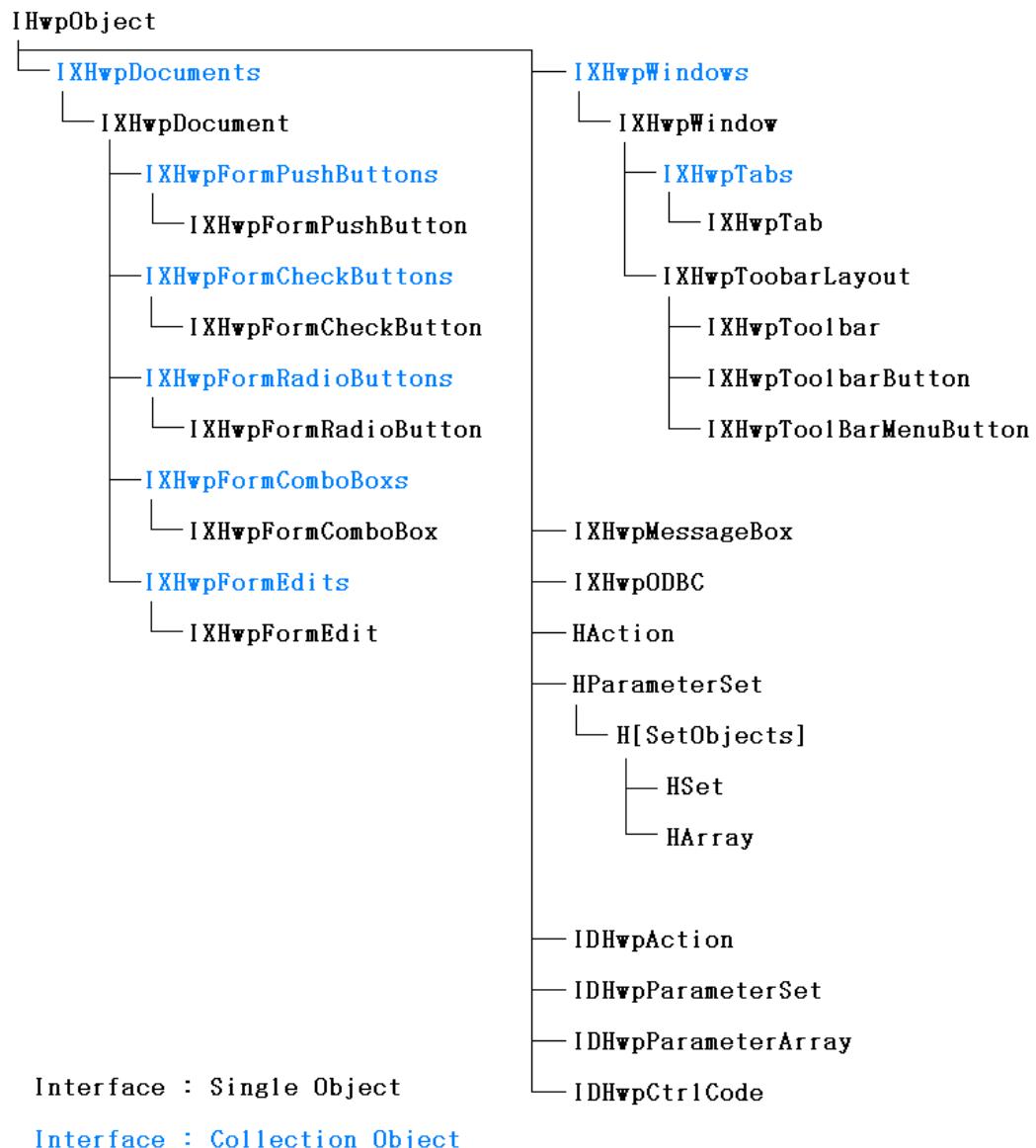


한글 OLE Automation

1.1. 한글 Objet Model



1.2. 한글 Objects

OLE Automation Object

Automation Object는 Automation Server가 Automation Client에게 제공하는 프로퍼티, 메소드, 이벤트로 구성된 개체이다.

IHwpObject는 한글 Automation Server의 최상위 오브젝트이며 단일 오브젝트와 Collection 오브젝트로 구성된 하위의 오브젝트는 모두 IHwpObject로부터 생성된다.

(JavaScript에서 IHwpObject 최상위 개체 생성)

```
var App = new ActiveXObject("HwpFrame.HwpObject.2");           //IXHwpObject 생성  
var Docs = App.XHwpDocuments;
```

(Visual C++에서 IHwpObject 최상위 개체 생성)

```
IHwpObject hwpobject;  
if (hwpobject.CreateDispatch("HwpFrame.HwpObject.2")) {  
    MessageBox("한글에 연결을 성공하였습니다.", "성공", MB_OK);  
} else {  
    MessageBox("한글에 연결을 실패하였습니다.", "오류", MB_OK);  
    PostQuitMessage(0);  
}
```

OLE Automation Collection Object

OLE Automation 단일 Object를 그룹화 하는 Object

한글에서 새글 또는 새탭으로 열려있는 문서는 IXHwpDocument Object에서 제어 가능하며 열려있는 문서를 모두 관리 할 수 있는 것은 IXHwpDocuments Collection Object이다.

IXHwpDocument Object는 IXHwpDocuments의 Add 메소드에 의하여 생성 될 수 있으며 생성된 Object는 Item 프로퍼티에 의하여 구할 수 있다.

