



# 28강. 비연결형 데이터베이스(1)

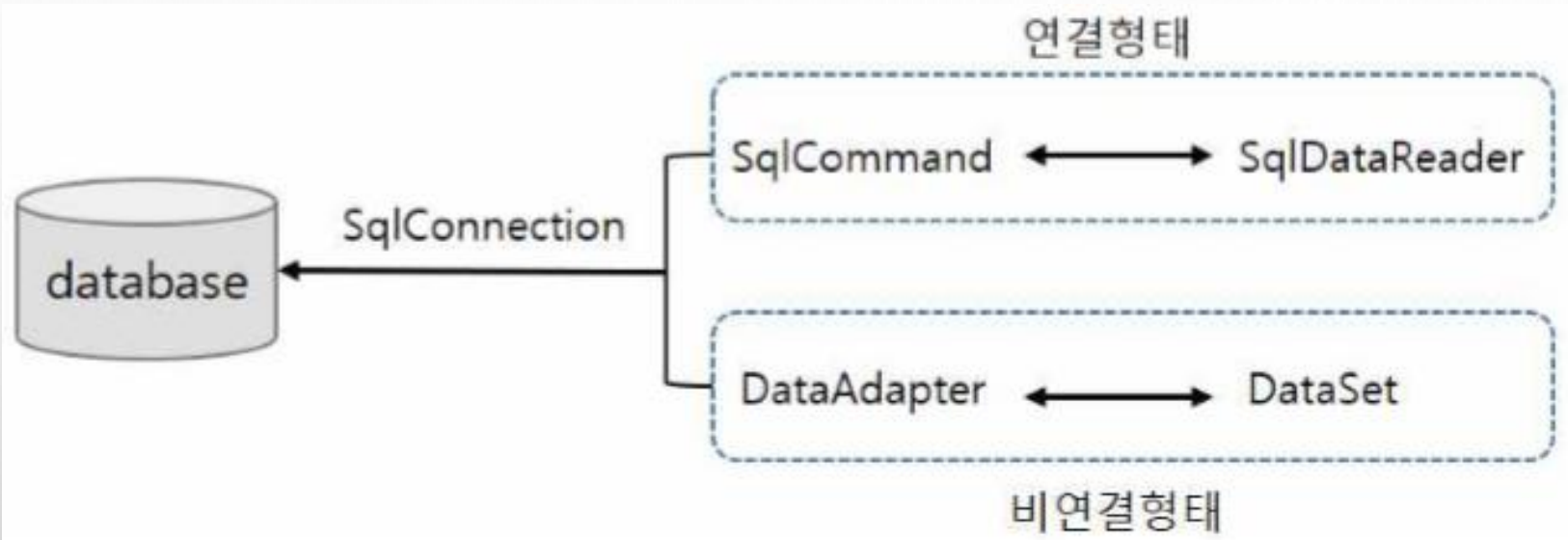
DataTable, DataGridView

# 목차

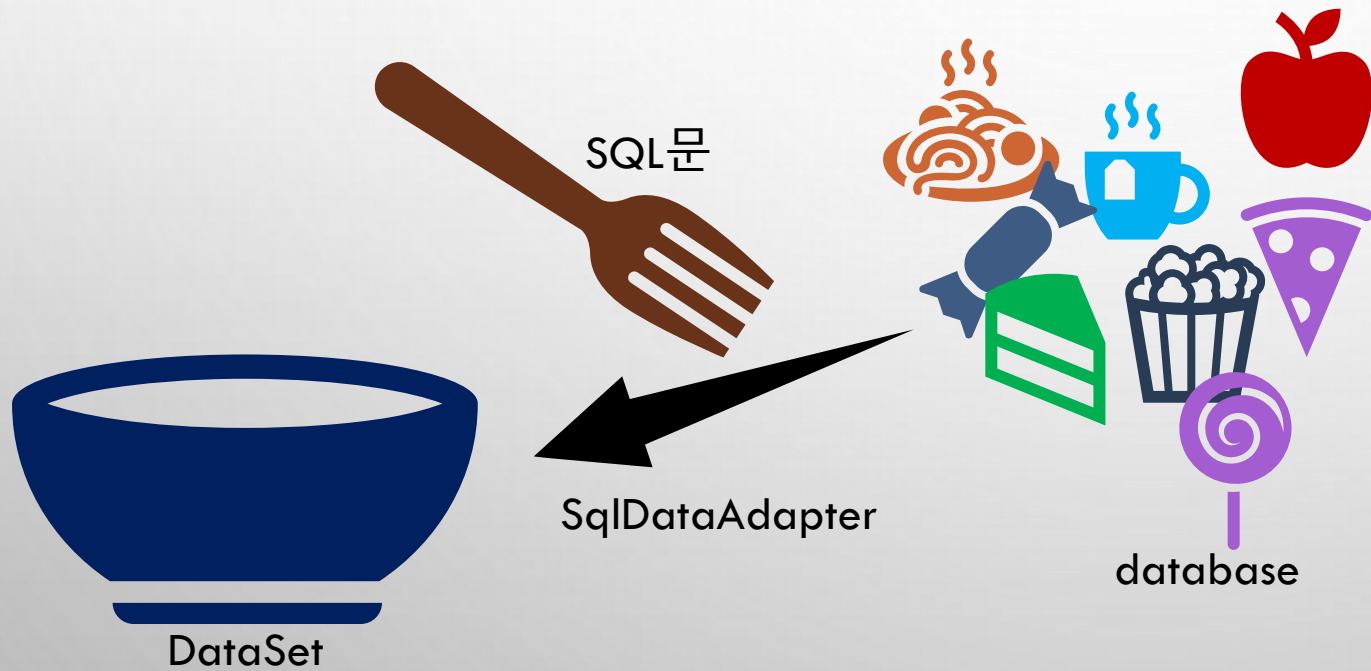
- 개요
- DataTable
- DataGridView

# 비연결형 데이터베이스

- 구성요소
  - SqlConnection : 연결
  - SqlDataAdapter : 조회, 저장, 수정, 삭제
    - SqlCommand : SQL 실행
  - DataSet : DataTable 집합
