

컨트롤을 사용해
프로그램을
똑딱똑딱
만들어보자!

Chapter 03

윈도 폼 컨트롤 1

도구 상자에 있는 기본 컨트롤을 이용하여 화면을 디자인하는 방법을 알아보자. 이 장을 마치고 나면 기본 컨트롤을 사용하여 간단한 프로그램 정도는 쉽게 작성할 수 있게 될 것이다.

SECTION 01 윈도 폼 컨트롤이란?

SECTION 02 버튼 컨트롤

SECTION 03 레이블 컨트롤

SECTION 04 텍스트박스 컨트롤

SECTION 05 체크박스, 라디오버튼, 그룹박스 컨트롤

SECTION 06 리스트박스, 콤보박스 컨트롤

요약

연습문제

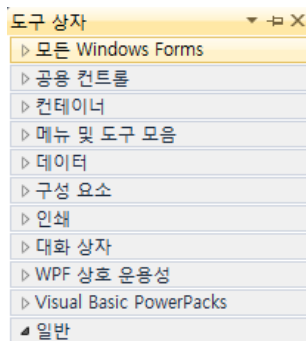
- 윈도 폼 컨트롤은 윈도 응용 프로그램의 화면을 구성하는 요소로 일정한 모양과 기능을 제공한다.

단순하고 익숙한 화면 구성은 사용자가 프로그램을 쉽게 이용하는 데 도움을 준다. 윈도 폼 컨트롤은 확인 버튼이나 문자 입력 박스처럼 윈도 응용 프로그램에서 자주 볼 수 있는 화면 구성 요소이다. 쉽게 가져다 사용할 수 있도록 일정한 모양과 기능이 정해진 상태로 제공되며, 이러한 컨트롤을 배치하여 화면(폼)을 디자인할 수 있다. 종이를 자를 때 가위를 사용하고 못을 박을 때 망치를 사용하는 것처럼, 용도에 맞는 컨트롤을 선택해서 사용하면 된다.

1 윈도 폼 컨트롤의 종류

윈도 폼 컨트롤은 통합 개발 환경의 도구 상자에서 찾을 수 있다. 컨트롤은 기능에 따라 공용 컨트롤, 컨테이너, 메뉴 및 도구 모음, 데이터, 구성 요소, 인쇄, 대화 상자 등으로 분류되어 있다. 각각의 컨트롤은 속성, 메서드, 이벤트를 가진다.

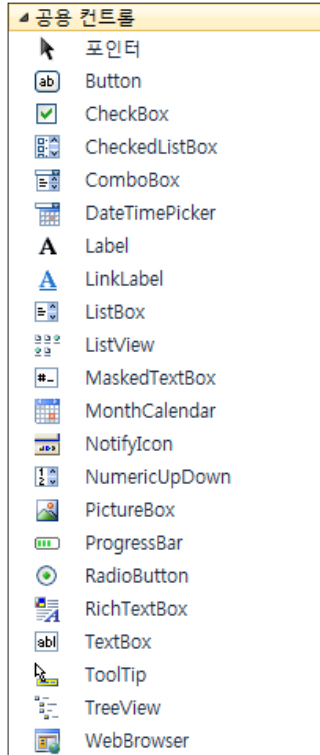
그림 3-1 윈도 폼 컨트롤의 종류



공용 컨트롤

공용 컨트롤은 비주얼 베이직(.Net)에 내장된 컨트롤로 총 21개가 제공된다. 이 책에서는 주요 공용 컨트롤을 중심으로 사용법을 설명하고, 다른 그룹에 포함된 컨트롤은 필요한 부분만 발췌하여 소개했다. 각 컨트롤의 기능과 사용법은 뒤에서 자세하게 살펴볼 것이다.

그림 3-2 공용 컨트롤의 종류



TIP 폼에 컨트롤을 배치하는 방법

- 도구 상자의 컨트롤을 더블클릭한다. 폼에 컨트롤이 추가되면 원하는 위치로 옮기고 크기를 조정한다.
- 도구 상자에서 컨트롤을 선택한 후 폼 위에서 마우스를 드래그한다.

2 컨트롤의 명명 규칙

비주얼 베이직 2010 한글 버전에서는 컨트롤의 이름이나 변수명에도 한글을 사용할 수 있다. 간단한 프로그램이라면 비주얼 베이직에서 정해주는 대로 TextBox1, Label1 등을 사용할 수도 있다. 하지만 사용하는 컨트롤이 많아지면 기억하기도 어렵고, 문제가 발생했을 때 해결하기도 힘들다. 더욱이 나중에 누구라도 기능 수정 및 향상이 가능해야 한다.

따라서 처음 선언할 때부터 올바른 이름을 붙이는 것이 좋은데 일반적으로 헝가리안 표기법으로 이름을 설정한다. 강제성은 없지만 많은 개발자들이 지키고 있는 암묵적인 약속이니 알고 있는 것이 좋다. [표 3-1]은 헝가리안 표기법으로 표현한 컨트롤 접두사의 예이다.

