

본 강의에서 수업자료로 이용되는 저작물은
저작권법 제25조 수업목적 저작물 이용 보상금제도에 의거,
한국복제전송저작권협회와 약정을 체결하고 적법하게 이용하고 있습니다.
약정범위를 초과하는 사용은 저작권법에 저촉될 수 있으므로
수업자료의 재 복제, 대중 공개·공유 및 수업 목적 외의 사용을 금지합니다.

2024. 8. 30.

부천대학교·한국복제전송저작권협회

C#

2주차 3교시

8장 컨트롤(2)

2주차 학습 내용

* 1교시

* 2장 C# 언어 구조 2

- * 자료형(값형:열거형, 참조형: 배열)
- * 형 변환
- * 박싱과 언박싱

* 3장 문장 종류

- * 제어문

* 2교시

* 8장 컨트롤

- * 윈폼 애플리케이션 작성 순서(7장 참조)
- * 버튼 기반 컨트롤(버튼, 라디오 버튼)

* 3교시

* 8장 컨트롤

- * 버튼 기반 컨트롤(체크 상자)
- * 레이블과 링크 레이블
- * 텍스트 상자

* 8장 Control

=> 멤버(속성 + 메서드 + 이벤트)

* Form

- * 속성 : Text (속성 값 : 문자열)

* Button

- * 속성 : Text (속성 값 : 문자열)
FlatStyle(속성 값 : 4개(FlatStyle 열거형))
- * 이벤트 : Click

* CheckBox와 RadioButton

- * 속성: Checked(속성 값: true, false)
- * 이벤트 : CheckChanged

* Label

- * 속성: Text, Font
BorderStyle(속성 값 3개(열거형))
TextAlign(속성 값 : 9개(열거형))

* LinkLabel

- * 속성: Text
- * 이벤트 : LinkClicked

* TextBox

- * 속성: Text, Multiline, PasswordChar

C# 프로그래밍



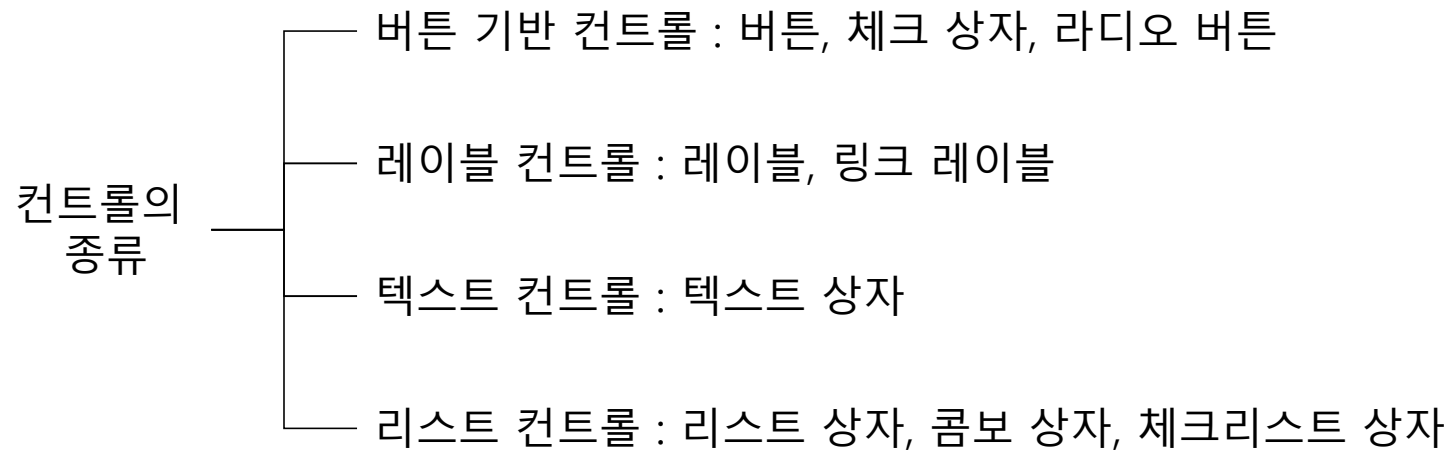
8. 컨트롤 (2)



컨트롤이란?

■ 컨트롤

- 화면에 표시되어 사용자와 상호작용을 수행하는 컴포넌트를 의미.
- 종류 : 버튼, 레이블, 텍스트, 리스트



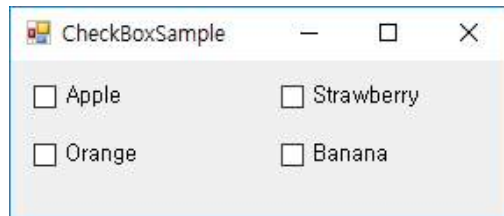


체크상자 [1/5]

■ 체크상자

- 주어진 항목들 중에서 선택할 수 있는 컨트롤
- 특징
 - 주어진 항목을 **복수로 선택할** 수 있음.