Property

OLE Automation Object의 상태에 대한 정보를 제어할 수 있는 기능을 제공

IXHwpDocument의EditMode 프로퍼티는 지정된 문서가 읽기 모드인지 일반 편집 모드인지 부분 편집 모드인지를 얻을 수 있거나 설정할 수 있다.

Method

OLE Automation Object가 특정한 동작을 수행하도록 하는 기능을 제공

IXHwpMessageBox의 DoModal 메소드는 메시지 박스를 보여준다.

Event

OLE Automation Object에서 Client로 특정 사건이 발생했음을 알려주는 기능을 제공 하며 IHwpObject에서 이벤트를 얻을 수 있다.

(Visual Basic에서 IHwpObject 최상위 개체로부터 종료 이벤트 얻기)

```
Private WithEvents objHwpObject As HwpObject  
Private Sub Form_Load()  
    Set objHwpObject = New HwpObject
```

```

    MsgBox “한글에 접속”
End Sub

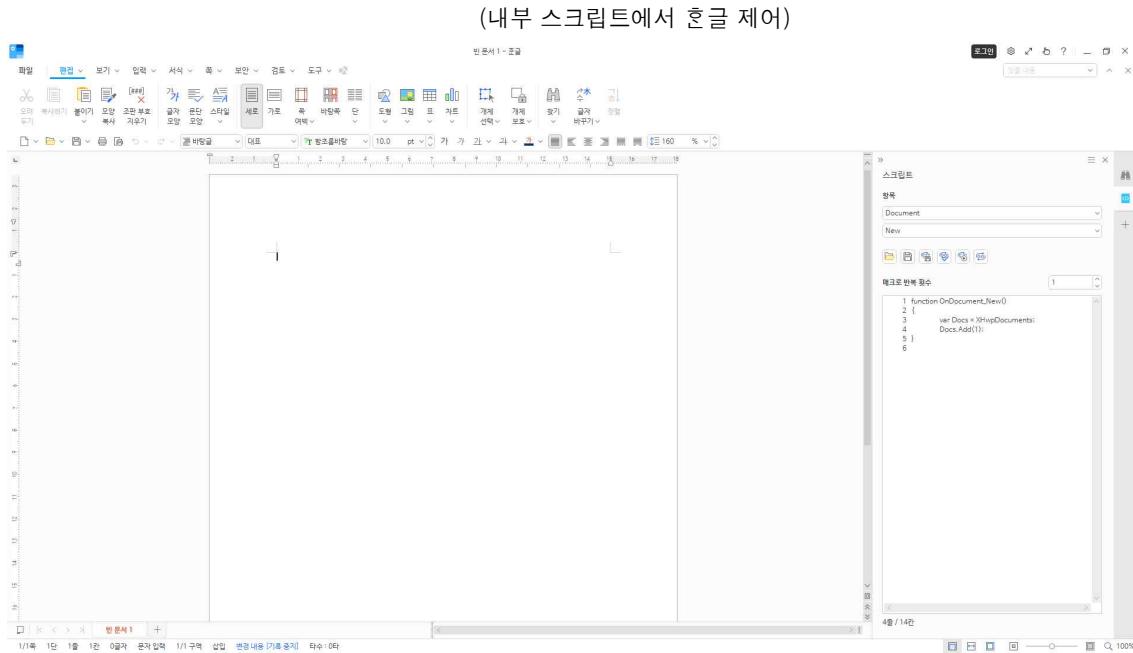
Private Sub objHwpObject_Quit()
    MsgBox "한글로부터 종료 이벤트를 받았습니다."
End Sub

```

1.3. 한글 내부 개체

내부 개체

한글은 Automation Client가 제어 할 수 있듯이 한글 내부에서도 스크립트로 제어할 수 있다.
스크립트는 JavaScript를 사용한다.
내부에서 사용되므로 IHwpObject는 이미 생성 되어있고 스크립트에서 최상위 IHwpObject 생성을 위한 코드는 필요가 없다.



JavaScript를 이용한 예제

```

//IHwpObject는 이미 생성되어있으므로 IHwpObject의 하위 항목은
//XHwpDocuments와 XHwpMessageBox같이 바로 사용 가능하다.
function OnDocument_Open()
{
    var Docs = XHwpDocuments;
    Docs.Add(1); //새 창으로 문서 생성 (0 = 새 창, 1 = 새 탭)
    var count = Docs.Count; //생성된 Document의 개수 얻기
    var msgbox = XHwpMessageBox;
    msgbox.String = count;
    msgbox.DoModal();
}

