico do aplicativo, como uma camada de negócios, uma camada de apresentação ou uma camada de acesso a dados. Os módulos são reunidos em um projeto principal que gerencia as dependências e a construção do aplicativo.

Qual é o papel das tecnologias JPA e EJB na criação de um aplicativo para a plataforma Web no ambiente Java?

JPA (Java Persistence API): Oferece uma interface padrão para acesso e gerenciamento de dados relacionais. Ele mapeia objetos Java para tabelas de banco de dados e vice-versa, facilitando o acesso e a persistência de dados.

EJB (Enterprise JavaBeans): Uma especificação que define componentes reutilizáveis que podem ser empregados em aplicativos corporativos distribuídos. Os EJBs oferecem serviços como gerenciamento de transações, segurança e acesso a recursos.

Como o NetBeans possibilita a melhoria de produtividade ao trabalhar com as tecnologias JPA e EJB?

- ✓ Modelos e preenchimento automático de código para classes JPA e EJB.
 - ✓ Navegação e validação de código facilitadas.
 - ✓ Integração com servidores de aplicativos para implantação e depuração.
 - ✓ Assistente de persistência para geração automática de classes de entidade JPA.
- Editor avançado com recursos como destaque de sintaxe e refatoração de código.

2º PROCEDIMENTO | INTERFACE CADASTRAL COM SERVLET E JSPS.

Análise e Conclusão:

Como opera o padrão Front Controller, e como ele é implementado em um aplicativo Web Java, na arquitetura MVC?

O Front Controller é um padrão de design que centraliza o tratamento de solicitações em um único componente, chamado de controlador. Ele gerencia o encaminhamento das solicitações, processa as requisições e coordena as ações do sistema.

No contexto da arquitetura MVC (Model-View-Controller), o Front Controller é geralmente implementado como um Servlet. Ele recebe todas as solicitações do cliente e determina qual ação realizar com base nas informações contidas na solicitação. O controlador então interage com o modelo (dados) e a visão (interface do usuário) conforme necessário, antes de enviar uma resposta de volta ao cliente.

Quais são as diferenças e semelhanças entre Servlets e JSPs?

Semelhanças: Ambos são componentes Java usados para desenvolvimento web, são executados no servidor, interagem com o cliente através do protocolo HTTP, têm acesso aos objetos de requisição e resposta, e são altamente extensíveis.

Diferenças: Servlets são predominantemente escritos em Java puro e são mais adequados para manipulação de lógica de negócios e processamento de solicitações. JSPs permitem a mistura de código Java e HTML, facilitando a criação de páginas web dinâmicas e são mais apropriados para a apresentação de dados e interação com o usuário final.

Qual é a diferença entre um redirecionamento simples e o uso do método forward, a partir do RequestDispatcher? Para que servem parâmetros e atributos nos objetos HttpRequest?

Redirecionamento simples: Envia uma resposta de redirecionamento para o navegador do cliente, instruindo-o a fazer uma nova solicitação para uma URL diferente. O navegador faz uma nova solicitação, e o cliente percebe que foi redirecionado.

Encaminhamento usando `RequestDispatcher`: Permite que o servlet encaminhe a solicitação e a resposta para outro componente no servidor, como outro servlet, JSP ou arquivo estático, sem a participação do cliente. O encaminhamento é interno e transparente para o cliente, que não percebe que está acontecendo.

Os parâmetros em objetos `HttpRequest` são dados enviados pelo cliente para o servidor como parte da solicitação HTTP. O servidor os utiliza para processar a solicitação e gerar a resposta. Eles podem incluir informações de formulários HTML ou parâmetros de URL.

Os atributos em objetos `HttpRequest` são objetos associados ao escopo da solicitação (`HttpServletRequest`) pelo servidor. São úteis para compartilhar dados entre diferentes componentes durante o processamento da solicitação. Exemplos de atributos incluem objetos de sessão ou atributos definidos pelo servlet durante o processamento da solicitação.

3º PROCEDIMENTO | MELHORANDO O DESIGN DA INTERFACE.

Análise e Conclusão:

Como o framework Bootstrap é utilizado?

O Bootstrap é usado para criar sites responsivos rapidamente, com componentes e estilos CSS e JavaScript prontos para uso. Para incluir: use links CDN no `<head>`, baixe do site oficial e adicione os arquivos ao projeto, ou instale via `npm/yarn` e importe no JavaScript.

Por que o Bootstrap garante a independência estrutural do HTML?

O Bootstrap garante a independência estrutural ao separar a apresentação (CSS) da estrutura (HTML). Suas classes predefinidas permitem aplicar estilos e comportamentos uniformes sem modificar o código HTML original.

Qual a relação entre o Bootstrap e a responsividade da página?

O Bootstrap usa um grid flexível baseado em colunas que se ajusta automaticamente ao tamanho da tela, permitindo que os layouts se adaptem a diferentes dispositivos e resoluções de forma eficiente.