TIP 이 책에서는 좀 더 가볍게 실습할 수 있도록 간단한 예제에서는 기본 이름을 사용하고, 프로젝트와 같이 규모가 큰 경우에만 헝가리안 표기법으로 명명했다.

표 3-1 비주얼 베이직에서 권장하는 컨트롤의 명명 규칙

개체 형식	접두사	사용 예
폼	frm	frmMain
프레임	fra	fraLanguage
버튼 컨트롤	btn	btnExit
레이블	lbl	lblHelpMessage
텍스트박스	txt	txtLastName
체크박스	chk	chkReadOnly
라디오버튼	rb	rbColor
이미지	img	imgIcon
리스트박스	lst	lstPolicyCodes

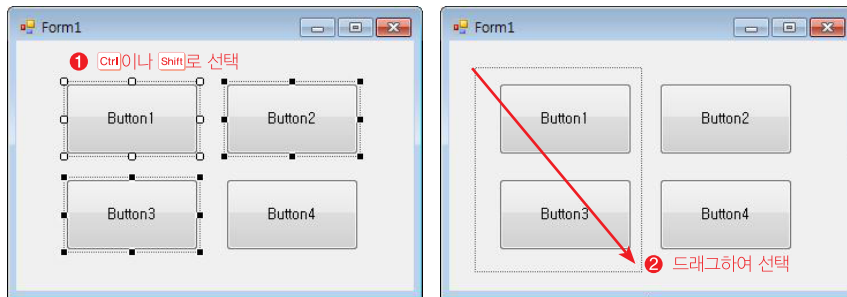
3 유용한 컨트롤 조작법

비주얼 베이직 프로그램을 작성하려면 도구 상자에 있는 개체(컨트롤)를 폼에 배치하는 것부터 시작해야 한다. 일반적으로 한 폼에 여러 개체가 놓인다. 구체적으로 각 컨트롤을 살펴보기에 앞서, 배치한 컨트롤을 효과적으로 다루는 방법을 알아보자.

여러 개체 선택하기

폼에 놓인 여러 개체를 선택하려면 **Ctrl**이나 **Shift**를 누른 상태에서 개체를 마우스로 클릭한다. 또는 여러 개체가 포함되도록 마우스로 드래그하여 영역을 지정한다.

그림 3-3 컨트롤 선택하기



개체 정렬하고 크기 조정하기

여러 개체를 선택한 상태에서 **[서식]** 메뉴를 이용하여 한꺼번에 정렬하고 크기를 조정할 수 있다. 가장 먼저 선택한 개체를 기준으로 추가 선택한 다른 개체의 크기나 위치가 변경된다.

그림 3-4 개체 정렬하기



- ① 맞춤 : 선택된 개체의 가로/세로 위치를 기준 개체의 왼쪽/가운데/오른쪽에 맞추거나 왼쪽 상단을 가장 가까운 모
눈에 맞춘다. 개체의 크기는 변하지 않는다.
- ② 같은 크기로 : 너비와 높이를 기준 개체와 같은 크기로 조정한다.
- ③ 가로 간격 조정/세로 간격 조정 : 개체 간의 상하 간격을 조정한다.
- ④ 폼의 가운데 맞춤 : 선택된 개체의 중심점을 폼의 중간에 있는 수평/수직 줄에 맞춘다.
- ⑤ 순서 : 여러 개체가 부분적으로 겹쳐 있을 때 선택된 개체의 순서를 변경한다.
- ⑥ 컨트롤 잠그기 : 폼에 배치된 컨트롤의 크기나 위치가 조정되지 않도록 잠근다.

키보드로 개체의 위치와 크기 조정하기

키보드를 이용하여 다음과 같이 개체의 위치와 크기를 조정할 수도 있다.

- 개체를 선택하고 **[Ctrl]**을 누른 상태에서 방향 키를 누르면 해당 개체가 원하는 방향으로 이동한다.
- 개체를 선택하고 **[Shift]**를 누른 상태에서 방향 키를 누르면 해당 개체의 크기를 조정할 수 있다.

4 컨트롤의 속성과 메서드 사용하기

컨트롤의 속성은 속성 창이나 코드 편집 창에서 설정할 수 있다. 속성 창을 이용하는 것은 디자인 모드 편집이라 하고, 프로그램 코드에서 속성을 설정하는 것은 실행 모드 편집이라 한다. 디자인 모드 편집에서는 속성 창에서 해당 속성의 값을 설정하면 되고, 실행 모드 편집에서는 다음과 같은 형식으로 값을 지정한다. 다음은 이름이 TEXT인 개체의 'Text' 속성 값을 'Text 속성'으로 설정하는 예이다.

형식 | 컨트롤명.속성=속성값

예 | Text.Text="Text 속성"

컨트롤의 속성 대부분을 위의 두 가지 방법으로 편집할 수 있지만 일부는 둘 중 한 모드에서만 편집이 가능하다. 실제 예는 각 컨트롤을 익히면서 알아보자.

메서드는 개체가 수행하는 동작이나 기능을 정의한 것으로 컨트롤에 속한 여러 함수를 의미한다. 속성처럼 컨트롤 이름에 마침표(.)를 붙이고 연결할 메서드를 기술하여 사용한다. 다음은

TEXT1 개체에 'SetFocus'라는 메서드를 실행하는 예이다.