    - DataTable : 데이터베이스 테이블
    - DataRow : 행 (레코드)
    - DataColumn : 열 (컬럼 단위)
  - DataSet  $\supset$  DataTable  $\supset$  DataRow,  
DataColumn

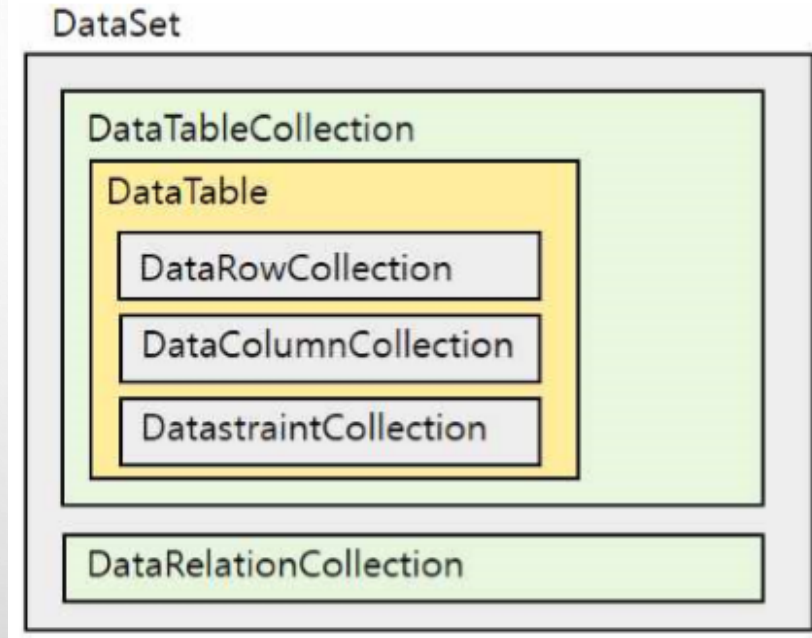


# 학습 과정



# DataTable

- 개념  
메모리형 데이터 집합
- 구조  
데이터베이스의 테이블  
과 동일한 구조
- 역할  
서버 테이블의 내용 저장  
& 관리(수정, 삽입, 삭제)



