

대화 상자

HyoJoon Han
동국대학교
han6343@dongguk.edu

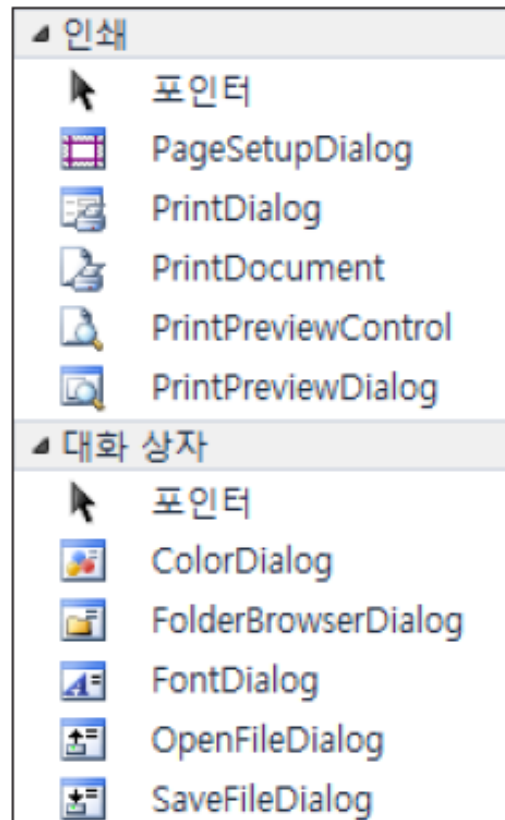


대화 상자 컨트롤

MessageBox 함수

- 파일 열기, 파일 저장, 글꼴 선택 등 윈도우 응용 프로그램에서 자주 사용하는 대화 상자를 윈도우 표준 대화 상자로 나타내는 데 사용

그림 10-1 비주얼 베이직에서 제공하는 대화 상자 컨트롤의 예



- [열기] 대화 상자 컨트롤
 - 사용자가 드라이브, 디렉터리, 파일의 이름과 확장자명 등을 선택할 수 있게 해준다.
 - ShowDialog 메서드를 이용하여 화면에 나타낸다.

표 10-1 [열기] 대화 상자 컨트롤의 주요 속성

속성	설명
CancelError	〈취소〉 버튼을 클릭했을 때 오류 발생 여부를 설정한다. 기본값은 False이며 오류가 발생하지 않는다.
DefaultExt	기본 확장자를 설정하거나 반환한다. 이 속성을 사용하여 .txt나 .hwp와 같은 파일 확장자를 지정한다.
FileName	'파일 이름' 입력란에 표시될 파일 이름을 설정한다.
Filter	대화 상자의 파일 형식 목록에 표시되는 파일 형식의 필터를 설정하거나 반환한다.
FilterIndex	필터의 인덱스를 지정하며, 파일 형식 목록에서 한 항목을 기본값으로 지정할 때 사용한다. 첫 번째 항목의 속성값을 1로 설정하면 목록에 표시된다.
InitialDirectory	기본 디렉터리를 설정하거나 반환한다.
Multiselect	파일을 다중으로 선택할 수 있게 한다.
Title	대화 상자의 제목을 설정한다.

표 10-2 [열기] 대화 상자 컨트롤의 주요 메서드

메서드	설명
Dispose	Open 대화 상자의 목록을 연결한다. 모든 리소스 해제
OpenFile	read-only 속성 파일로 연결된다.
Reset	모든 속성값이 기본값으로 지정된다.
ShowDialog	대화 상자를 나타낸다.

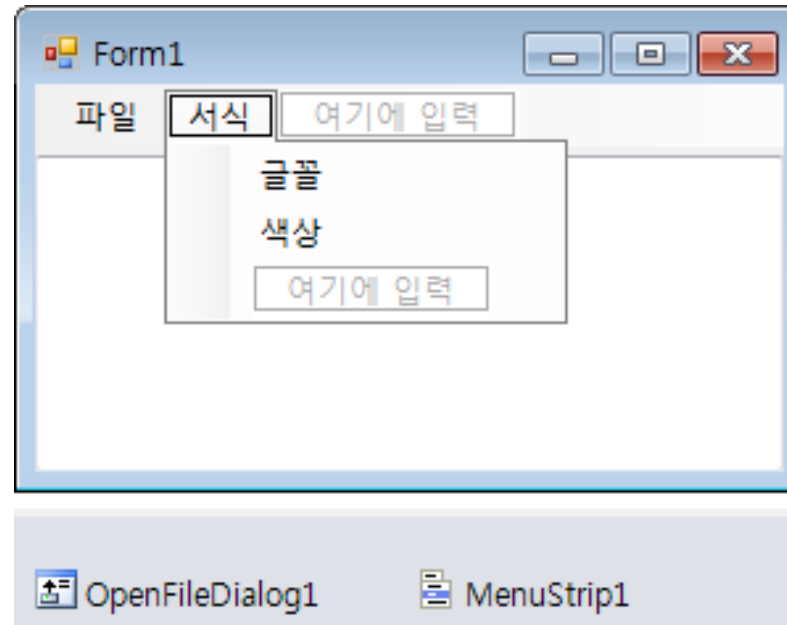
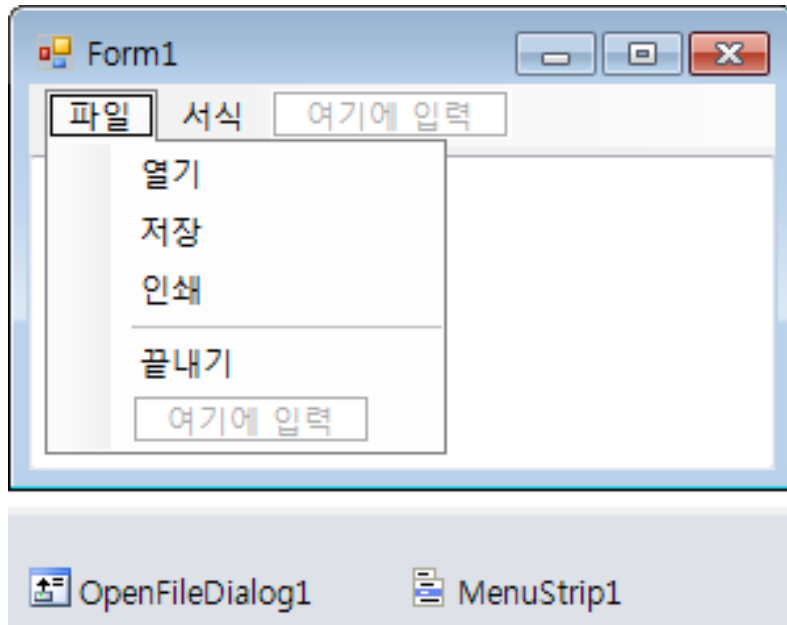
형식 | 개체명.Filter = "설명_1 | 필터_1 | 설명_2 | 필터_2 | ..."

