

비주얼베이직을 이용한 한글컨트롤 사용법

(The HwpCtrl Manual for Visual Basic)

목차

1. 개요
2. 예제 1: 문서 내 글자입력 후 글자모양 및 색
변경하기
3. 예제 2: 작성 문서 이미지파일로 저장하기
4. 예제 3: 표 그리기

1. 개요

비주얼 베이직을 이용한 한글컨트롤 프로그래밍 시작에 앞서 함께 제공된 ‘한글컨트롤 프로그래밍 이해와 활용.hwp’ 파일을 참고하시길 바랍니다. 해당 매뉴얼에서 한글 매크로 기능을 이용한 액션 ID 유추 및 API 문서 이용법 등을 상세하게 설명하고 있습니다.

개요에서는 본격적인 한글컨트롤 프로그래밍 이전에 공통적으로 수행해야 하는 작업을 설명합니다.

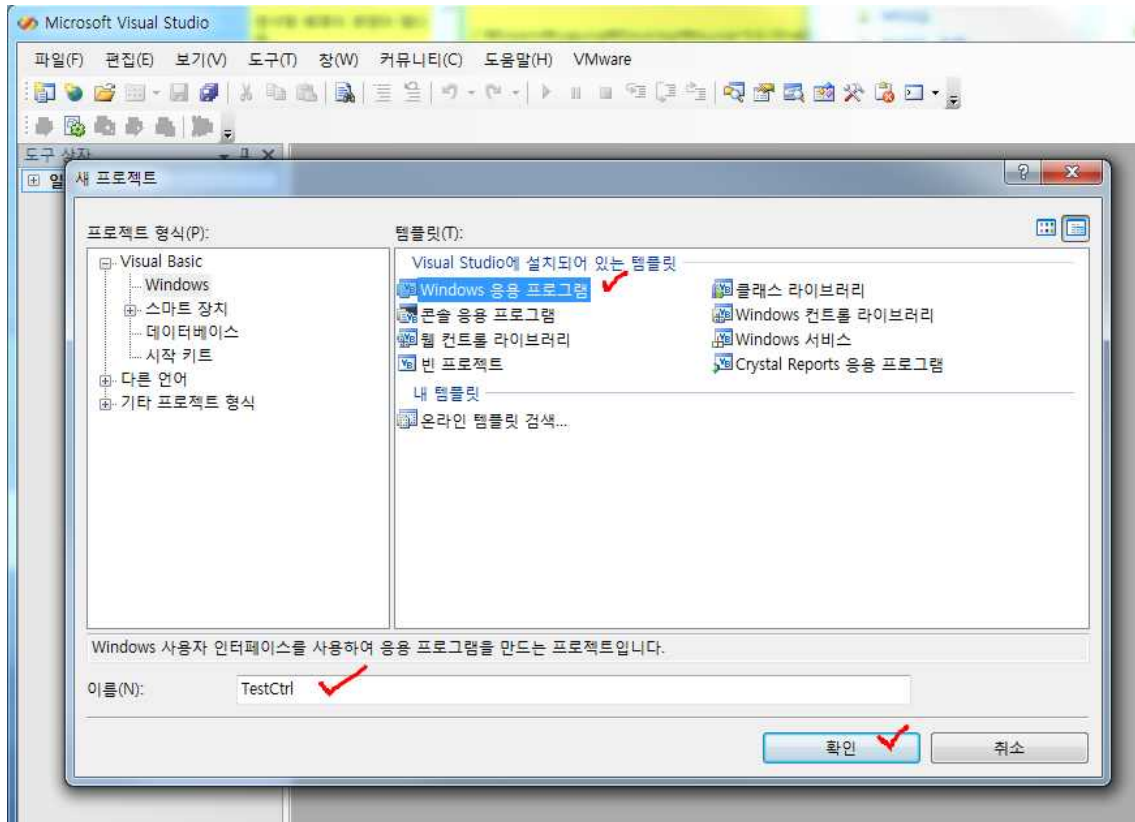


그림 1. 새 프로젝트 만들기

그림 1은 비주얼베이직에서 새 프로젝트를 만드는 화면입니다. Windows 응용프로그램을 선택을 한 후 원하는 이름의 새 프로젝트를 생성합니다. 새 프로젝트가 생성이 되면, 그림 2 처럼 좌측 도구상자에서 항목선택을 클릭하여 한글컨트롤을 도구상자에 추가합니다.