형식 | 컨트롤명.메서드

예 | Text1.SetFocus 'Text1' 텍스트박스에 포커스를 둔다.

TIP / 비주얼 베이직에서 주석(보충 설명)은 작은따옴표(아포스트로피, ')로 표시한다. 자세한 내용은 2장 52쪽을 참조하기 바란다.

SECTION
02

버튼 컨트롤

- 마우스로 클릭하여 명령을 지시하거나 취소할 때 사용하는 버튼 컨트롤의 사용법을 익힌다.

버튼(Button) 컨트롤(**ab**)은 컴퓨터가 명령을 수행하도록 방어쇠 역할을 한다. 사용자가 버튼을 클릭하여 이벤트를 발생시키면 버튼에 연결된 명령을 처리하는 것이다. 버튼 컨트롤의 주요 속성, 메서드, 이벤트는 [표 3-2]와 같다.

표 3-2 버튼 컨트롤의 주요 속성, 메서드, 이벤트

파란색은 기본값

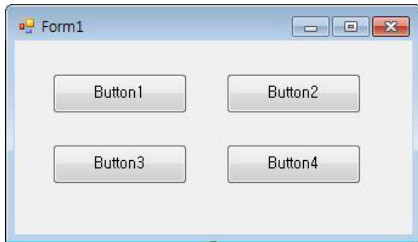
종류	이름	설명
속성	AutoSize	버튼의 크기를 텍스트 값에 따라 자동 조정할 수 있는지를 결정한다. <ul style="list-style-type: none"> • True : 버튼 크기 자동 조정 가능 • False : 버튼 크기 자동 조정 불가
	BackColor	버튼의 배경 색을 설정한다. <ul style="list-style-type: none"> • Control
	BackgroundImage	버튼에 배경 이미지를 넣는다. <ul style="list-style-type: none"> • 기본 값 없음
	Enabled	버튼의 활성화(폼에서 사용할 수 있도록 할 것인지) 여부를 설정한다. <ul style="list-style-type: none"> • True : 버튼 활성화 • False : 버튼 비활성화
	Font	버튼에 표시되는 문자열의 글꼴을 설정한다. <ul style="list-style-type: none"> • 굴림
	Image	버튼에 표시할 이미지를 설정한다. <ul style="list-style-type: none"> • 기본 값 없음
	Locked	버튼을 이동하거나 크기를 조정할 수 있는지를 설정한다. <ul style="list-style-type: none"> • True : 버튼 이동 및 크기 조정 가능 • False : 버튼 조정 불가
	Picture	그림 파일을 읽어오기 위해 설정한다. 단, Style 속성이 '1-그래픽'으로 설정되어 있어야 적용된다. <ul style="list-style-type: none"> • 기본 값 없음
	Text	버튼에 표시될 문자열을 설정한다. <ul style="list-style-type: none"> • Button1
	TextAlign	버튼에 표시되는 문자열의 위치를 설정한다. <ul style="list-style-type: none"> • TopLeft, TopCenter, TopRight • MiddleLeft, MiddleCenter, MiddleRight • BottomLeft, BottomCenter, BottomRight
	Visible	버튼 컨트롤이 보이게 할지 여부를 설정한다. <ul style="list-style-type: none"> • True : 컨트롤이 보이게 함 • False : 컨트롤이 보이지 않게 함

종류	이름	설명
메서드	SetFocus	버튼에 포커스를 둔다.
	Click	마우스 버튼을 눌렀을 때 발생한다.
이벤트	KeyPress	키보드의 키를 눌렀다 놓았을 때 발생한다.
	KeyDown	키보드의 키를 눌렀을 때 발생한다.
	KeyUp	키보드의 키를 놓았을 때 발생한다.

실습하기 3-1 버튼 속성 변경하기

버튼의 속성을 다양하게 설정한 후 결과를 직접 확인해보자.

1] 새 프로젝트를 열고 다음과 같이 폼을 디자인한다.



2] 컨트롤의 속성을 다음과 같이 설정한다.

컨트롤	속성	속성값
Button1	Text	기본 버튼
Button2	Text	배경색 변경버튼
	BackColor	Gold
Button3	Text	이미지 버튼
	Image	(임의의 이미지 파일 선택)
Button4	TextAlign	BottomCenter
	Text	배경이미지 버튼
	BackgroundImage	(임의의 이미지 파일 선택)

3] F5를 눌러 프로그램을 실행한다. 변경된 속성에 따른 결과를 확인할 수 있다.



SECTION
03

레이블 컨트롤

- 문자열을 출력하거나 다른 컨트롤의 용도를 표기하는 데 사용하는 레이블 컨트롤의 사용법을 익힌다.

레이블^{Label} 컨트롤(A)은 문자를 출력하는 컨트롤로, 출력되는 문자열을 사용자가 편집할 수 없기 때문에 다른 컨트롤의 제목이나 용도를 표시하는 데 주로 사용한다. 예를 들어 폼에 빈 텍스트박스만 있으면 무엇을 입력해야 할지 알 수 없다. 따라서 레이블 컨트롤을 사용하여 이름, 나이, 주소와 같이 텍스트박스에 입력할 정보를 알려준다. 레이블 컨트롤의 주요 속성은 [표 3-3]과 같다.