예 | CommonDialog1.Filter = "비트맵 파일(*.bmp) | *.bmp"

CommonDialog2.Filter = "그림 파일(bmp, gif, jpg) | *.bmp; *.gif; *.jpg | 비트
맵 파일(*.bmp) | *.bmp"

- 실습 – [열기] 대화 상자 컨트롤을 이용하여 파일 열기

1) 윈도우 폼 디자인



2) 속성 설정

컨트롤	메뉴	하위 메뉴	속성	속성값
MenuStrip1	파일	파일	Text	파일
		열기	Text	열기
		저장	Text	저장
		인쇄	Text	인쇄
		–	Text	–
		끝내기	Text	끝내기
	서식	서식	Text	서식
		글꼴	Text	글꼴
		색상	Text	색상

컨트롤	속성	속성값
OpenFileDialog1	Filter	텍스트파일(*.txt) *.txt All Files(*.*)**
	FilterIndex	1
	Title	열기
	InitialDirectory	C:\
	FileName	(빈칸)
	Text	(빈칸)
TextBox1	Multiline	True 설정 후 Form 크기만큼 영역을 확장한다.

3) 코드 작성

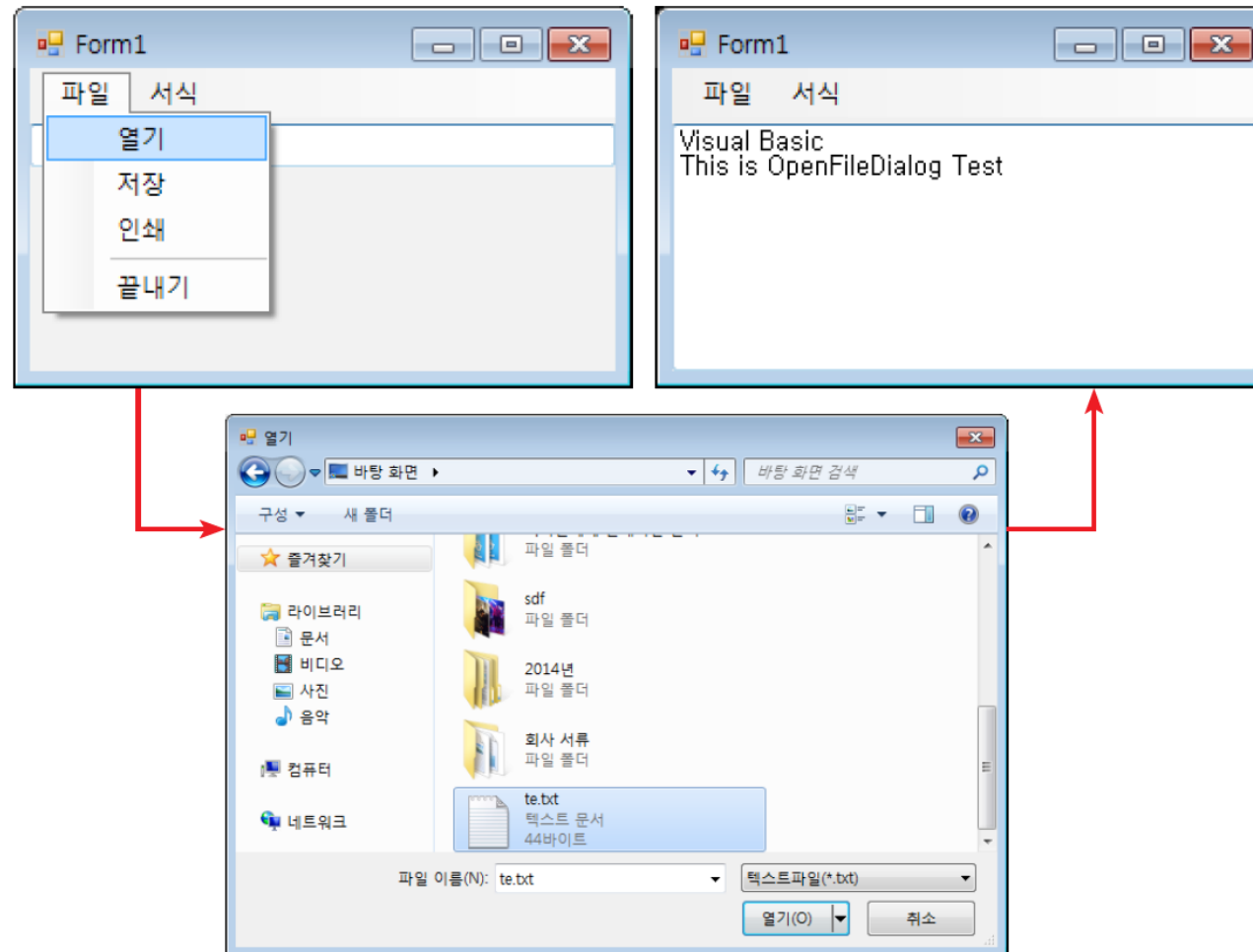
```
using System.Windows.Forms;
using System.IO;

namespace Chapter10
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void 열기ToolStripMenuItem_Click(object sender, EventArgs e)
        {
            var dialog = openFileDialog1.ShowDialog();
            if(dialog == DialogResult.OK)
            {
                string strfname = openFileDialog1.FileName;
                string strfcon = File.ReadAllText(strfname);
                textBox1.Text = strfcon;
            }
        }
    }
}
```

ReadAllText 메소드는 파일의 경로와 파일명을 인자로 받아 해당 파일내의 모든 텍스트를 읽어온다.

