

CRUD JAVA: INSERIR, EXCLUIR, ATUALIZAR, LISTAR COM STATEMENT

MARÇO 28, 2014 MARCOS

Conteúdo [Esconder]

- 1 Desenvolvimento Simples Inserir, Excluir, Atualizar, Listar com a Classe Statement em MySQL no Netbeans
- 2 Vídeo de aprendizagem
- 3 Inserir, Excluir, Atualizar e Listar em Banco em MySQL
 - 3.1 Siga os Passos
 - 3.1.1 Criação de um Banco de Dados e Tabela Aluno
 - 3.1.2 Criação da Aplicação JAVA no Netbeans
 - 3.2 Evento do Botão Inserir
 - 3.3 Evento do Botão Atualizar
 - 3.4 Evento do Botão Excluir
- 4 Exercícios
- 5 Referências

Desenvolvimento Simples Inserir, Excluir, Atualizar, Listar com a Classe Statement em MySQL no Netbeans

Nesta Pequena Postagem você vai aprender de maneira muito básica e simples como criar uma aplicação **CRUD JAVA** (Create, Read, Update e Delete) no Netbeans para atender tarefas simples e rápidas como inserir, atualizar, excluir e listar banco de dados em MySQL com a classe [Statement](#) em uma tabela chamada aluno. Esta pequena solução foi feita para aqueles que precisam rapidamente aprender a realizar estas tarefas de maneira simples e objetiva. Faça o teste! Considere isto um

desafio. Você vai aprender e gostar.



Caso queira uma orientação para um aprendizado JAVA basta seguir esta orientação de aprenda JAVA de maneira simples

Vídeo de aprendizagem

Vídeos para aplicação CRUD JAVA com Statement

Inserir, Excluir, Atualizar e Listar em Banco em MySQL


Clique e saiba o que é MySQL

Clique e saiba o que é JAVA

Atenção: Siga os passos abaixo. Brinque e aprenderá. Tente fazer o que é solicitado. Muita atenção aos comentários. É importante faça para ver o que acontece. Irá descobrir algumas coisas por si só. Considere isto um DESAFIO. Perceba e já estará aprendendo muito. Pode ter certeza! Você pode aprender muito

mais do que pensa! 

Siga os Passos

DICA: Encapsule (– / +) os tópicos à medida que for aprendendo. Isto vai facilitar seu aprendizado. 

Criação de um Banco de Dados e Tabela Aluno

Passos para criar a tabela Aluno em um Banco de Dados Escola

Caso precise entender mais sobre este tópico você pode acessar: [Criar Conexão MySQL no Netbeans](#).

O MySQL deve estar instalado em sua máquina

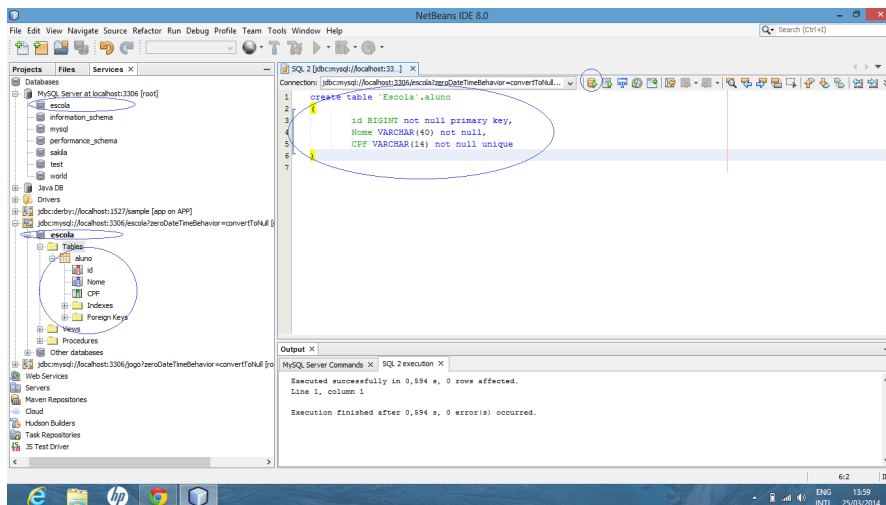
Será necessário criar a tabela em MySQL chamada [Aluno](#). Para isto, você pode seguir os passos apresentados abaixo.

