

Chapter 09 인터페이스

01 인터페이스 소개

02 인터페이스 생성

03 인터페이스 멤버

04 인터페이스 다중 상속

05 함께하는 응용 예제

06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기

요약

연습문제

Section 01 인터페이스 소개(1)

- 특정 클래스 만들 때 사용하는 규약(실수하지 않게 도와주는 기능)
- 특별한 기능이 아니라 그냥 '최소 사항' 만듬

■ Comparable 인터페이스

- 인터페이스는 모두 대문자 I로 시작
- 클래스에서 해당 인스턴스 정렬 및 순서를 지정하기 위해 구현하는 일반화된 형식별 비교 메서드 정의
- Comparable 필수 CompareTo 메서드 정의

Section 01 인터페이스 소개(1)

■ 기본예제 9-1 IComparable 인터페이스 활용(교재 426p)

/9장/IComparableBasic

```
namespace IComparableBasic
{
    class Program
    {
        class Product
        {
            public string Name { get; set; }
            public int Price { get; set; }

            public override string ToString()
            {
                return Name + " : " + Price + "원";
            }
        }
    }
}
```

```
static void Main(string[] args)
{
    List<Product> list = new List<Product>()
    {
        new Product() { Name = "고구마", Price = 1500 },
        new Product() { Name = "사과", Price = 2400 },
        new Product() { Name = "바나나", Price = 1000 },
        new Product() { Name = "배", Price = 3000 }
    };
    list.Sort();
    foreach (var item in list) { Console.WriteLine(item); }
```

Product 클래스를 어떤 기준으로
정렬해야 할지 몰라 오류발생

Section 01 인터페이스 소개(2)

- ① 기본적인 클래스와 자료 생성하기
- ② Comparable 인터페이스 상속하기

Product 클래스의 정렬 기준을 정해주기 위해 Comparable 인터페이스 구현

코드 9-2 Comparable 인터페이스 상속

```
01 class Product : Comparable
02 {
03     public string Name { get; set; }
04     public int Price { get; set; }
05
06     public override string ToString()
07     {
08         return Name + " : " + Price + "원";
09     }
10 }
```

Section 01 인터페이스 소개(3)

③ IComparable 인터페이스의 메서드 생성하기

IComparable 클래스에 커서 놓아 생기는 파란색 상자 클릭, 또는 커서를 놓고 **Ctrl** + **.** 단축키



그림 9-1 IComparable 인터페이스의 메서드 생성

Section 01 인터페이스 소개(4)

④ CompareTo() 메서드 구현하기

IComparable 인터페이스는 CompareTo() 메서드를 구현해야 함

코드 9-4

CompareTo() 메서드 구현

```
01 public int CompareTo(object obj)
02 {
03     return this.Price.CompareTo((obj as Product).Price);
04 }
```

실행 결과

바나나 : 1000원
고구마 : 1500원
사과 : 2400원
배 : 3000원

■ 인터페이스 인스턴스화

- 인터페이스는 실체 없는 규칙, 인스턴스화 할 수 없음(오류 발생)

코드 9-5 인터페이스 인스턴스화(오류 발생)

/9장/Interfaces

```
Comparable comparable = new Comparable();
```

Section 01 인터페이스 소개(5)

■ IDisposable 인터페이스

- using 블록 사용할 때 자동으로 호출되는 규약
- **기본예제 9-2** IDisposable 인터페이스 활용(교재 430p)

[/9장/IDisposableBasic](#)

① IDisposable 인터페이스 상속하기

코드 9-6

IDisposable 인터페이스 상속

```
01 class Program
02 {
03     class Dummy : IDisposable
04     {
05
06     }
07
08     static void Main(string[] args)
09     {
10
11     }
12 }
```


Section 01 인터페이스 소개(6)

② IDisposable 인터페이스의 메서드 생성하기

IDisposable 클래스에 커서 놓아 생기는 파란색 상자 클릭 또는 커서놓고 **Ctrl** + **.** 단축키 클릭



그림 9-2 IDisposable 인터페이스의 메서드 생성

Section 01 인터페이스 소개(7)

③ Dispose () 메서드 구현 및 호출하기

코드 9-8

Dispose() 메서드 구현 및 호출

```
01 class Program
02 {
03     class Dummy : IDisposable
04     {
05         public void Dispose()
06         {
07             Console.WriteLine("Dispose() 메서드를 호출합니다.");
08         }
09     }
10
11     static void Main(string[] args)
12     {
13         Dummy dummy = new Dummy();
14         dummy.Dispose();
15     }
16 }
```

Section 01 인터페이스 소개(8)

- ④ using 키워드와 IDisposable 인터페이스
 - using 블록을 벗어날 때 자동으로 Dispose() 메서드 호출한다.