```

}

양식 개체와 내부 개체

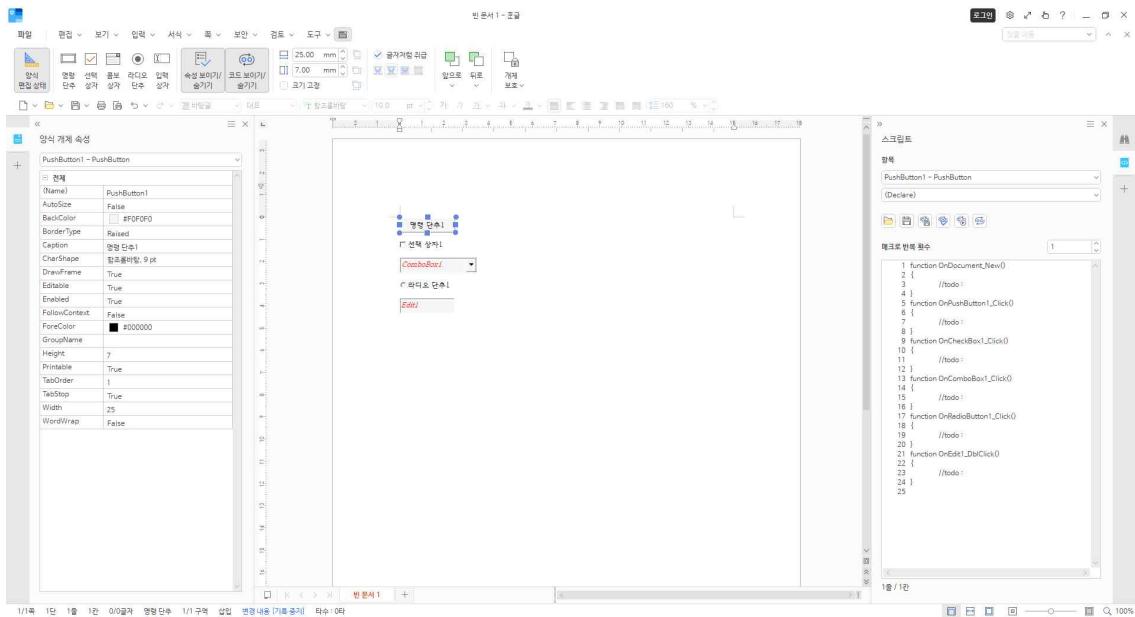
흔들은 양식 개체와 내부 개체를 연동하여 사용할 수 있다.

양식 개체는 PushButton, CheckBox, ComboBox, RadioButton, Edit가 있다.

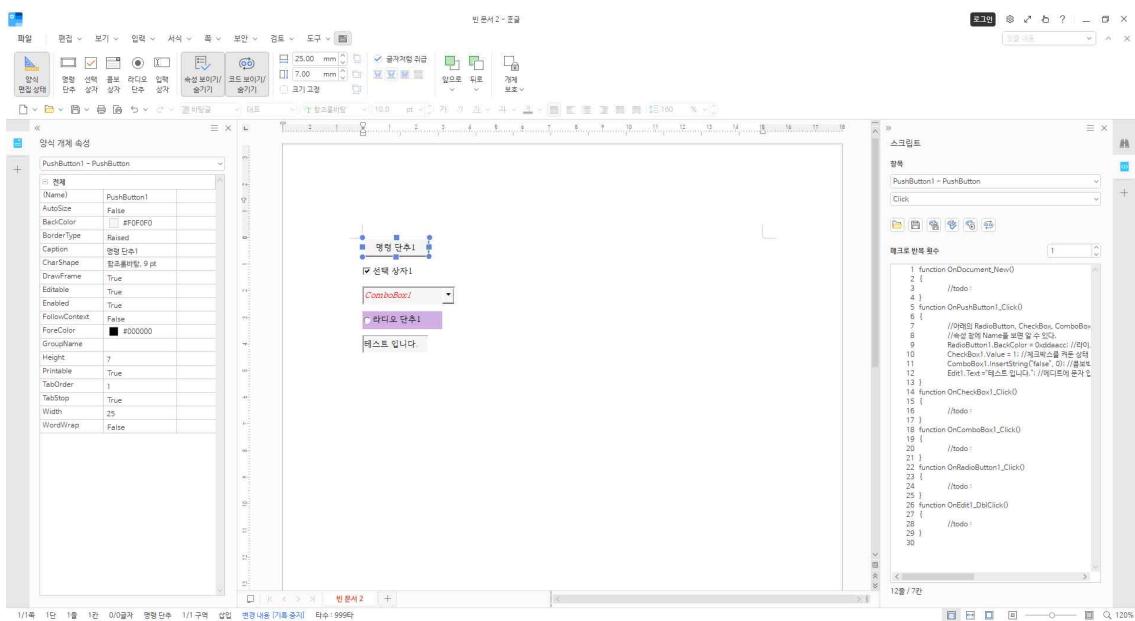
각 양식 개체는 Automation Object로 되어있으므로 스크립트로 제어가 가능하며

양식개체의 특정 이벤트에따라 스크립트가 수행되도록 할 수 있다.

(스크립트 적용 전)



(스크립트 적용 후)



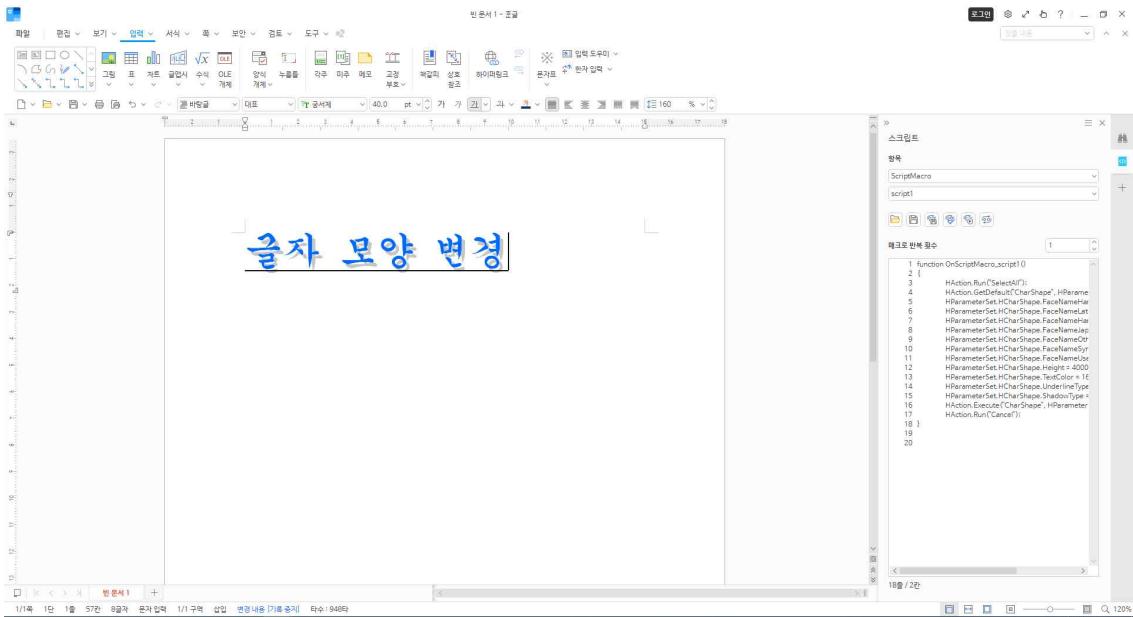
양식개체 스크립트 적용 예

```
function OnPushButton_Click()          //PushButton을 클릭 했을때 수행되는 함수
{
    //아래의 RadioButton, CheckBox, ComboBox등의 Name은
    //속성 창에 Name을 보면 알 수 있다.
    RadioButton.BackColor = 0xddaacc; //라이오버튼의 색깔을 변경
    CheckBox.Value = 1; //체크박스를 켜둔 상태
    ComboBox.InsertString("false", 0); //콤보박스에 false, true 문자 입력
    ComboBox.InsertString("true", 1);
    Edit.Text = "테스트입니다."; //에디트에 문자 입력
}
```