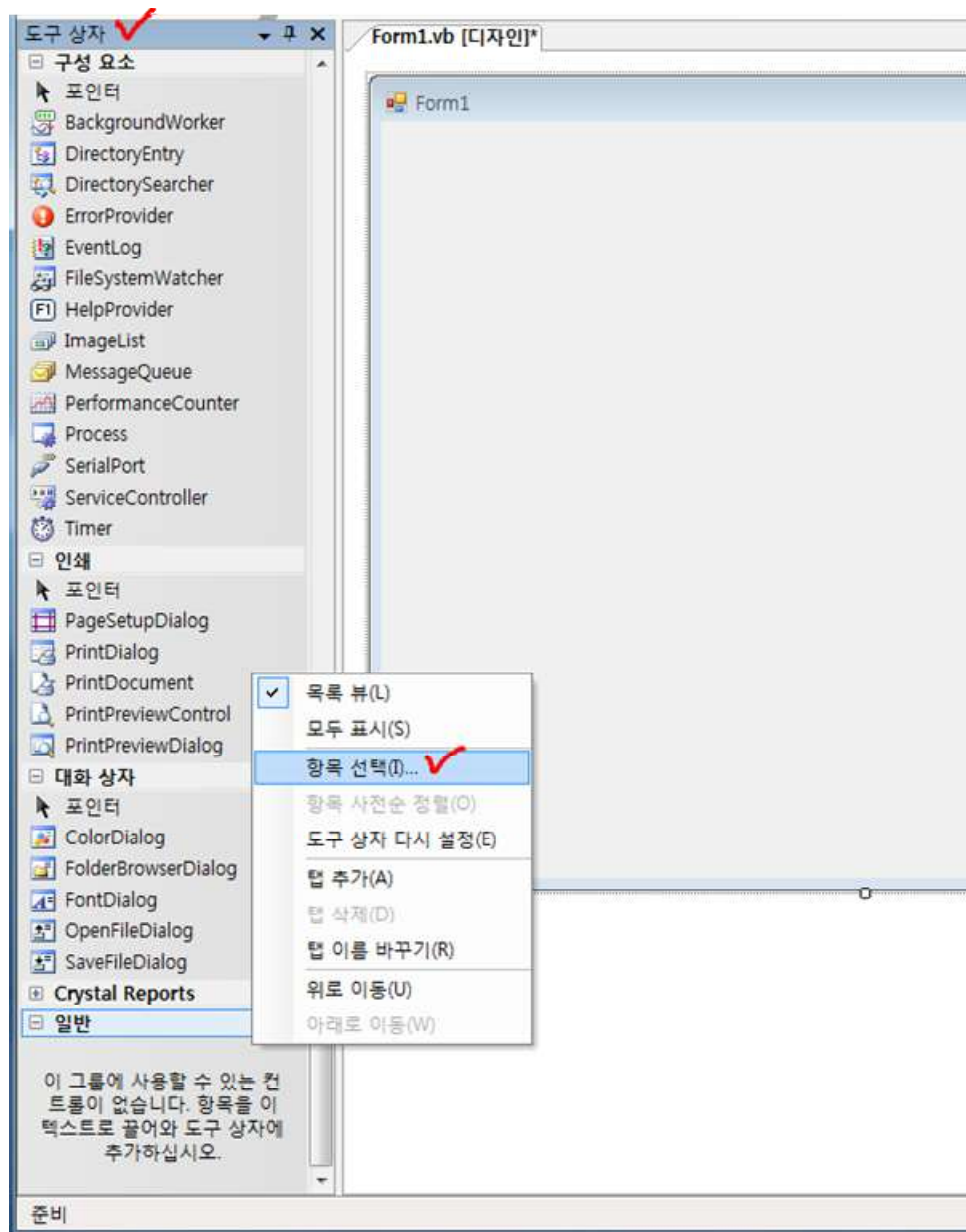


그림 2. 도구모음에 한글컨트롤 추가하기

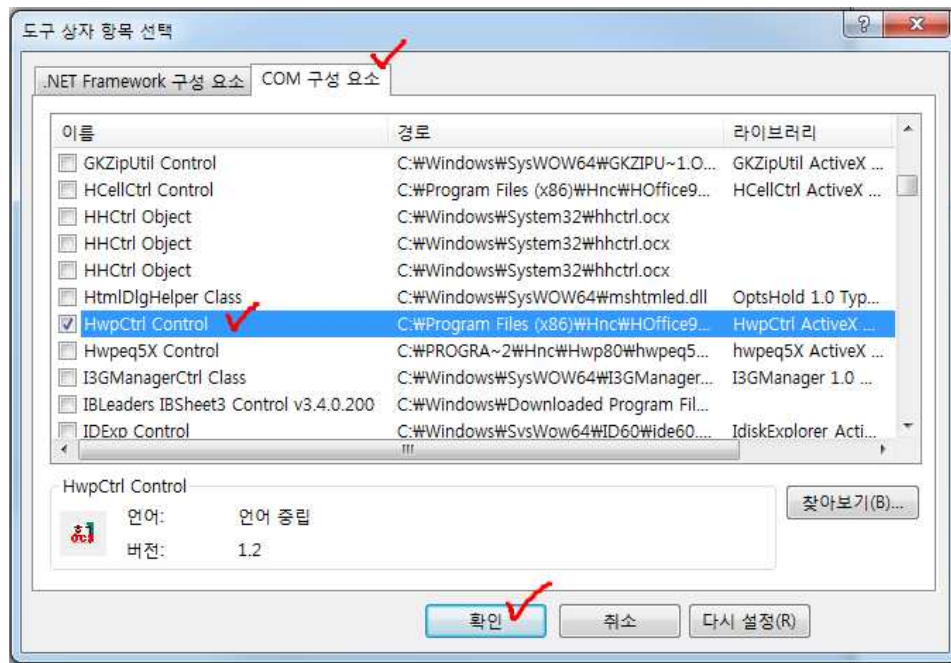


그림 3. 한글컨트롤 추가하기

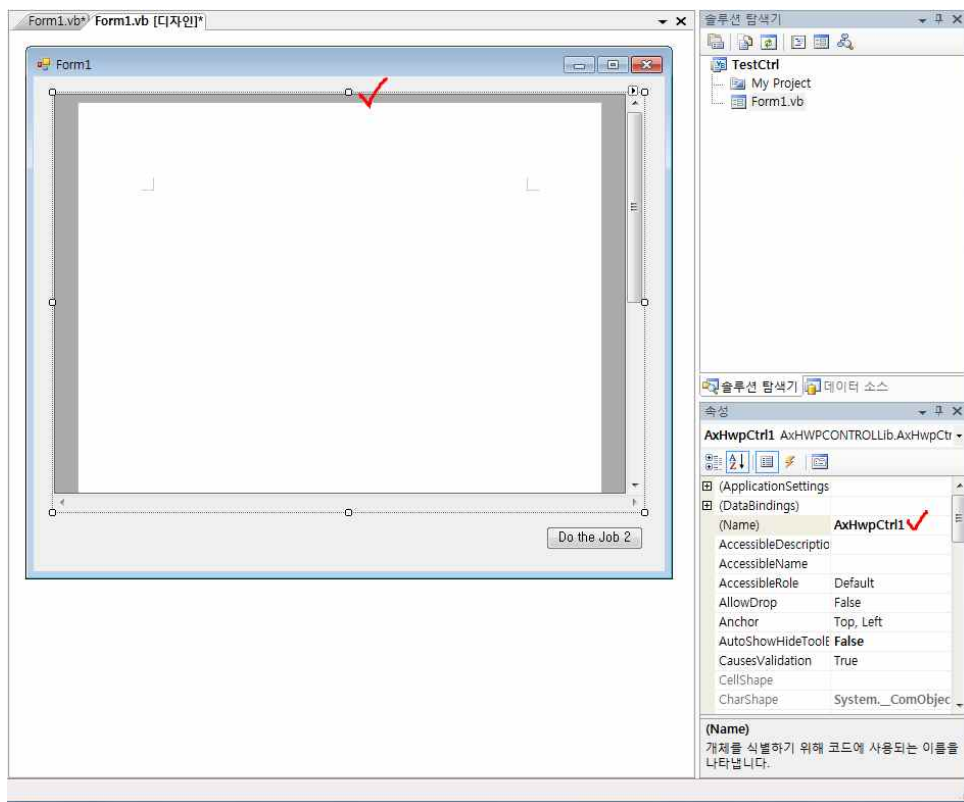


그림 4. 추가된 한글컨트롤

그림 3에서 한글컨트롤을 찾아서 추가하고 생성된 도구모음에서 클릭하면, 그림 4처럼 폼 디자인 부분에 한글컨트롤이 추가됩니다. 해당 한글컨트롤의 이름은 AxHwpCtrl1로 정의가 되어 있는 것을 우측 속성에서 확인할 수 있습니다. 해당 한글컨트롤 이름을 통해 한글컨트롤의 제어가 가능하게 됩니다.

