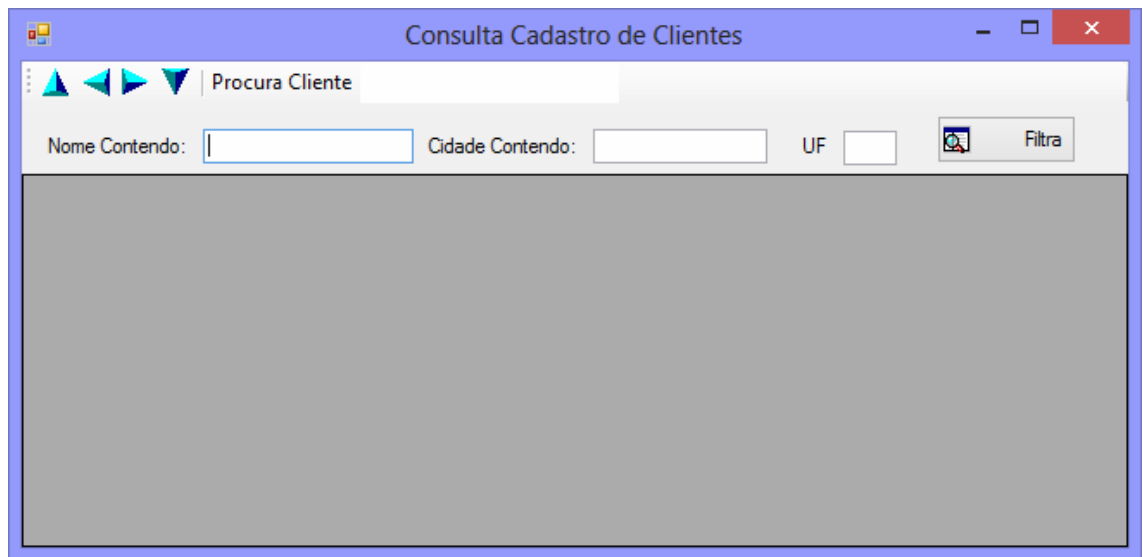


Exercícios de laboratório

Laboratório 1

A – Abra o projeto ConsultaClientes fornecido no kit de exercícios.



Nesta tela vai permitir consultas à tabela CLIENTES do banco de dados PEDIDOS. Os parâmetros de consulta são:

- Campo NOME contendo o dado digitado no TextBox correspondente ao nome.
- Campo CIDADE contendo o dado digitado no TextBox correspondente a cidade.
- Campo ESTADO começando com dado digitado no TextBox correspondente a UF.

O comando SELECT já está escrito no evento Click do botão Filtro, mas comentado, precisamos fazê-lo funcionar:

```

private void btnFiltrar_Click(object sender, EventArgs e)
{
    //      @"SELECT CODCLI,NOME,ENDERECO,BAIRRO,CIDADE,ESTADO,CEP,
    //      FONE1,FAX,E_MAIL,CNPJ,INSCRICAO
    //      FROM CLIENTES
    //      WHERE NOME LIKE ? AND CIDADE LIKE ? AND ESTADO LIKE ?
    //      ORDER BY NOME";
}

```

ROTEIRO

1. Criar App.config definindo o string de conexão OleDb
2. Criar a classe Conexoes contendo
 - 2.1. propriedadeConnectionString
 - 2.2. método GetConnection
3. No form, declarar 3 variáveis para toda o form
 - 2.1. OleDbConnection
 - 2.2. DataTable
 - 2.3. BindingSource
4. Concluir o botão filtra, fazendo o SELECT funcionar e ser exibido no DataGridView chamado dgvClientes.
5. Fazer o evento Load do formulário para forçar o Click no botão Filtra e configurar os títulos de colunas do DataGridView
6. Fazer os eventos Click dos botões de movimentação
7. Criar método Locate para fazer pesquisa incremental
8. Evento TextChanged de tbxProcura para concluir a pesquisa incremental

Laboratório e

A – Abra o projeto ConsultaEmpregados fornecido no kit de exercícios.

Consulta Cadastro de Empregados

Procura Nome:

Nome Contendo: Depto começando com: Cargo começando com: Salário Entre Admitido entre: Filtro

Código:	Nome:	Dt. Admissão:	Salário:	Departamento	Cargo:
*					

Nesta tela vai permitir consultas à tabela EMPREGADOS do banco de dados PEDIDOS. Os parâmetros de consulta são:

- Campo NOME contendo o dado digitado no TextBox correspondente ao nome.
- Campo DEPTO começando com o dado digitado no TextBox correspondente ao departamento.
- Campo CARGO começando com o dado digitado no TextBox correspondente ao cargo.
- Campo SALARIO com valor entre os dados contidos nos NumericUpDown correspondentes aos salários.
 - **Neste caso a propriedade do NumericUpDown é Value e não Text.**
 - **Não vamos concatenar o "%" porque a instrução SELECT usa BWTEWEEN e não LIKE.**
- Campo DATA_ADMISSAO com valor entre os dados contidos nos DateTimePicker correspondentes às datas.
 - **Neste caso a propriedade do DateTimePicker é Value e não Text.**
 - **Não vamos concatenar o "%" porque a instrução SELECT usa BWTEWEEN e não LIKE.**

O comando SELECT já está escrito no evento Click do botão Filtra, mas comentado, precisamos fazê-lo funcionar:

```
private void btnFiltra_Click(object sender, EventArgs e)
{
    @"SELECT E.CODFUN, E.NOME, E.DATA_ADMISSAO,
    E.SALARIO, D.DEPTO, C.CARGO, E.OBS
    FROM EMPREGADOS E
    JOIN TABELADEP D ON E.COD_DEPTO = D.COD_DEPTO
    JOIN TABELACAR C ON E.COD_CARGO = C.COD_CARGO
    WHERE E.NOME LIKE ? AND D.DEPTO LIKE ?
    AND C.CARGO LIKE ? AND E.SALARIO BETWEEN ? AND ?
    AND E.DATA_ADMISSAO BETWEEN ? AND ?
    ORDER BY E.NOME";
}
```

ROTEIRO

Obs.: Apesar do roteiro sugerir as classes OleDb, o procedimento será exatamente o mesmo caso você queira fazer usando as classes Sql, basta substituir as interrogações da instrução SELECT por nomes começando por @.

1. Criar App.config definindo o string de conexão OleDb
2. Criar a classe Conexoes contendo
 - 2.1. propriedade ConnectionString
 - 2.2. método GetConnection
3. No form, declarar 3 variáveis para toda o form
 - 2.1. OleDbConnection
 - 2.2. DataTable
 - 2.3. BindingSource
4. Concluir o botão filtra, fazendo o SELECT funcionar e ser exibido no DataGridView chamado dgvEmpregados.
5. Fazer o evento Load do formulário para forçar o Click no botão Filtra e configurar os títulos de colunas do DataGridView
6. Fazer os eventos Click dos botões de movimentação
7. Criar método Locate para fazer pesquisa incremental
8. Evento TextChanged de tbxProcura para concluir a pesquisa incremental