

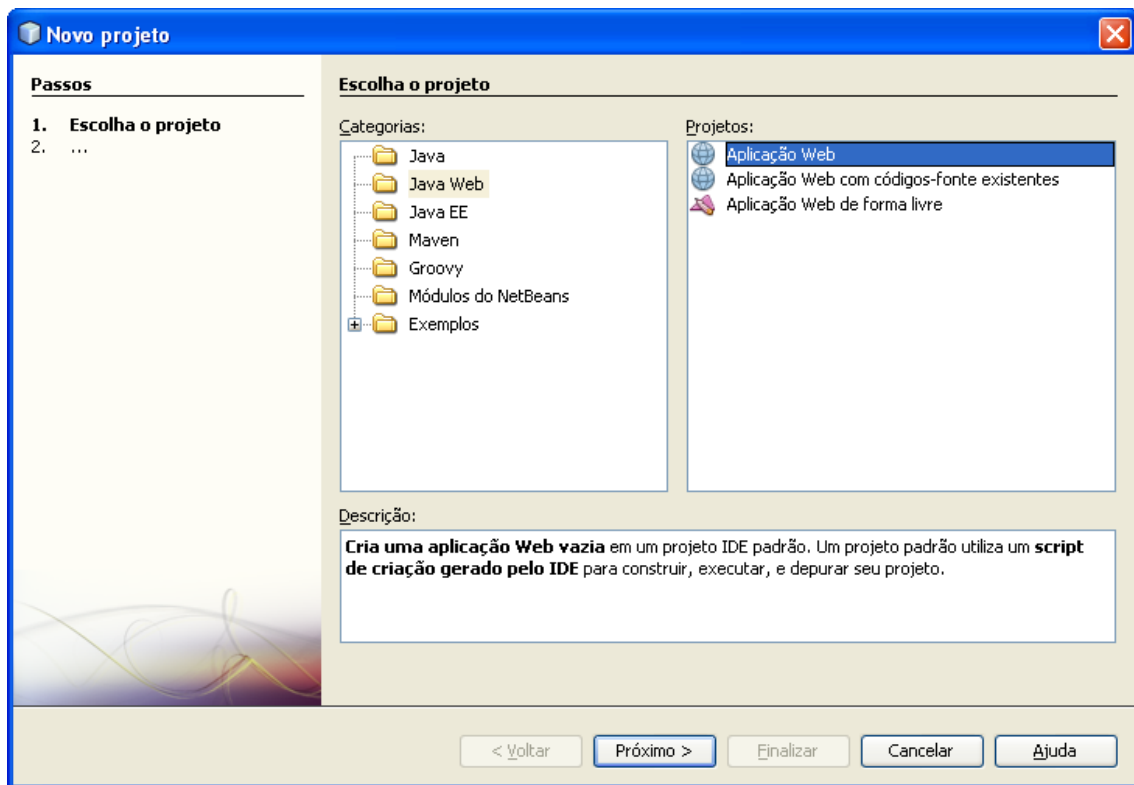
1. Criação do Projeto de Aplicação

O objetivo deste exercício é criar um novo projeto da aplicação Web voltado à administração de uma locadora de vídeos, usando as vantagens dos recursos do modelo de persistência Java.

Para este tutorial você usará formulários em páginas JavaServer Faces (JSF) para ler e modificar os dados no banco de dados, seguindo estes passos para a configuração básica da aplicação:

1. Escolha Arquivo > Novo projeto (Ctrl-Shift-N). Selecione Aplicação Web na categoria Web e clique em Próximo.
2. Digite um nome para o projeto e defina a localização dele.
3. Desmarque a opção Usar pasta dedicada, se ela estiver selecionada e clique em Próximo.
4. Defina o servidor como GlassFish e defina a versão de Java EE Java EE 5 e clique em Próximo.
5. Selecione a caixa de verificação JavaServer Faces e clique em Terminar.

(Seqüência de tela para os passos acima)



Novo de aplicações Web

Passos

1. Escolha o projeto

2. **Nome e localização**

3. Servidor e configurações

4. Frameworks

Nome e localização

Nome do projeto: WebVideoLocadora

Local do projeto: C:\pessoal\aulas\Uninove\JPA\ExemploJPA Procurar...

Pasta do projeto: C:\pessoal\aulas\Uninove\JPA\ExemploJPA\WebVideoLocadora

☐ Usar pasta dedicada para armazenar bibliotecas

Pasta de bibliotecas: Procurar...

Usuários e projetos diferentes podem compartilhar as mesmas bibliotecas de compilação (consulte a Ajuda para obter detalhes).

☒ Configurar como projeto principal

< Voltar

Próximo >

Finalizar

Cancelar

Ajuda

Novo de aplicações Web

Passos

1. Escolha o projeto

2. Nome e localização

3. **Servidor e configurações**

4. Frameworks

Servidor e configurações

Adicionar ao aplicativo corporativo: <Nenhuma>

Servidor: GlassFish v2.1 Adicionar...