4) 실행 결과 확인



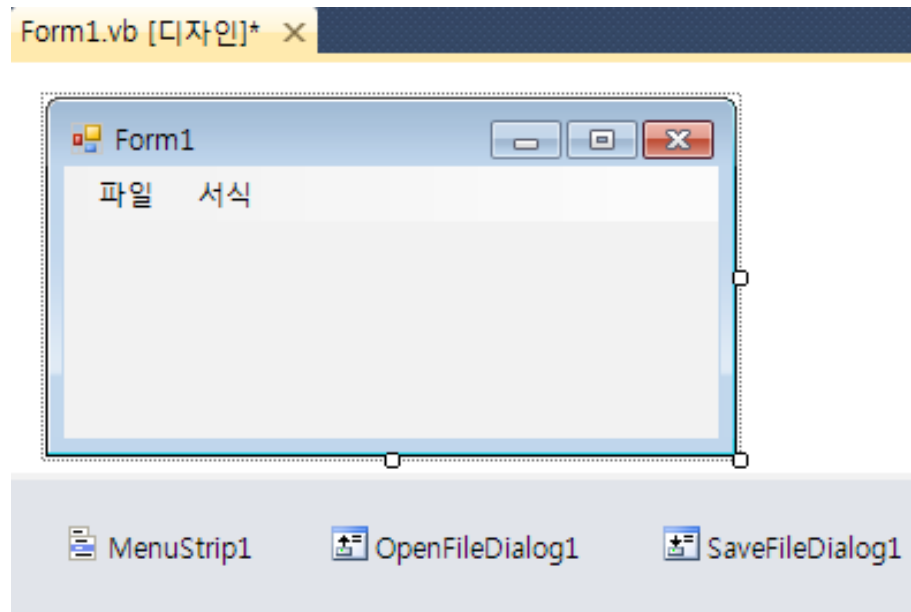
- [저장] 대화 상자 컨트롤
 - 기존 파일을 열어 덮어쓰거나 새 파일로 저장할 수 있게 해준다.
 - 사용하는 방법과 속성은 [열기] 대화 상자 컨트롤과 유사

표 10-3 [저장] 대화 상자 컨트롤의 주요 속성

속성	설명
CreatePrompt	존재하지 않는 파일을 지정할 때 파일을 새로 만들 것인지 사용자에게 묻는 메시지가 대화 상자에 표시되는지 여부를 나타내는 값을 가져오거나 설정한다.
OverwritePrompt	이미 존재하는 파일 이름을 지정할 때 Save As 대화 상자에 경고가 표시되는지 여부를 나타내는 값을 가져오거나 설정한다.

- 실습 - [저장] 대화 상자 컨트롤을 이용하여 파일 저장하기

1) 위에서 만든 프로그램에 다음과 같이 [저장] 대화 상자 컨트롤을 추가하고 아래 표를 참고하여 속성을 설정한다.