■ 체크상자의 기본적인 형태





체크상자 [2/5]

■ 체크 상자의 프로퍼티

- Control 클래스를 상속받은 컨트롤이기 때문에 버튼과 대부분 동일.
- 체크 상자의 체크 상태를 나타내는 Checked 프로퍼티가 추가로 제공.

■ Checked 프로퍼티

- 참으로 설정하면 네모부분에 '✓'가 표시.
- Checked 프로퍼티의 값이 변경될 때마다 CheckedChanged 이벤트 발생.



체크상자 [3/5]

■ 예제 8.2 [w2_ex8_2_xxxx]

- 4개의 항목을 갖는 체크 상자를 만들고 선택된 항목의 이름을 출력하는 예제.

예제 8.2 [w2_ex8_2_xxxx]

1) 디자인

컨트롤 : (Name)	프로퍼티	값
Form : Form1	Text	CheckBoxApp
CheckBox : checkBox1	Text	Apple
	Checked	True
CheckBox : checkBox2	Text	Strawberry
	Checked	False
CheckBox : checkBox3	Text	Orange
	Checked	False
CheckBox : checkBox4	Text	Banana
	Checked	False
Label : label1	Text	Apple



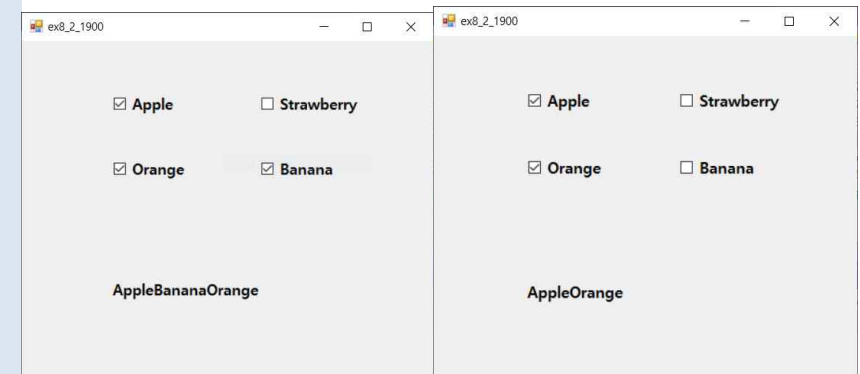
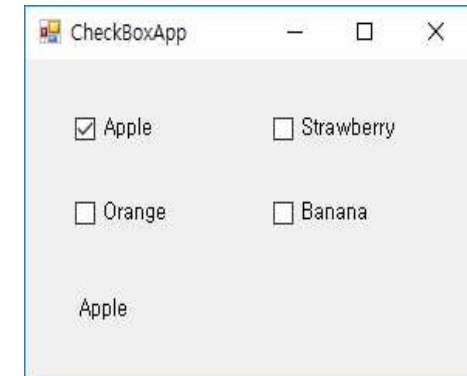
체크상자 [4/5]

컨트롤 : (Name)	이벤트	메소드명
CheckBox : checkBox1	CheckedChange	checkBox1_CheckedChanged()
CheckBox : checkBox2	CheckedChange	checkBox2_CheckedChanged()
CheckBox : checkBox3	CheckedChange	checkBox3_CheckedChanged()
CheckBox : checkBox4	CheckedChange	checkBox4_CheckedChanged()

2) 코드

```
private string strTemp;
private void UpdateLabel(string s, bool b) {
    if (b) { label1.Text += s; }
    else {
        strTemp = label1.Text;
        int i = strTemp.IndexOf(s);
        label1.Text = strTemp.Remove(i, s.Length);
    }
}
private void checkBox1_CheckedChanged(object sender, EventArgs e) {
    UpdateLabel(checkBox1.Text, checkBox1.Checked);
}
private void checkBox2_CheckedChanged(object sender, EventArgs e) {
    UpdateLabel(checkBox2.Text, checkBox2.Checked);
}
private void checkBox3_CheckedChanged(object sender, EventArgs e) {
    UpdateLabel(checkBox3.Text, checkBox3.Checked);
}
private void checkBox4_CheckedChanged(object sender, EventArgs e) {
    UpdateLabel(checkBox4.Text, checkBox4.Checked);
}
```

실행 방법 : 체크 상자를 클릭하여 선택하거나 해제한다.
실행 결과 :

