

DAL

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.SqlClient;

namespace ParaCasa1
{
    class LivroDAL
    {
        private static string strConexao = "Data Source=DESKTOP-3LVM070\\MSSQLSERVER1;Initial Catalog=LivrariaDB;Integrated Security=True";
        private static SqlConnection conn = new SqlConnection(strConexao);
        private static SqlCommand strSQL;
        private static SqlDataReader result;

        public static void conecta()
        {
            try
            {
                conn.Open();
            }
            catch (Exception)
            {
                Erro.setMsg("Problemas ao se conectar ao Banco de Dados");
            }
        }

        public static void desconecta()
        {
            conn.Close();
        }
    }
}
```

```
}
```

```
public static void inseriUmLivro(Livro umlivro)
```

```
{
```

```
    string aux = "INSERT INTO TabLivro(codigo, titulo, autor, editora, ano) VALUES  
(@codigo, @titulo, @autor, @editora, @ano)";
```

```
    strSQL = new SqlCommand(aux, conn);
```

```
    strSQL.Parameters.AddWithValue("@codigo", int.Parse(umlivro.getCodigo()));
```

```
    strSQL.Parameters.AddWithValue("@titulo", umlivro.getTitulo());
```

```
    strSQL.Parameters.AddWithValue("@autor", umlivro.getAutor());
```

```
    strSQL.Parameters.AddWithValue("@editora", umlivro.getEditora());
```

```
    strSQL.Parameters.AddWithValue("@ano", int.Parse(umlivro.getAno()));
```

```
    Erro.setErro(false);
```

```
    try
```

```
    {
```

```
        strSQL.ExecuteNonQuery();
```

```
    }
```

```
    catch (Exception)
```

```
    {
```

```
        Erro.setMsg("Chave Duplicada!");
```

```
    }
```

```
}
```

```
public static void excluiUmLivro(Livro umlivro)
```

```
{
```

```
    string aux = "DELETE FROM TabLivro WHERE codigo = @codigo";
```

```
    strSQL = new SqlCommand(aux, conn);
```

```
    strSQL.Parameters.AddWithValue("@codigo", int.Parse(umlivro.getCodigo()));
```

```
    strSQL.ExecuteNonQuery();
```

```
}
```

```
public static void atualizaUmLivro(Livro umlivro)
```

```

{
    string aux = "UPDATE TabLivro SET titulo = @titulo, autor = @autor, editora = @editora,
ano = @ano WHERE codigo = @codigo";

    strSQL = new SqlCommand(aux, conn);

    strSQL.Parameters.AddWithValue("@titulo", umlivro.getTitulo());
    strSQL.Parameters.AddWithValue("@autor", umlivro.getAutor());
    strSQL.Parameters.AddWithValue("@editora", umlivro.getEditora());
    strSQL.Parameters.AddWithValue("@ano", int.Parse(umlivro.getAno()));
    strSQL.Parameters.AddWithValue("@codigo", int.Parse(umlivro.getCodigo()));

    strSQL.ExecuteNonQuery();
}

```

public static void consultaUmLivro(Livro umlivro)

```

{
    string aux = "SELECT * FROM TabLivro WHERE codigo = @codigo";

    strSQL = new SqlCommand(aux, conn);

    strSQL.Parameters.AddWithValue("@codigo", int.Parse(umlivro.getCodigo()));

    result = strSQL.ExecuteReader();

    Erro.setErro(false);

    if (result.Read())
    {
        umlivro.setTitulo(result.GetString(1));
        umlivro.setAutor(result.GetString(2));
        umlivro.setEditora(result.GetString(3));
        umlivro.setAno(result.GetInt32(4).ToString());
    }
    else
    {
        Erro.setMsg("Livro não cadastrado.");
    }

    result.Close();
}

```

}

}

}