

# UT 3 - Componentes

# **CONTROLES WINDOWS FORMS**

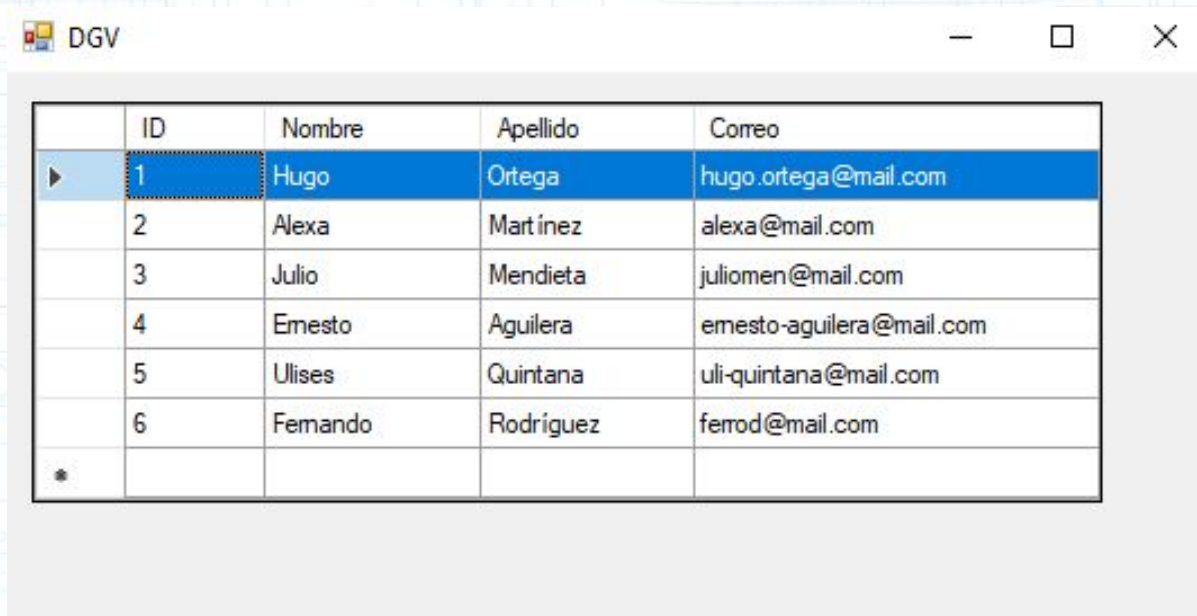
## Visual Studio 2019 - C#

- ✓ **DataGridView**
- ✓ **Conexión BBDD (MySQL)**
- ✓ **Filter**

# Visual Studio 2019 - C#



## DataGridView



	ID	Nombre	Apellido	Correo
▶	1	Hugo	Ortega	hugo.ortega@mail.com
	2	Alexa	Martínez	alexa@mail.com
	3	Julio	Mendieta	juliomen@mail.com
	4	Ernesto	Aguilera	ernesto-aguilera@mail.com
	5	Ulises	Quintana	uli-quintana@mail.com
	6	Fernando	Rodríguez	ferrod@mail.com
*				



# Visual Studio 2019 - C#

## ✓ DataGridView

### Añadir datos:

DGV

	ID	Nombre	Apellido	Correo
▶	1	Hugo	Ortega	hugo.ortega@mail.com
	2	Alexa	Martinez	alexa@mail.com
	3	Julio	Mendieta	julioen@mail.com
	4	Ernesto	Aguilera	ernesto-aguilera@mail.com
	5	Ulises	Quintana	uli-quintana@mail.com
	6	Fernando	Rodriguez	ferrod@mail.com
•				

```
//Añadimos líneas a Datagridview
//Añadiendo una fila y sus valores
dataGridView2.Rows.Add("10", "e3", "e3", "camarero", "12/12/2019", "");
//Asignando valores a las celdas
dataGridView2.Rows[1].Cells[0].Value = "11";
dataGridView2.Rows[1].Cells[1].Value = "r4";
dataGridView2.Rows[1].Cells[2].Value = "r4";
dataGridView2.Rows[1].Cells[3].Value = "cocinero";
dataGridView2.Rows[1].Cells[4].Value = "12/12/2019";
dataGridView2.Rows[1].Cells[5].Value = "12/12/2019";
```

# Visual Studio 2019 - C#

## ✓ DataGridView

Añadir datos:

```
dataGridView1.DataSource = controller.loadUsersDA();
```

**DataSource Property** → The **object** that contains data for the DataGridView to display.

DGV

	ID	Nombre	Apellido	Correo
▶	1	Hugo	Ortega	hugo.ortega@mail.com
	2	Alexa	Martinez	alexa@mail.com
	3	Julio	Mendieta	julioMEN@mail.com
	4	Ernesto	Aguilera	ernesto-aguilera@mail.com
	5	Ulises	Quintana	uli-quintana@mail.com
	6	Fernando	Rodriguez	ferrod@mail.com
•				