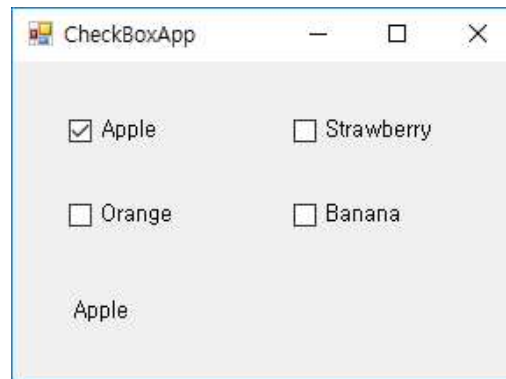




체크상자 [5/5]

실행 방법 : 체크 상자를 클릭하여 선택하거나 해제한다.

실행 결과 :





레이블 [1/5]

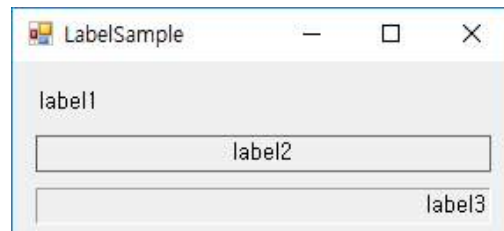
■ 레이블

- 각종 정보를 폼에 표시할 때 사용하는 컨트롤

- 레이블의 용도

1. 폼에 있는 컨트롤을 식별하는 **정보를 표시**.
2. 애플리케이션의 실행에 대한 응답 정보를 표시.
3. 특정 컨트롤을 클릭했을 때 실행되는 작업을 설명하는 메시지를 표시.

■ 레이블의 기본적인 형태





레이블 [2/5]

■ 레이블의 프로퍼티

- Font, BorderStyle, TextAlign

■ BorderStyle

- 레이블의 테두리를 설정할 때 사용하는 프로퍼티
- BorderStyle 열거형

기호상수	순서 값	설명
None	0	테두리가 없음
FixedSingle	1	단일 선 테두리
Fixed3D	2	3차원 테두리



레이블 [3/5]

■ TextAlign

- 레이블과 같은 컨트롤에서 표시되는 문자열에 대한 정렬을 위한 프로퍼티
- System.Drawing 네임스페이스에 포함된 ContentAlignment 열거형을 사용하여 지정할 수 있음.
 - ContentAlignment 열거형

기호상수	순서 값	설명
TopLeft	0x0001	좌측 상단 정렬
TopCenter	0x0002	가운데 상단 정렬
TopRight	0x0004	우측 상단 정렬
MiddleLeft	0x0010	좌측 중앙 정렬
MiddleCenter	0x0020	가운데 중앙 정렬
MiddleRight	0x0040	우측 중앙 정렬
BottomLeft	0x0100	좌측 하단 정렬
BottomCenter	0x0200	가운데 하단 정렬
BottomRight	0x0400	우측 중앙 정렬



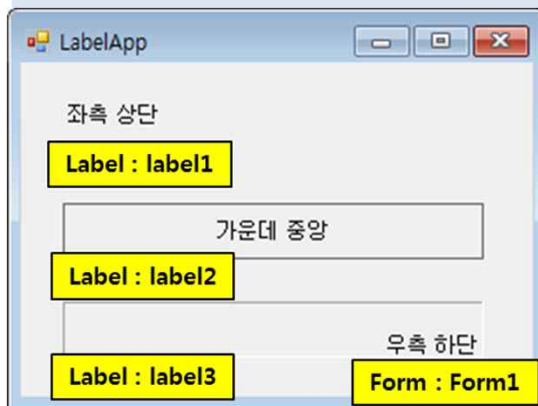
레이블 [4/5]

■ 예제 8.4 [w2_ex8_4_xxxx]

- 3개의 테두리와 9개의 문자열 정렬 중 대표적인 3개의 정렬을 사용한 예제.

예제 8.4 [w2_ex8_4_xxxx]

1) 디자인



컨트롤 : (Name)	프로퍼티	값
Form : Form1	Text	LabelApp
Label : label1	Text	좌측 상단
	BorderStyle	None
Label : label2	TextAlign	TopLeft
	Text	가운데 중앙
	BorderStyle	FixedSingle
Label : label3	TextAlign	MiddleCenter
	Text	우측 하단
	BorderStyle	Fixed3D
	TextAlign	BottomRight

실행 결과 :