☐ Usar pasta de biblioteca dedicada para arquivos JAR do servidor

Versão do Java EE: Java EE 5

Caminho do contexto: /WebVideoLocadora

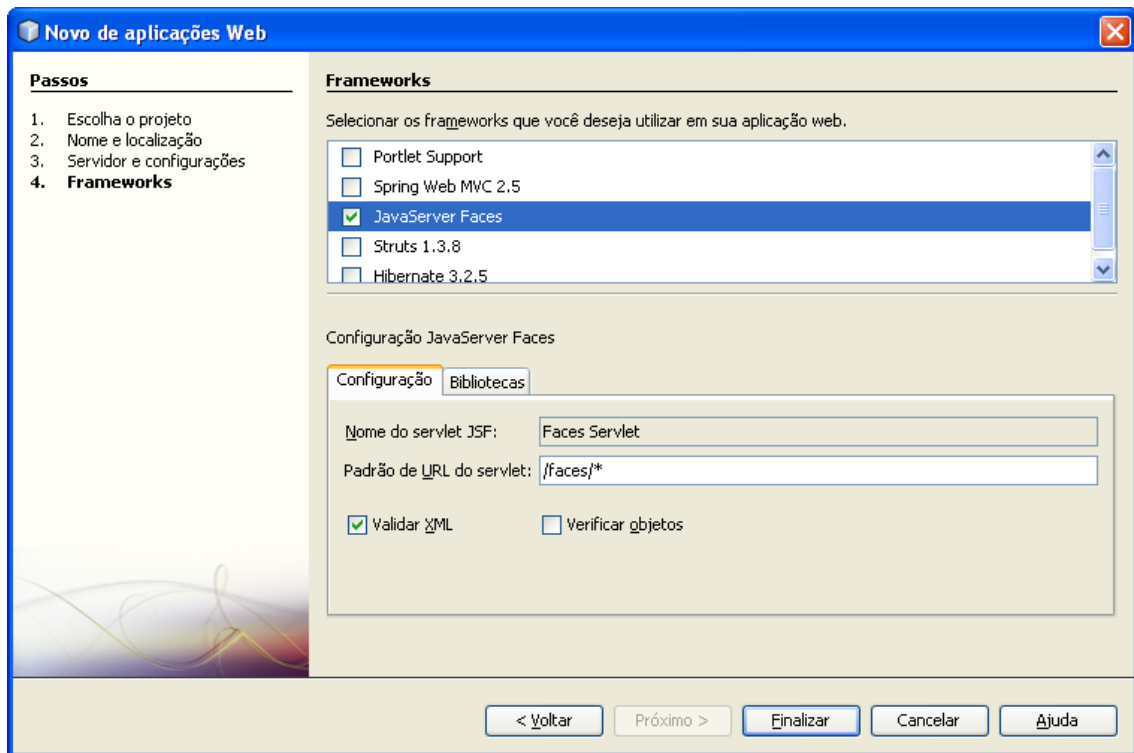
< Voltar

Próximo >

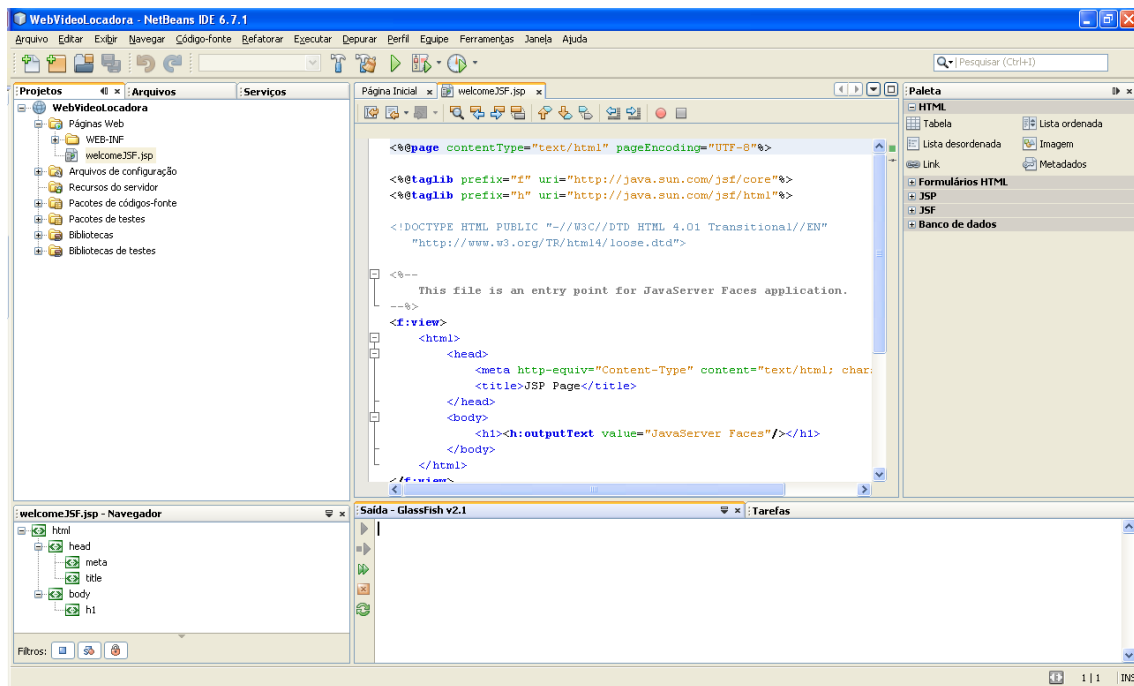
Finalizar

Cancelar

Ajuda



Ao final deste processo, a tela ficará com a seguinte aparência:



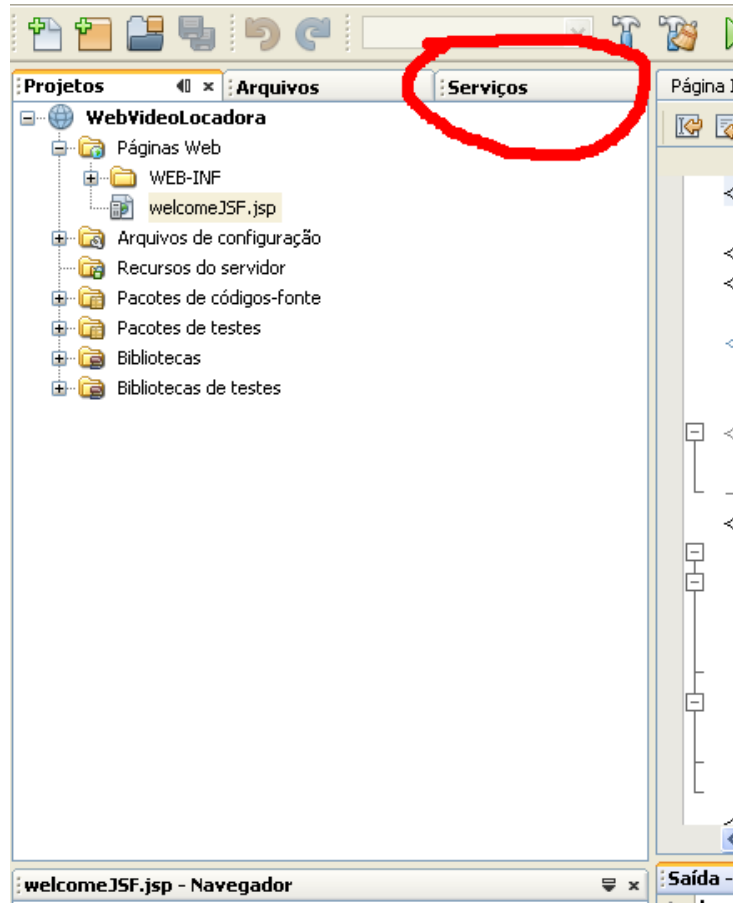
2. Criação da Base de Dados

O passo seguinte para o nosso projeto é a criação de uma base de dados, que armazenará os dados da aplicação, e o driver de conexão a essa base. O banco de dados será criado