# DataTable

- 네임스페이스

System.Data

- 생성자

DataTable()

DataTable(SerializationInfo,  
StreamingContext)

DataTable(String)

DataTable(String, String)

# DataTable

- DataTable 생성과 해제

```
DataTable table = new DataTable("table1");
```

```
.....
```

```
table.Dispose();
```



# DataColumn : 개별 항목

## ■ DataColumn ?

테이블의 항목과 데이터형 설정

테이블

컬럼

|   | Name | Number   | sign | Network |
|---|------|----------|------|---------|
|   | 이태성  | 12345678 |      | 76      |
| ▶ | 홍길동  | 23456789 |      | 86      |
| * |      |          |      |         |

레코드

필드

# DataColumn : 개별 항목

## ■ DataColumn 생성

DataColumn()

DataColumn(String)

DataColumn(String, Type)

DataColumn(String, Type, String)

DataColumn(String, Type, String,  
MappingType)

## DataType

Boolean, Byte, Char,  
DateTime, Decimal,  
Double, Guid, Int16,  
Int32, Int64, SByte,  
Single, String,  
TimeSpan, UInt16,  
UInt32, UInt64

# DataColumn

## ■ DataType 비교

|                 |          |
|-----------------|----------|
| INT             | Int32    |
| BIGINT          | Int64    |
| NVARCHAR, NCHAR | String   |
| DATETIME        | DateTime |
| BIT             | Boolean  |
| DECIMAL         | Decimal  |
| VARBINARY       | byte[]   |

SQL 서버 타입

닷넷 타입

# DataColumn : 개별 항목

## ■ DataColumn 항목 생성과 설정

|        |              |     |      |
|--------|--------------|-----|------|
| 열이름    | Name         | Age | Male |
| 데이터 형식 | nvarchar(10) | int | bit  |

```
static void Main(string[] args)
{
    // 1 단계 DataTable과 DataColumn 만들기
    DataTable table1 = new DataTable("table1");
    DataColumn Col1 = new DataColumn("Name", typeof(string));
    Col1.MaxLength = 10;
    Col1.AllowDBNull = false;
    table1.Columns.Add(Col1);
    Col1 = new DataColumn("Age", typeof(int));
    table1.Columns.Add(Col1);
    Col1 = new DataColumn("Male", typeof(bool));
}
```

# DataRow : 레코드

- DataRow  
행 데이터

- DataRow 생성과 데이터 입력

```
DataRow row = new table1.NewRow(); // 테이블과 같은 속성  
row["Name"] = "철수";                // 의 행  
row["Age"] = 12;  
row["Male"] = true;  
table1.Rows.Add(row);
```