표 3-3 레이블 컨트롤의 주요 속성

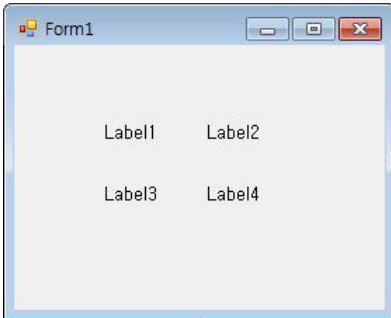
파란색은 기본값

속성	설명
AutoSize	레이블 너비의 자동 조정 여부를 설정한다. <ul style="list-style-type: none"> • True : 크기 임의 조정 불가, 텍스트 내용에 맞게 크기 자동 조정 • False : 크기 임의 조정 가능, 크기 초과 시 내용이 잘림
BorderStyle	테두리 모양을 설정한다. <ul style="list-style-type: none"> • None : 테두리 관련 요소 없음 • FixedSingle : 테두리를 평면 직선으로 표시 • Fixed3D : 테두리를 입체로 표시
Font	레이블의 글꼴을 설정한다.
ForeColor	레이블의 글자 색을 설정한다.
Text	레이블에 실제로 출력되는 내용을 표시한다.
TextAlign	레이블에 표시되는 텍스트의 위치를 설정한다. <ul style="list-style-type: none"> • TopLeft, TopCenter, TopRight • MiddleLeft, MiddleCenter, MiddleRight • BottomLeft, BottomCenter, BottomRight
Visible	레이블이 보이게 할지 여부를 설정한다. <ul style="list-style-type: none"> • True : 레이블이 보이게 함 • False : 레이블이 보이지 않게 함

실습하기 3-2 레이블 속성 변경하기

레이블의 속성을 다양하게 설정하고 결과를 직접 확인해보자. Text, TextAlign, AutoSize, Visible, BorderStyle 등 주요 속성을 사용하여 레이블의 내용을 변경할 수 있다.

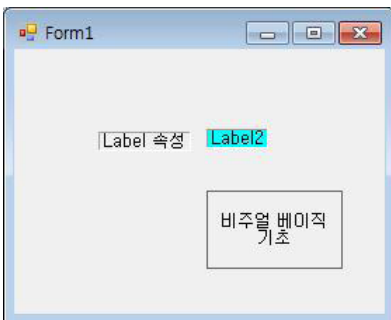
- 1] 새 프로젝트를 열고 다음과 같이 레이블 컨트롤을 4개 배치한다.



- 2] 컨트롤의 속성을 다음과 같이 설정한다.

컨트롤	속성	속성값
Label1	BorderStyle	Fixed3D
	Text	Label 속성
Label2	BorderStyle	Fixed3D
	BackColor	Cyan
Label3	Visible	False
	AutoSize	False
	BorderStyle	FixedSingle
Label4	Text	비주얼 베이직 기초
	TextAlign	MiddleCenter
	Size	120, 70

- 3] F5를 눌러 프로그램을 실행하면 변경된 속성에 따른 결과를 확인할 수 있다.



SECTION
04

텍스트박스 컨트롤

● 사용자로부터 값을 입력받거나 내용을 출력할 때 사용하는 텍스트박스 컨트롤의 사용법을 익힌다.

텍스트박스(TextBox) 컨트롤(☐)은 텍스트를 입력하고 편집할 때 사용한다. 기본적으로 한 줄 텍스트만 허용하지만 설정을 변경하면 여러 줄을 입력하고 편집할 수도 있다. 입력하는 문자열이 컨트롤의 범위를 넘어가면 스크롤바를 표시할지 여부도 설정할 수 있다. 텍스트박스 컨트롤의 주요 속성, 메서드, 이벤트는 [표 3-4]와 같다.