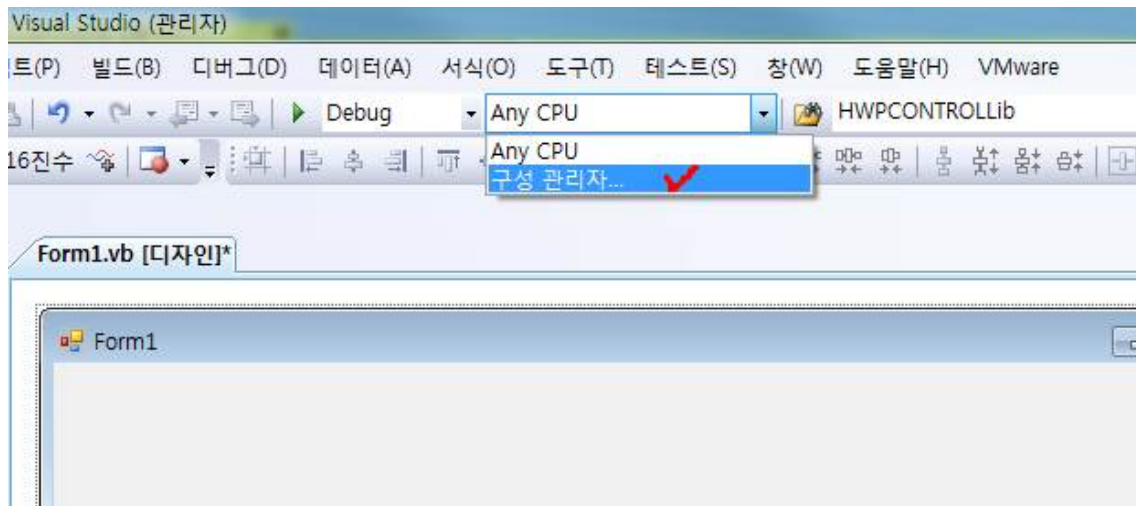


그림 5. x86 컴파일 설정

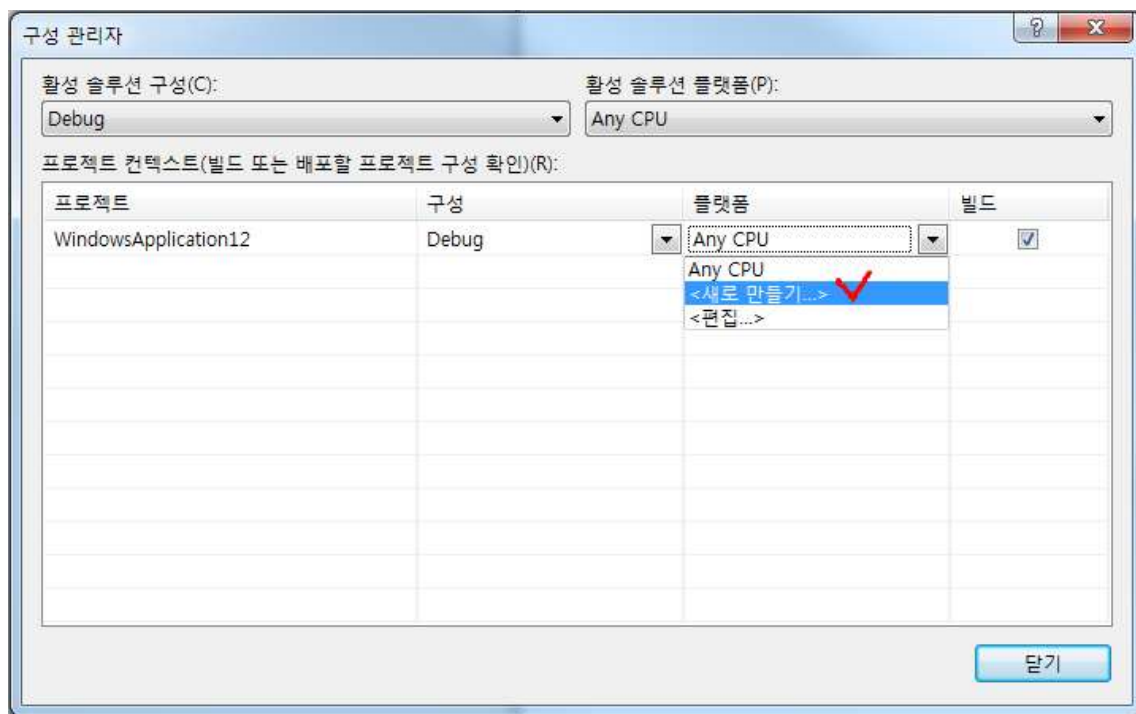


그림 6. x86 설정추가하기

그림 5의 경우처럼 상단 디버그 메뉴바에 x86이나 x64 대신 Any CPU라고 되어 있는 경우 빌드 시 문제가 발생될 수 있습니다. 그림 5처럼 구성관리자로 들어간 후 그림 6의 새로 만들기 클릭하여 x86설정을 추가합니다.

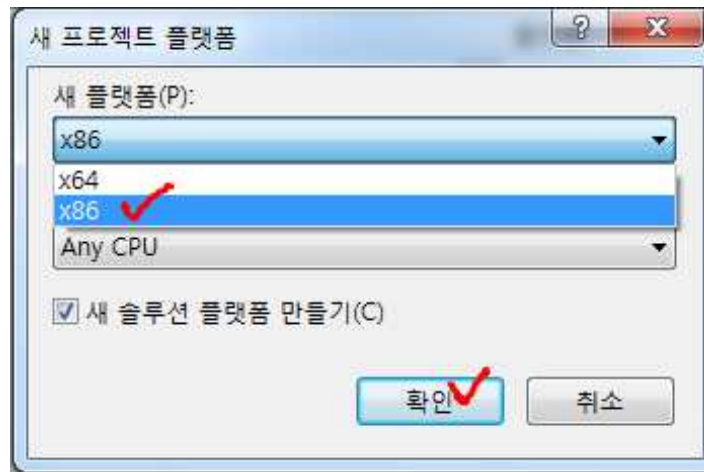


그림 7. x86 설정하기

여기까지 컴파일과 빌드가 되는지 확인하고, 이제 아래 3가지 예제를 통해 한글컨트롤 프로 그래밍을 해봅니다.