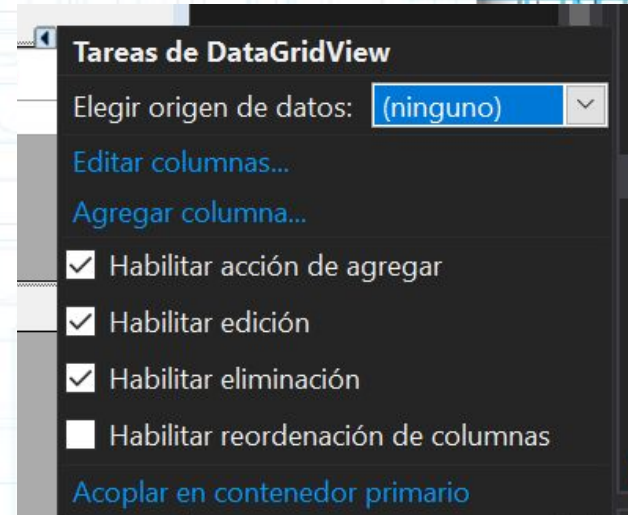
## Visual Studio 2019 - C#

### ✓ **DataGridView**

**Podemos editar las columnas** :(igual que Listview)

**Ojo: con DataSource**

→ ya vienen solas





## Visual Studio 2019 - C#

### ✓ **Conexión BBDD (MySQL)**





## Visual Studio 2019 - C#

### ✓ **Conexión BBDD (MySQL)**

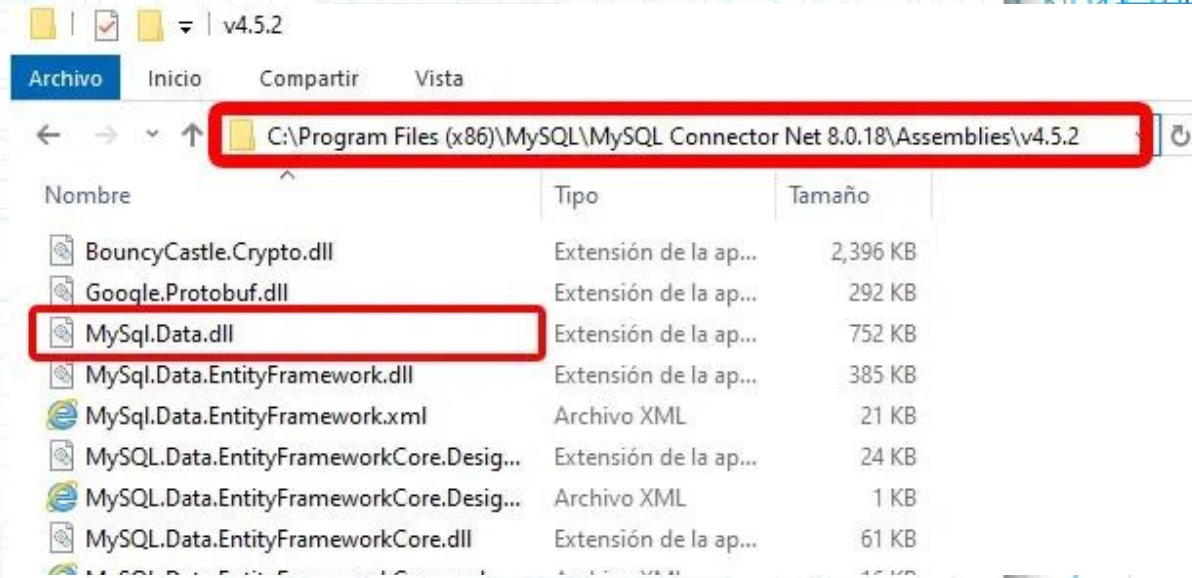
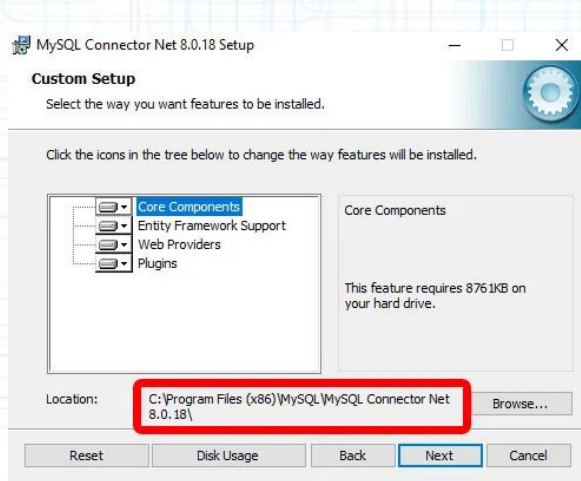
Conectaremos con la BBDD que teneis en la máquina , sino , Plan B



# Visual Studio 2019 - C#

## ✓ **Conexión BBDD (MySQL)**

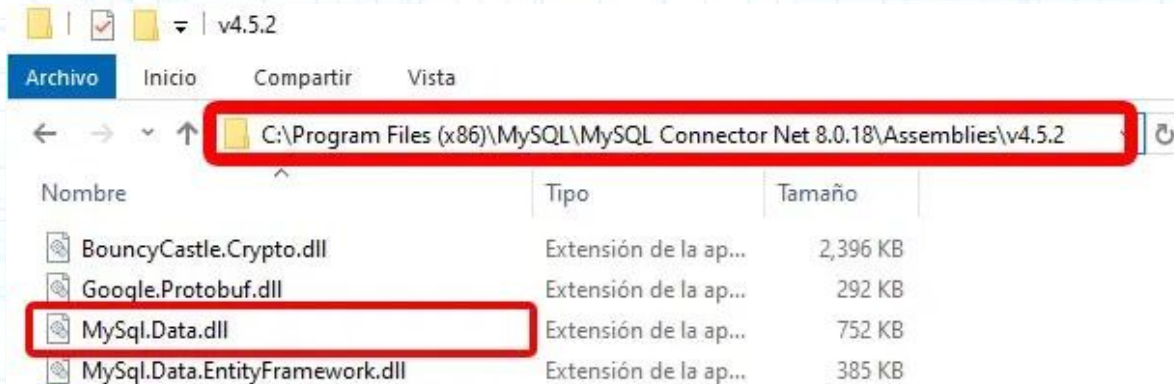
### 1. Necesitamos MySQL Connector/NET 8.x



# Visual Studio 2019 - C#

## ✓ **Conexión BBDD (MySQL)**

2. Creamos un Proyecto con .NET Framework IGUAL que el del connector que hemos instalado **OJO**  
**Porque sino no funcionará**
3. Agregamos referencia al proyecto a `MySQL.Data.dll`