1.4. 외부 클라이언트와 훈글

훈글은 외부 클라이언트(응용 프로그램)와 연결되어 외부 클라이언트는 훈글을 제어할 수 있다.
외부 클라이언트가 훈글에 접속을 요청할 때 훈글은 자동으로 실행 된다.

(내부 스크립트로 글자 모양을 변경한 예)



(사용된 스크립트 – 스크립트 매크로로 생성)

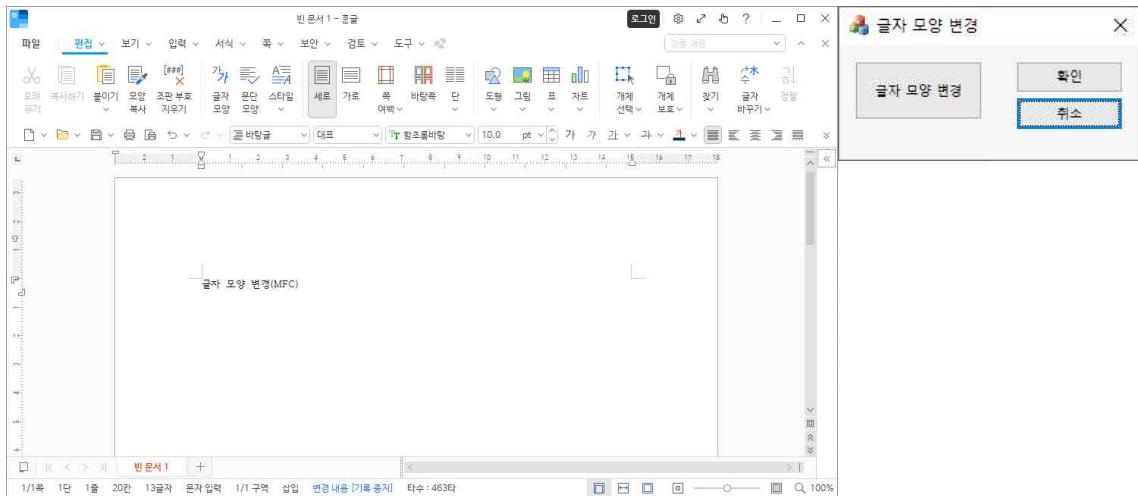
```
HAction.Run("SelectAll");
HAction.GetDefault("CharShape", HParameterSet.HCharShape.HSet);
HParameterSet.HCharShape.FaceNameHangul = "궁서체";
HParameterSet.HCharShape.FaceNameLatin = "궁서체";
HParameterSet.HCharShape.FaceNameHanja = "궁서체";
HParameterSet.HCharShape.FaceNameJapanese = "궁서체";
HParameterSet.HCharShape.FaceNameOther = "궁서체";
HParameterSet.HCharShape.FaceNameSymbol = "궁서체";
HParameterSet.HCharShape.FaceNameUser = "궁서체";
HParameterSet.HCharShape.Height = 4000;
HParameterSet.HCharShape.TextColor = 16737792;
HParameterSet.HCharShape.UnderlineType = 1;
```

```

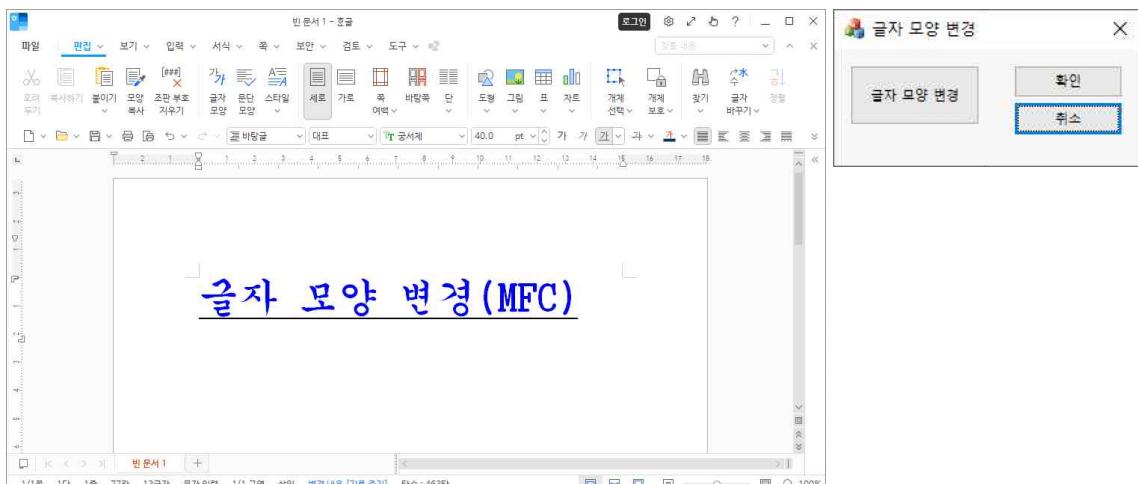
HParameterSet.HCharShape.ShadowType = 1;
HAction.Execute("CharShape", HParameterSet.HCharShape.HSet);
HAction.Run("Cancel");