코드 9-9 using 키워드와 IDisposable 인터페이스

```
01 static void Main(string[] args)
02 {
03     using (Dummy dummy = new Dummy())
04     {
05
06     }
07 }
```

실행 결과

Dispose() 메서드를 호출합니다.

Section 02 인터페이스 생성(1)

- 인터페이스 생성 방법

```
interface [인터페이스 이름]
{
}
```

- 인터페이스는 클래스와 동급의 카테고리
- 클래스를 생성하는 위치라면 어디든지 만들 수 있음
- Interface 키워드 사용

Section 02 인터페이스 생성(1)

■ 같은 파일에 인터페이스 생성

- 클래스를 생성할 때와 같은 위치에 인터페이스를 만듦 → 거의 없음

```
using System;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace InterfaceBasic
{
    interface ITest
    {
    }
    class Program
    {
        static void Main(string[] args)
        {
        }
    }
}
```

인터페이스를 생성합니다.

Section 02 인터페이스 생성(2)

■ 다른 파일에 인터페이스 생성

- 프로젝트 이름 마우스 오른쪽 버튼 - [추가] - [새 항목] 눌러 새 항목 추가 대화상자 오픈

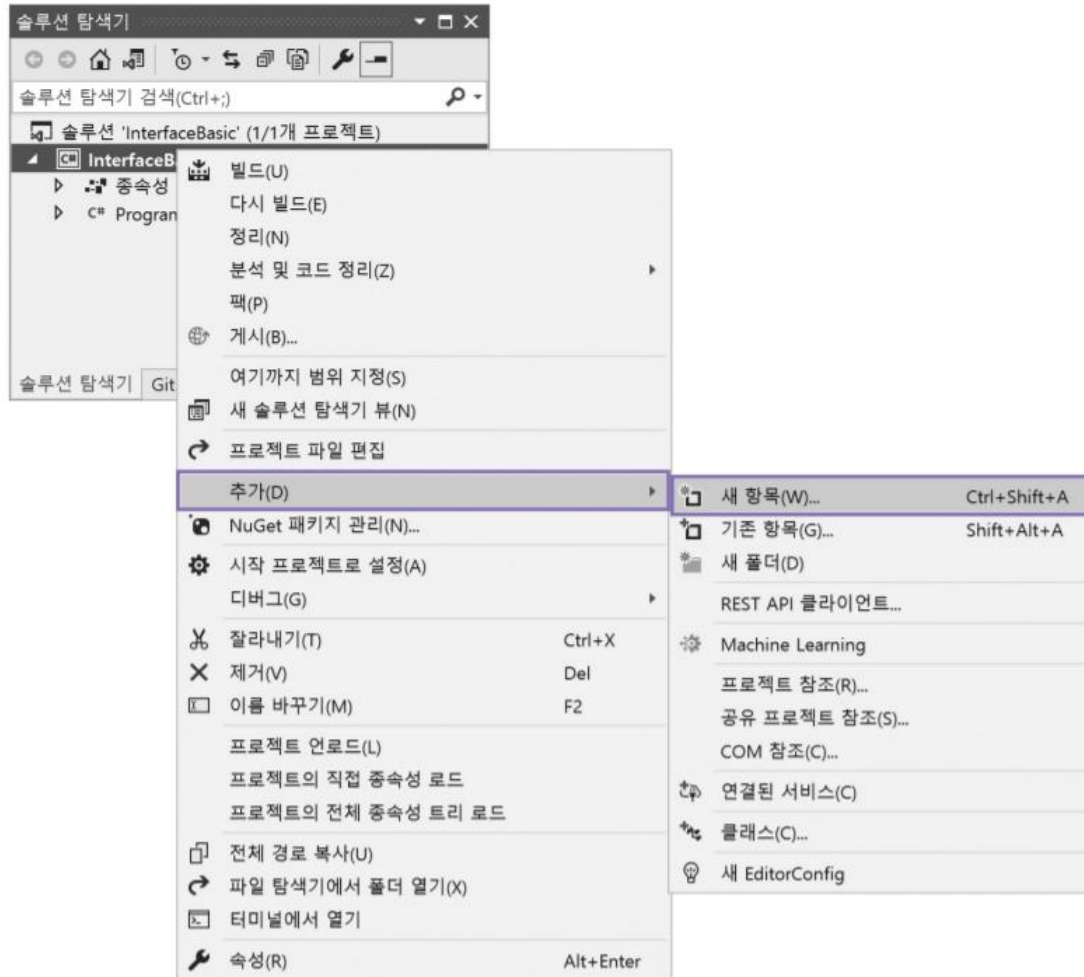


그림 9-3 새 항목 추가

Section 02 인터페이스 생성(3)