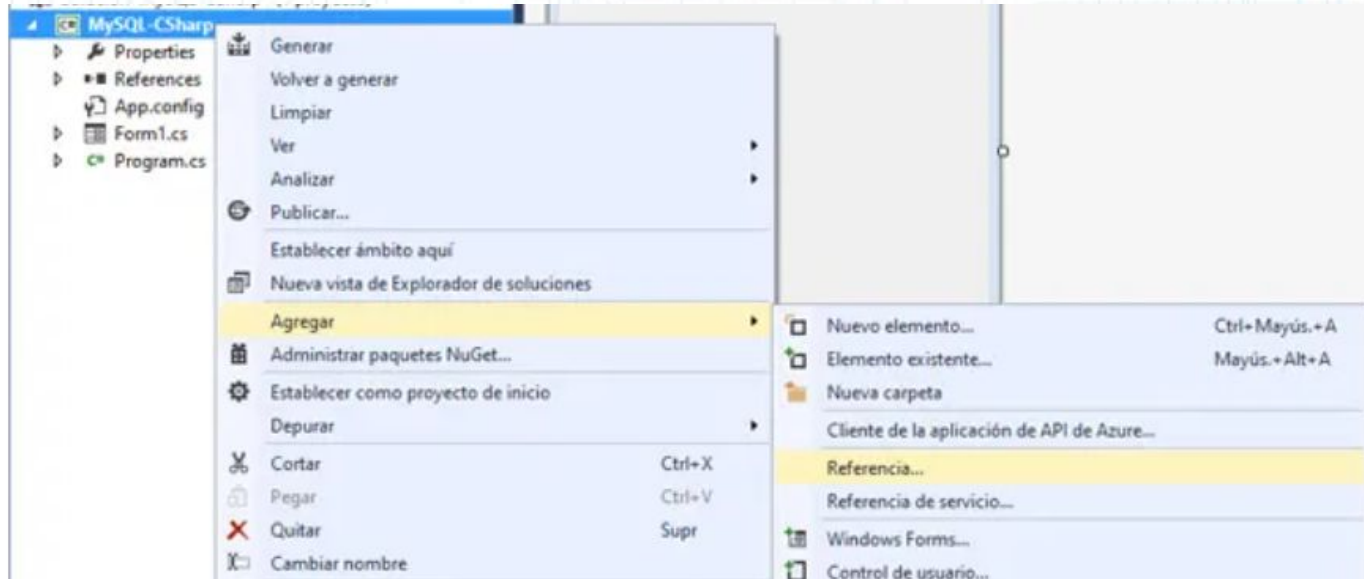




# Visual Studio 2019 - C#

## ✓ **Conexión BBDD (MySQL)**

2. Agregamos referencia al proyecto a MySql.Data.dll

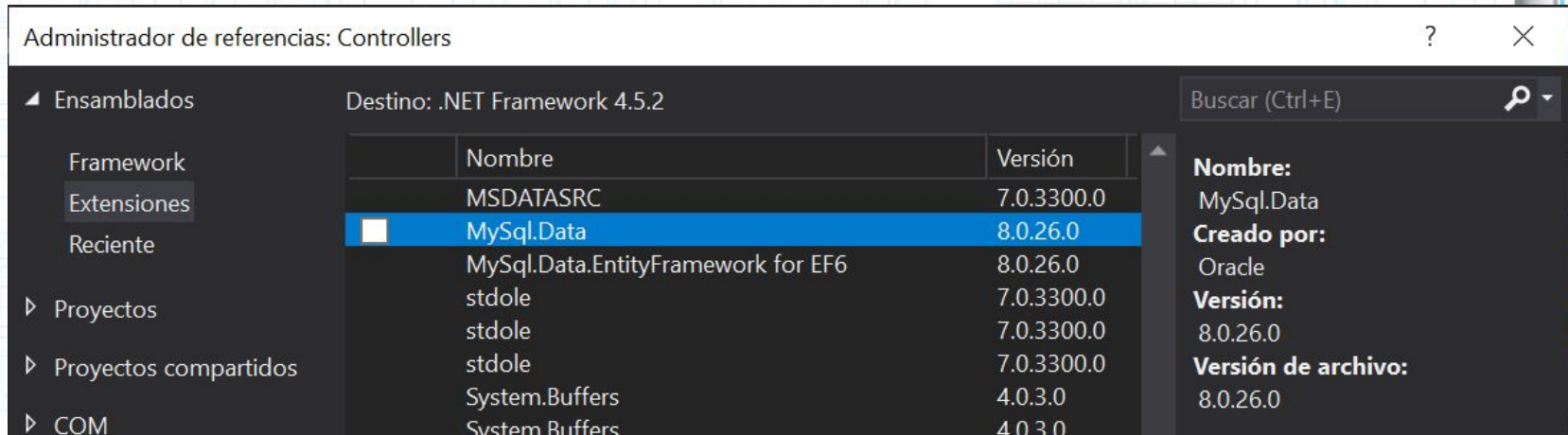




# Visual Studio 2019 - C#

## ✓ **Conexión BBDD (MySQL)**

2. Agregamos referencia al proyecto a `MySql.Data.dll`

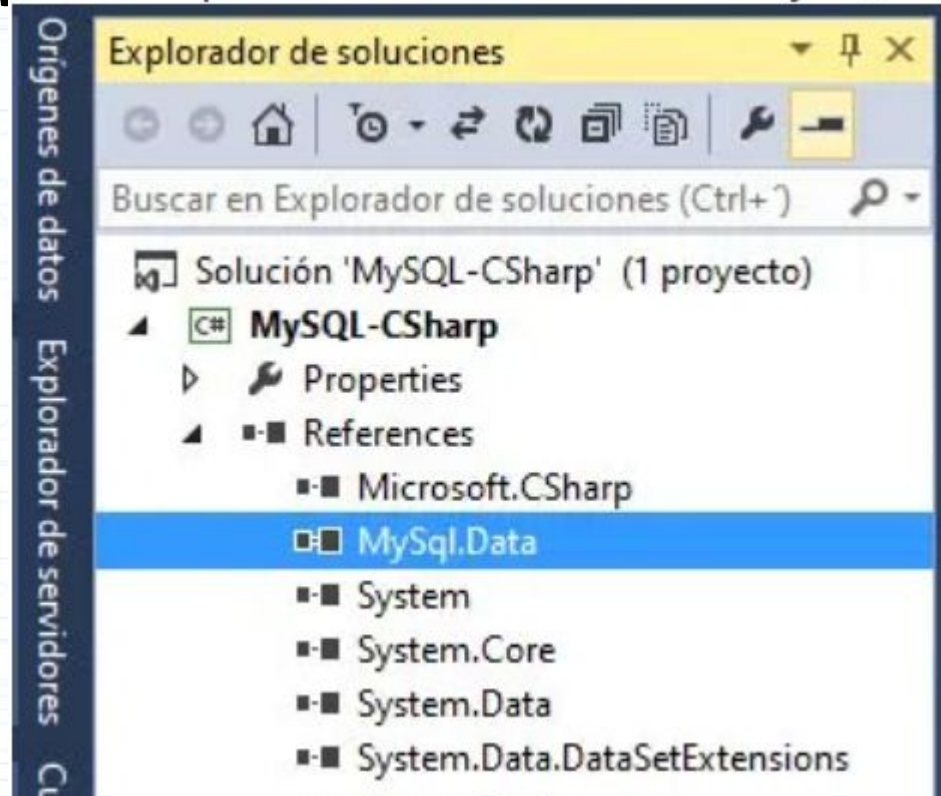


# Visual Studio 2019 - C#

## ✓ **Conexión BBDD (MySQL)**

2. Quedará así:

[Web Instalación](#)

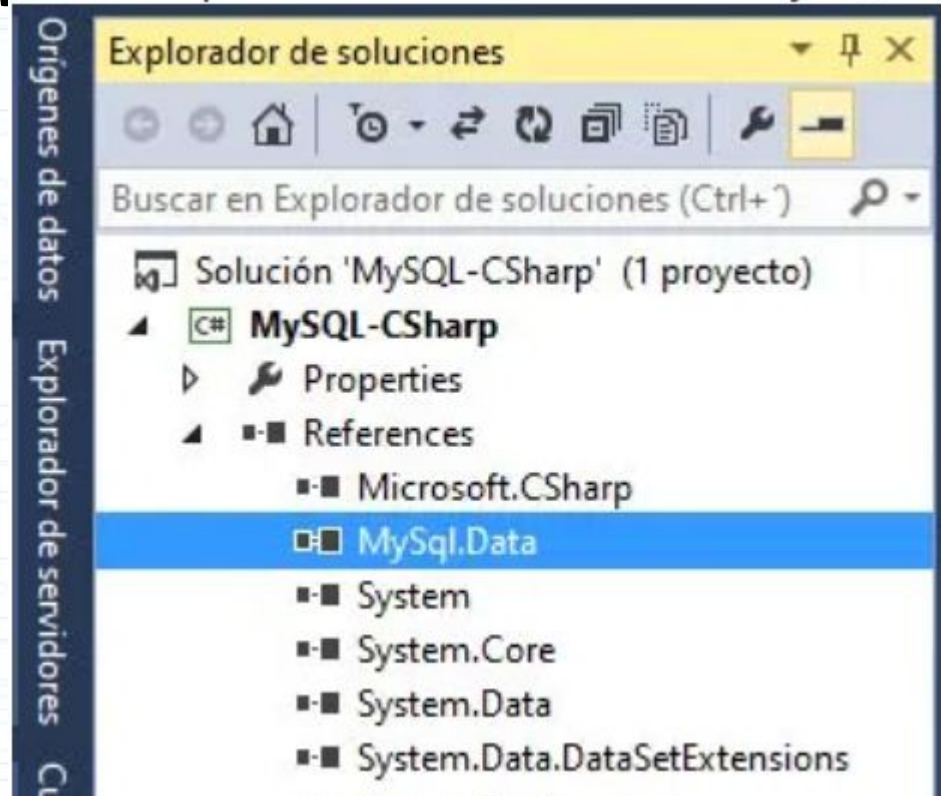


# Visual Studio 2019 - C#

## ✓ **Conexión BBDD (MySQL)**

2. Quedará así:

[Web Instalación](#)