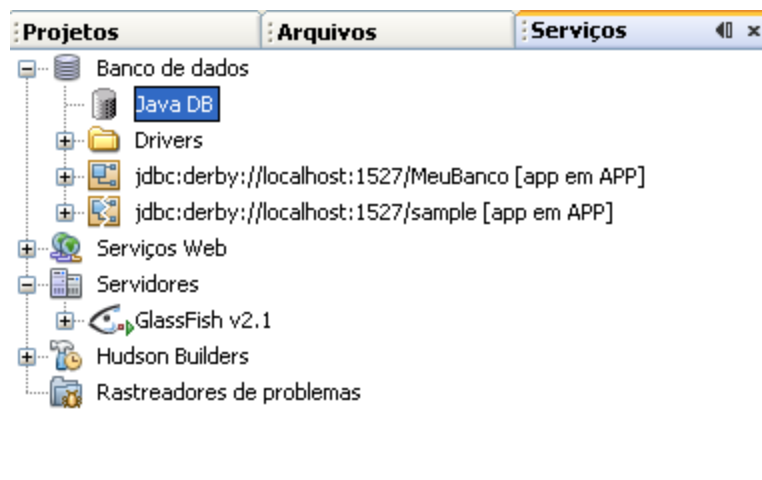
dentro do Java DB, que é uma base para execução de testes durante o desenvolvimento de uma aplicação, e quer serve para o nosso exemplo.

Neste momento, criaremos um banco vazio no Java DB, sem tabelas, que serão criadas mais adiante, durante a implantação do projeto.

Para isso, selecione a aba Serviços no netbeans.

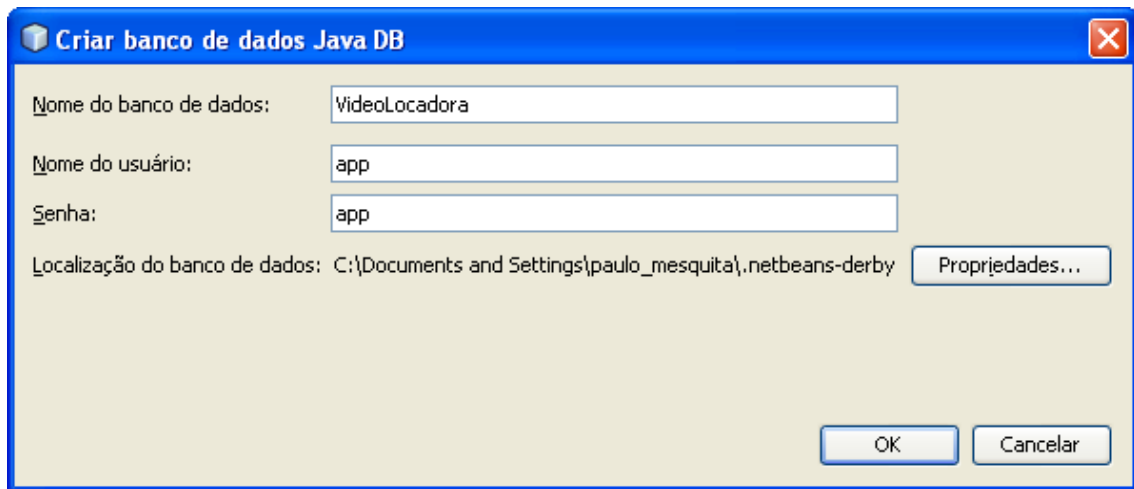


A tela ficará com a seguinte aparência:

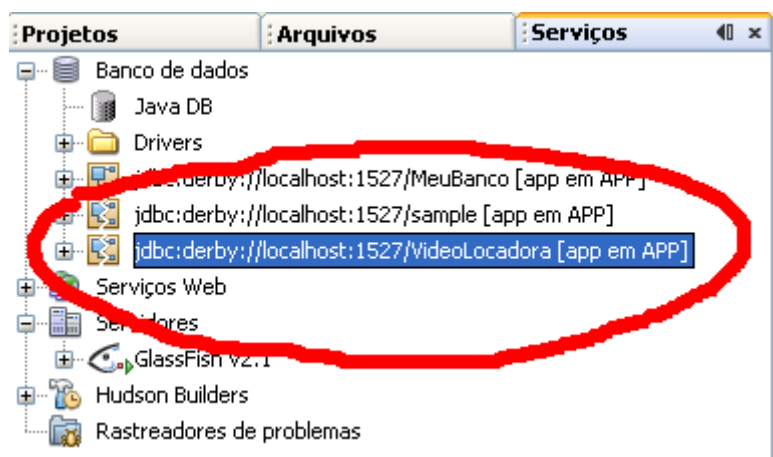


Para criar o banco de dados, execute os seguintes passos:

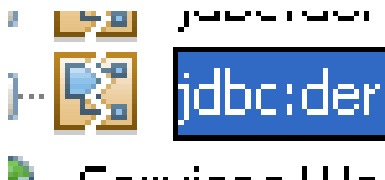
1. Clique em Java DB como o botão direito do mouse, e selecione Criar banco de dados.
2. Na janela Criar banco de dados Java DB, especifique um nome para o banco de dados, e defina o nome do usuário do banco e a senha como app, e clique em OK



Após a criação do banco, a tela criará uma nova ponte para o banco de dados criado.



Neste momento, ainda não existe uma conexão ao bando, como atestado pela linha branca que quebra o ícone da conexão.



Para usar o banco de dados, é necessário conectar-se a ele. Para isso, com o botão direito do mouse, clique na linha correspondente ao banco que foi criado, e selecione Conectar.



