**GUÍA TÉCNICA: CONEXIÓN A BASE DE DATOS CON ADO .NET Y C#**

**Autor:** Luciano **Tecnología:** .NET / SQL Server **Tema:** Arquitectura en Capas y ADO .NET

### 1. Introducción Teórica

Para conectar una aplicación de escritorio (WinForms) con una base de datos SQL Server, utilizamos la tecnología **ADO .NET**. El proceso se divide en tres capas lógicas para mantener el código ordenado:

1. **Modelo (Clase Entidad):** Representa el objeto del mundo real (ej. Pokemon). Es el "molde".
2. **Negocio (Acceso a Datos):** Es la capa encargada de la lógica pesada. Aquí se escriben las consultas SQL y se manejan las conexiones.
3. **Presentación (Formulario):** Es lo que ve el usuario. Solo se encarga de pedir los datos al Negocio y mostrarlos.

---

### 2. Requisitos Previos

Antes de escribir el código, es necesario importar la librería oficial de Microsoft para SQL Server.

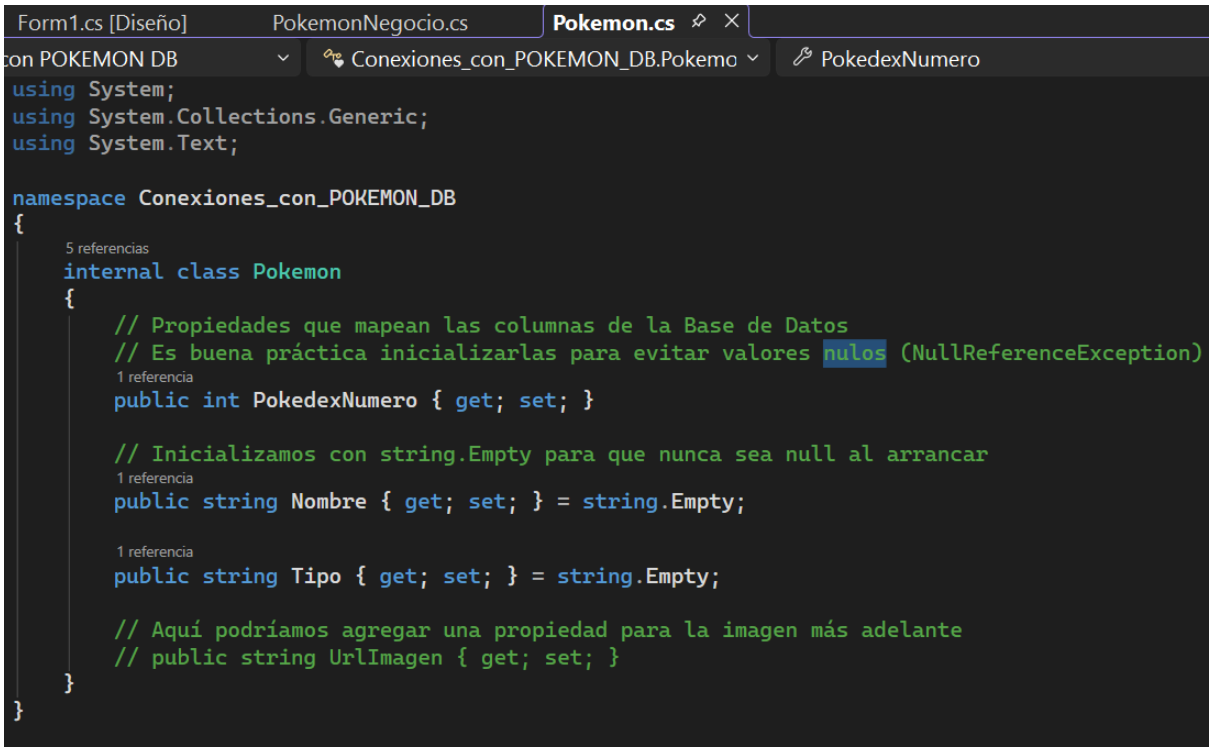
- **Paquete NuGet:** Microsoft.Data.SqlClient (o System.Data.SqlClient en versiones anteriores).
- **Namespace requerido:** using Microsoft.Data.SqlClient;

---

### 3. Implementación Paso a Paso

#### PASO A: La Clase Entidad (Modelo)

Esta clase sirve simplemente para transportar los datos. Sus propiedades deben coincidir con lo que queremos traer de la base de datos.