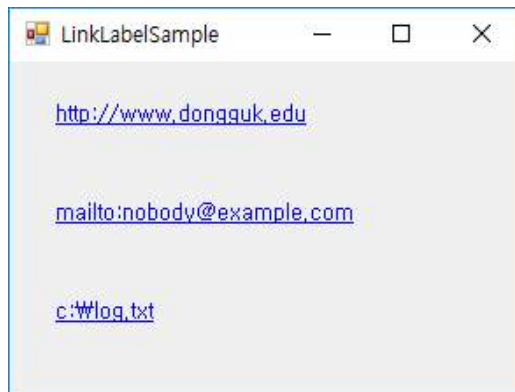


링크 레이블 [1/3]

■ 링크 레이블

- 하이퍼링크(hyper link)를 설정할 수 있는 레이블 컨트롤
- 레이블 컨트롤을 상속받았기 때문에 레이블의 기본적인 기능을 모두 가지고 있음.
- 링크 레이블은 링크를 클릭했을때 발생하는 LinkClicked 이벤트에 대한 이벤트 처리기를 작성.
 - 이벤트 처리를 위해서는 **Process** 클래스의 정적 메소드인 **Start()** 메소드를 사용함.

■ 링크 레이블의 기본적인 형태





링크 레이블 [2/3]

■ 예제 8.5 [w2_ex8_5_xxxx]

- 웹 브라우저를 실행 링크 레이블, 이메일 클라이언트를 실행하는 링크 레이블과 메모장 프로그램을 실행하는 링크 레이블을 구현하는 예제.

[예제 8.5 [w2_ex8_5_xxxx]

1) 폼 설계



컨트롤 : (Name)	프로퍼티	값
Form : Form1	Text	LinkLabelApp
LinkLabel : linkLabel1	Text	http://www.dongguk.edu
LinkLabel : linkLabel2	Text	mailto:nobody@example.com
LinkLabel : linkLabel3	Text	C:\log.txt

컨트롤 : (Name)	이벤트	메소드명
LinkLabel : linkLabel1	LinkClicked	linkLabel1_LinkClicked()
LinkLabel : linkLabel2	LinkClicked	linkLabel2_LinkClicked()
LinkLabel : linkLabel3	LinkClicked	linkLabel3_LinkClicked()



링크 레이블 [3/3]

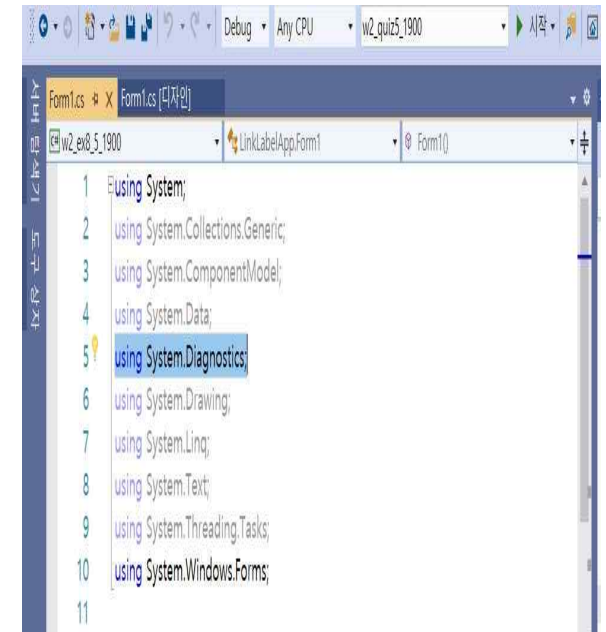
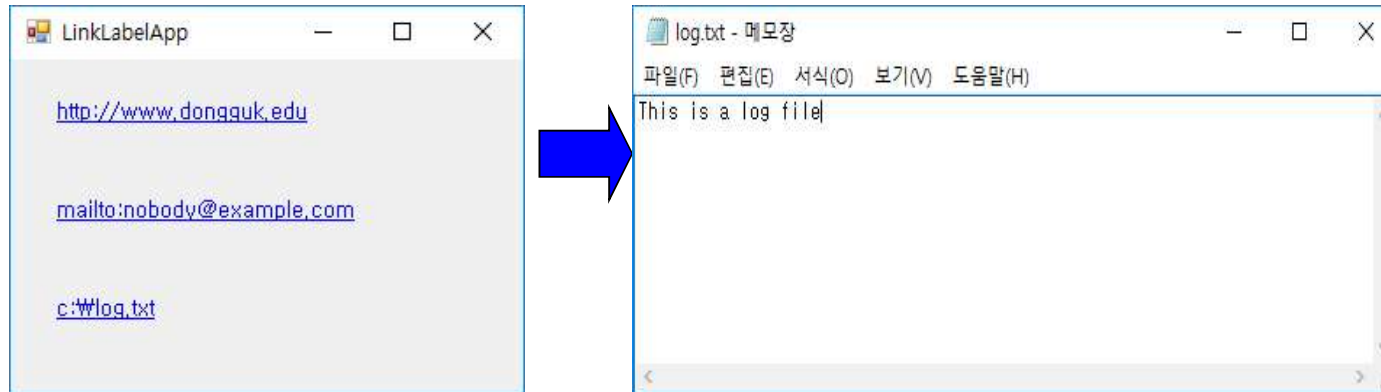
2) 코드