## Visual Studio 2019 - C#

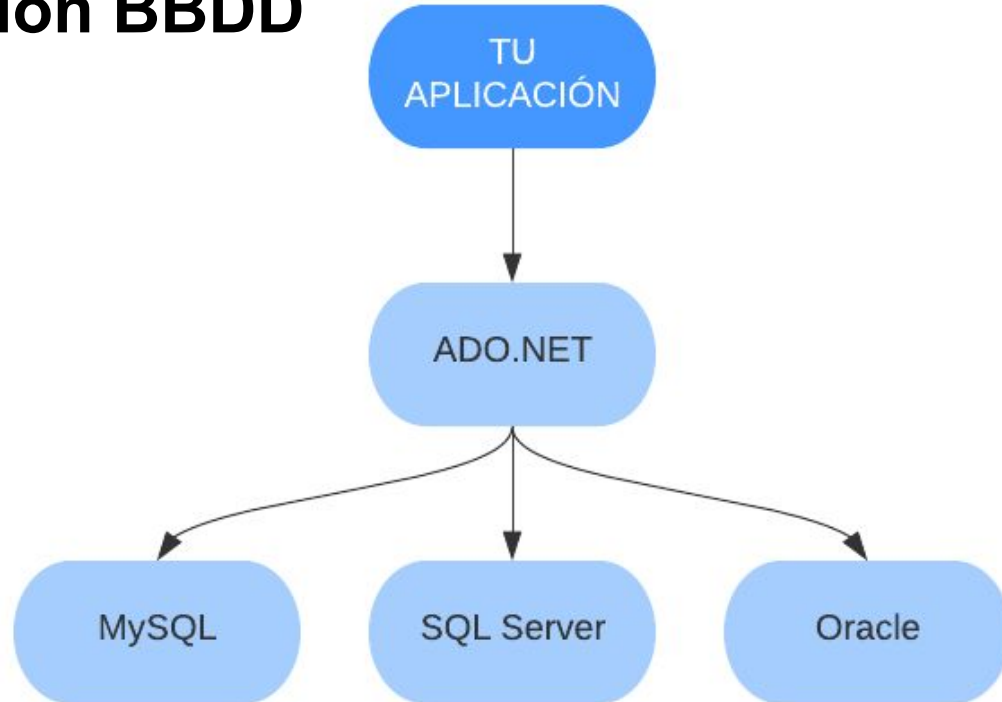
### ✓ **Conexión BBDD (MySQL)**

1. Conectamos con la BBDD (ejemplo de conexión)
2. Hacemos la solicitud de datos a la BBDD (ejemplo)
3. Recogemos y procesamos los datos