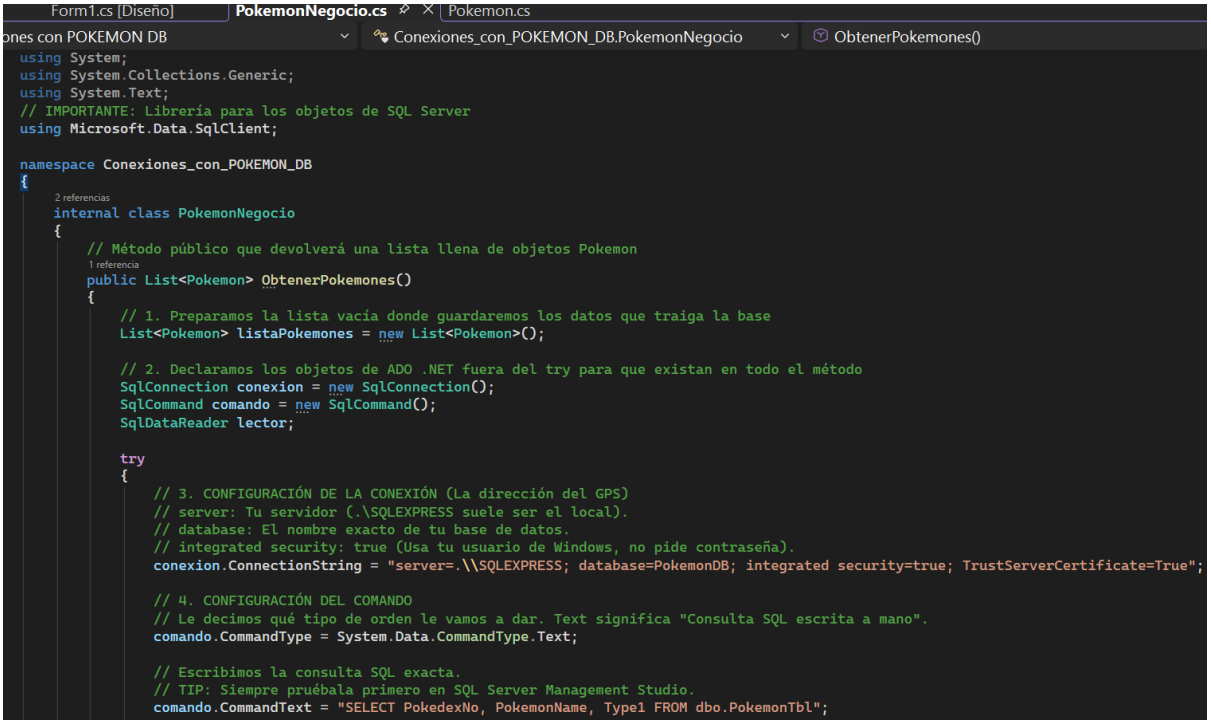


**PASO B: La Clase de Negocio (Lógica de Acceso a Datos)**

Esta es la clase más importante. Aquí ocurre la magia de ADO .NET. Se conecta, busca los datos, los transforma en objetos C# y cierra la conexión.

