

DataGridView

Visual c#에서 데이터를 다루기 위한 컨트롤
(데이터 표시와 조작)



2024.08 하성호

목차

1. DataGridView의 개요

2. 주요 기능

3. 사용 예제

4. 고급 기능 및 팁

DataGridView의 개요

- 정의

- DataGridView는 Windows Forms 애플리케이션에서 테이블 형태의 데이터를 표시하고 조작할 수 있는 컨트롤입니다.

- 주요특징

- 데이터 바인딩

- 다양한 데이터 소스(SQL, XML, 객체 리스트 등)와 바인딩하여 데이터를 손쉽게 표시합니다.

- 다양한 데이터 형식 지원

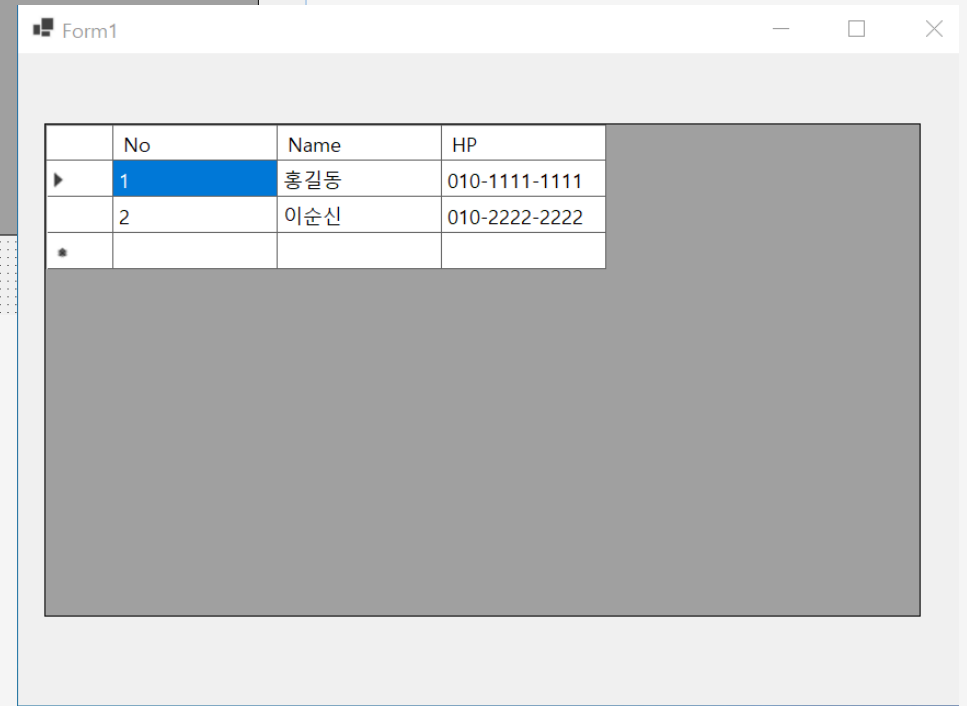
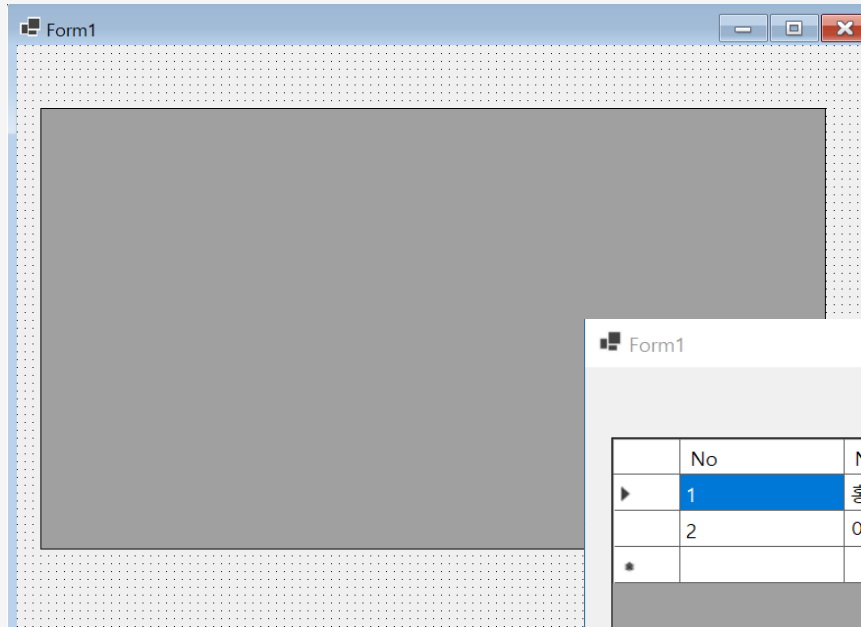
- 텍스트, 이미지, 버튼, 체크박스 등 다양한 형식의 데이터를 표시할 수 있습니다.