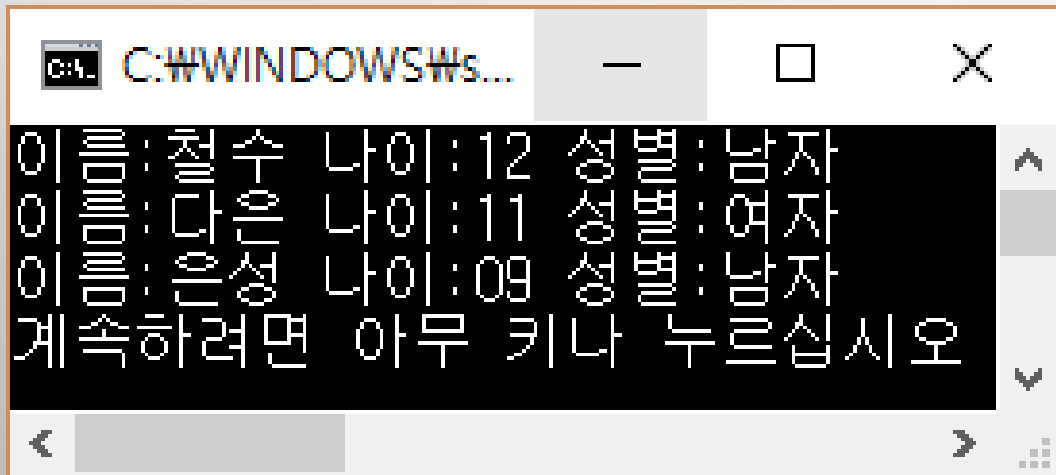


DataTable.DataRowCollection

# DataTable

- DataRow 원소 다루기  
배열 형식 + 컬럼명  
`row["컬럼명"]` → 결과는 모두 string 형
- 행의 내용 출력하기

[소스 보기](#)



```
C:\WINDOWS#s...
이름:철수 나이:12 성별:남자
이름:다은 나이:11 성별:여자
이름:은성 나이:09 성별:남자
계속하려면 아무 키나 누르십시오
```

# DataTable 행 검색 #1

- DataTable.Select()

`public DataRow[] Select( string filterExpression )`

- 필터 만들기

- 숫자는 비교문으로 문자열 만듬

`string Filter = "Age >= 12";`

- 문자열 비교는 작은 따옴표 명시

`string Filter = "Name = '철수'";`

`string Filter = "Name = '" + textBox1.Text + "'";`

[소스 보기](#)



# DataTable 행 검색 #2

- DataTable.Rows.Find()
  - DataTable.Select()에 비해 활용도가 낮음
  - 반드시 기본 키 값이 있어야만 함
- 기본키 설정
  - DataTable.PrimaryKey 에 컬럼 설정

```
public DataColumn[] PrimaryKey { get; set; }
```

[소스 보기](#)