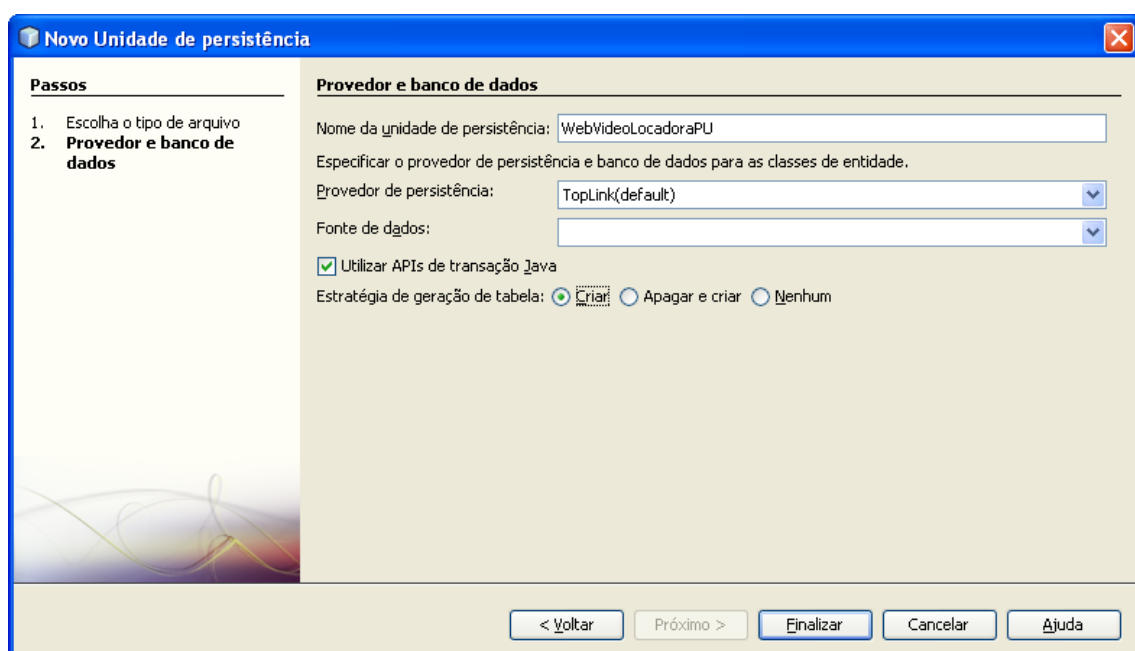
Agora, o banco de dados criado já pode ser usado pela aplicação. Agora, é necessário criar uma unidade de persistência.

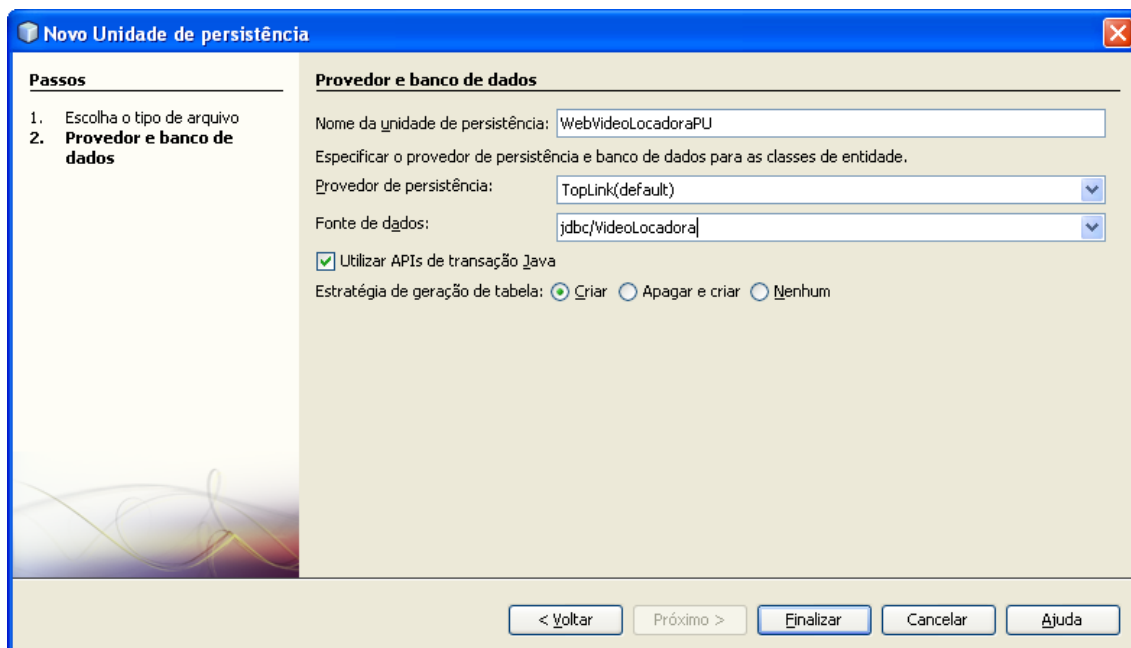
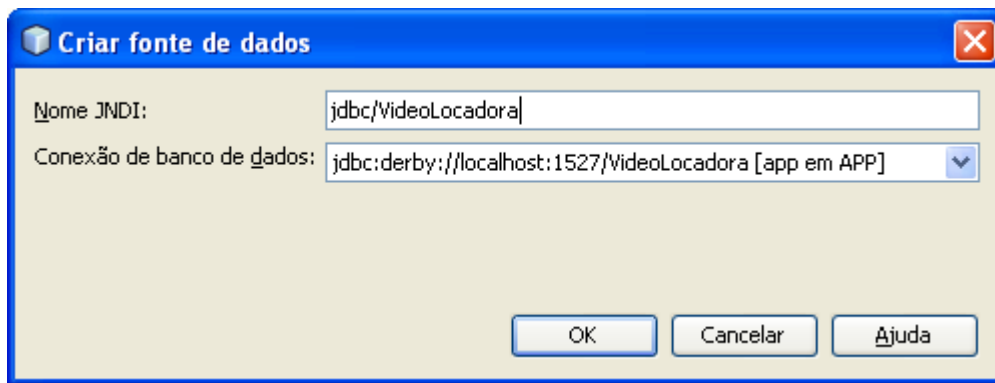
3. Criação da unidade de persistência

Você cria a unidade de persistência, definindo suas propriedades no arquivo `persistence.xml`, que fica dentro do módulo Web. Depois de especificar o nome da unidade de persistência, você especificará o provedor de persistência que o contêiner usa para gerenciar instâncias de entidade. Você também precisa especificar as fontes de dados e uma estratégia de geração de tabelas. Você criará a unidade de persistência, usando o assistente para Nova unidade de persistência.