2. 예제 1: 문서 내 글자 입력 후 글자모양 및 색 변경하기

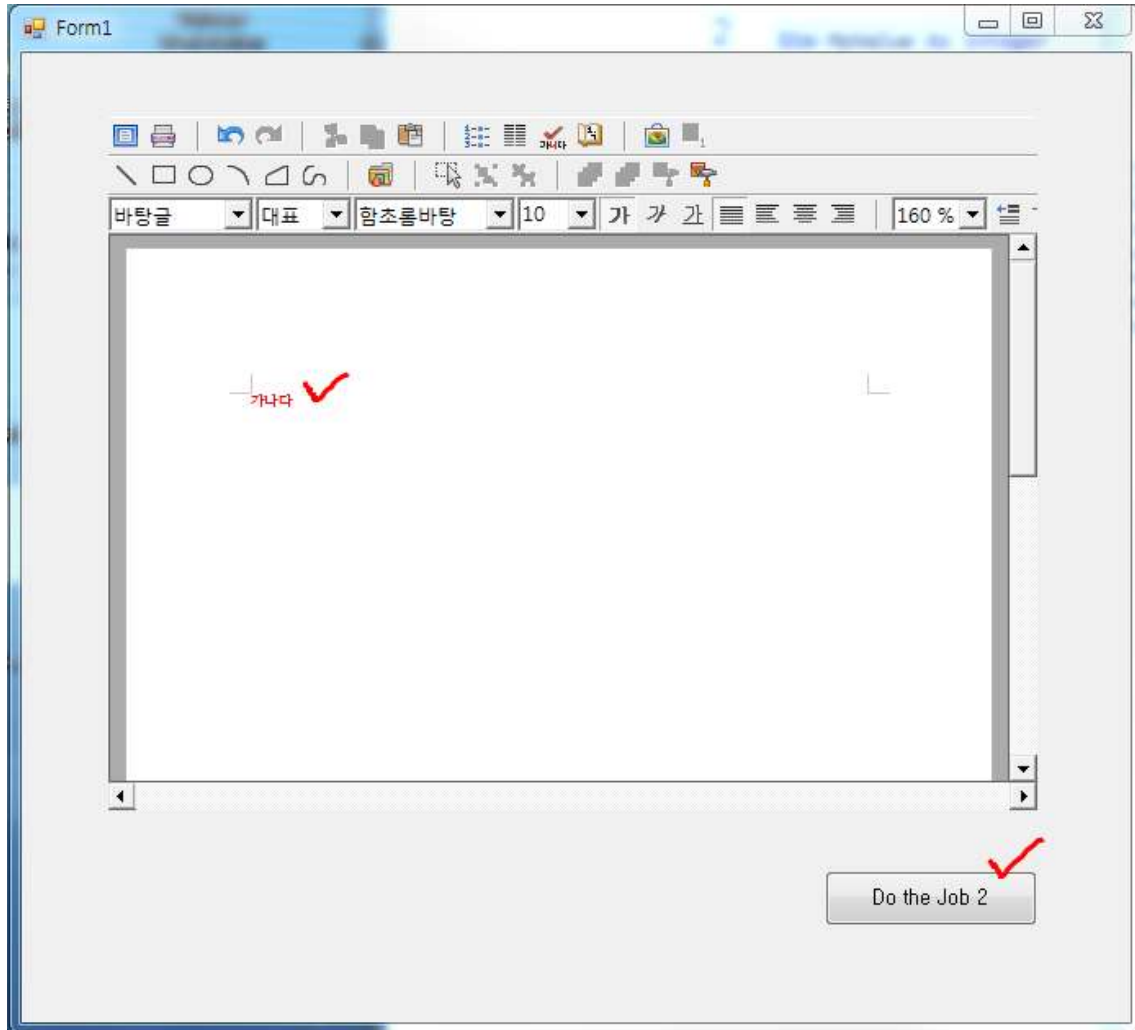


그림 8. (예제 1) 글자 관련 예제

그림 8은 비주얼 베이직으로 만든 응용프로그램의 우하단 'Do the Job 2' 버튼을 클릭하면 빈 한글컨트롤 문서 내에 '가나다' 글자가 자동으로 입력된 후 글씨체를 굵게 만들고 글자색을 빨강색으로 변경하는 예제입니다. 이 예제 프로그래밍에 사용된 액션 ID (Action ID) 및 파라미터 셋 ID (Parameter Set ID)는 다음 표와 같습니다.

표 1. 예제 1 액션 ID & 파라미터 셋 ID

액션 ID	파라미터 셋 ID	파라미터 셋 속성	파라미터 값	파라미터 형(타입)
InsertText	InsertText	Text	‘가나다’	PIT_BSTR (스트링, 문자열)
MoveSelLineBegin	없음			
CharShapeBold	없음			
CharShape	CharShape	TextColor	0x000000FF	PIT_UI4 (4바이트 비트값)

```

1 Public Class Form1
2
3     Dim Arr1() As Object = {"FilePreview", "Print", "Separator", "Undo", "Redo", "Separator", "Cut", "Copy", "Paste",
4     Dim Arr2() As Object = {"DrawObjCreatorLine", "DrawObjCreatorRectangle", "DrawObjCreatorEllipse", "DrawObjCreator",
5     Dim Arr3() As Object = {"StyleCombo", "CharShapeLanguage", "CharShapeTypeFace", "CharShapeHeight", "CharShapeBold"}
6
7
8     Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
9
10        Dim myStr As String = "가나다"
11        Dim vAct As Object = AxHwpCtrl1.CreateAction("InsertText")
12        Dim vSet As Object = vAct.CreateSet()
13        vAct.GetDefault(vSet)
14        vSet.SetItem("Text", CStr(myStr)) 'CStr(): Variant 타입의 명시적 지정
15        vAct.Execute(vSet)
16
17        AxHwpCtrl1.Run("MoveSelLineBegin")
18        AxHwpCtrl1.Run("CharShapeBold")
19
20        Dim vColor As Integer = &HFF '0x000000FF
21        'Not RGB, it's BGR (Blue=0, Green=0, Red=255) order.
22        'Green=0x0000FF00, Blue=0x00FF0000, Red=0x000000FF
23
24        Dim vAct2 As Object = AxHwpCtrl1.CreateAction("CharShape")
25        Dim vSet2 As Object = vAct2.CreateSet()
26        vAct2.GetDefault(vSet2)
27        vSet2.SetItem("TextColor", Cint(vColor)) 'Cint(): Variant 타입의 명시적 지정
28        vAct2.Execute(vSet2)
29
30    End Sub
31
32    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
33        AxHwpCtrl1.SetToolBar(0, Arr1)
34        AxHwpCtrl1.SetToolBar(1, Arr2)
35        AxHwpCtrl1.SetToolBar(2, Arr3)
36        AxHwpCtrl1.ShowToolBar(True)
37    End Sub
38 End Class
39
  
```

그림 9. 예제 1 소스코드

그림 9는 예제 1을 구현한 소스코드입니다. Button1_Click 함수가 해당 기능을 수행합니다. 예제 1에 대한 보다 상세한 설명은 ‘한글컨트롤 프로그래밍 이해와 활용.hwp’ 파일을 참고하시길 바랍니다.