표 3-4 텍스트박스 컨트롤의 주요 속성, 메서드, 이벤트

파란색은 기본값

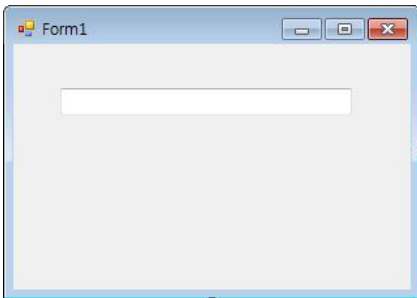
종류	이름	설명
속성	AcceptsReturn	☐ Enter 입력 시 기능을 설정한다. • True : MultiLine 속성이 True인 경우 여러 줄을 입력하기 위해 줄 바꿈 기능을 하도록 한다. • False : Form 속성의 AcceptButton과 연동할 수 있다. AcceptButton에 특정 버튼을 지정해 놓은 상태에서 ☐ Enter를 누르면 지정된 버튼을 클릭한 효과가 나타난다.
	BorderStyle	테두리 모양을 설정한다. • None-없음 : 테두리나 테두리에 관련된 요소가 없음 • FixedSingle-단일 고정 : 테두리가 있는 형태 • Fixed3D-3D 고정 : 테두리가 없는 3D 형태
	Font	입력 글자의 모양과 크기를 설정한다.
	Locked	텍스트박스에 있는 텍스트의 편집 가능 여부를 설정한다. • True : 편집 불가, 문자열 이동이나 반전 표시는 가능 • False : 편집 가능
	MaxLength	텍스트박스에 입력할 수 있는 최대 문자 수를 설정한다. • 32767 : 최대 문자 수
	MultiLine	텍스트박스에 텍스트를 여러 줄 입력할 수 있는지 여부를 설정한다. • True : 여러 줄 텍스트 허용 • False : 한 줄 텍스트만 허용
	PasswordChar	텍스트박스에 입력한 문자가 화면에 나타나지 않고 이 속성에 설정한 문자(예를 들면 *)가 대신 화면에 출력된다. 주로 암호를 입력하는 프로그램에서 사용한다. • 기본값 없음
	ScrollBars	텍스트박스에 수평/수직 스크롤바를 생성할지 여부를 설정한다. MultiLine 속성이 True로 설정되어 있어야 적용된다. • None-없음 : 스크롤바를 생성하지 않음 • Horizontal-수평 : 수평 스크롤바 생성 • Vertical-수직 : 수직 스크롤바 생성 • Both-양방향 : 수직, 수평 스크롤바 모두 생성

종류	이름	설명
속성	Text	텍스트박스에 입력된 텍스트를 나타낸다. • 기본값 없음
	TextAlign	텍스트박스에 입력된 텍스트의 위치를 설정한다. • Left – 왼쪽 맞춤 : 왼쪽 정렬 • Right – 오른쪽 맞춤 : 오른쪽 정렬 • Center – 가운데 맞춤 : 가운데 정렬
	Visible	실행 시 텍스트박스를 보이게 할지 여부를 설정한다. • True : 텍스트박스가 보이게 함 • False : 텍스트박스가 보이지 않게 함
	WordWrap	여러 줄을 입력할 때 자동 줄 바꿈 기능을 제공한다. • True : 자동 줄 바꿈 • False : 자동 줄 바꿈 하지 않음
메서드	Move	폼에서 해당 텍스트박스의 위치를 설정한다.
	SetFocus	해당 텍스트박스에 포커스를 둔다.
이벤트	Click	텍스트박스를 마우스로 클릭했을 때 발생한다.
	TextChanged	Text 속성의 값이 변하면 발생한다.
	KeyPress	텍스트박스에 포커스를 둔 상태에서 특수 키([Alt], [Ctrl], [Shift]) 외의 키를 눌렀을 때 발생한다.

실습하기 3-3 텍스트박스에 텍스트 여러 줄 입력하기

텍스트박스 속성을 이용하여 많은 양의 텍스트가 잘리지 않고 여러 줄로 표시되도록 설정해보자.

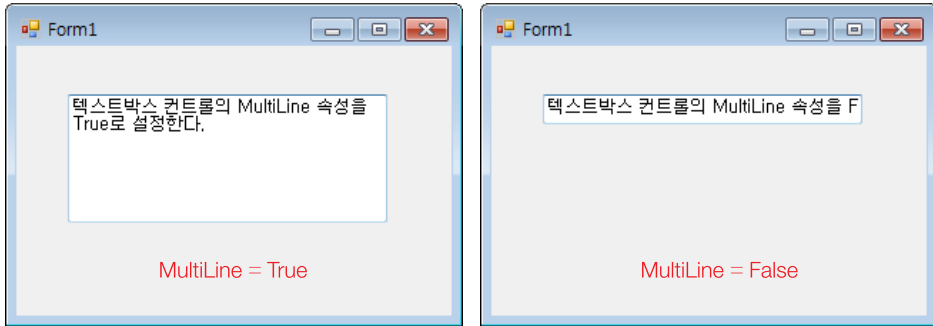
- 1] 새 프로젝트를 열고 텍스트박스 컨트롤 하나를 배치한다.



- 2] 컨트롤의 속성을 다음과 같이 설정한다.

컨트롤	속성	속성값	기능
TextBox1	Text	(빈칸)	
	AcceptsReturn	True	[Enter]를 누르면 줄 바꿈이 되도록 한다.
	MultiLine	True	여러 줄 입력 기능을 활성화한다.
	WordWrap	True	자동 줄 바꿈 기능을 활성화한다.

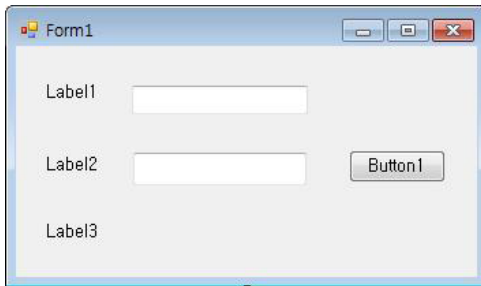
- 3] **F5**를 눌러 프로그램을 실행하고 텍스트를 입력해보자. 텍스트가 오른쪽 끝에 닿으면 자동으로 줄이 바뀐다. 또는 **Enter**를 눌러 강제로 줄을 바꾸고 입력할 수도 있다.