# Visual Studio 2019 - C#

## ✓ **Conexión BBDD**



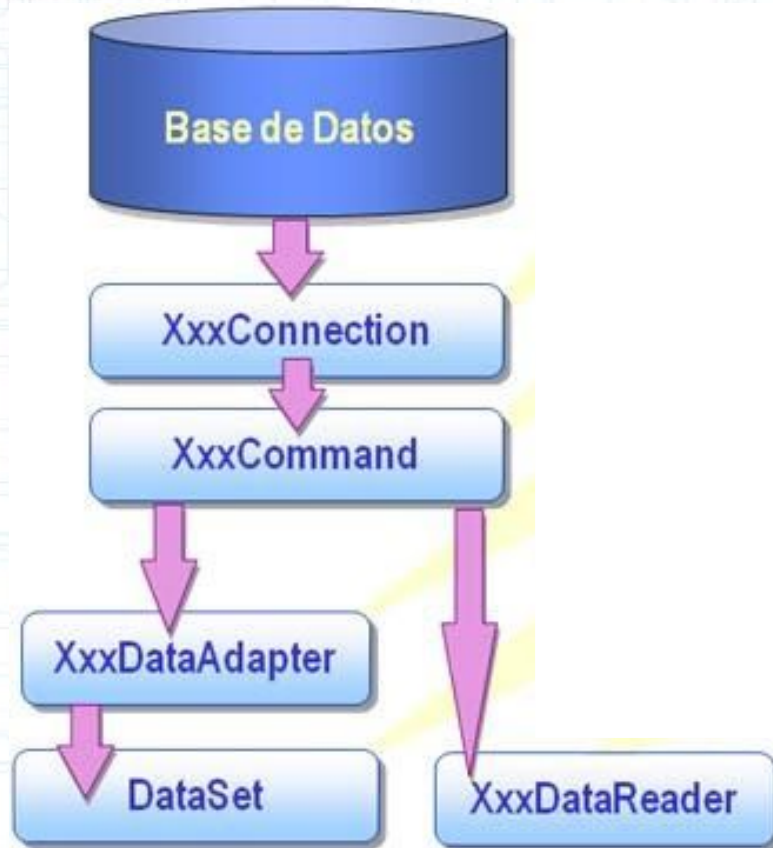
# Visual Studio 2019 - C#



## Arquitectura del .NET Framework

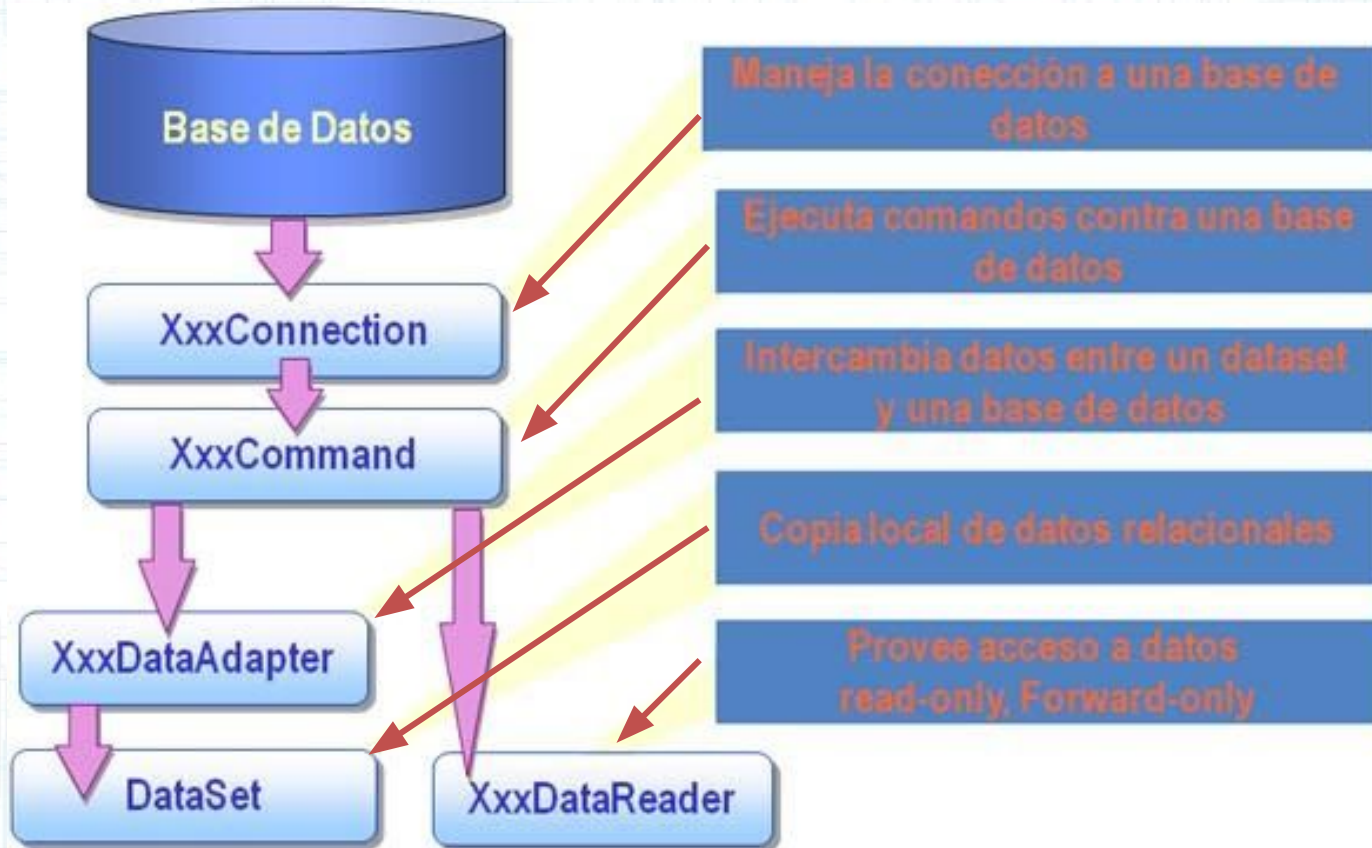


## Visual Studio 2019 - C#





# Visual Studio 2019 - C#





## Visual Studio 2019 - C#

### ✓ **Conexión BBDD (MySQL)**

1. <https://docs.microsoft.com/es-es/dotnet/framework/data/adonet/ado-net-code-examples>

## Visual Studio 2019 - C#

### ✓ **Conexión BBDD (MySQL)**

1. Objetos que utilizaremos para manipular la BBDD
  - a. **MySqlConnection**
  - b. **MySqlCommand**
  - c. **MySqlDataAdapter o MySqlReader**

## Visual Studio 2019 - C#

✓ **MySqlDataAdapter o MySqlDataReader**

# Visual Studio 2019 - C#

## ✓ **MySqlDataReader**

- Mantiene la conexión abierta hasta que termines (**¡no olvides cerrarla!**).
- Normalmente sólo se puede iterar una vez
- No es tan útil para actualizar la base de datos

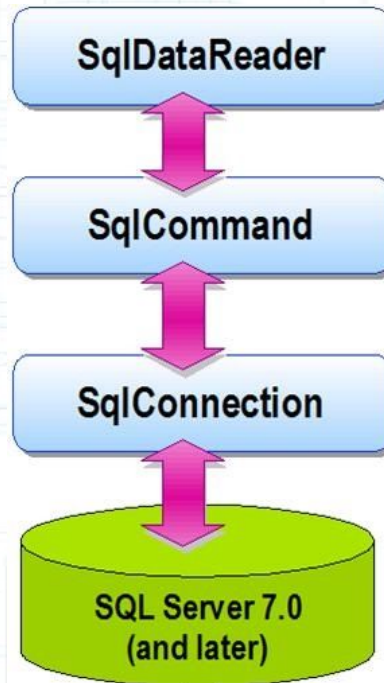
Por otro lado:

- Sólo tiene un registro en la memoria a la vez en lugar de un conjunto de resultados completo (esto puede ser **ENORME**)