- 다양한 데이터 소스와의 호환성

- 컬럼, 셀의 스타일을 자유롭게 변경 가능하며, 사용자 정의 컨트롤을 삽입할 수 있습니다.

주요 기능

- 데이터 바인딩
 - 단일 데이터 소스
 - 복합 데이터 소스
- 편집
 - 셀 편집
 - 행 추가/삭제
- 정렬 및 필터링
 - 정렬
 - 필터링
- 스타일링

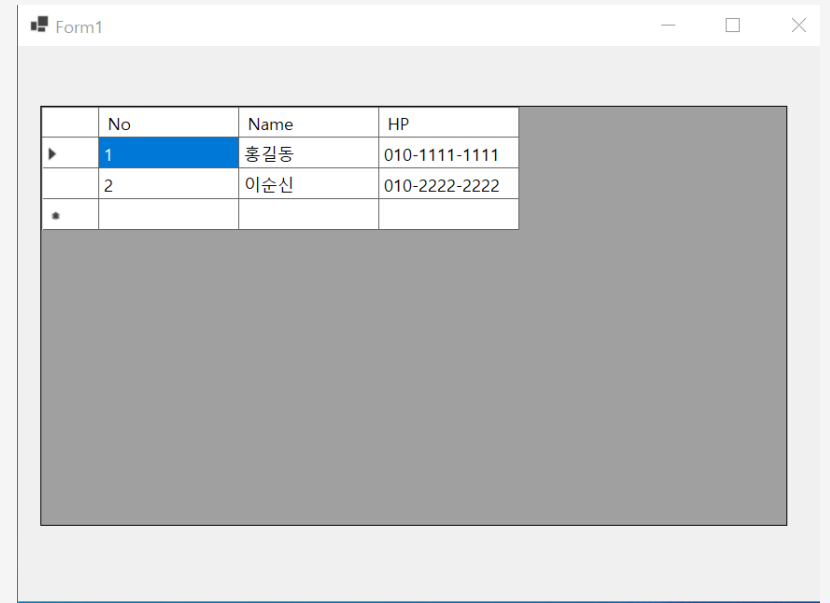


실습 -1 DataGridView 사용하기 (Form_Load) 방식

- DataGridView 컨트롤을 Form에 올리고 이름을 "DataGridView"로 수정합니다.

```
private void Form1_Load(object sender, EventArgs e)
{
    this.dataGridView.Columns.Add("No", "No");
    this.dataGridView.Columns.Add("Name", "Name");
    this.dataGridView.Columns.Add("HP", "HP");

    this.dataGridView.Rows.Add("1", "홍길동", "010-1111-1111");
    this.dataGridView.Rows.Add("2", "이순신", "010-2222-2222");
}
```



실습-1 버튼에 삽입 기능을 넣기 (버튼 이벤트에 넣기)

```
private void Form1_Load(object sender, EventArgs e)
{
    dataGridView1.Columns.Add("No", "번호");
    dataGridView1.Columns.Add("Name", "이름");
}
```

```
private void button1_Click(object sender, EventArgs e)
{
    dataGridView1.Rows.Add("1", "홍길동");
}
```