실습하기 3-4 아이디와 비밀번호를 입력하면 환영 메시지 출력하기

사용자가 아이디와 비밀번호를 입력한 후 <로그인>을 클릭하면 환영 메시지를 새 창에 출력하는 프로그램을 작성해보자. 텍스트박스, 레이블, 버튼을 사용하고 비밀번호에 입력한 글자는 보이지 않도록 처리한다.

- 1] 새 프로젝트를 열고 다음과 같이 컨트롤을 배치한다.



- 2] 컨트롤의 속성을 다음과 같이 설정한다.

컨트롤	속성	속성값
Label1	Text	아이디
Label2	Text	비밀번호
Label3	Text	(빈칸)
TextBox1	Text	(빈칸)
TextBox2	Text	(빈칸)
	PasswordChar	*
Button1	Text	로그인


3] <로그인>을 클릭하면 환영 메시지를 출력하도록, <로그인>을 더블클릭하여 다음과 같이 코드를 작성한다.

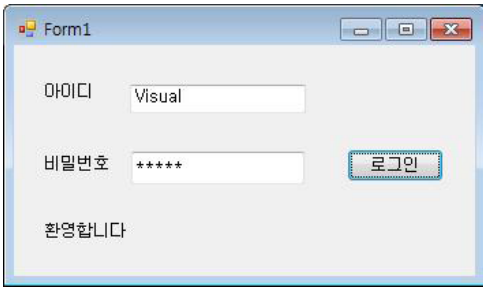
```

01 Private Sub Button1_Click(sender As System.Object, e As System.EventArgs)
    Handles Button1.Click
02     Label3.text = "환영합니다"
03 End Sub

```

02행 : Lable3에 '환영합니다'를 출력한다.

4] 를 눌러 프로그램을 실행한다. 아이디와 비밀번호를 입력한 후 <로그인>을 클릭하면 메시지가 나타난다. 값을 입력하지 않고 <로그인>만 클릭해도 메시지가 출력된다. 반드시 값을 입력하게끔 처리하는 방법은 나중에 제어문을 배운 다음 적용해보자.



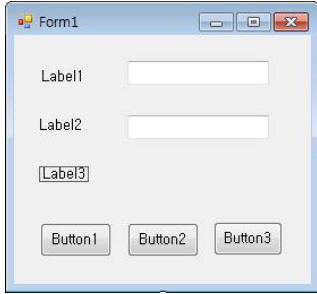
실습하기 3-5 아이디, 비밀번호 입출력 화면에 버튼 추가하기

아이디와 비밀번호를 입력하고 출력하는 [실습하기 3-4]의 프로그램에 버튼을 추가하여 확장해보자. <확인>을 클릭하면 입력한 아이디와 비밀번호를 출력하고, <취소>를 클릭하면 다시 입력할 수 있도록 입력 텍스트 상자의 내용을 지우며, <종료>를 클릭하면 프로그램을 종료하도록 프로그램을 작성한다.

▼ 실행 결과



- 1] 새 프로젝트를 열고 다음과 같이 폼을 디자인한다.



- 2] 컨트롤의 속성을 다음과 같이 설정한다.

컨트롤	속성	속성값
Label1	Text	아이디
Label2	Text	비밀번호
Label3	BorderStyle	FixedSingle
	Text	(빈칸)
TextBox1	Text	(빈칸)
TextBox2	Text	(빈칸)
	PasswordChar	*
Button1	Text	확인
Button2	Text	취소
Button3	Text	종료

- 3] 다음과 같이 코드를 작성한다.

```

01 Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.
    EventArgs) Handles Button1.Click
02     Label3.Text = TextBox1.Text + " 님 환영합니다"
03 End sub

04 Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.
    EventArgs) Handles Button2.Click
05     Label3.Text = "다시 입력해주세요"
06     TextBox1.Text = ""
07     TextBox2.Text = ""
08     TextBox1.Focus()
09 End Sub

10 Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.
    EventArgs) Handles Button3.Click
11     End
12 End Sub

```

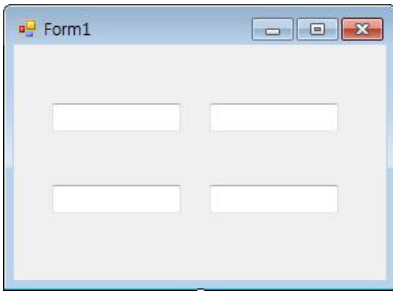
- 01~03행 : <확인>을 클릭하면 환영 메시지를 레이블에 출력한다.
- 04행 : <취소>를 클릭하면
- 05행 : 레이블에 '다시 입력해주세요'라는 메시지를 출력한다.
- 06~07행 : 텍스트박스의 값을 지운다.
- 08행 : TextBox1에 커서를 놓는다.
- 10~12행 : <종료>를 클릭하면 End 명령을 수행하여 프로그램을 종료한다.