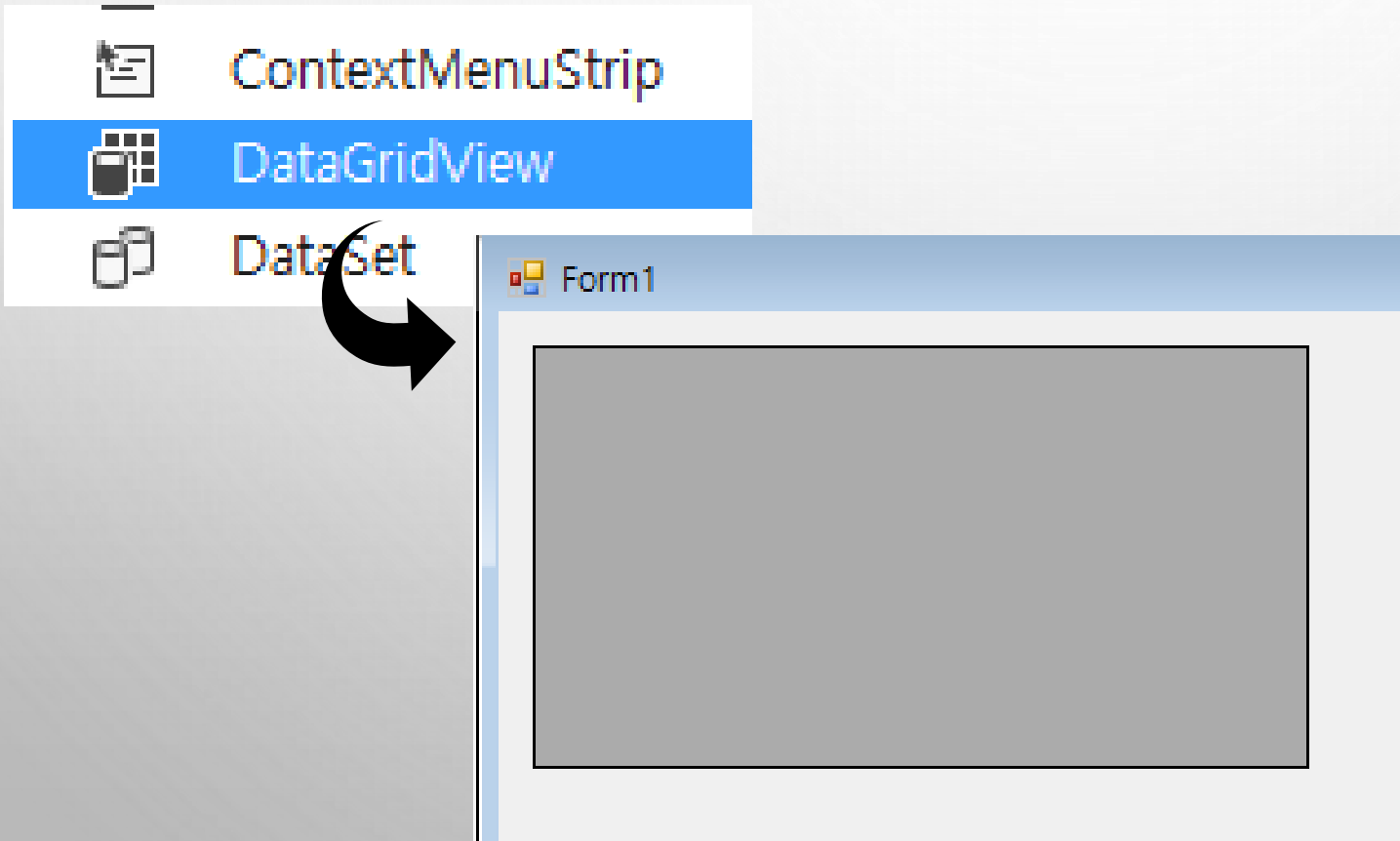


# DataGridView

# DataGridView

- DataGridView

- 표에 데이터를 표시하는 컨트롤
- 지정한 DataTable의 모든 내용을 표에 출력



# DataGridView

- 주요 속성

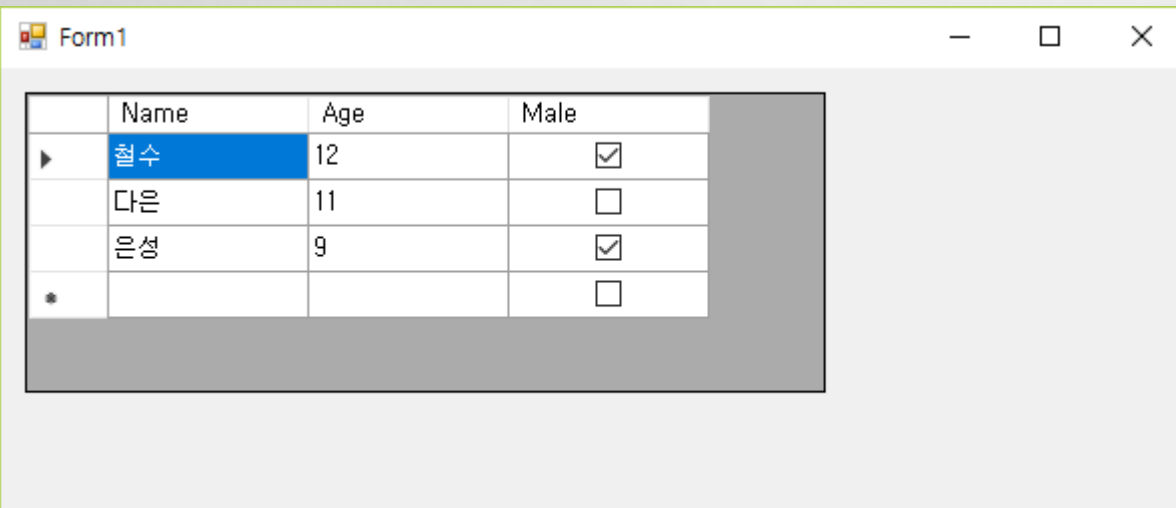
- DataGridView.DataSource**

- 데이터 원본 설정

- 출력할 DataTable 설정

- `DataGridView1.DataSource = table1;`

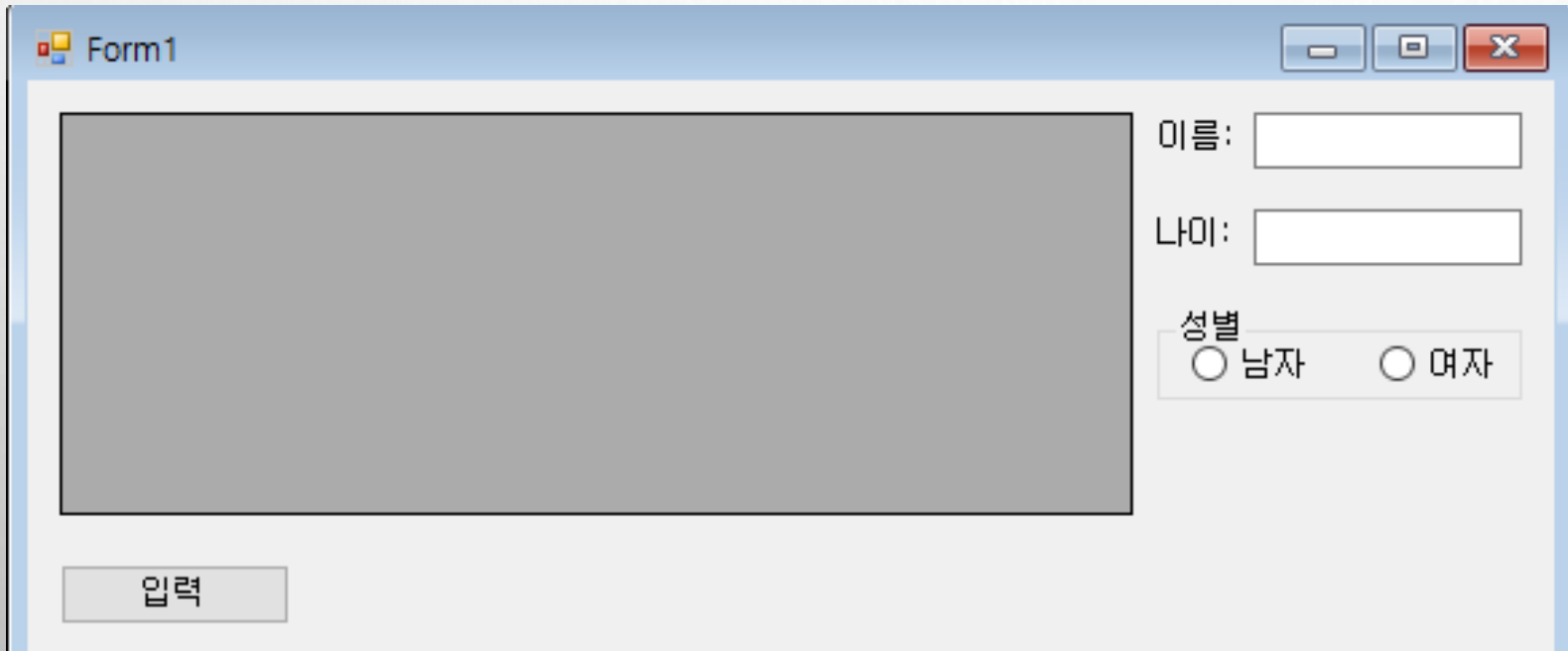
[소스 보기](#)



|   | Name | Age | Male                                |
|---|------|-----|-------------------------------------|
| ▶ | 철수   | 12  | <input checked="" type="checkbox"/> |
|   | 다은   | 11  | <input type="checkbox"/>            |
|   | 은성   | 9   | <input checked="" type="checkbox"/> |
| * |      |     | <input type="checkbox"/>            |

# DataGridView

- DataGridView에 데이터 입력  
별도의 메서드 호출 없이 Table에 Row추가하면 됨



The screenshot shows a Windows Form titled "Form1". On the left side of the form is a large, empty DataGridView. To the right of the DataGridView are three input fields: "이름:" (Name) with a text box, "나이:" (Age) with a text box, and "성별" (Gender) with two radio buttons labeled "남자" (Male) and "여자" (Female). At the bottom left of the form is a button labeled "입력" (Input).