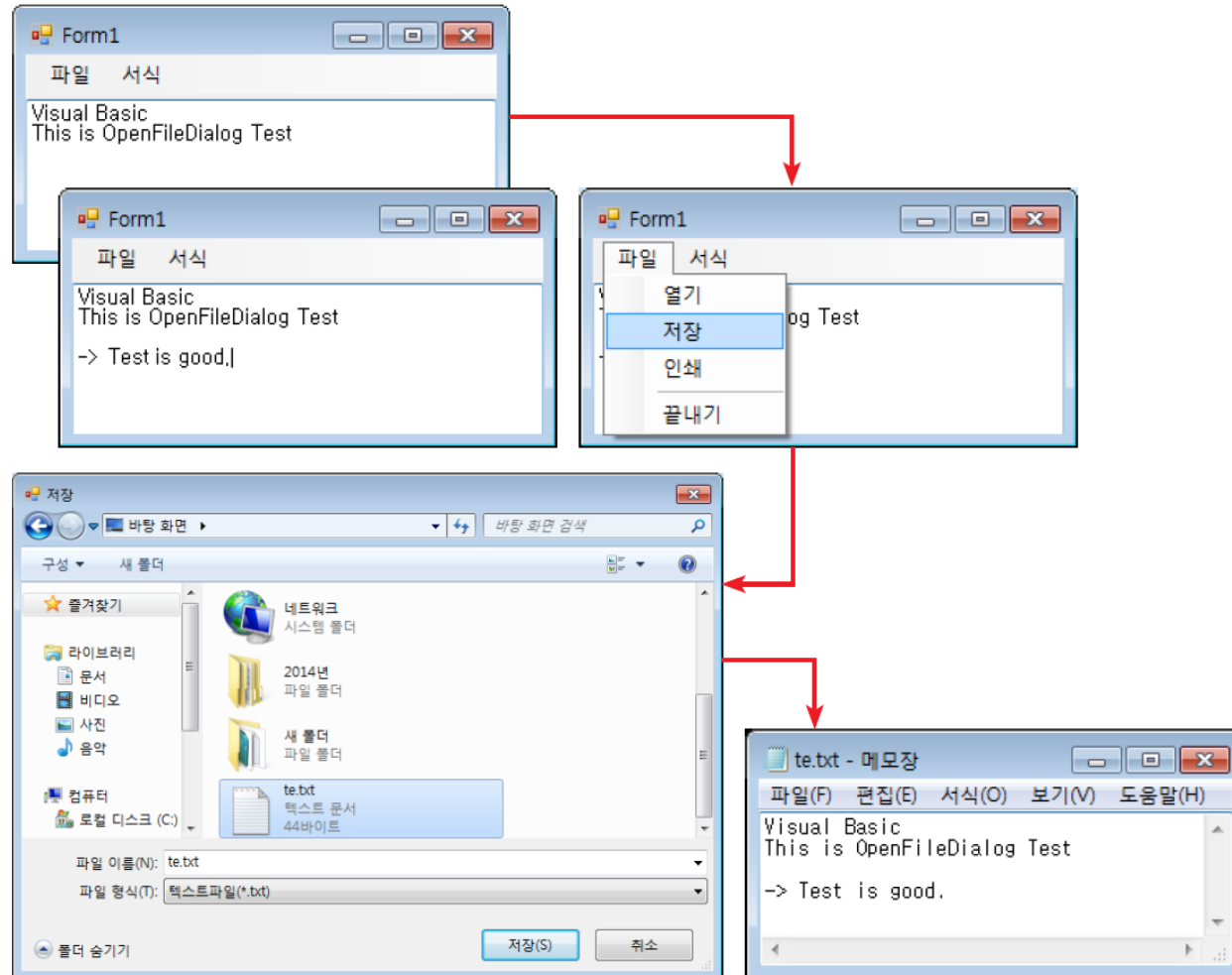


컨트롤	속성	속성값
SaveFileDialog1	Filter	텍스트 파일(*.txt) *.txt All Files(*.*) *.*
	FilterIndex	1
	Title	저장
	InitialDirectory	C:\
	FileName	(빈칸)

2) 코드 작성

```
private void 저장ToolStripMenuItem_Click(object sender, EventArgs e)
{
    var dialog = saveFileDialog1.ShowDialog();
    if(dialog == DialogResult.OK)
    {
        string strfname = saveFileDialog1.FileName;
        File.WriteAllText(strfname, textBox1.Text, System.Text.Encoding.UTF8);
    }
}
```

3) 실행 결과 확인



- [글꼴] 대화 상자 컨트롤
 - 글꼴의 종류, 크기, 색상, 스타일 등을 고를 수 있게 해준다.
 - ShowDialog 메서드를 이용한다.
 - 주요 속성과 메서드는 [열기] 대화 상자 컨트롤과 유사

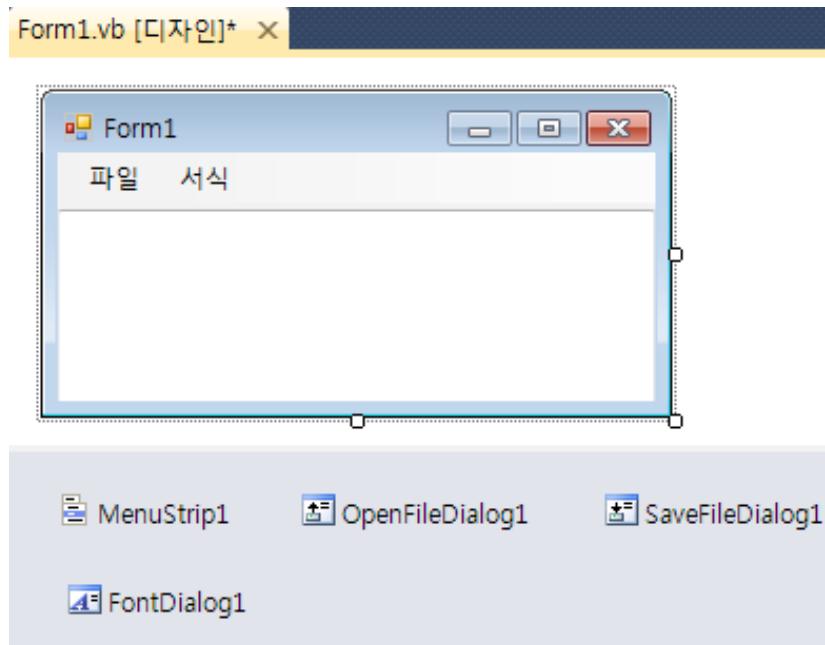
표 10-4 [글꼴] 대화 상자 컨트롤의 주요 속성

파란색은 기본값

속성	설명
Color	현재 선택된 색을 표시한다. • 기본값: 검은색
Font	현재 선택된 글꼴을 표시한다.
ShowColor	글꼴 창에 색상 선택 조건 표시 여부를 나타낸다. • 기본값: False
ShowEffects	글꼴 창에 밑줄, 취소선 조건 표시 여부를 나타낸다. • 기본값: True

- 실습 - [글꼴] 대화 상자 컨트롤을 이용하여 글꼴 변경하기

1) 위에서 만든 프로그램에 다음과 같이 [글꼴] 대화 상자 컨트롤을 추가하고 아래 표를 참고하여 속성을 설정한다.