Form1

	번호	이름
	1	홍길동
	1	홍길동
	1	홍길동
▶▶		

컬럼추가

실습-2 DataGridView 사용하기

- connectionString을 로컬인 경우 간단하게 설정할 수 있다.
- Data객체를 만들어서 표현할 수 있다.

```
private void Form1_Load(object sender, EventArgs e)
{
    //데이터 바인딩
    var dataList = new List<Data>
    {
        new Data { No = "1", Name = "홍길동", HP = "010-1111-1111"},
        new Data { No = "2", Name = "이순신", HP = "010-2222-2222"},
        new Data { No = "3", Name = "강감찬", HP = "010-3333-3333"}
    };
}
```

```
this.dataGridView.DataSource = dataList;
}
```

```
public class Data
{
    public string No { get; set; }
    public string Name { get; set; }
    public string HP { get; set; }
}
```

실습-3 Oracle에서 데이터 가져오기

```
private string connectionString = "User Id=scott;Password=tiger;Data Source=localhost:1521/XE;";

private void Form1_Load(object sender, EventArgs e)
{
    using (OracleConnection connection = new OracleConnection(connectionString))
    {
        try
        {
            connection.Open();
            string query = "SELECT id, name, hp FROM student";
            OracleDataAdapter adapter = new OracleDataAdapter(query, connection);

            DataTable dataTable = new DataTable();
            adapter.Fill(dataTable);

            dataGridView.DataSource = dataTable;
        }
        catch (Exception ex)
        {
            MessageBox.Show($"에러: {ex.Message}");
        }
    }
}
```


실습-3 계속

```
public class Data
{
    public string No { get; set; }
    public string Name { get; set; }
    public string HP { get; set; }
}
```

- 데이터 그리드 이벤트

```
private void dataGridView_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
    MessageBox.Show($"버튼이 클릭된 행: {e.RowIndex}");
}
```

실습-4 DataGridView만 가지고 삽입, 삭제, 검색

```
private void Form1_Load(object sender, EventArgs e)
{
    dataGridView1.Columns.Add("No", "번호");
    dataGridView1.Columns.Add("Name", "이름");
    dataGridView1.Columns.Add("HP", "핸드폰번호");
}
```

실습-4 DataGridView만 (삽입)

```
private void buttonAdd_Click(object sender, EventArgs e)
{
    string no = textBoxNo.Text;
    string name = textBoxName.Text;
    string hp = textBoxHP.Text;

    if (!string.IsNullOrEmpty(no) && !string.IsNullOrEmpty(name) && !string.IsNullOrEmpty(hp))
    {
        dataGridView1.Rows.Add(no, name, hp);
        textBoxNo.Clear();
        textBoxName.Clear();
        textBoxHP.Clear();
    }
    else
    {
        MessageBox.Show("모든 필드를 입력해 주세요.");
    }
}
```

실습-4 DataGridView만 (삭제)

```
private void buttonDelete_Click(object sender, EventArgs e)
{
    foreach (DataGridViewRow row in dataGridView1.SelectedRows)
    {
        dataGridView1.Rows.Remove(row);
    }
}
```

실습-4 DataGridView만 (검색)

```
private void buttonSearch_Click(object sender, EventArgs e)
{
    string searchValue = textBoxSearch.Text;

    foreach (DataGridViewRow row in dataGridView1.Rows)
    {
        if (row.Cells["Name"].Value != null && row.Cells["Name"].Value.ToString().Equals(searchValue))
        {
            row.Selected = true;
            return;
        }
    }

    MessageBox.Show("해당 이름을 찾을 수 없습니다.");
}
```

실습-5 오라클 연동

- 테이블 모양

```
CREATE TABLE student (  
    id number(5) primary key,  
    name varchar2(30) not null,  
    hp varchar2(13)  
);
```

- 기본모양
 - **DataGridView**: dataGridView1
 - **TextBox**: textBoxNo, textBoxName, textBoxHP, textBoxSearch
 - **Button**: buttonAdd, buttonDelete, buttonSearch

실습-5 전체화면

- DataGridView를 활용한 Windows 프로그램

The screenshot shows a Windows application window titled "Form1". Inside the window, there is a DataGridView table with the following data:

	번호	이름	핸드폰번호
▶	1	홍길동	010-1111-1111
	2	강감찬	010-2222-2222
	3	이순신	010-3333-3333
*			

Below the table, there are input fields and buttons for managing the data:

- Input field: 4
- Input field: 을지문덕
- Input field: 010-5555-5555
- Button: 추가 (Add)
- Button: 삭제 (Delete)
- Input field: 홍길동
- Button: 검색 (Search)

실습-5 연결자 설정

```
private string connectionString = "Data  
Source=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=127.0.0.1)(PORT=1521)))(CONNECT_DA  
TA=(SERVER=DEDICATED)(SERVICE_NAME=xe)));User Id=scott;Password=tiger";
```

```
private void Form1_Load(object sender, EventArgs e)  
{  
    LoadData();  
}
```


실습-5 DBMS에 기록된 내용을 DataGridView에 불러오기 (화면갱신)

```
private void LoadData()
{
    using (OracleConnection connection = new OracleConnection(connectionString))
    {
        try
        {
            connection.Open();
            string query = "SELECT id, name, hp FROM student";
            OracleDataAdapter adapter = new OracleDataAdapter(query, connection);

            DataTable dataTable = new DataTable();
            adapter.Fill(dataTable);

            dataGridView1.DataSource = dataTable;

            // 컬럼 이름 설정
            dataGridView1.Columns["Id"].HeaderText = "번호";
            dataGridView1.Columns["Name"].HeaderText = "이름";
            dataGridView1.Columns["HP"].HeaderText = "핸드폰번호";
        }
        catch (Exception ex)
        {
            MessageBox.Show($"데이터 로드 중 오류: {ex.Message}");
        }
    }
}
```

실습-5 데이터 추가

```
private void buttonAdd_Click(object sender, EventArgs e)
{
    using (OracleConnection connection = new OracleConnection(connectionString))
    {
        try
        {
            connection.Open();
            string query = "INSERT INTO student (id, name, hp) VALUES (:id, :name, :hp)";
            using (OracleCommand command = new OracleCommand(query, connection))
            {
                command.Parameters.Add("id", OracleDbType.Int32).Value = int.Parse(textBoxNo.Text);
                command.Parameters.Add("name", OracleDbType.Varchar2).Value = textBoxName.Text;
                command.Parameters.Add("hp", OracleDbType.Varchar2).Value = textBoxHP.Text;

                command.ExecuteNonQuery();
            }

            LoadData();

            textBoxNo.Clear();
            textBoxName.Clear();
            textBoxHP.Clear();
        }
        catch (Exception ex)
        {
            MessageBox.Show($"데이터 삽입 중 오류: {ex.Message}");
        }
    }
}
```

실습-5 오라클을 이용한 데이터 삭제

```
private void buttonDelete_Click(object sender, EventArgs e)
{
    int IdNum;
    if (int.TryParse(textBoxNo.Text, out IdNum))
    {
        using (OracleConnection connection = new OracleConnection(connectionString))
        {
            try
            {
                connection.Open();
                string query = "DELETE FROM student WHERE id = :id";
                using (OracleCommand command = new OracleCommand(query, connection))
                {
                    command.Parameters.Add("id", OracleDbType.Int32).Value = IdNum;
                    int rowsAffected = command.ExecuteNonQuery();

                    if (rowsAffected > 0)
                    {
                        MessageBox.Show("데이터가 성공적으로 삭제되었습니다.");
                        LoadData();
                    }
                    else
                    {
                        MessageBox.Show("삭제할 데이터가 없습니다.");
                    }
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show($"데이터 삭제 중 오류: {ex.Message}");
            }
        }
        textBoxNo.Clear();
    }
}
```

실습-5 기본검색 DataGridView 위주

```
private void buttonSearch_Click(object sender, EventArgs e)
{
    string searchValue = textBoxSearch.Text;
    bool found = false;

    foreach (DataGridViewRow row in dataGridView1.Rows)
    {
        if (row.Cells["Name"].Value != null && row.Cells["Name"].Value.ToString().Equals(searchValue))
        {
            row.Selected = true;
            found = true;
            MessageBox.Show("데이터를 찾았습니다.");
            break;
        }
    }

    if (!found)
    {
        MessageBox.Show("해당 이름을 찾을 수 없습니다.");
    }
}
```

실습-5