3. 예제 2: 작성 문서 이미지파일로 저장하기

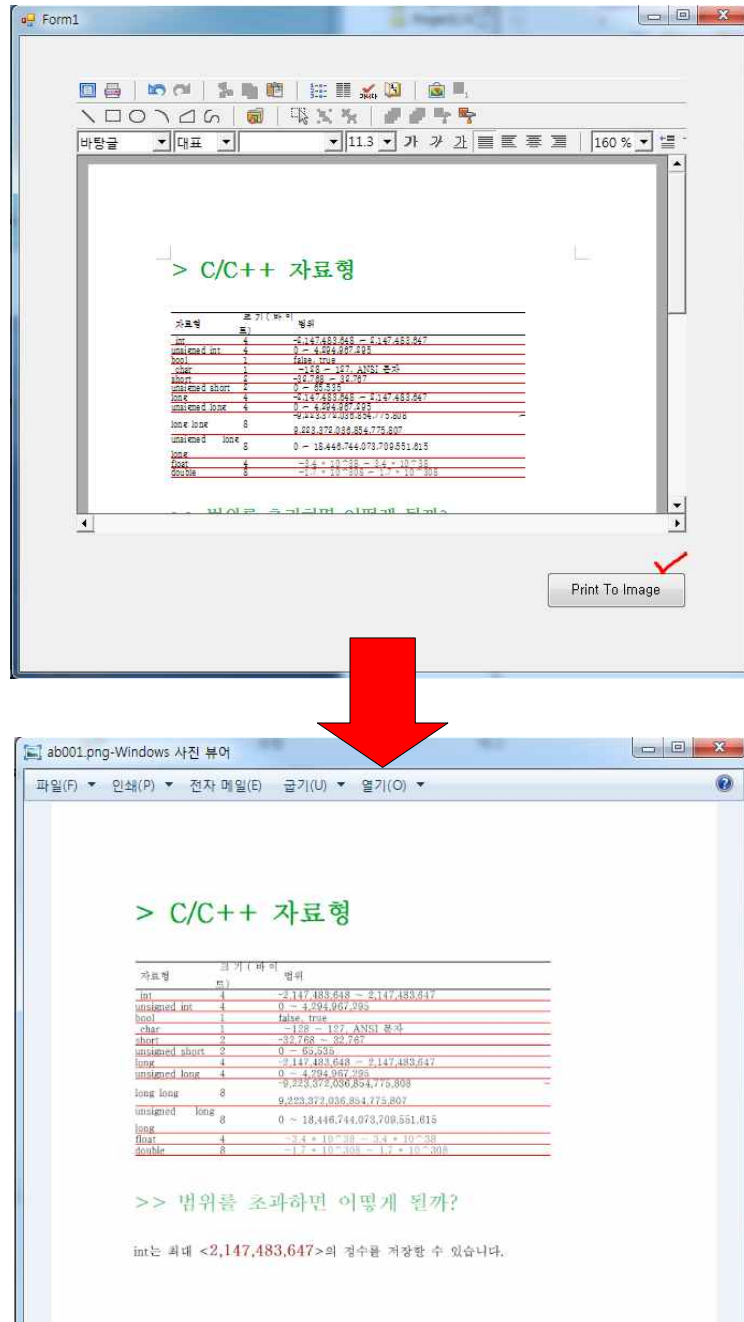


그림 10. (예제 2) 이미지 파일로 저장 예제

그림 10은 비주얼 베이직으로 만든 응용프로그램 우하단 'Print To Image' 버튼을 클릭하면 한글컨트롤 문서 내에 작성한 문서가 자동으로 이미지 파일로 저장되는 예제입니다. 이 예제 프로그래밍에 사용된 액션 ID (Action ID) 및 파라미터 셋 ID (Parameter Set ID)는 다음 표와 같습니다.