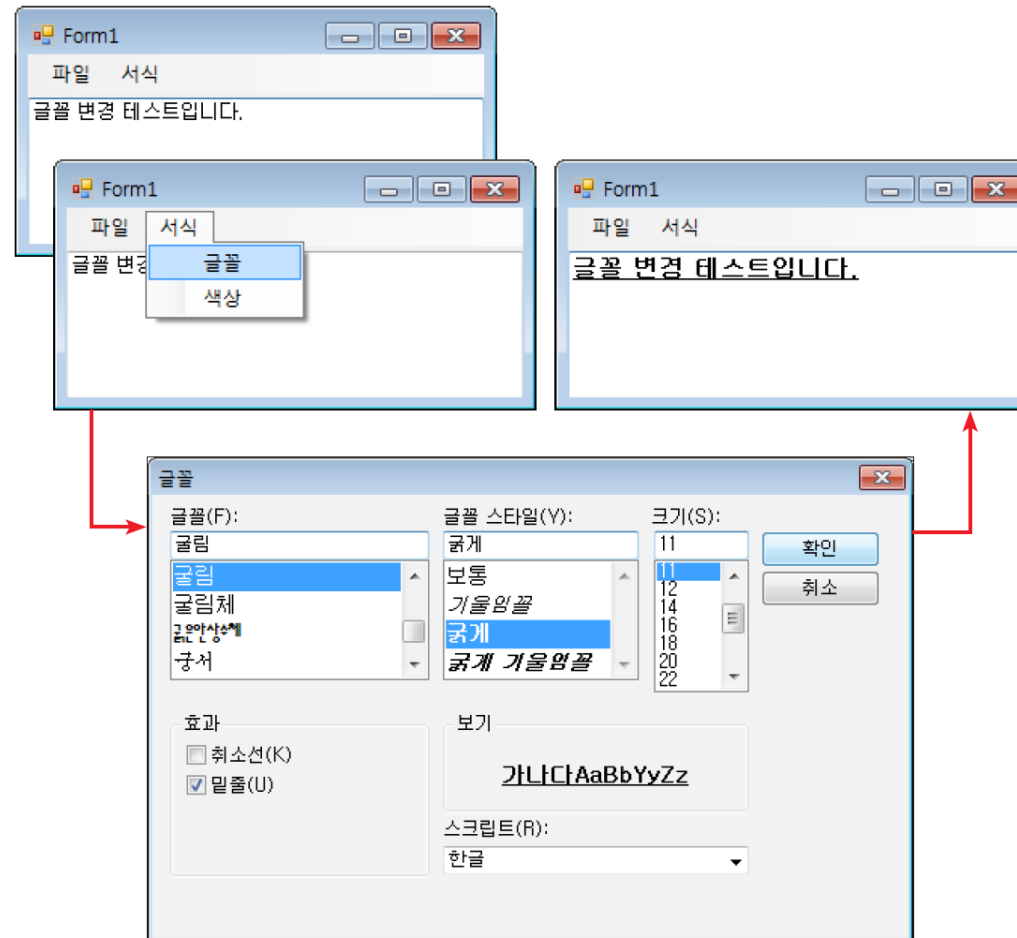


컨트롤	속성	속성값
FontDialog1	ShowColor	True

2) 코드 작성

```
private void 글꼴ToolStripMenuItem_Click(object sender, EventArgs e)
{
    var dialog = fontDialog1.ShowDialog();
    if(dialog == DialogResult.OK)
    {
        textBox1.Font = fontDialog1.Font;
        textBox1.ForeColor = fontDialog1.Color;
    }
}
```


3) 실행 결과 확인



- [색상] 대화 상자 컨트롤
 - 새로운 색상을 선택할 수 있게 해준다.
 - ShowDialog 메서드를 이용한다.
 - 주요 속성과 메서드는 [열기] 대화 상자 컨트롤과 유사

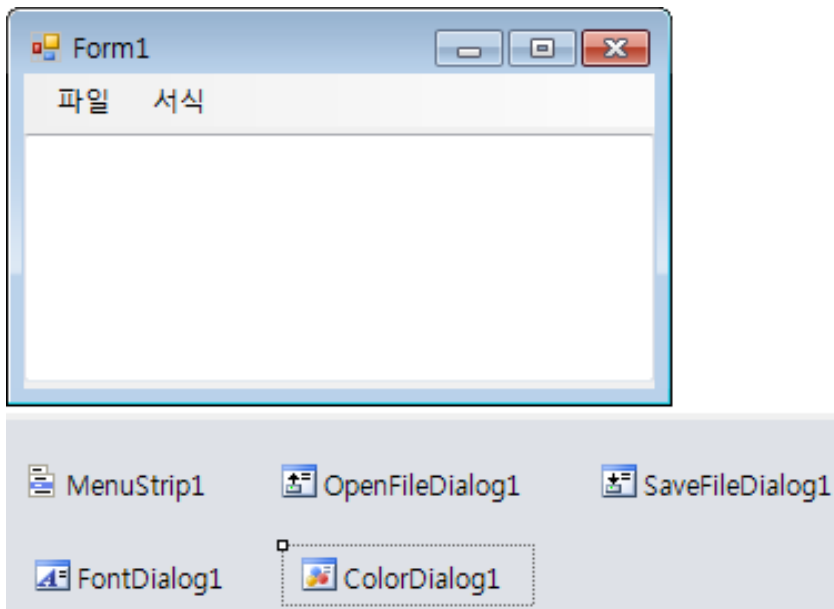
표 10-5 [색상] 대화 상자 컨트롤의 주요 속성

파란색은 기본값

속성	설명
AllowFullOpen	현재 정의된 모든 색상을 사용하도록 나타낸다. • 기본값: True
Color	선택한 색상을 반환하거나 설정한다.