- 새 항목 대화상자 뜨면 [인터페이스] 선택, 원하는 이름으로 인터페이스 추가



그림 9-4 인터페이스 추가

Section 02 인터페이스 생성(4)

- 아래와 같은 인터페이스 생성됨

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace InterfaceBasic
{
    interface IBasic
    {

    }
}
```


Section 03 인터페이스 멤버(1)

■ 인터페이스 구현

- 클래스 내부에 넣을 수 있었던 인스턴스 메서드, 속성을 넣을 수 있다.
- 단, 내부 구현하는 코드는 넣을 수 없다.
- 인터페이스를 상속 받아서 사용!

- **기본예제 9-3** 인터페이스 구현(교재 436p)

/9장/InterfaceBasic

① 인터페이스 생성하기

코드 9-10

인터페이스 생성

```
01 interface IBasic
02 {
03     int TestInstanceMethod(); —— 메서드에 내부 구현을 입력할 수 없습니다.
04     int TestProperty { get; set; } —— 속성에도 마찬가지로 내부 구현을 입력할 수 없습니다.
05 }
```

Section 03 인터페이스 멤버(2)

② 인터페이스 상속하기

코드 9-11

인터페이스 상속

```
01 class Program
02 {
03     class TestClass : IBasic
04     {
05
06     }
07
08     static void Main(string[] args)
09     {
10
11     }
12 }
```

Section 03 인터페이스 멤버(3)

③ 인터페이스 구현하기

인터페이스 주변에 마우스 커서 올려 뜨는 파란색 커서 또는 **Ctrl** + **.** 단축키 클릭
자동으로 내부 코드가 생성되고, 이후에 내부 코드를 입력하면 된다.



그림 9-5 인터페이스 구현

Section 03 인터페이스 멤버(4)

■ 인터페이스 다형성

- 인터페이스를 구현한 클래스는 인터페이스를 상속한다.

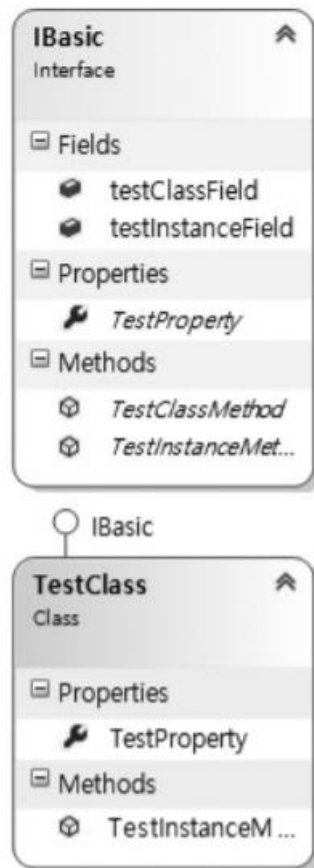


그림 9-6 인터페이스 상속

코드 9-13

인터페이스 다형성

/9장/Interfaces

```
01 static void Main(string[] args)
02 {
03     IBasic basic = new TestClass();
04 }
```

Section 04 인터페이스 다중 상속(1)

- 하나의 클래스가 여러 부모 클래스 가질 수 있게 되는 것
- 클래스 상속과 인터페이스 상속 함께 활용, 하나의 클래스가 여러 다형성 가지게 됨

코드 9-14 다중 상속

/9장/Interfaces