표 2. 예제 2 액션 ID & 파라미터 셋 ID

액션 ID	파라미터 셋 ID	파라미터 셋 속성	파라미터 값	파라미터 형(타입)
PrintToImage	PrintToImage	Format	3 (PNG파일)	PIT_UI1 (숫자 값)
PrintToImage	PrintToImage	FileName	"C:\\ab.png"	PIT_BSTR (문자열, 스트링 값)
PrintToImage	PrintToImage	ColorDepth	24 (24비트)	PIT_UI1 (숫자 값: 1,4,8,24)
PrintToImage	PrintToImage	Resolution	200 (200 dpi)	PIT_UI2 (숫자 값)



```

1 Public Class Form1
2
3     Dim Arr1() As Object = {"FilePreview", "Print", "Separator", "Undo", "Redo", "Separator", "Cut", "Copy", "Paste"}
4     Dim Arr2() As Object = {"DrawObjCreatorLine", "DrawObjCreatorRectangle", "DrawObjCreatorEllipse", "DrawObjCreator"}
5     Dim Arr3() As Object = {"StyleCombo", "CharShapeLanguage", "CharShapeTypeFace", "CharShapeHeight", "CharShapeBo"}
6
7
8     Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
9
10        vAct As Object = AxHwpCtrl1.CreateAction("PrintToImage")
11        vSet As Object = vAct.CreateSet()
12        vAct.GetDefault(vSet)
13        vSet.SetItem("Format", 3)
14        vSet.SetItem("FileName", "C:\\ab.png")
15        vSet.SetItem("ColorDepth", 24) ' 1,4,8,24 bit
16        vSet.SetItem("Resolution", 200) ' DPI 100, 200, 300, ...
17        vAct.Execute(vSet)
18
19    End Sub
20
21    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
22        AxHwpCtrl1.SetToolBar(0, Arr1)
23        AxHwpCtrl1.SetToolBar(1, Arr2)
24        AxHwpCtrl1.SetToolBar(2, Arr3)
25        AxHwpCtrl1.ShowToolBar(True)
26    End Sub
27 End Class
28

```

그림 11. 예제 2 소스코드

그림 11는 예제 2을 구현한 소스코드입니다. PrintToImage 파라미터 셋은 파라미터 속성으로 Format, FileName, ColorDepth, Resolution 등을 가집니다. 이는 각각 파일확장자형식, 파일저장경로 및 이름, 색심도(색상품질), 해상도(DPI)를 나타냅니다.

4. 예제 3: 표 그리기

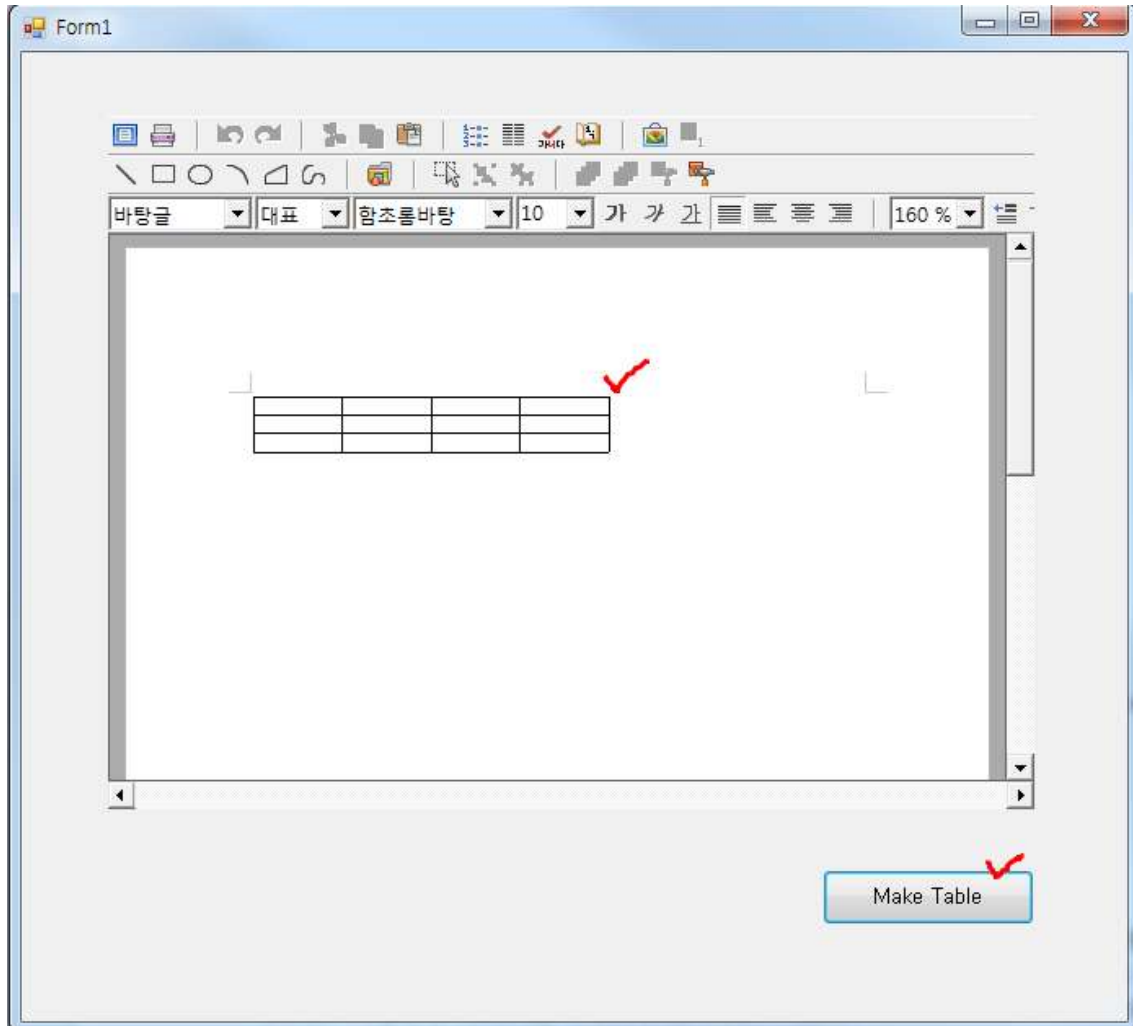
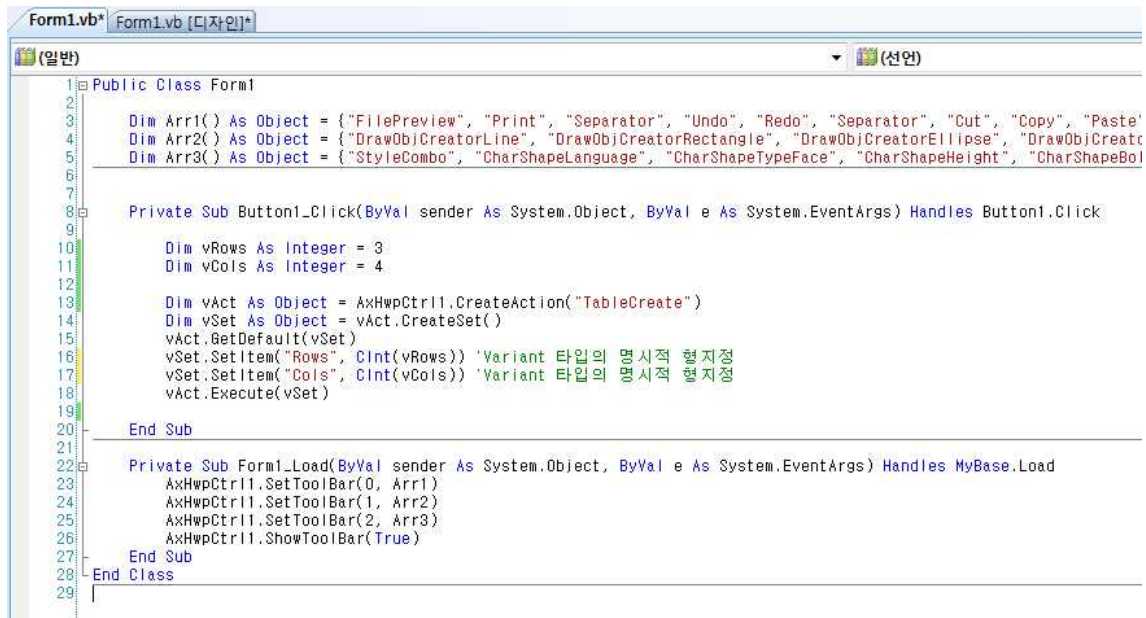