- 실습 - [색상] 대화 상자 컨트롤을 이용하여 텍스트박스의 글자 색 변경하기

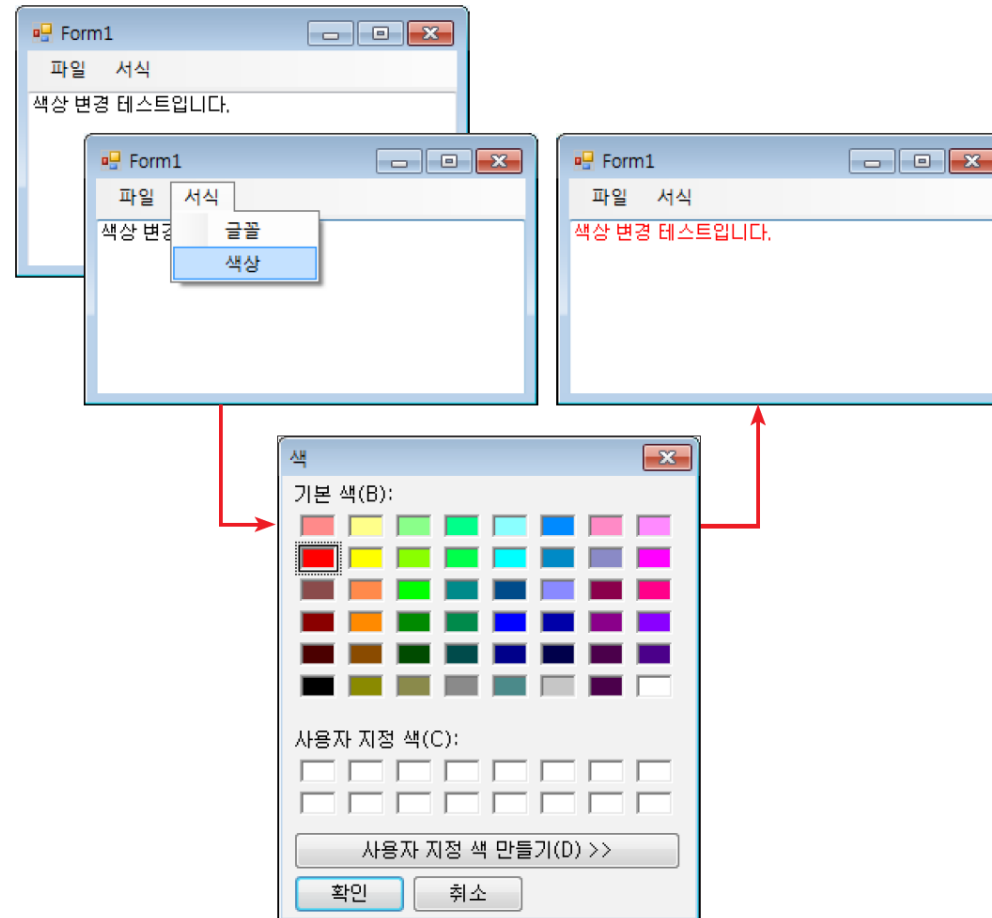
1) 위에서 만든 프로그램에 다음과 같이 [색상] 대화 상자 컨트롤을 추가한다. ColorDialog1 컨트롤의 속성은 기본값을 그대로 사용한다.



2) 코드 작성

```
private void 색상ToolStripMenuItem_Click(object sender, EventArgs e)
{
    var dialog = colorDialog1.ShowDialog();
    if(dialog == DialogResult.OK)
    {
        textBox1.ForeColor = colorDialog1.Color;
    }
}
```

3) 실행 결과 확인



- [인쇄] 대화 상자 컨트롤
 - 인쇄 방법, 인쇄 범위, 인쇄 품질, 인쇄 매수 등을 지정할 수 있게 해준다.
 - ShowDialog 메서드를 이용한다.
 - 주요 속성과 메서드는 [열기] 대화 상자 컨트롤과 유사

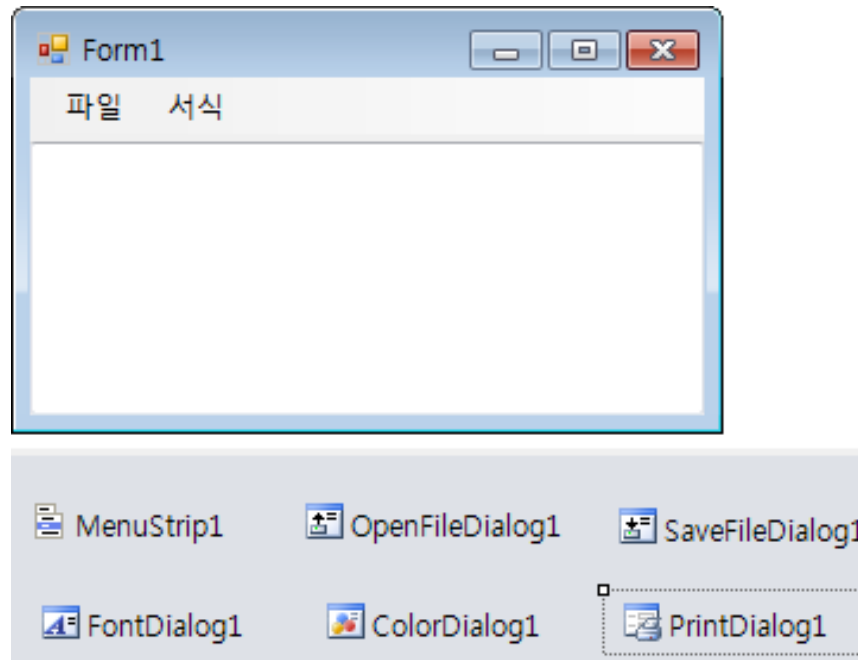
표 10-6 [인쇄] 대화 상자 컨트롤의 주요 속성

파란색은 기본값

속성	설명
AllowCurrentPage	현재 페이지만 출력되게 한다. • 기본값 : False
AllowPrintToFile	출력을 파일로 저장한다. • 기본값 : True