```
01 class Program
02 {
03     class Parent { }
04
05     class Child : Parent, IDisposable, IComparable
06     {
07         public void Dispose()
08         {
09             throw new NotImplementedException();
10         }
11
12         public int CompareTo(object obj)
13         {
14             throw new NotImplementedException();
15         }
16     }
17
18     static void Main(string[] args)
19     {
20
21     }
22 }
```

한 개의 클래스와 두 개의 인터페이스를 상속받습니다.

IDisposable 인터페이스 구현

IComparable 인터페이스 구현

Section 04 인터페이스 다중 상속(2)

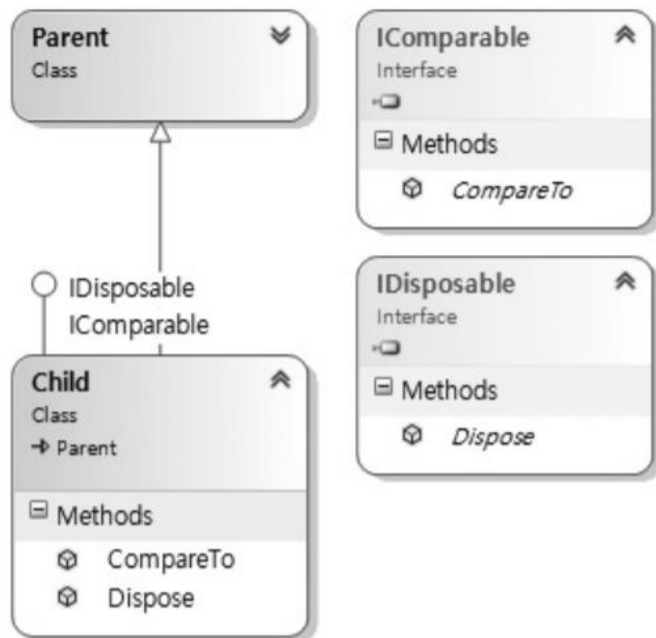
■ 자료형 변환

코드 9-15 다중 상속과 다형성

/9장/Interfaces

```
01 Child child = new Child();  
02 Parent childAsParent = new Child();  
03 IDisposable childAsDisposable = new Child();  
04 IComparable childAsComparable = new Child();
```

■ 인터페이스를 활용하면 코드에 규약을 부여해 여러 사람과 작업 시 안정성 높임



다소 어려운
개념!

그림 9-7 [코드 9-14]의 상속 관계

Section 05 함께 하는 응용 예제(1)

■ 응용예제 9-1 파일처리(교재 442p)

/9장/FileProcess

① 한번에 읽고 쓰기 – 파일에 문자열 쓰기

- @기호를 붙여 문자열을 만들면 내부에서 이스케이프 문자는 전혀 사용할 수 없다.
대신 'W' 기호를 한 번만 입력해도 되므로 코드의 가독성이 좋아짐
- **test** 폴더를 미리 만들어 놓는다 → 없으면 예외 발생!

```
class Program
{
    static void Main(string[] args)
    {
        // 한 번에 쓰고 읽기
        File.WriteAllText(@"c:\Wtest\Wtest.txt", "문자열을 파일에 씁니다");
        Console.WriteLine(File.ReadAllText(@"c:\Wtest\Wtest.txt"));
    }
}
```

실행 결과

문자열을 파일에 씁니다.

Section 05 함께 하는 응용 예제(1)

■ 응용예제 9-1 파일처리(교재 442p)

/9장/FileProcess

② 스트림으로 쓰기 – 한 줄 씩 쓰기(StreamWriter 클래스 사용)

using 구문에서 StreamWriter 클래스의 인스턴스 생성

```
class Program
{
    static void Main(string[] args)
    {
        // 스트림으로 쓰기
        using (StreamWriter writer = new StreamWriter(@"c:\test\test.txt"))
        {
            writer.WriteLine("안녕하세요");
            writer.WriteLine("StreamWriter 클래스를 사용해");
            writer.WriteLine("글자를 여러 줄 입력해봅니다");

            for (int i = 0; i < 10; i++)
            {
                writer.WriteLine("반복문 - " + i);
            }
        }
        Console.WriteLine(File.ReadAllText(@"c:\test\test.txt"));
    }
}
```

실행 결과

```
안녕하세요
StreamWriter 클래스를 사용해
글자를 여러 줄 입력해봅니다
반복문 - 0
반복문 - 1
반복문 - 2
반복문 - 3
반복문 - 4
반복문 - 5
반복문 - 6
반복문 - 7
반복문 - 8
반복문 - 9
```


Section 05 함께 하는 응용 예제(1)

■ 응용예제 9-1 파일처리(교재 442p)

/9장/FileProcess

③ 스트림으로 읽기 – StreamReader 클래스 사용

- 여러 줄 읽기(반복문 사용)