- Es lo más rápido que se puede conseguir para esa iteración
- Permite empezar a procesar los resultados antes (**una vez que el primer registro está disponible**). Para algunos tipos de consulta esto también puede ser muy importante.



## Visual Studio 2019 - C#

### ✓ **Conexión BBDD**



# Visual Studio 2019 - C#

## ✓ **MySqlDataAdapter**

- Le permite cerrar la conexión tan pronto como termine de cargar los datos, e incluso puede cerrarla por usted automáticamente
- Todos los resultados están disponibles en la memoria
- Puedes iterar sobre él tantas veces como necesites, o incluso buscar un registro específico por índice
- Tiene algunas facultades incorporadas para la actualización de la base de datos

### **A costa de:**

- *Mucho* mayor uso de la memoria
- Hay que esperar a que se carguen todos los datos antes de utilizarlo

## Visual Studio 2019 - C#

### ✓ **Conexión BBDD (MySQL)**

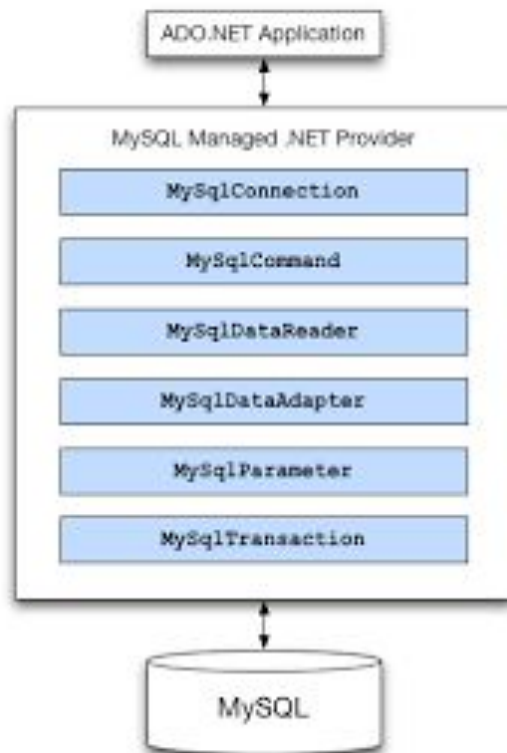
1. Objetos que utilizaremos para manipular la BBDD

#### **a. MySqlConnection**

Nos va a permitir “adaptar los datos de una BBDD a una estructura de datos de C# **DataSet o DataTable**