1. Acesse, no Netbeans, a janela [Serviços \(Services\)](#) e clique com o botão direito do mouse sobre [Banco de Dados \(DataBases\)](#) e registre o servidor MySQL.
2. Após aparecer [Servidor MySQL em localhost \(Server MySQL at localhost\)](#), clique com o botão direito do mouse sobre ele, Clique em [conectar \(Connect\)](#) e crie um banco de dados ([Create Database](#)) chamado [Escola](#).
3. Com o botão direito do mouse sobre [Escola](#) e faça a conexão ([Connect](#)) sem senha e uma conexão JDBC MySQL aparecerá.
4. Clique com o botão direito do mouse sobre [jdbc:mysql://localhost:3306/escola?...](#) e faça a conexão. Também, com o botão direito do mouse sobre [Escola -> Tabelas](#), acesse [Executar Comando... \(Execute Command ...\)](#) para que apareça um editor [Comando SQL](#) em branco.

5. Digite no editor (Você pode usar **Ctrl-C** **Ctrl-V**), o script de criação da tabela **Aluno**, mostrado abaixo, e rode-o com o botão **Executar SQL**. A tabela **Aluno** será criada.

```
1 create table `Escola`.aluno
2 (
3     id BIGINT not null primary key,
4     Nome VARCHAR(40) not null,
5     CPF VARCHAR(14) not null unique
6 )
```

6. Resultado Final

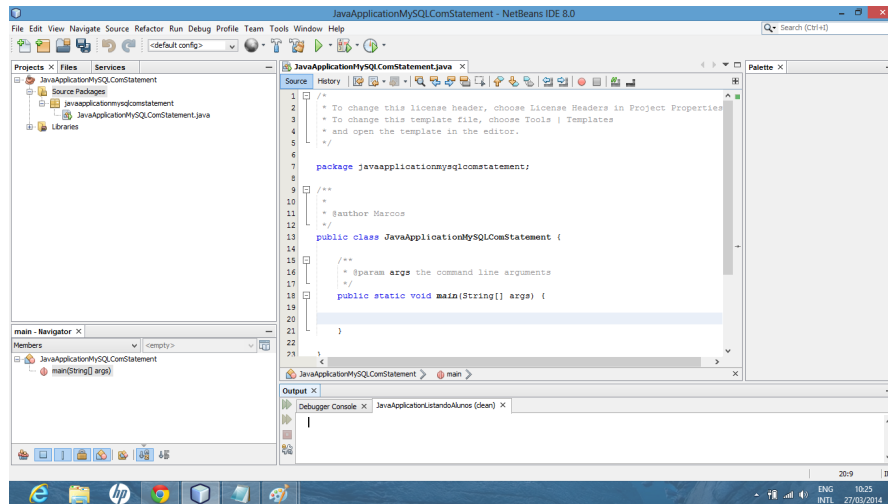


Criação da Aplicação JAVA no Netbeans

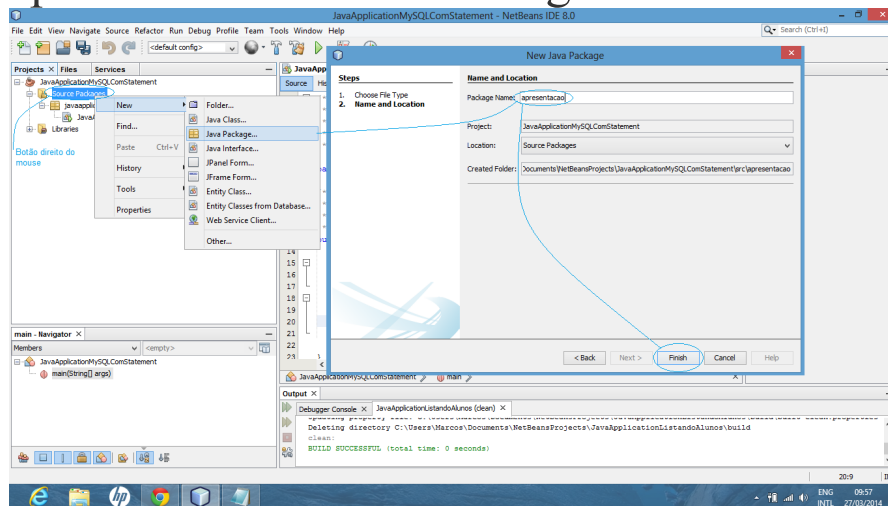
Como criar uma Aplicação JAVA CRUD no Netbeans