[소스 보기](#)

# DataGridView

## ■ DataGridView 행 선택 모드 변경

|               |               |
|---------------|---------------|
| ScrollBars    | Both          |
| SelectionMode | FullRowSelect |
| SelectionMode | FullRowSelect |

The screenshot shows the Microsoft Developer Network (MSDN) website. The navigation bar at the top includes links for '다운로드' (Downloads), '프로그램' (Programs), '커뮤니티' (Community), and '설명서' (Documentation). The '설명서' link is circled in red. Below the navigation bar, the breadcrumb path 'MSDN Library > .NET 개발 > .NET Framework 4.6 and 4.5' is highlighted with a red circle. A red arrow points from this breadcrumb to the '개발 가이드' (Development Guide) link in the left sidebar, which is also circled in red. A yellow box on the right side of the page contains the text: '문서를 영문으로 보려면 영문 호트를 표시할 수도 있습니다.' (You may also want to display the English hotkey to view the document in English.) The main content area shows the title '.NET Framework'.

## ▼ 클라이언트 응용 프로그램

▶ Windows Presentation  
Foundation

▶ Windows Forms

▶ 일반 클라이언트 기술

콘솔 응용 프로그램 만들기

## ▼ Windows Forms 시작

Windows Forms 개요

- ▶ 새 Windows Form 만들기
- ▶ Windows Forms에서 이벤트 처리  
기 만들기
- ▶ Windows Forms의 크기 및 배율  
조정
- ▶ Windows Forms의 모양 변경
- ▶ Windows Forms 컨트롤
- ▶ Windows Forms에 사용자 입력

## ▼ Windows Forms 컨트롤

- ▶ Windows Forms에서 컨트롤 정렬
- ▶ Windows Forms에 컨트롤 넣기
- ▶ 개별 Windows Forms 컨트롤 레이블 지정 및 바로 가기 제공
- ▶ Windows Forms에 사용할 수 있는 컨트롤
- ▶ .NET Framework에서 사용자 지정 Windows Forms 컨트롤 개발
- ▶ .NET Framework의 기능별 Windows Forms 컨트롤
- ▶ 디자인할 때 Windows Forms 컨트롤 개발

## ▼ Windows Forms에 사용할 수 있는 컨트롤

기능별 Windows Forms 컨트롤

소유자가 그린 기본 제공 컨트롤 지원

- ▶ BackgroundWorker 구성 요소
- ▶ BindingNavigator 컨트롤
- ▶ BindingSource 구성 요소
- ▶ Button 컨트롤
- ▶ CheckBox 컨트롤
- ▶ CheckedListBox 컨트롤
- ▶ ColorDialog 구성 요소
- ▶ ComboBox 컨트롤
- ▶ ContextMenu 구성 요소
- ▶ ContextMenuStrip 컨트롤
- ▶ DataGrid 컨트롤
- ▶ DataGridView 컨트롤

# DataGridView

## ■ DataGridView 행 선택 모드 변경

|               |               |
|---------------|---------------|
| ScrollBars    | Both          |
| SelectionMode | FullRowSelect |
| SelectionMode | FullRowSelect |

## ■ DataGridView를 이용한 행 삭제

### - DataRow를 삭제하는 방법

#### 1) DataRow.Delete()

해당 행의 RowState속성을 Deleted로 바꿀뿐  
삭제 하지 않음, 삭제 취소가 가능

➔ AcceptChanges() 호출 : 영구히 삭제

[소스 보기](#)



# DataGridView

- DataGridView를 이용한 행 삭제
    - DataGridView에서 삭제하는 방법
- DataGridView와 DataTable이 서로 연동

DataGridViewRow로 삭제   
DataGridView.Rows.Remove( )

[소스 보기](#)