1. Clique com o botão direito do mouse no nome do projeto WebVideoLocadora na aba Projetos e escolha Novo > Unidade de persistência para abrir o assistente Novo unidade de persistência.
2. Preserve o nome da unidade de persistência padrão.
3. Para o Provedor de persistência, use `TopLink (default)`. Ele identifica o provedor padrão como `TopLink Essential.jar`, que contém as bibliotecas da persistência Java.
4. Para Fonte de dados, selecione Nova fonte de dados.
5. Na janela Criar fonte de dados, especifique o Nome JNDI do banco, e a conexão para o mesmo e clique Ok
6. Verifique se a unidade de persistência está usando a API de transação de Java e se a Estratégia de geração de tabela está definida como Criar, de modo que as tabelas baseadas nas classes de entidade sejam criadas quando a aplicação for implantada.
7. Clique em Terminar.

(Sequência de telas para este procedimento)





Ao clicar em Terminar, o IDE cria `persistence.xml` e o abre no Editor de código-fonte na exibição Design. Você pode clicar em XML na barra de ferramentas do editor de código-fonte para ver o XML de `persistence.xml`. Esse arquivo contém as informações que o contêiner Java EE 5 precisa para gerenciar as entidades e a persistência do aplicativo.

4. Criação das classes de entidade

Agora serão criadas duas classes de entidade, `Cliente.java` e `Video.java`, que representam as tabelas no banco de dados relacional que é usado. Em seguida, você irá definir alguns campos nas classes para representar os dados, usando anotações para fornecer informações para o contêiner sobre os campos, tais como informações de mapeamento de objeto relacional.

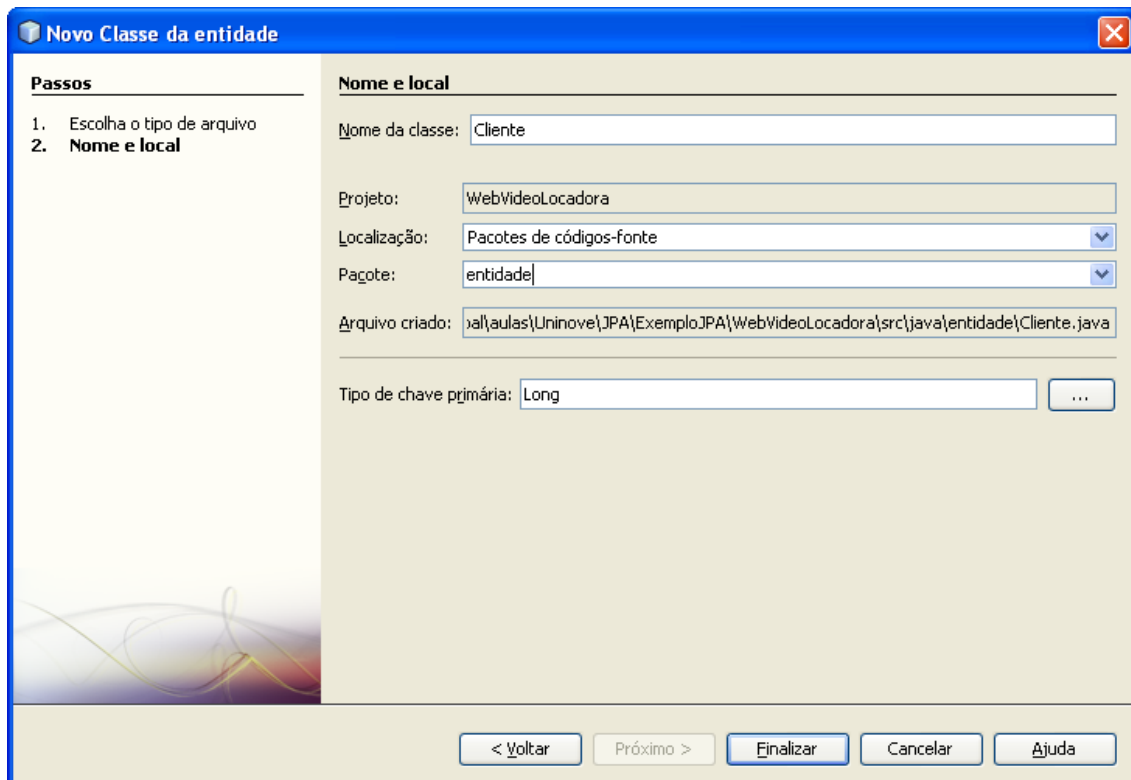
Quando você cria a classe de entidade, o IDE adiciona a anotação `@Entity` para definir a classe como uma classe de entidade. Depois de criar a classe, você criará os campos na classe para representar os dados que deseja em sua tabela, e usará anotações para fornecer informações adicionais sobre alguns dos campos.