```

(외부 클라이언트(exe 파일)가 한글을 제어 – 적용 전)



(외부 클라이언트(exe 파일)가 한글을 제어 – 적용 후)



(적용 코드)

```

void CTestHwpOleDlg::OnButton1()
{
    CDHwpParameterSet set;
    CDHwpAction dAction = myHwpObj.CreateAction(L"CharShape");
    set = dAction.CreateSet();
    dAction.GetDefault(set);

    myHwpObj.Run(L"SelectAll");
    set.SetItem(L"FaceNameHangul", COleVariant((LPCWSTR)"궁서체"));
    set.SetItem(L"FaceNameLatin", COleVariant((LPCWSTR)"궁서체"));
    set.SetItem(L"FaceNameHanja", COleVariant((LPCWSTR)"궁서체"));
    set.SetItem(L"FaceNameJapanese", COleVariant((LPCWSTR)"궁서체"));
    set.SetItem(L"FaceNameOther", COleVariant((LPCWSTR)"궁서체"));
}

```

```

set.SetItem(L"FaceNameSymbol", COleVariant((LPCWSTR)"궁서체"));
set.SetItem(L"FaceNameUser", COleVariant((LPCWSTR)"궁서체"));

set.SetItem(L"FontTypeHangul", COleVariant((LPCWSTR)myHwpObj.FontType(L"TTF")));
set.SetItem(L"FontTypeLatin", COleVariant((LPCWSTR)myHwpObj.FontType(L"TTF")));
set.SetItem(L"FontTypeHanja", COleVariant((LPCWSTR)myHwpObj.FontType(L"TTF")));
set.SetItem(L"FontTypeJapanese", COleVariant((LPCWSTR)myHwpObj.FontType(L"TTF")));
set.SetItem(L"FontTypeOther", COleVariant((LPCWSTR)myHwpObj.FontType(L"TTF")));
set.SetItem(L"FontTypeSymbol", COleVariant((LPCWSTR)myHwpObj.FontType(L"TTF")));
set.SetItem(L"FontTypeUser", COleVariant((LPCWSTR)myHwpObj.FontType(L"TTF")));

set.SetItem(L"Height", COleVariant((long)myHwpObj.PointToHwpUnit(40)));
set.SetItem(L"TextColor", COleVariant((long)myHwpObj.RGBColor(255, 0, 0)));
set.SetItem(L"UnderlineType", COleVariant((long)1));

dAction.Execute(set);
myHwpObj.Run(L"Cancel");
}

```

1.5. 외부 클라이언트와 실행 중인 훈글

이미 실행 중인 훈글에 외부 클라이언트가 접속하여 훈글을 제어할 수 있다. (Running Object Table을 액세스 할 수 있는 프로그래밍 언어로 작성 된 프로그램에서 가능함)

예제 코드

```

void CTestRunningHwpDlg::OnConnectButton()
{
    WCHAR szobjectname[] = L"!HwpObject.130";
    WCHAR* szname = NULL;

    IEnumMoniker* pEnumMoniker = NULL;
    IRunningObjectTable* pRot = NULL;
    IMoniker* iMoniker = NULL;
    IUnknown* iUnknown = NULL;
    IBindCtx* iBindCtx = NULL;
    ULONG uelement = 0;

    CHwpObject cActiveHwpObj;

    HRESULT hr = GetRunningObjectTable(0, &pRot);
    if (FAILED(hr)) {
        return;
    }

    hr = pRot->EnumRunning(&pEnumMoniker);
    if (FAILED(hr)) {
        pRot->Release();
        return;
    }
}