## Visual Studio 2019 - C#

### ✓ **Conexión BBDD**



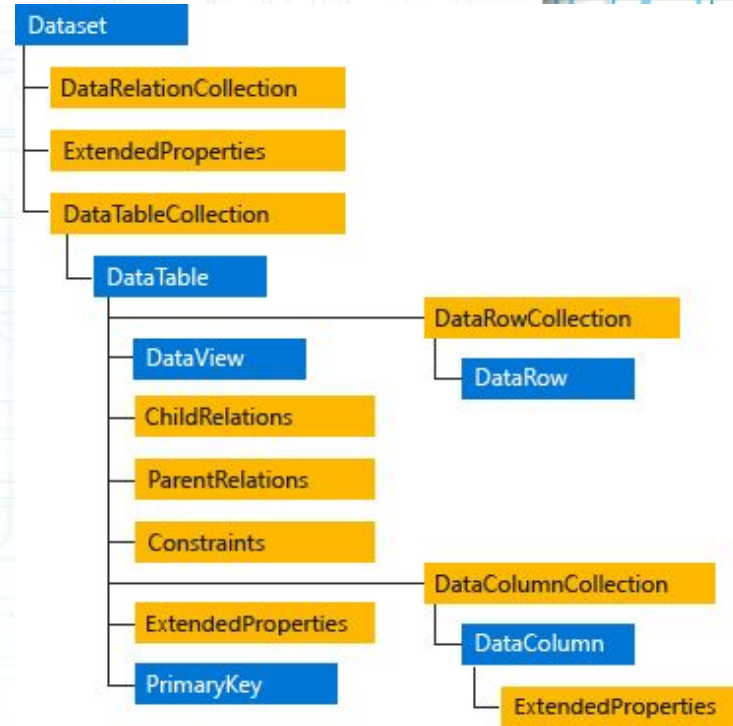


# Visual Studio 2019 - C#



## **DataSet:**

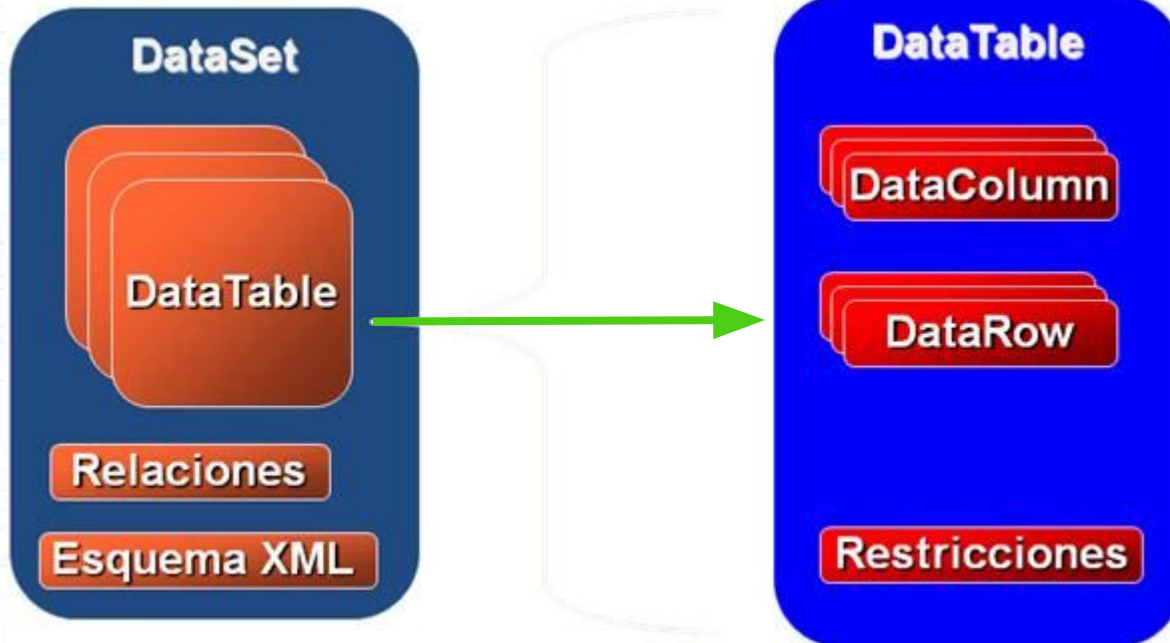
- Es una representación de datos residentes en memoria que proporciona un modelo de programación relacional coherente independientemente del origen de datos.
- Representa un conjunto completo de datos, incluidas las tablas relacionadas, las restricciones y las relaciones entre las tablas



# Visual Studio 2019 - C#



**DataSet:**



# Visual Studio 2019 - C#

## ✓ SqlCommand

### ❑ SqlCommand.ExecuteReader

Envía la propiedad `CommandText` a `Connection` y crea un objeto `SqlDataReader`.

### ❑ SqlCommand.ExecuteNonQuery:

Ejecuta una instrucción de Transact-SQL en la conexión y devuelve el número de filas afectadas.

### ❑ SqlCommand.ExecuteScalar

Ejecuta la consulta y devuelve la primera columna de la primera fila del conjunto de resultados devueltos por la consulta. Las demás nada

## Visual Studio 2019 - C#

### ✓ Añadir datos a la BBDD

```
// Assumes connection is a valid SqlConnection.  
connection.Open();  
  
string queryString = "INSERT INTO Customers " +  
"(CustomerID, CompanyName) Values('NWIND', 'Northwind Traders')";  
  
SqlCommand command = new SqlCommand(queryString, connection);  
Int32 recordsAffected = command.ExecuteNonQuery();|
```



## Visual Studio 2019 - C#

### ✓ **Parámetros de la consulta**

```
using (SqlConnection connection = new SqlConnection(connectionString))
{
    SqlCommand command = new SqlCommand(commandText, connection);
    command.Parameters.Add("@ID", SqlDbType.Int);
    command.Parameters["@ID"].Value = customerID;

    // Use AddWithValue to assign Demographics.
    // SQL Server will implicitly convert strings into XML.
    command.Parameters.AddWithValue("@demographics", demoXml);
}
```