Using System.Diagnostics; //Process 클래스를 사용하려면 System.Diagnostics 네임스페이스 추가

```
private void linkLabel1_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e) {  
    Process.Start(linkLabel1.Text);  
}  
private void linkLabel2_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e) {  
    Process.Start(linkLabel2.Text);  
}  
private void linkLabel3_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e) {  
    Process.Start("notepad.exe",linkLabel3.Text);  
}
```

실행 방법 : 각각의 링크를 클릭

실행 결과 :





텍스트 상자 [1/4]

■ 텍스트 상자

- 사용자가 직접 텍스트를 입력할 수 있는 기본적인 컨트롤로서 사용자로부터 값을 입력 받거나 애플리케이션의 실행 결과를 출력하고자 할 때 유용하게 사용할 수 있음.

■ 텍스트 상자의 기본적인 형태





텍스트 상자 [2/4]

■ 텍스트 상자의 프로퍼티

프로퍼티	설명
MaxLength	텍스트 상자에 입력할 수 있는 최대 문자 수를 설정.
Multiline	텍스트 상자의 영역을 여러 줄로 설정.
PasswordChar	텍스트 상자의 암호 입력에 사용할 문자를 설정.
ReadOnly	텍스트 상자의 텍스트를 변경하지 못하도록 설정.
WordWrap	텍스트 상자의 텍스트가 영역을 초과할 경우 자동으로 줄을 바꾸도록 설정.
ScrollBars	Multiline 프로퍼티가 참인 경우 텍스트 상자에 스크롤 바를 설정 (None Horizontal Vertical Both)



텍스트 상자 [3/4]

■ 예제 8.6 [w2_ex8_6_xxxx]

- 사용자 이름과 패스워드를 입력받아 출력하는 예제.

예제 8.6 [w2_ex8_6_xxxx]

1) 디자인

컨트롤 : (Name)	프로퍼티	값
Form : Form1	Text	TextBox1App
Label : label1	Text	Enter your name
Label : label2	Text	Enter your password
TextBox : textBox1	Text	
TextBox : textBox2	Text	
	PasswordChar	*
TextBox : textBox3	Text	
	Multiline	True
Button : button1	Text	OK

컨트롤 : (Name)	이벤트	메소드명
Button : button1	Click	Button1_Click()



퀴즈 5번(미니 계산기)

■ 퀴즈 5 [w2_quiz5_xxxx]

- 다음과 같은 미니 계산기 프로그램을 작성하시오.

퀴즈 5 [w2_quiz5_xxxx]

1) 디자인

컨트롤 : (Name)	프로퍼티	값
TextBox : textBox1	Text	
TextBox : textBox2	Text	
TextBox : textBox3	Text	
	ReadOnly	True
나머지 생략		

2) 실행 결과

2주차 학습 내용 정리

* 1교시

* 2장 C# 언어 구조 2

- * 자료형(값형:열거형, 참조형: 배열)
- * 형 변환
- * 박싱과 언박싱

* 3장 문장 종류

- * 제어문

* 2교시

* 8장 컨트롤

- * 윈폼 애플리케이션 작성 순서(7장 참조)
- * 버튼 기반 컨트롤(버튼, 라디오 버튼)

* 3교시

* 8장 컨트롤

- * 버튼 기반 컨트롤(체크 상자)
- * 레이블과 링크 레이블
- * 텍스트 상자

* 8장 Control

=> 멤버(속성 + 메서드 + 이벤트)

* Form

- * 속성 : Text (속성 값 : 문자열)

* Button

- * 속성 : Text (속성 값 : 문자열)
FlatStyle(속성 값 : 4개(FlatStyle 열거형))
- * 이벤트 : Click

* CheckBox와 RadioButton

- * 속성: Checked(속성 값: true, false)
- * 이벤트 : CheckedChanged

* Label

- * 속성: Text, Font
BorderStyle(속성 값 3개(열거형))
TextAlign(속성 값 : 9개(열거형))

* LinkLabel

- * 속성: Text
- * 이벤트 : LinkClicked

* TextBox

- * 속성: Text, Multiline, PasswordChar