```

```

    }

    hr = CreateBindCtx(0, &iBindCtx);
    if (FAILED(hr)) {
        pEnumMoniker->Release();
        pRot->Release();
        return;
    }

    while (SUCCEEDED(pEnumMoniker->Next(1, &iMoniker, &uelement))) {
        if (uelement == 0)
            return;

        pRot->GetObject(iMoniker, &iUnknown);

        iMoniker->GetDisplayName(iBindCtx, NULL, (OLECHAR**)&szname);
        CString strobject;
        strobject = szname;
        if (strobject.Find(szobjectname, 0) != -1) {
            LPDISPATCH pdisp;
            iUnknown->QueryInterface(IID_IDispatch, (void**)&pdisp);
            if (pdisp) {
                cActiveHwpObj.AttachDispatch(pdisp);
                cActiveHwpObj.Run(L"SelectAll");

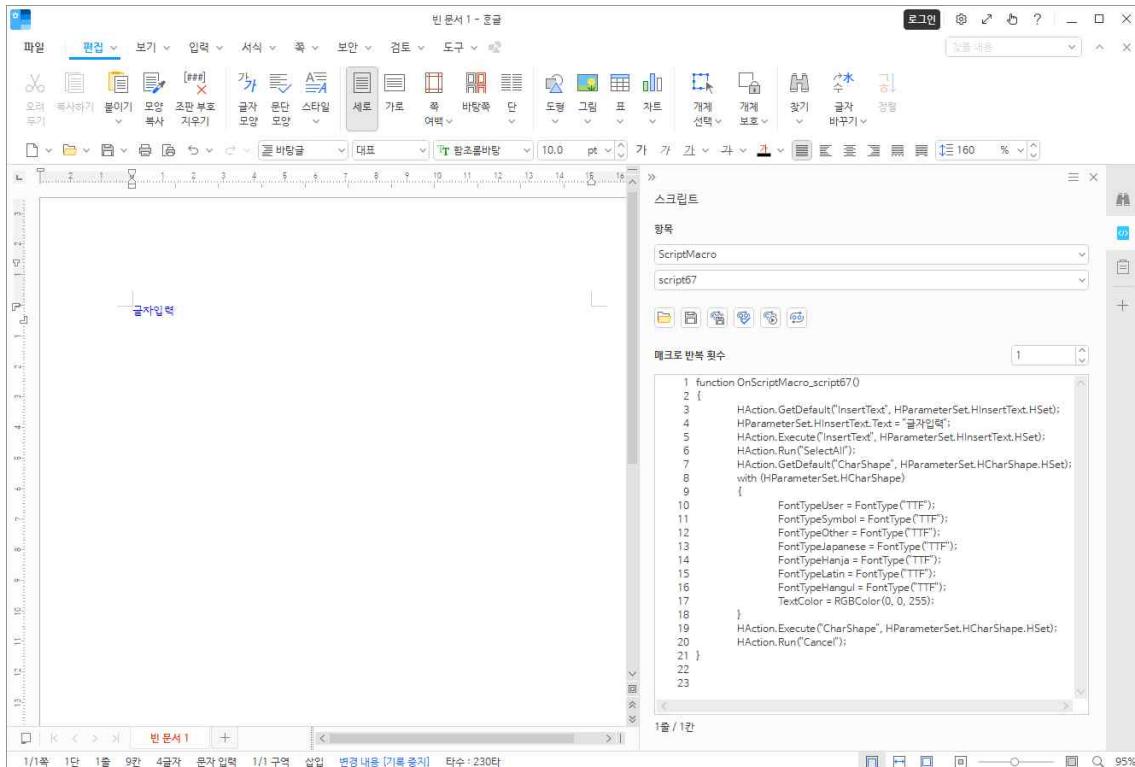
            }
        }
    }
}

```

1.6. 스크립트 매크로를 이용한 코드 적용 예제

스크립트 매크로 기록으로 코드 생성

1. "글자입력" 문자열 입력
2. 전체 블록 설정
3. 글자모양에서 색깔을 파란색으로 설정
4. 블록 설정 해제



스크립트 매크로로 생성된 스크립트 코드

```
function OnScriptMacro_script67()
{
    HAction.GetDefault("InsertText", HParameterSet.HInsertText.HSet);
    HParameterSet.HInsertText.Text = "글자입력";
    HAction.Execute("InsertText", HParameterSet.HInsertText.HSet);
    HAction.Run("SelectAll");
    HAction.GetDefault("CharShape", HParameterSet.HCharShape.HSet);
    with (HParameterSet.HCharShape)
    {
        FontTypeUser = FontType("TTF");
        FontTypeSymbol = FontType("TTF");
        FontTypeOther = FontType("TTF");
        FontTypeJapanese = FontType("TTF");
        FontTypeHanja = FontType("TTF");
        FontTypeLatin = FontType("TTF");
        FontTypeHangul = FontType("TTF");
        TextColor = RGBColor(0, 0, 255);
    }
}
```

```
        }  
        HAction.Execute("CharShape", HParameterSet.HCharShape.HSet);  
        HAction.Run("Cancel");  
    }  
}
```

생성된 스크립트 코드를 인터넷 익스플로러에 적용한 예제

```
<HTML>
<SCRIPT LANGUAGE="JSCRIPT">
var App = new ActiveXObject("HwpFrame.HwpObject.2");
function OnScriptMacro_script1()
{
    App.HAction.GetDefault("InsertText", App.HParameterSet.HInsertText.HSet);
    App.HParameterSet.HInsertText.Text = "글자입력";
    App.HAction.Execute("InsertText", App.HParameterSet.HInsertText.HSet);
    App.HAction.Run("MoveSelLineBegin");
    App.HAction.GetDefault("CharShape", App.HParameterSet.HCharShape.HSet);
    App.HParameterSet.HCharShape.TextColor = 16750848;
    App.HAction.Execute("CharShape", App.HParameterSet.HCharShape.HSet);
    App.HAction.Run("Cancel");
}
</SCRIPT>
<BODY>
<BUTTON OnClick="OnScriptMacro_script1()"> 글자 테스트</BUTTON>
</BODY>
</HTML>
```