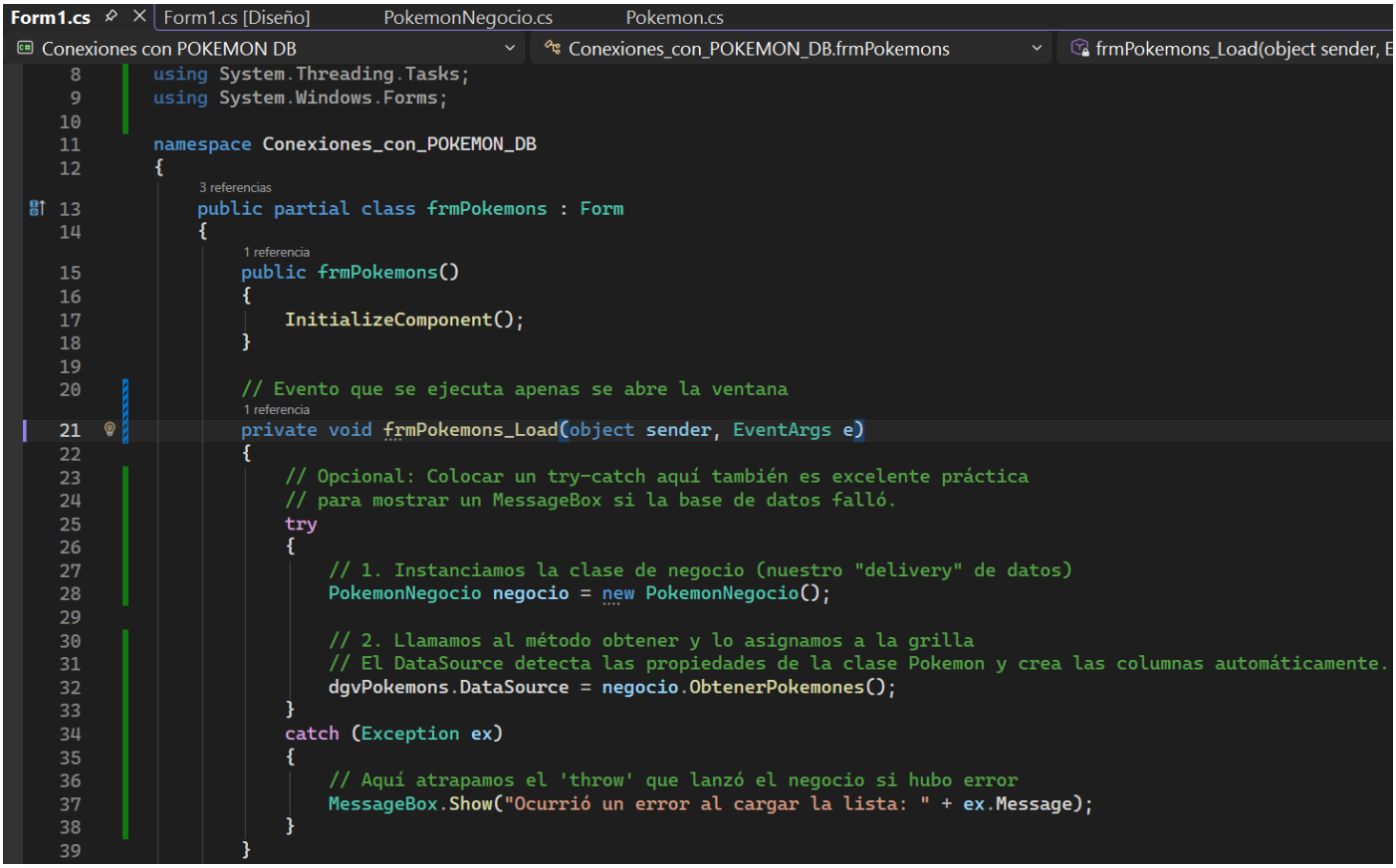
**Conceptos Clave:**

- **SqlConnection:** El puente hacia el servidor.
- **SqlCommand:** El mensajero que lleva la consulta SQL.
- **SqlDataReader:** El lector que recibe los resultados en formato de tabla.




**PASO C: La Capa de Presentación (El Formulario)**

El formulario no debe saber nada de SQL. Su única responsabilidad es llamar al Negocio y pedirle la lista ya fabricada.




**4. Resumen de flujo de datos**

1. **Usuario** abre el formulario (frmPokemons).
2. El evento Load llama a negocio.ObtenerPokemones().
3. **PokemonNegocio** crea la conexión, el comando y abre la puerta a SQL Server.
4. **SQL Server** ejecuta el SELECT y devuelve los datos crudos.
5. El SqlDataReader recorre los datos y los convierte en objetos **Pokemon**.
6. La lista de objetos viaja de regreso al formulario.
7. El DataGridView muestra la lista en pantalla.
8. El bloque finally cierra la conexión pase lo que pase.