그림 12. (예제 3) 표 그리기 예제

그림 12은 비주얼 베이직으로 만든 응용프로그램 우하단 'Make Table' 버튼을 클릭하면 버튼 하단의 한글컨트롤 문서 내에 3 x 4의 표를 자동으로 생성하는 예제입니다. 이 예제 프로그래밍에 사용된 액션 ID (Action ID) 및 파라미터 셋 ID (Parameter Set ID)는 다음 표와 같습니다.

표 3. 예제 3 액션 ID & 파라미터 셋 ID

액션 ID	파라미터 셋 ID	파라미터 셋 속성	파라미터 값	파라미터 형(타입)
TableCreate	TableCreation	Rows	3	PIT_UI2 (숫자 값)
TableCreate	TableCreation	Cols	4	PIT_UI2 (숫자 값)



```

1 Public Class Form1
2
3     Dim Arr1() As Object = {"FilePreview", "Print", "Separator", "Undo", "Redo", "Separator", "Cut", "Copy", "Paste"}
4     Dim Arr2() As Object = {"DrawObjCreatorLine", "DrawObjCreatorRectangle", "DrawObjCreatorEllipse", "DrawObjCreator"}
5     Dim Arr3() As Object = {"StyleCombo", "CharShapeLanguage", "CharShapeTypeFace", "CharShapeHeight", "CharShapeBol"}
6
7
8     Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
9
10        Dim vRows As Integer = 3
11        Dim vCols As Integer = 4
12
13        Dim vAct As Object = AxHwpCtrl1.CreateAction("TableCreate")
14        Dim vSet As Object = vAct.CreateSet()
15        vAct.GetDefault(vSet)
16        vSet.SetItem("Rows", CInt(vRows)) 'Variant 타입의 명시적 형 지정
17        vSet.SetItem("Cols", CInt(vCols)) 'Variant 타입의 명시적 형 지정
18        vAct.Execute(vSet)
19
20    End Sub
21
22    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
23        AxHwpCtrl1.SetToolBar(0, Arr1)
24        AxHwpCtrl1.SetToolBar(1, Arr2)
25        AxHwpCtrl1.SetToolBar(2, Arr3)
26        AxHwpCtrl1.ShowToolBar(True)
27    End Sub
28 End Class
29
  
```

그림 13. 예제 3 소스코드

그림 13는 예제 3을 구현한 소스코드입니다. 예제 3은 단순히 버튼을 누르면 표를 생성해주는 형태입니다. 하지만 표 속성이나 셀 속성, 셀 내 값 변경 등을 제어하는 프로그램을 소스코드로 작성하기 위해서는 TableProperties, TablePropertyDialog, PutFieldText, InsertText 등의 좀 더 복잡한 형태의 코딩이 요구됩니다.