생성된 스크립트 코드를 MFC에 적용한 예제

```
IHwpObject pHwpObject;
void CTestText::OnButtonClick()
{
    BOOL bres = pHwpObject.CreateDispatch("HwpFrame.HwpObject.2");
    if (bres == FALSE)
        return ;
    HAction haction;
    haction.AttachDispatch(pHwpObject.GetHAction());
    HParameterSet hparameterset;
    hparameterset.AttachDispatch(pHwpObject.GetHParameterSet());
    HInsertText hinserertext;
    hinserertext.AttachDispatch(hparameterset.GetHInsertText());
    HSet hset1;
    hset1.AttachDispatch(hinserertext.GetHSet());
    haction.GetDefault("InsertText", hset1);
    hinserertext.Text = "글자입력";
    haction.Execute("InsertText", hset1);
    haction.Run("MoveSelLineBegin");
    HCharShape hcharshape;
    hcharshape.AttachDispatch(hparameterset.GetHCharShape());
    HSet hset2;
    hset2.AttachDispatch(hcharshape.GetHSet());
    haction.GetDefault("CharShape", hset2);
    hcharshape.TextColor = 16750848;
    haction.Execute("CharShape", hset2);
    haction.Run("Cancel");
}
```

1.7. OLE Automation 오브젝트

IHwpObject : 최상위 개체 (모든 Automation Object의 최상위 Object이다.

Item Name	Description
IsModified(Property)	<p>Description 문서가 변경되어있는지 나타낸다. (읽기 전용)</p> <p>Remark 0 = 변경되지 않은 깨끗한 상태 1 = 변경된 상태 2 = 변경되었으나 자동 저장된 상태</p>
IsEmpty(Property)	<p>Description 아무 내용도 들어있지 않은 빈 문서인지 여부를 나타낸다.(읽기 전용)</p>
EditMode(Property)	<p>Description 현재 편집 모드</p> <p>Remark 0 : 읽기 전용 1 : 일반 편집모드 2 : 양식 모드(양식 사용자 모드) : Cell과 누름틀 중 양식 모드에서 편집 가능 속성을 가진 것만 편집 가능하다. 16 : 배포용 문서 (SetEditMode로 지정 불가능)</p>
SelectionMode(Property)	<p>Description 문서의 내용이 어떤 Selection 상태인가를 알려준다.(읽기 전용)</p> <p>Remark 일반 블록이 아닌 F3키나 F4키에 의해 블록이 지정된 경우, HWPSEL_STRICT_MODE (= 0x10, 십진수 16)으로 OR 마스크되어 오기때문에, 항상 0x0F(십진수15)로 AND 마스크한 결과로 판단하도록 한다.</p>
CurFieldState(Property)	<p>Description 캐럿이 위치한 필드의 상태 정보를 구한다.(읽기 전용)</p> <p>Remark bit 5 ~ 31 = 예약 bit 4 = 필드명의 존재 여부 (1 = 있음, 0 = 없음) bit 0 ~ 3 = 필드의 종류 (0 = 없음, 1 = 셀, 2 = 누름틀)</p>
PageCount(Property)	<p>Description 문서 페이지 수 (읽기 전용)</p> <p>Remark 문서의 전체 페이지 수를 나타낸다. 문서 전체에 대한 pagination이 수행되지 않을 상태에서 이 property를 참조하면 먼저 문서 전체의 pagination을 먼저 수행하므로 긴 문서에 대해 문서 내용 변경과 참조를 반복하면 속도가 심각하게 느려질 수 있다.</p>
CellShape(Property)	<p>Description 현재 선택되어있는 표와 셀의 모양 정보를 나타낸다.</p> <p>Remark ParameterSet/Table로 표의 속성에 대한 기본 정보를 나타내며, 이 가운데 "Cell" 아이템이 ParameterSet/Cell로 셀의 속성을 나타낸다. 셀 블록이 잡혀있지 않은 상태이면 현재 캐럿이 위치한 셀 하나만을 대상으로 한다. 현재 표 내부에 캐럿이 위치하지 않으면 에러가 발생한다.</p>
CharShape(Property)	<p>Description 현재 Selection의 글자 모양을 나타낸다.</p> <p>Remark property get을 수행하면 현재 selection 내의 글자 모양을 구할 수 있다. selection이 존재하지 않으면 현재 캐럿이 위치한 곳의 글자 모양을 돌려준다. 글자 모양 중 특정 항목이 selection 내에서 서로 다른</p>