## Visual Studio 2019 - C#

### ✓ **Parámetros de la consulta**

```
mysqlDataAdapter.UpdateCommand = new MySqlCommand(  
    "UPDATE users SET username = @username WHERE id = @id", connection);  
  
mysqlDataAdapter.UpdateCommand.Parameters.Add(  
    "@username", MySqlDbType.VarChar, 15, "username");  
  
MySqlParameter parameter = mysqlDataAdapter.UpdateCommand.Parameters.Add(  
    "@id", MySqlDbType.UInt32);  
parameter.SourceColumn = "id";  
parameter.SourceVersion = DataRowVersion.Original;
```

# Visual Studio 2019 - C#

## ✓ **Parámetros de la consulta**

**add()-->**

<https://docs.microsoft.com/es-es/dotnet/api/system.data.sqlclient.sqlparametercollection.add?view=dotnet-plat-ext-6.0>

```
sql = "INSERT INTO users (id, username, password, role, createdOn, lastLogin) " +  
      "VALUES (@id, @username, @password, @role, @createdOn, @lastLogin)";  
using (mysqlCmd = new MySqlCommand(sql, connection))  
{  
    mysqlCmd.Parameters.AddWithValue("@id", id);  
    mysqlCmd.Parameters.AddWithValue("@username", username);  
    mysqlCmd.Parameters.AddWithValue("@password", password);  
    mysqlCmd.Parameters.AddWithValue("@role", role);  
}
```



```
public void addUser(String id, String username, String password, String role, String createOn)
{
    String sql;
    using (connection = dataSource.getConnection())
    {
        connection.Open();
        sql = "INSERT INTO users (id, username, password, role, createdOn, lastLogin) " +
            "VALUES (@id, @username, @password, @role, @createdOn, @lastLogin)";
        using (mysqlCmd = new MySqlCommand(sql, connection))
        {
            mysqlCmd.Parameters.AddWithValue("@id", id);
            mysqlCmd.Parameters.AddWithValue("@username", username);
            mysqlCmd.Parameters.AddWithValue("@password", password);
            mysqlCmd.Parameters.AddWithValue("@role", role);
            mysqlCmd.Parameters.AddWithValue("@createdOn", createOn);
            mysqlCmd.Parameters.AddWithValue("@lastLogin", "");
            int rowsAffected = mysqlCmd.ExecuteNonQuery();
        }
    }
}
```



# Visual Studio 2019 - C#



## **DataGridView.SelectedRows**

```
private void DataGridView1_UserDeletingRow(object sender,
    DataGridViewRowCancelEventArgs e)
{
    DataGridViewRow startingBalanceRow = DataGridView1.Rows[0];

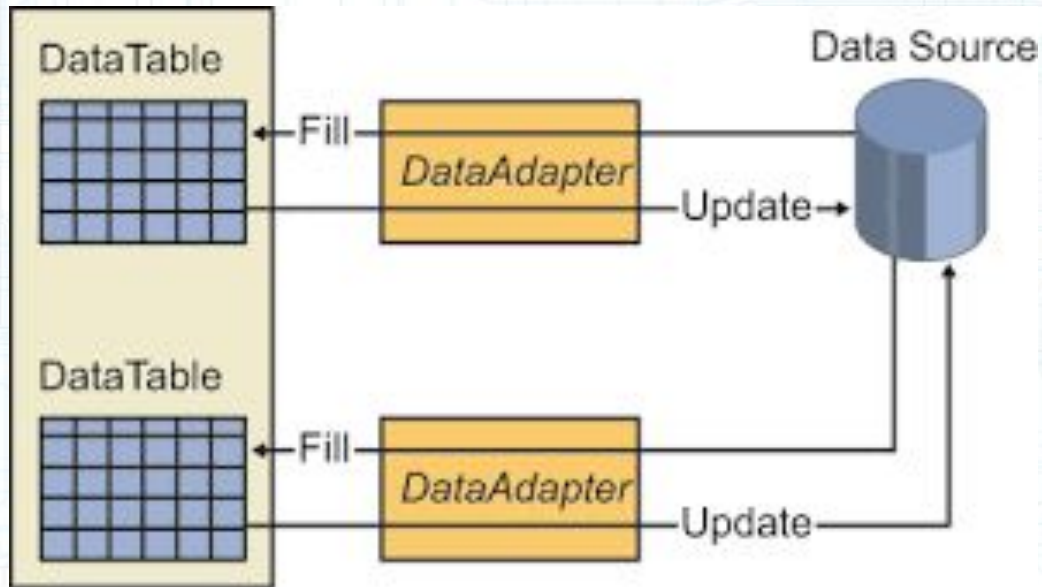
    // Check if the Starting Balance row is included in the selected rows
    if (DataGridView1.SelectedRows.Contains(startingBalanceRow))
    {
        // Do not allow the user to delete the Starting Balance row.
        MessageBox.Show("Cannot delete Starting Balance row!");

        // Cancel the deletion if the Starting Balance row is included.
        e.Cancel = true;
    }
}
```

Más info--> [aqui](#)

## Visual Studio 2019 - C#

### ✓ **MySqlDataAdapter para Modificaciones BBDD**



## Visual Studio 2019 - C#

### ✓ Recuperar Datos de DataReader

```
while (mysqlReader.Read())
{
    Console.WriteLine("{0}\t{1}",
        mysqlReader.GetInt32(0),
        mysqlReader.GetString(1));
}
```

```
while (mysqlReader.Read())
{
    int id = Convert.ToInt32(mysqlReader["id"]);
    string username = mysqlReader["username"].ToString();
    string password = mysqlReader["password"].ToString();
    string role = mysqlReader["role"].ToString();
}
```

<https://docs.microsoft.com/es-es/dotnet/api/microsoft.data.sqlclient.sqldatareader?view=sqlclient-dotnet-standard-4.0>

<https://docs.microsoft.com/es-es/sql/connect/ado-net/retrieve-data-by-datareader?view=sql-server-ver15>



## Visual Studio 2019 - C#

### ✓ **MySqlDataAdapter para Modificaciones BBDD**