- 실습 - [인쇄] 대화 상자 컨트롤을 이용하여 인쇄 대화 상자 표시하기

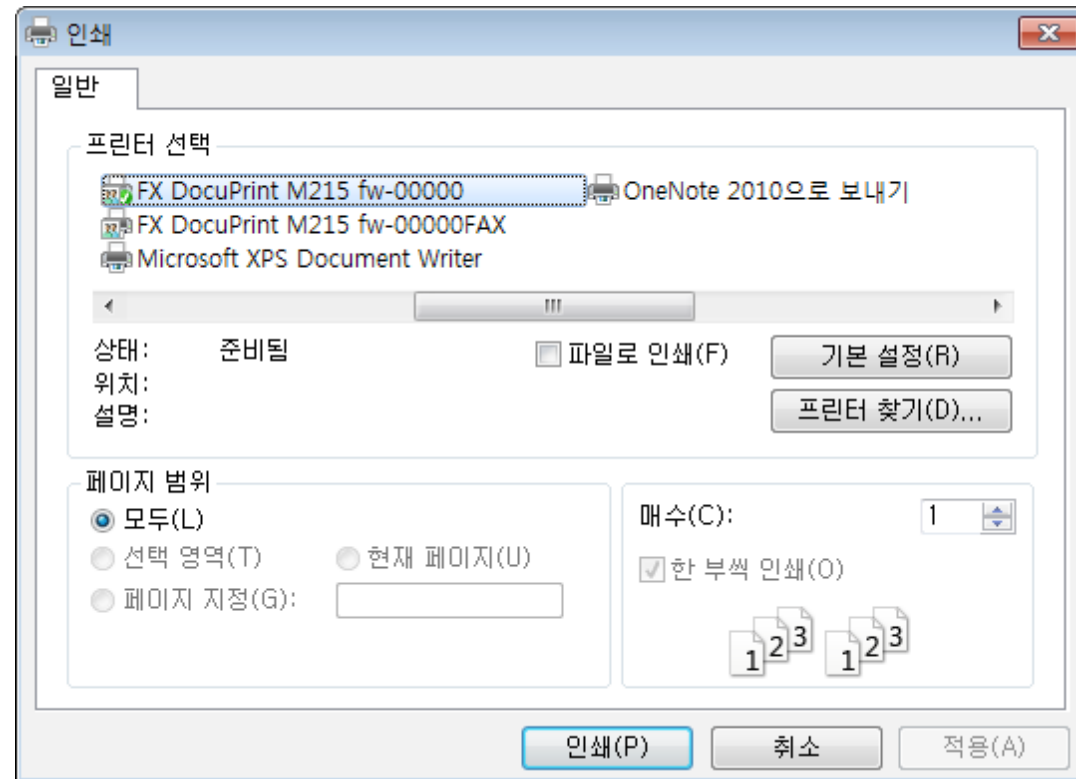
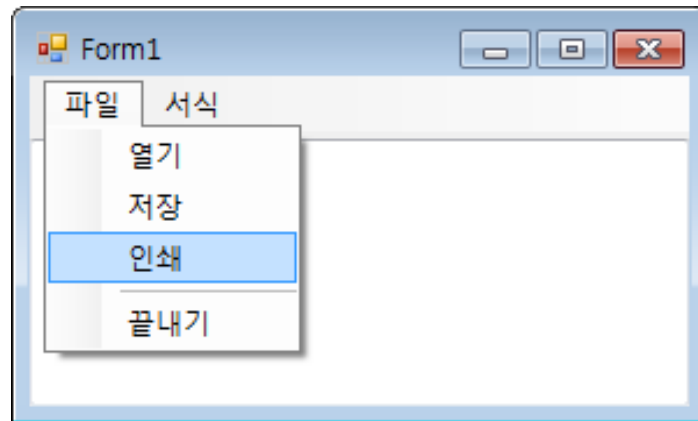
1) 위에서 만든 프로그램에 다음과 같이 [인쇄] 대화 상자 컨트롤을 추가한다. PrintDialog1 컨트롤의 속성은 기본값을 그대로 사용한다.



2) 코드 작성

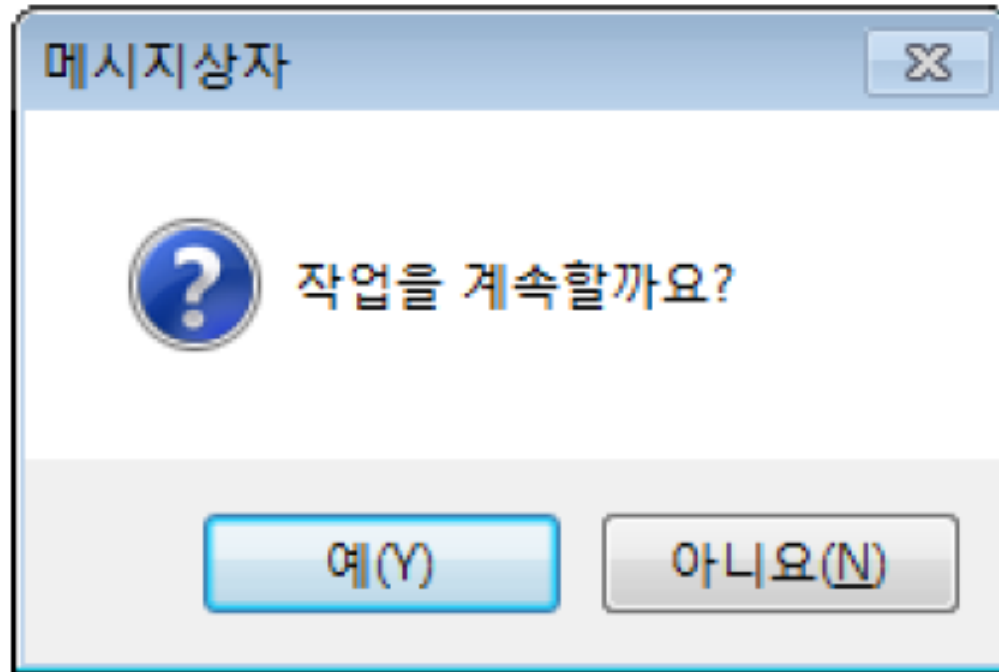
```
private void 인쇄ToolStripMenuItem_Click(object sender, EventArgs e)
{
    printDialog1.ShowDialog();
}
```


3) 실행 결과 확인



- MessageBox 함수
 - 오류나 경고 메시지 상자처럼 사용자에게 필요한 정보와 명령을 알려주는 대화 상자(메시지 상자)를 이용할 수 있게 해준다.