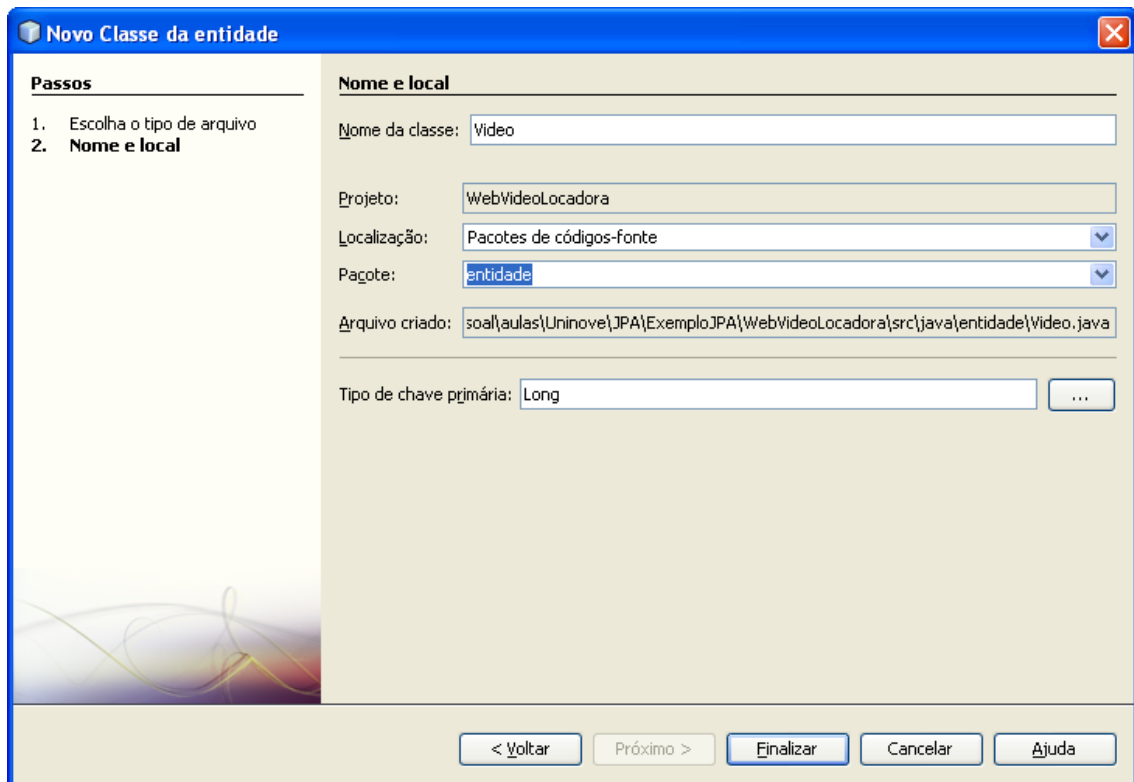
Cada classe de entidade deve ter uma chave primária. Quando você cria a classe de entidade, o IDE adiciona a anotação @Id para declarar o campo a ser usado como chave primária. O IDE também adiciona a anotação @Generated para especificar a estratégia de geração de chave para o Id primário.

Para criar as classes de entidade, faça o seguinte:

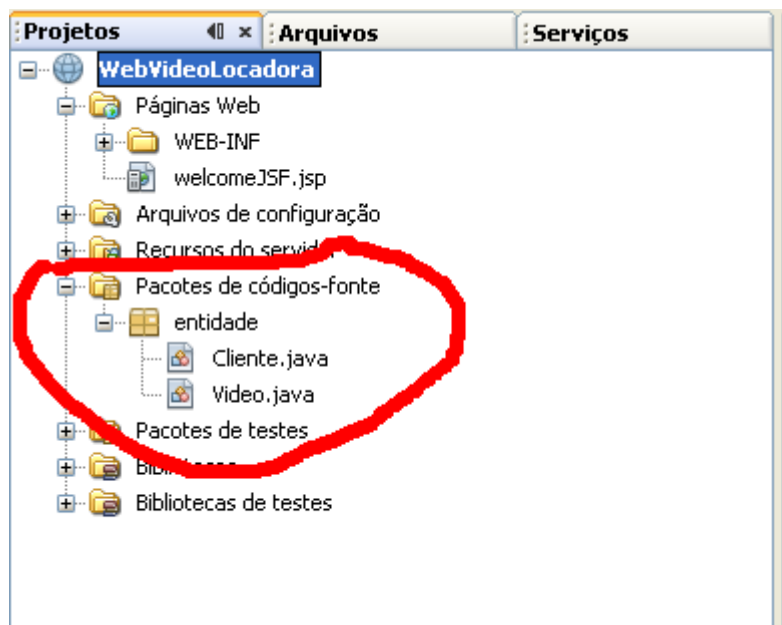
1. Clique com o botão direito do mouse no nó do projeto WebVideoLocadora e escolha Novo > Classe da Entidade.
2. Digite `Cliente` para o nome da classe, digite `entidade` para o pacote e deixe o Tipo de chave primária como `Long`. Clique em Terminar.
3. Repita as etapas acima para criar a classe `Video` no pacote de códigos-fonte `entidade`.

(Sequência de telas para estas operações)





Quando você clica em Terminar, as novas classes de entidade se abrem no editor. Se expandir o nó entidade sob Pacotes de códigos-fonte na janela Projetos, você pode ver que as classes de entidade `Cliente` e `Video` estão lá.

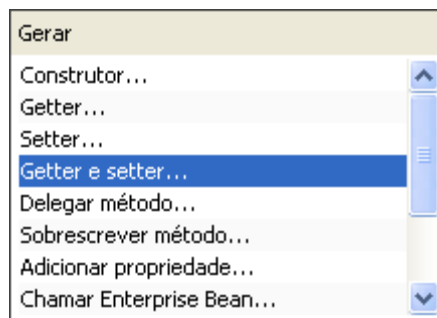


Para definição dos campos na Classe `Cliente`, abra classe `Cliente` no editor e faça o seguinte:

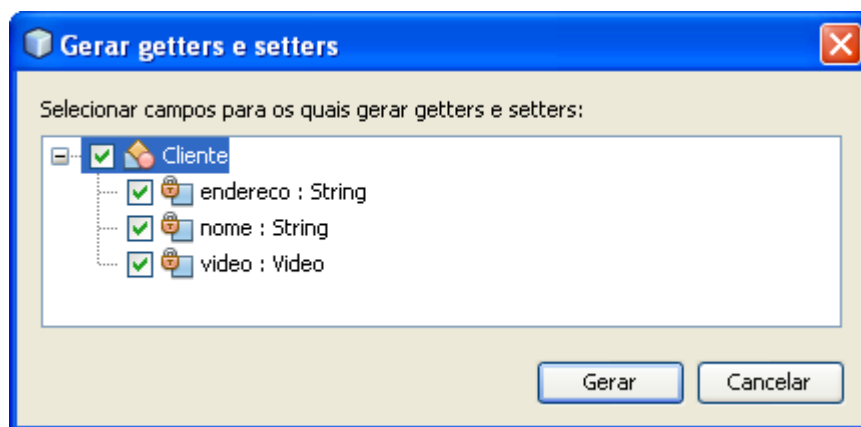
1. Adicione as seguintes declarações de campo à classe

```
private static final long serialVersionUID = 1L;
@Id
@GeneratedValue(strategy = GenerationType.AUTO)
private Long id;
private String nome;
private String endereco;
private Video video;
```

2. Clique com o botão direito do mouse no Editor de código-fonte e escolha Inserir código (Alt-Insert) e, em seguida, selecione Getter e Setter para abrir a caixa de diálogo Gerar getters e setters.