1. Sobre **Arquivo (File) -> Novo Projeto... (New Project...)** selecione **Java -> Aplicação Java (Java Application)** e clique em **Próximo (Next)**.
2. Em **Nome do Projeto (Project Name)** coloque o nome, ou outro que deseje, **JavaApplicationMySQLComStatement** e clique em **Próximo (Next)**. Resultado

abaixo:

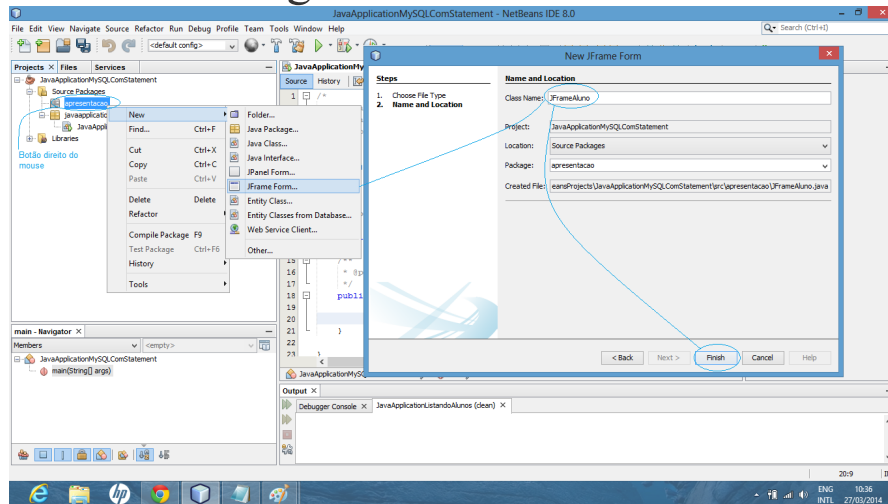


3. Crie o pacote **apresentação** como apontado na imagem abaixo:

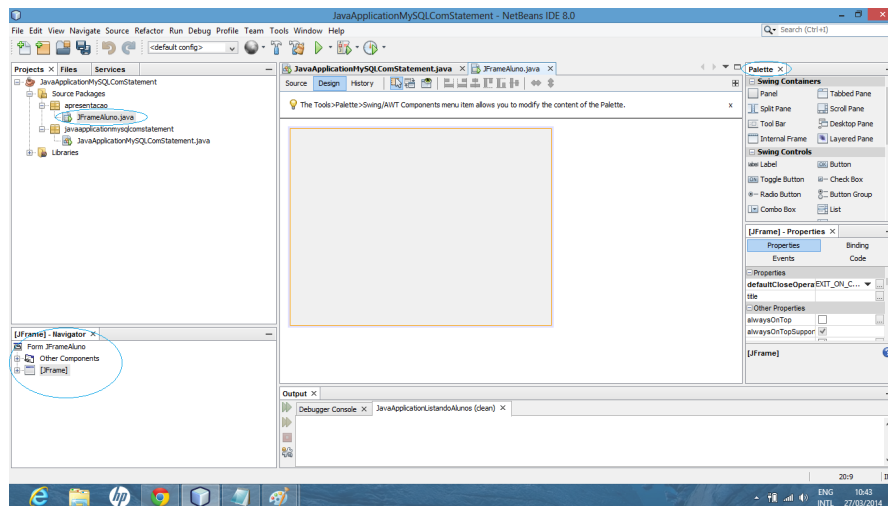


4. Crie um **JFrame** chamado **JFrameAluno** seguindo

a imagem abaixo:

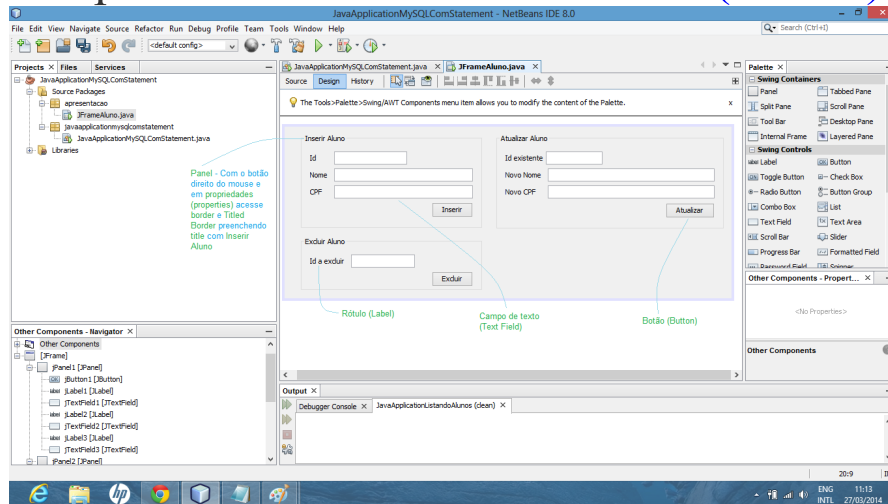


5. Resultado

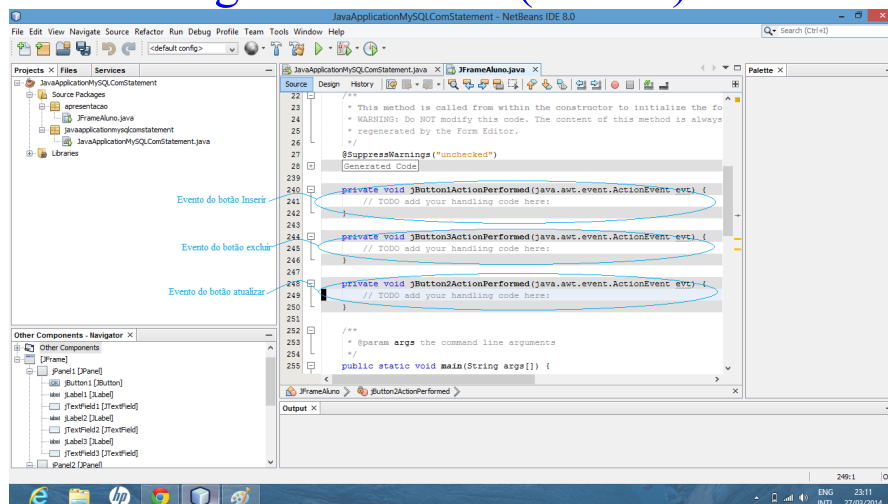


6. Crie os componentes, no **JFrame**, como apresentado na imagem a seguir. Esteja atento aos detalhes de quais são estes

componentes na Paleta (Pallette).