그림 10-2 MsgBox 함수를 이용한 메시지 상자의 예



- MessageBox 함수의 인수

형식 | MessageBox.Show(메시지 [,제목] [,버튼] [,아이콘])

표 10-7 MsgBox 함수의 인수

파란색은 필수, 나머지는 선택

인수	설명
메시지	메시지 상자에 표시할 텍스트로 최대 길이는 1,024자이다. 두 줄 이상 입력할 때는 캐리지 리턴 문자 Chr(13), 라인 피드 문자 Chr(10), 캐리지 리턴 및 라인 피드 문자 Chr(13)&Chr(10)을 사용한다.
버튼	표시할 버튼의 종류, 사용할 아이콘의 형태, 기본 버튼 지정 여부, 메시지 상자의 양식을 지정하는 값을 차례로 연결해서 버튼의 스타일을 설정한다. 생략하면 0-VbOkOnly, 즉 <확인> 버튼으로 표시된다.
제목	대화 상자의 제목 표시 줄에 나타나는 텍스트로, 생략하면 응용 프로그램의 이름이 나타난다.

MessageBox 함수

- 버튼 인수에 사용되는 상수값

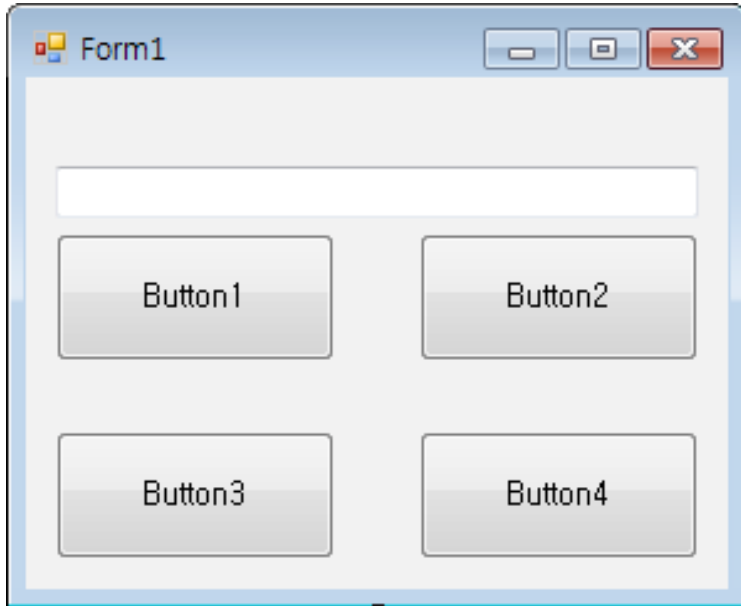
구분	상수	값	설명
버튼 MessageBox Button	Ok	0	<확인> 버튼만 표시
	OkCancel	1	<확인>, <취소> 버튼 표시
	AbortRetryIgnore	2	<중단>, <재시도>, <무시> 버튼 표시
	YesNoCancel	3	<예>, <아니요>, <취소> 버튼 표시
	YesNo	4	<예>, <아니요> 버튼 표시
	RetryCancel	5	<재시도>, <취소> 버튼 표시
아이콘 MessageBox Icon	Error or Hand or Stop	16	중대 오류 메시지 표시
	Question	32	질의 경고 표시
	Exclamation or Warning	48	메시지 경고 표시
	Asterisk or Information	64	메시지 정보 표시
초기 버튼 MessageBox DefaultButton	Button1	0	첫째 버튼이 기본 버튼
	Button2	256	둘째 버튼이 기본 버튼
	Button3	512	셋째 버튼이 기본 버튼
문자열 MessageBox Options	RightAlign	524288	문자열을 오른쪽으로 정렬
	RtlReading	1048576	문자열을 오른쪽에서 왼쪽으로 읽을 수 있게 설정

- MessageBox의 반환 상수 값

상수	값	설명
OK	1	<확인>
Cancel	2	<취소>
Abort	3	<중단>
Retry	4	<재시도>
Ignore	5	<무시>
Yes	6	<예>
No	7	<아니요>

- 실습 – 메시지 상자 사용하기

1) 윈도우 폼 디자인



2) 속성 설정

컨트롤	속성	속성값
TextBox1	Text	(빈칸)
Button1	Text	기본 메시지
Button2	Text	아이콘 추가
Button3	Text	반환값 계산
Button4	Text	변수 가져오기

3) <기본 메시지> 버튼을 클릭하면 일반 메시지가 출력되도록 코드 작성

```
private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show("기본 메시지 상자입니다.", "메시지 상자", MessageBoxButtons.OK);
}
```

4) <아이콘 추가> 버튼을 클릭하면 <예>, <아니요> 버튼과 아이콘이 출력되도록 코드 작성

```
private void button2_Click(object sender, EventArgs e)
{
    MessageBox.Show("작업을 계속할까요?", "메시지 상자",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
}
```

5) <반환값 계산> 버튼을 클릭하면 메시지 상자의 각 버튼에 해당하는 값을 반환 받아
TextBox1 컨트롤의 Text 속성으로 설정하는 코드 작성

```
private void button3_Click(object sender, EventArgs e)
{
    var returnVal = MessageBox.Show("작업을 계속할까요?", "메시지 상자",
        MessageBoxButtons.YesNoCancel);
    if(returnVal == DialogResult.OK)
    {
        textBox1.Text = "<예>를 눌렀습니다.";
    }else if(returnVal == DialogResult.No)
    {
        textBox1.Text = "<아니요>를 눌렀습니다.";
    }else if(returnVal == DialogResult.Cancel)
    {
        textBox1.Text = "<취소>를 눌렀습니다.";
    }
}
```


6) <변수 가져오기> 버튼을 클릭하면 텍스트박스에 입력된 값을 가져와 제곱 계산을 하도록 코드를 작성한다.

```
private void button4_Click(object sender, EventArgs e)
{
    var returnVal = MessageBox.Show(textBox1.Text + "이 값이 맞습니까?: "
        , "", MessageBoxButtons.OK);
    if(returnVal == DialogResult.OK)
    {
        int num = int.Parse(textBox1.Text);
        num = num * num;
        textBox1.Text = num.ToString();
    }
}
```

7) 실행 결과 확인