3. Na caixa de diálogo Gerar getters e setters, selecione todos os campos e clique em Gerar.



4. Agora você deseja alterar o nome das colunas que serão criadas na tabela Cliente de forma que a coluna seja chame clienteNome em vez de nome. Você pode usar anotações para especificar o nome da coluna gerada, adicionando a anotação seguinte acima da declaração do campo nome:

```
@Column(name="clienteNome")
private String nome;
```

5. A coluna video em nossa tabela Cliente também deve ter uma relação muitos para um. Você pode fazer isso usando anotações, adicionando a anotação seguinte acima da declaração video.

```
@ManyToOne
private Video video;
```

6. Pressione Alt-Shift-I para gerar as instruções de importação necessárias para a classe.
7. Salve as alterações

Para definir os campos na Classe Vídeo, abra a classe video no editor e faça o seguinte:

1. Adicione as seguintes declarações de campo à classe:

```
public class Video implements Serializable {
    private static final long serialVersionUID = 1L;
    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private Long id;
    private String nome;
    private String genero;
    private Collection<Cliente> clientes;
```

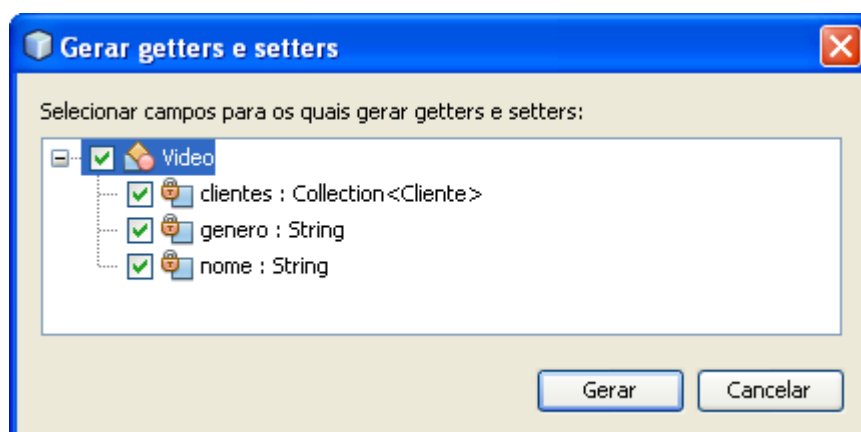
2. Adicione a seguinte anotação acima da declaração nome para alterar o nome da coluna gerada:

```
@Column(name="videoNome")
private String nome;
```

3. Adicione a notação seguinte acima da coleção clientes para especificar uma relação Um para muitos para a entrada

```
@OneToMany(mappedBy="video")
private Collection<Cliente> clientes;
```

4. Clique com o botão direito do mouse no editor e escolha Inserir código (Alt-Insert) e, em seguida, selecione Getter e setter
5. Na caixa de diálogo Gerar getters e setters, selecione todos os campos e clique em Gerar.