```
class Program
{
    static void Main(string[] args)
    {
        // 스트림으로 읽기
        using (StreamReader reader = new StreamReader(@"c:\Wtest\Wtest.txt"))
        {
            string line;
            while ((line = reader.ReadLine()) != null)
            {
                Console.WriteLine(line);
            }
        }
    }
}
```

실행 결과

안녕하세요
StreamWriter 클래스를 사용해
글자를 여러 줄 입력해봅니다
반복문 - 0
반복문 - 1
반복문 - 2
반복문 - 3
반복문 - 4
반복문 - 5
반복문 - 6
반복문 - 7
반복문 - 8
반복문 - 9

■ 레이블과 링크 레이블

- 레이블과 링크 레이블의 차이점
 - 글자에 밑줄이 그어지는 것

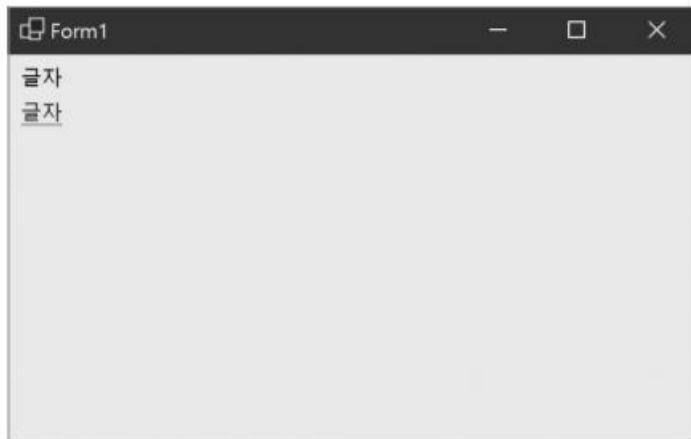


그림 9-9 레이블과 링크 레이블

코드 9-23 레이블과 링크 레이블 생성

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();

        // 레이블과 링크 레이블을 생성합니다.
        Label label = new Label()
        {
            Text = "글자",
            Location = new Point(10, 10),
        };
        LinkLabel linkLabel = new LinkLabel()
        {
            Text = "글자",
            Location = new Point(10, 50)
        };

        // 요소에 이벤트를 연결합니다.
        linkLabel.Click += LabelClick;

        // 요소를 화면에 추가합니다.
        Controls.Add(label);
        Controls.Add(linkLabel);
    }

    private void LabelClick(object sender, EventArgs e)
    {
        throw new NotImplementedException();
    }
}
```

Section 06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (2)

- 웹페이지로의 링크

코드 9-24 웹 페이지로의 링크

```
01 private void LabelClick(object sender, EventArgs e)
02 {
03     System.Diagnostics.Process.Start("http://hanb.co.kr");
04 }
```

- 응용 프로그램으로의 링크

코드 9-25 응용 프로그램으로의 링크

```
01 private void LabelClick(object sender, EventArgs e)
02 {
03     System.Diagnostics.Process.Start("notepad.exe");
04 }
```

■ 체크 박스 : 한 번에 여러 개의 조건 선택할 수 있는 요소

코드 9-26 체크 박스 생성

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();

        // 라디오 버튼을 생성합니다.
        CheckBox checkBox1 = new CheckBox();
        CheckBox checkBox2 = new CheckBox();
        CheckBox checkBox3 = new CheckBox();
        Button button = new Button();

        // 요소의 속성을 설정합니다.
        checkBox1.Text = "감자";
        checkBox2.Text = "고구마";
        checkBox3.Text = "토마토";
        button.Text = "클릭";

        checkBox1.Location = new Point(10, 10);
        checkBox2.Location = new Point(10, 40);
        checkBox3.Location = new Point(10, 70);
        button.Location = new Point(10, 100);
    }
}
```

Section 06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (4)

```
// 요소에 이벤트를 연결합니다.  
button.Click += ButtonClick;
```

```
// 요소를 화면에 추가합니다.  
Controls.Add(checkBox1);  
Controls.Add(checkBox2);  
Controls.Add(checkBox3);  
Controls.Add(button);
```

```
}
```

```
private void ButtonClick(object sender, EventArgs e)  
{  
    throw new NotImplementedException();  
}  
}
```



그림 9-10 체크 박스

Section 06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (5)

■ 체크 박스 활용

코드 9-27 체크 박스 활용