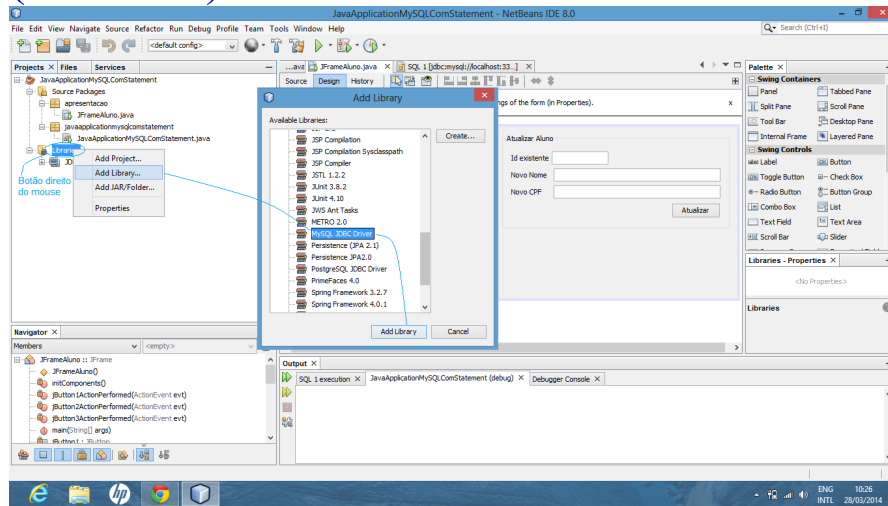
7. Gere os eventos de cada botão com duplo clique dobre eles ou clicando com o botão direito do mouse e acessando **Eventos (Events)** -> **Ação (Action)** -> **actionPerformed**. Veja os eventos na guia de **código fonte (Source)** abaixo



8. Adicione o **Driver JDBC MySQL (MySQL JDBC Driver)** em **Bibliotecas**

(Libraries).

Abaixo:



9. Digite estes códigos para o evento de cada botão no JFrame:

FIQUE MUITO ATENTO AOS
COMENTÁRIOS PARA NÃO TER
PROBLEMAS

Evento do Botão Inserir

ATENÇÃO: Renomeie os componentes (Botão direto do mouse sobre o componente no JFrame e **Alterar Nome da Variável...** (Change Variable Name...)) do painel **Inserir Aluno** conforme indicado no código abaixo:

Exemplo **jTextField1** para **jTextFieldId**, **jTextField2** para **jTextFieldNome**, **jTextField3** para **jTextFieldCPF**

```
1      try {
2          //Registra JDBC driver
3          Class.forName("com.mysql.jdbc.Driver");
4
5          //Abrindo a conexão: ATENÇÃO OS DOIS PARÂMETROS V
6          Connection conn = DriverManager.getConnection("jd
7
8          //Executa a query de inserção
9          java.sql.Statement st = conn.createStatement();
10         st.executeUpdate("INSERT INTO aluno (id,nome,cpf)
11             + this.jTextFieldId.getText() + ','
```



```

12         + this.jTextFieldNome.getText() + "',''"
13         + this.jTextFieldCPF.getText() + "')");
14
15         JOptionPane.showMessageDialog(rootPane, "Aluno in
16     } catch (SQLException | ClassNotFoundException e) {
17         JOptionPane.showMessageDialog(rootPane, e);
18     } //Fim try

```

Evento do Botão Atualizar

Esteja atento à necessidade de mudança o nome das variáveis como no caso acima.

```

1     try {
2         //Registra JDBC driver
3         Class.forName("com.mysql.jdbc.Driver");
4
5         //Abrindo a conexão
6         Connection conn = DriverManager.getConnection("jd
7
8         //Executa a query de atualização
9         java.sql.Statement st = conn.createStatement();
10        st.executeUpdate("UPDATE aluno SET nome='"
11            + this.jTextFieldNovoNome.getText() + "','
12            + this.jTextFieldNovoCPF.getText()
13            + "' WHERE id='" + this.jTextFieldIdExist
14        JOptionPane.showMessageDialog(rootPane, "Aluno at
15    } catch (SQLException | ClassNotFoundException e) {
16        JOptionPane.showMessageDialog(rootPane, e);
17    } //Fim try

```

Evento do Botão Excluir

Esteja atento à necessidade de mudança o nome das variáveis como no caso acima.

```

1     try {
2         //Registra JDBC driver
3         Class.forName("com.mysql.jdbc.Driver");
4
5         //Abrindo a conexão
6         Connection conn = DriverManager.getConnection("jd
7
8         //Executa a query de exclusão
9         java.sql.Statement st = conn.createStatement();
10        st.executeUpdate("DELETE FROM Aluno WHERE id='" +
11        JOptionPane.showMessageDialog(rootPane, "Aluno ex
12    } catch (SQLException | ClassNotFoundException e) {
13        JOptionPane.showMessageDialog(rootPane, e);
14    } //Fim try

```