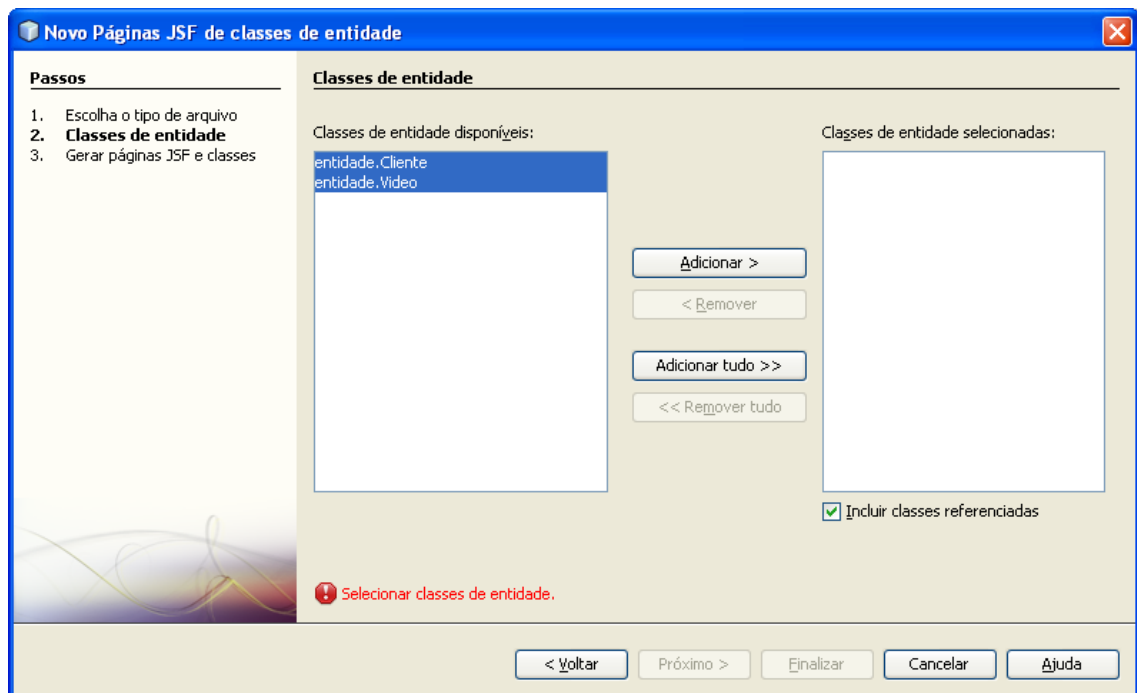


6. Pressione Alt-Shift-I para gerar quaisquer instruções de importação ausentes.
7. Salve as alterações.

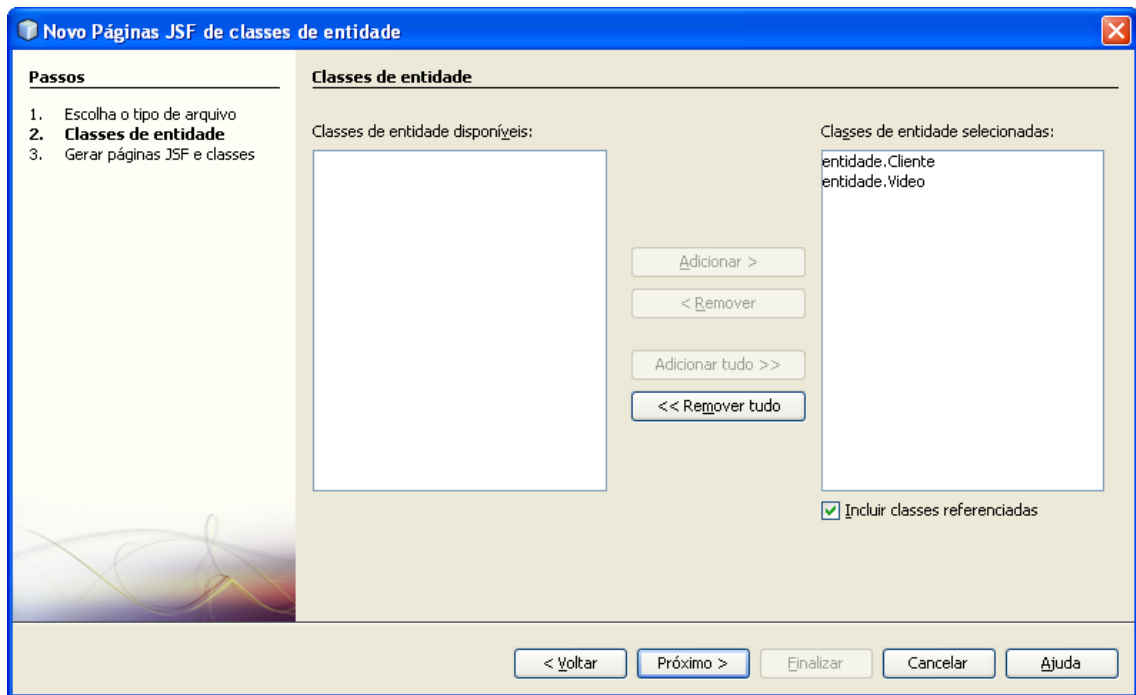
5. Criação de um projeto web para teste

Agora você precisa criar algumas páginas da Web simples para ver se as tabelas do banco de dados foram criadas e se você pode adicionar dados. Você adicionará páginas Java Server Faces (JSF) à aplicação e usará as JSF Pages do assistente para Classe de entidade para criar rapidamente uma interface da Web simples.

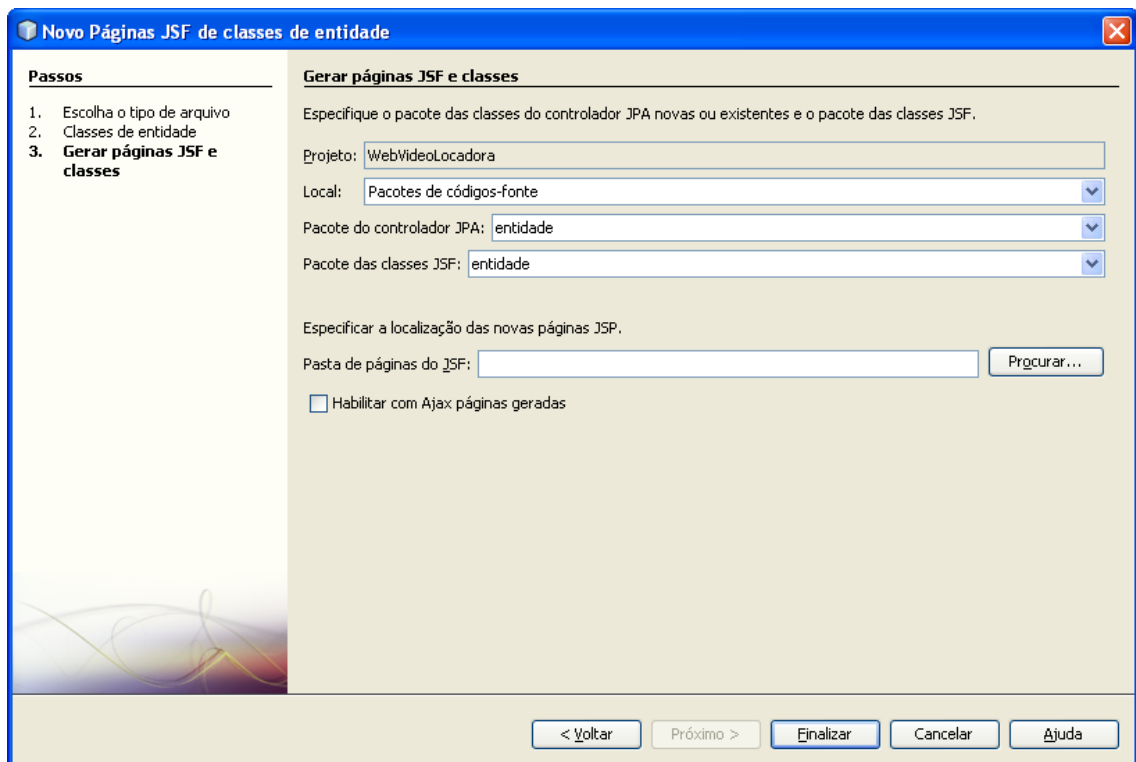
1. Escolha Arquivo > Novo no menu principal. Selecione JSF Pages em Classe de entidade e clique em Próximo.



2. Nas Novas páginas JSF do assistente para Classe de entidade, clique em Adicionar tudo para selecionar duas classes de entidade e clique em Próximo.



3. Deixe o campo de texto Pasta de páginas JSF vazio para salvar os arquivos JSF na localização padrão.



4. Especifique entidade como o pacote das classes geradas e clique em Terminar.

Quando você clica em Terminar, o IDE gera os arquivos JavaServer Faces necessários de forma que você possa executar e testar a WebVideoLocadora.

6. Testando a aplicação

Para testar a aplicação, inicialize o servidor de aplicações e o Java DB, e com um clique com o botão direito do mouse no nó do projeto WebVideoLocadora, escolha Executar projeto.