4] F5를 눌러 프로그램을 실행한다. 각 버튼을 클릭하여 기능이 실행되는지 확인해보자.

실습하기 3-6 디자인 모드 편집과 실행 모드 편집에서 컨트롤 속성 설정하기

앞서 살펴본 예제처럼 속성 창에서 속성값을 설정하기도 하지만, 속성을 설정하는 코드를 삽입하여 프로그램상에서 변경할 수도 있다. 이를 어떻게 처리하는지 알아보자.

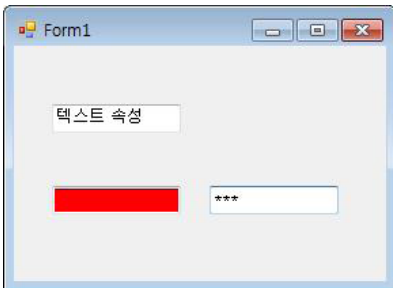
1] 새 프로젝트를 열고 텍스트박스 컨트롤을 4개 배치한다.



2] 디자인 모드 편집 : 속성 창에서 다음과 같이 속성을 설정한다.

컨트롤	속성	속성값	기능
TextBox1	Text	텍스트 속성	
TextBox2	Text	보여요	
	Visible	False	보이지 않게 한다.
TextBox3	BackColor	RED	배경 색을 붉게 칠한다.
TextBox4	PasswordChar	*	입력한 글자를 *로 표시한다.


3] F5를 눌러 프로그램을 실행하고 각 텍스트박스 컨트롤이 속성값에 따라 어떻게 변했는지 확인해본다. TextBox4에 글자를 입력하면 *로 표시된다.

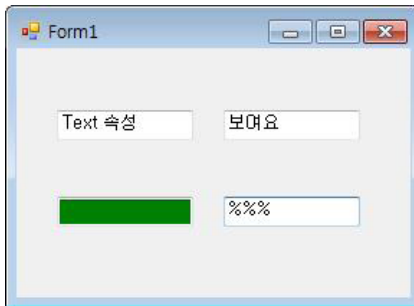


- 4] 실행 모드 편집 : 폼을 더블클릭하여 코드 편집 창을 열고 다음과 같이 코드를 작성한다.

```
01 Private Sub Form1_Load(sender As System.Object, e As System.EventArgs)
    Handles MyBase.Load
02     TextBox1.Text = "Text 속성"
03     TextBox2.Visible = True
04     TextBox3.BackColor = Color.Green
05     TextBox4.PasswordChar = "x"
06 End Sub
```

01행 : Form_Load는 폼이 실행될 때 수행되는 이벤트 프로시저로 보통 폼을 초기화하는 데 사용한다.
03-06행 : 각각의 텍스트박스에 해당하는 속성값을 할당한다.

- 5] 를 눌러 프로그램을 다시 실행한다. 속성 창에서 설정한 속성이 사라지고 코드로 설정한 속성이 적용된다. 이는 어떤 속성을 속성 창과 코드 편집 창에서 서로 다르게 설정하면 코드로 설정한 속성이 우선 적용됨을 보여준다.



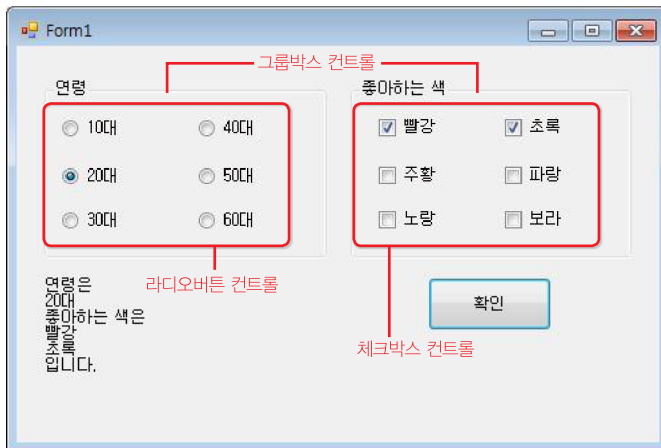
SECTION
05

체크박스, 라디오버튼, 그룹박스 컨트롤

- 마우스로 원하는 항목을 선택하는 체크박스과 라디오버튼, 그리고 컨트롤을 그룹으로 묶는 그룹박스의 사용법을 익힌다.

체크박스(CheckBox)와 라디오버튼(RadioButton) 컨트롤은 데이터 입력 시 문자를 직접 입력하지 않고 사용자가 선택할 수 있는 예시를 보여주어 마우스 클릭만으로 필요한 데이터를 입력할 수 있도록 하는 컨트롤이다. 그룹박스(GroupBox) 컨트롤은 여러 개의 컨트롤을 그룹으로 묶어 서로 연관된 항목임을 보여줄 때 사용한다.

그림 3-5 체크박스, 라디오버튼, 그룹박스 컨트롤의 사용 예



1 체크박스과 라디오버튼 컨트롤

체크박스 컨트롤()은 그룹으로 묶인 여러 항목 중 하나 이상을 선택할 때 사용한다. True(참)와 False(거짓) 중 한 가지 상태만 가질 수 있으며, 체크박스를 마우스로 클릭하면 V가 표시되고 다시 한 번 클릭하면 V가 사라진다.