```
01 private void ButtonClick(object sender, EventArgs e)
02 {
03     // 리스트를 생성합니다.
04     List<string> list = new List<string>();
05
06     // 리스트에 체크된 요소를 추가하고자 반복을 돌립니다.
07     foreach (var item in Controls)
08     {
09         if(item is CheckBox)
10         {
11             CheckBox checkBox = (CheckBox)item;
12             if(checkBox.Checked)
13             {
14                 list.Add(checkBox.Text);
15             }
16         }
17     }
18
19     // 리스트를 붙여 문자열을 만듭니다.
20     MessageBox.Show(string.Join(",", list));
21 }
```

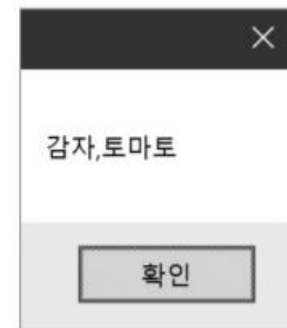


그림 9-11 체크 박스 활용

■ 라디오 버튼 : 하나의 그룹에서 한 번에 하나의 버튼만 선택 가능

코드 9-28 라디오 버튼 생성

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();

        // 라디오 버튼을 생성합니다.
        RadioButton radio1 = new RadioButton();
        RadioButton radio2 = new RadioButton();
        RadioButton radio3 = new RadioButton();
        Button button = new Button();

        // 요소의 속성을 설정합니다.
        radio1.Text = "감자";
        radio2.Text = "고구마";
        radio3.Text = "토마토";
        button.Text = "클릭";
```

Section 06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (7)

```
radio1.Location = new Point(10, 10);  
radio2.Location = new Point(10, 40);  
radio3.Location = new Point(10, 70);  
button.Location = new Point(10, 100);
```

```
// 요소에 이벤트를 연결합니다.
```

```
button.Click += ButtonClick;
```

```
// 요소를 화면에 추가합니다.
```

```
Controls.Add(radio1);
```

```
Controls.Add(radio2);
```

```
Controls.Add(radio3);
```

```
Controls.Add(button);
```

```
}
```

```
private void ButtonClick(object sender, EventArgs e)
```

```
{
```

```
    throw new NotImplementedException();
```

```
}
```

```
}
```



그림 9-12 라디오 버튼

Section 06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (8)

■ 라디오 버튼 활용

코드 9-29 라디오 버튼 활용

```
private void ButtonClick(object sender, EventArgs e)
{
    foreach (var item in Controls)
    {
        if(item is RadioButton)
        {
            RadioButton radioButton = (RadioButton)item;
            if(radioButton.Checked)
            {
                MessageBox.Show(radioButton.Text);
            }
        }
    }
}
```

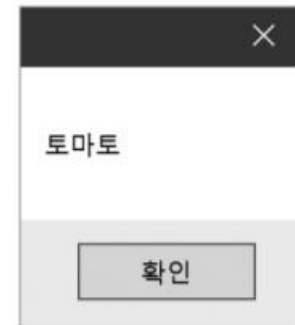


그림 9-13 라디오 버튼 활용

■ 그룹 박스 : 그룹을 나누고 그룹마다 하나씩 선택하고 싶을 때 사용

코드 9-30 라디오 박스 생성

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();

        // 라디오 버튼을 생성합니다.
        RadioButton radio1 = new RadioButton();
        RadioButton radio2 = new RadioButton();
        RadioButton radio3 = new RadioButton();
        RadioButton radio4 = new RadioButton();
        Button button = new Button();

        // 요소의 속성을 설정합니다.
        radio1.Text = "감자";
        radio2.Text = "고구마";
        radio3.Text = "광어";
        radio4.Text = "우럭";
        button.Text = "클릭";
```

Section 06 원도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (10)

```
radio1.Location = new Point(10, 10);  
radio2.Location = new Point(10, 40);  
radio3.Location = new Point(150, 10);  
radio4.Location = new Point(150, 40);  
button.Location = new Point(10, 70);
```