라디오버튼 컨트롤()은 여러 항목 중 하나만 선택할 때 사용한다. 동시에 여러 항목을 선택할 수 있는 체크박스 컨트롤과 달리 한 항목만 선택할 수 있다. 체크박스과 라디오버튼 컨트롤을 비교하면 [표 3-5]와 같다. 속성도 [표 3-6]과 같이 매우 유사하다.

표 3-5 체크박스과 라디오버튼 컨트롤의 비교

체크박스 컨트롤	라디오버튼 컨트롤
여러 항목 선택 가능	한 항목만 선택 가능
세 가지 상태 사용	두 가지 상태 사용
선택의 의미는 True	선택의 의미는 True

표 3-6 체크박스과 라디오버튼 컨트롤의 주요 속성

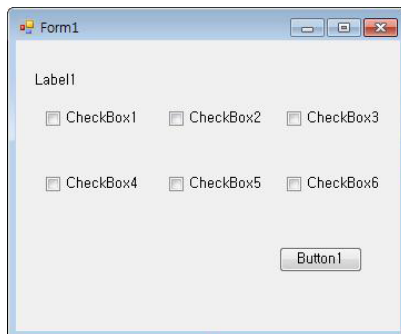
파란색은 기본값

속성	설명
AutoSize	체크박스과 라디오버튼의 크기 자동 조정 여부를 설정한다. <ul style="list-style-type: none"> • True : 자동 조정 가능 • False : 임의 조정 가능
BackColor	체크박스과 라디오버튼의 배경 색을 설정한다.
BackgroundImage	체크박스과 라디오버튼의 배경 이미지를 설정한다.
Enabled	체크박스과 라디오버튼의 활성화 여부를 설정한다. <ul style="list-style-type: none"> • True : 체크박스과 라디오버튼 활성화 • False : 체크박스과 라디오버튼 비활성화
Text	체크박스과 라디오버튼 옆에 나오는 문자열을 설정한다.
Checked	체크박스과 라디오버튼의 선택 여부를 나타낸다. <ul style="list-style-type: none"> • True : 선택된 경우 • False : 선택되지 않은 경우
체크박스의 경우	CheckState 체크박스의 초기 선택 상태를 설정한다. <ul style="list-style-type: none"> • Unchecked : 체크되지 않은 상태 • Checked : 체크된 상태 • Indeterminate : 체크되고 회색인 상태(비활성화 상태)
	ThreeState 체크박스에 Indeterminate를 사용 가능하게 하는지 여부를 설정한다. <ul style="list-style-type: none"> • True : 세 가지 선택 상태 • False : 두 가지 선택 상태

실습하기 3-7 체크박스 컨트롤을 원하는 색 출력하기

좋아하는 색을 선택하고 <확인>을 클릭하면 레이블 컨트롤에 선택한 색을 출력하는 프로그램을 만들어보자.

1] 새 프로젝트를 열고 다음과 같이 폼을 디자인한다.



2] 컨트롤의 속성을 다음과 같이 설정한다.

컨트롤	속성	속성값
Label1	Text	좋아하는 색을 선택하세요.
CheckBox1	Text	빨강
CheckBox2	Text	주황
CheckBox3	Text	노랑
CheckBox4	Text	초록
CheckBox5	Text	파랑
CheckBox6	Text	보라
Label2	Text	(빈칸)
Button1	Text	확인(&O)

3] 원하는 색의 체크박스를 선택하고 <확인>을 클릭하면 선택한 색이 출력되도록, <확인>을 더블클릭하여 다음과 같이 코드를 작성한다.

```

01 Private Sub Button1_Click_1(ByVal sender As System.Object, ByVal e As System.
    EventArgs) Handles Button1.Click
02     Dim str As String

03     If CheckBox1.Checked = True Then str = CheckBox1.Text & Chr(13)
04     If CheckBox2.Checked = True Then str = str & CheckBox2.Text & Chr(13)
05     If CheckBox3.Checked = True Then str = str & CheckBox3.Text & Chr(13)
06     If CheckBox4.Checked = True Then str = str & CheckBox4.Text & Chr(13)
07     If CheckBox5.Checked = True Then str = str & CheckBox5.Text & Chr(13)
08     If CheckBox6.Checked = True Then str = str & CheckBox6.Text & Chr(13)

09     Label2.Text = str
10 End Sub
    
```

02행 : 출력할 문자열을 저장할 변수 str을 선언한다.

03-08행 : <확인>을 클릭했을 때 체크박스의 문자열을 레이블(Label2)에 보여주도록 각 체크박스의 선택 여부를 확인한 후, 변수 str에 선택한 색상명을 추가한다.

09행 : 선택한 색상명(체크박스의 텍스트)을 레이블(Label2)에 출력한다.

TIP / &는 문자열을 연결하는 연산자, Chr(13)은 줄 바꿈을 의미한다. 변수에 관해서는 4장에서 자세하게 살펴본다.