```
// 요소에 이벤트를 연결합니다.
```

```
button.Click += ButtonClick;
```

```
// 요소를 화면에 추가합니다.
```

```
Controls.Add(radio1);
```

```
Controls.Add(radio2);
```

```
Controls.Add(radio3);
```

```
Controls.Add(radio4);
```

```
Controls.Add(button);
```

```
}
```

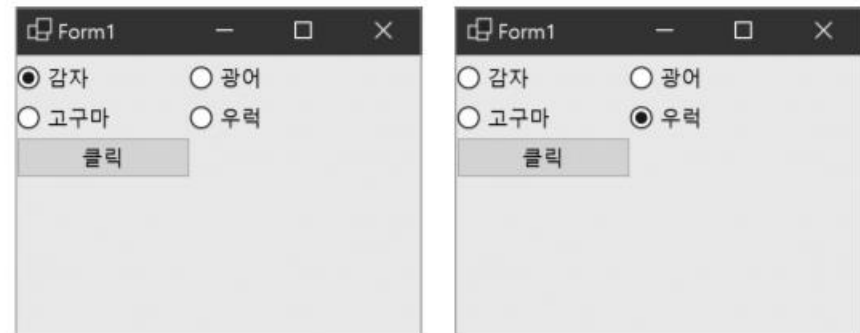


그림 9-14 하나밖에 선택하지 못하는 라디오 버튼

```
private void ButtonClick(object sender, EventArgs e)
```

```
{
```

```
    /* 생략 */
```

```
}
```

```
}
```

Section 06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (11)

코드 9-31 그룹 박스 생성

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();

        // 라디오 버튼을 생성합니다.
        GroupBox groupBox1 = new GroupBox();
        GroupBox groupBox2 = new GroupBox();
        RadioButton radio1 = new RadioButton();
        RadioButton radio2 = new RadioButton();
        RadioButton radio3 = new RadioButton();
        RadioButton radio4 = new RadioButton();
        Button button = new Button();

        // 요소의 속성을 설정합니다.
        groupBox1.Text = "식물";
        groupBox2.Text = "물고기";
        radio1.Text = "감자";
        radio2.Text = "고구마";
        radio3.Text = "광어";
        radio4.Text = "우럭";
        button.Text = "클릭";
```

Section 06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (12)

```
groupBox1.Size = new Size(120, 100);  
groupBox2.Size = new Size(120, 100);  
groupBox1.Location = new Point(10, 10);  
groupBox2.Location = new Point(140, 10);
```

```
radio1.Location = new Point(10, 30);  
radio2.Location = new Point(10, 60);  
radio3.Location = new Point(10, 30);  
radio4.Location = new Point(10, 60);  
button.Location = new Point(10, 120);
```

// 요소에 이벤트를 연결합니다.

```
button.Click += ButtonClick;
```

Section 06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (13)

```
// 라디오 버튼을 그룹 박스에 추가합니다.  
groupBox1.Controls.Add(radio1);  
groupBox1.Controls.Add(radio2);
```

```
groupBox2.Controls.Add(radio3);  
groupBox2.Controls.Add(radio4);
```

```
// 요소를 화면에 추가합니다.
```

```
Controls.Add(groupBox1);  
Controls.Add(groupBox2);  
Controls.Add(button);
```

```
}
```

```
private void ButtonClick(object sender, EventArgs e)  
{  
    throw new NotImplementedException();  
}  
}
```

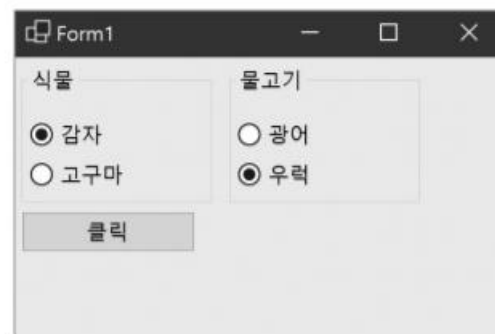


그림 9-15 그룹에서 하나씩 선택되는 라디오 버튼

Section 06 윈도 폼: 레이블, 링크 레이블, 체크 박스, 라디오 버튼, 그룹 박스 사용하기 (14)

- 그룹 박스와 라디오 버튼 활용

코드 9-32 그룹 박스와 라디오 버튼 활용

```
private void ButtonClick(object sender, EventArgs e)
{
    foreach (var outerItem in Controls)
    {
        if(outerItem is GroupBox)
        {
            foreach(var innerItem in ((GroupBox)outerItem).Controls)
            {
                RadioButton radioButton = innerItem as RadioButton;
                if(radioButton != null && radioButton.Checked)
                {
                    MessageBox.Show(radioButton.Text);
                }
            }
        }
    }
}
```



그림 9-16 그룹마다 선택되는 라디오 버튼