	<p>속성을 가지고 있으면 아예 아이템 자체가 존재하지 않는다. property set을 수행하면 아이템이 존재하는 항목에 대해서만 속성을 설정한다. (ParameterSet의 형식은 ParameterSet/CharShape 참조.)</p>
HeadCtrl(Property)	<p>Description 문서 중 첫 번째 컨트롤(읽기 전용) Remark 문서 중의 모든 컨트롤(표, 그림 등의 특수 문자들)은 linked list로 서로 연결되어 있는데, 그 list의 시작 컨트롤을 나타낸다. 이 컨트롤로부터 시작, Ctrl.Next를 이용해 forward iteration을 수행할 수 있다.</p>
LastCtrl(Property)	<p>Description 문서 중 마지막 컨트롤 (읽기 전용) Remark 문서 중의 모든 컨트롤(표, 그림 등의 특수 문자들)은 linked list로 서로 연결되어 있는데, 그 list의 마지막 컨트롤을 나타낸다. 이 컨트롤로부터 시작, Ctrl.Prev를 이용해 backward iteration을 수행할 수 있다.</p>
CurSelectedCtrl(Property)	<p>Description 현재 선택되어 있는 컨트롤(읽기 전용)</p>
ParaShape(Property)	<p>Description 현재 Selection의 문단 모양을 나타낸다. Remark 개념과 사용법은 CharShape과 동일하다. (ParameterSet의 형식은 ParameterSet/ParaShape 참조.)</p>
ParentCtrl(Property)	<p>Description 현재 캐럿의 상위 컨트롤(읽기 전용) Remark 상위 컨트롤은 현재 캐럿이 위치한 리스트를 보유한 컨트롤이다. 예를 들어 셀 내부에 위치하면 표, 각주 내용에 위치하면 각주, 바탕쪽이면 구역 컨트롤이 상위 컨트롤이다. 현재 캐럿이 본문 레벨에 위치해 상위 컨트롤이 없을 때는 NULL이 리턴된다.</p>
ViewProperties(Property)	<p>Description 뷰의 상태 정보 Remark 조판 부호, 화면 확대 비율과 같은 view에 관련된 정보를 나타낸다. (ParameterSet의 형식은 ParameterSet/ViewProperties 참조.)</p>
Path(Property)	<p>Description 문서 파일의 경로</p>
IsPrivateInfoProtected(Property)	<p>Description 개인정보가 보호되어 있는지 여부를 알 수 있는 속성(읽기 전용) Remark 문서 내에 개인정보가 보호된 문서인지 아닌지를 알 수 있다.</p>
Open(Method)	<p>Description 문서 파일을 연다. Declaration Open(BSTR path, BSTR format, BSTR arg) Parameters path : 문서 파일의 경로(URL 사용 가능) format : 문서 형식. 별도 설명 참조. 빙 문자열을 지정하면 자동으로 인식한다. 생략하면 빙 문자열이 지정된다. arg : 세부 옵션. 의미는 format에 지정한 파일 형식에 따라 다르다. 생략하면 빙 문자열이 지정된다. Remark format에 지정할 수 있는 문서 형식은 현재 시스템에 설치된 문서 필터 (*.dft)의 종류에 따라 달라진다. 일반적으로 설치되는 형식들에는 다</p>

음과 같은 종류가 있다.

HWP	워디안 native format
HWP30	한글 3.X/96/97
HTML	인터넷 문서
TEXT	아스키 텍스트 문서
UNICODE	유니코드 텍스트 문서
HWP20	한글 2.0
HWP21	한글 2.1/2.5
HWP15	한글 1.X
HWPML1X	HWPML 1.X 문서 (Open만 가능)
HWPML2X	HWPML 2.X 문서 (Open / SaveAs 가능)
RTF	서식있는 텍스트 문서
DBF	DBASE II/III 문서
HUNMIN	훈민정음 3.0/2000
MSWORD	마이크로소프트 워드 문서
HANA	하나워드 문서
ARIRANG	아리랑 문서
ICHITARO	一太郎 문서 (일본 워드프로세서)
WPS	WPS 문서
DOCIMG	인터넷 프레젠테이션 문서(SaveAs만 가능)
SWF	Macromedia Flash 문서(SaveAs만 가능)

arg에 지정할 수 있는 옵션의 의미는 필터가 정의하기에 따라 다르지만, 선택지는 다음과 같이 공통된 형식을 사용한다.

key:value;key:value;...

* key는 A-Z, a-z, 0-9, _로 구성된다.

* value는 타입에 따라 다음과 같은 3 종류가 있다.

boolean: ex) fullsave:true (== fullsave)
integer: ex) type:20
string: ex) prefix:_This_

* value는 생략 가능하며, 이때는 콜론도 생략한다.

* arg에 지정할 수 있는 옵션

"모든 파일"	setcurdir(boolean, FALSE) : 로드한 후 해당 파일이 존재하는 폴더로 현재 위치를 변경한다. hyperlink 정보가 상대적인 위치로 되어 있을 때 유용하다.
"HWP"	<ul style="list-style-type: none"> - lock (boolean, TRUE) : 로드한 후 해당 파일을 계속 오픈한 상태로 lock을 걸지 여부 - notext (boolean, FALSE) : 텍스트 내용을 읽지 않고 헤더 정보만 읽을지 여부. (스타일 로드 등에 사용) - template (boolean, FALSE): 새로운 문서를 생성하기 위해 템플릿 파일을 오픈한다. 이 옵션이 주어지면 lock은 무조건 FALSE로 처리된다. - suspendpassword (boolean, FALSE): TRUE로 지정하면, 암호가 있는 파일일 경우 암호를 묻지 않고 무조건 읽기기에 실패한 것으로 처리한다. - forceopen (boolean, FALSE): TRUE로 지정하면, 읽기 전용으로 읽어야 하는 경우 대화상자를 띄우지 않는다. - versionwarning (boolean, TRUE): FALSE로 지정하면 “상위 버전에서 작성한 문서입니다” 메세지가 안나옴.
"HTML"	<ul style="list-style-type: none"> - code(string, codepage) : 문서 변환시 사용되는 코드페이지를 지정할 수 있으며 code키가 존재할 경우 필터 사용시 사용자 디아일로그를 띄우지 않는다. - textunit(boolean, pixel) : Export될 Text의 크기의 단위 결정(pixel, point, mili 지정 가능.)

	<ul style="list-style-type: none"> - formatunit(boolean, pixel) : Export될 문서 포맷 관련 (마진, Object 크기 등) 단위 결정. pixel, point, mili 지정 가능 																								
"DOCIMG"	<ul style="list-style-type: none"> - asimg(boolean, FALSE) : 저장할 때 페이지를 image로 저장 - ashtml(boolean, FALSE) : 저장할 때 페이지를 html로 저장 <p>[codepage 종류]</p> <p>ks : 한글 KS 완성형 kssm : 한글 조합형 sjis : 일본 utf8 : UTF8 unicode : 유니코드 gb : 중국 간체 big5 : 중국 번체 acp : Active Codepage 현재 시스템의 코드 페이지</p>																								
"TEXT"	<ul style="list-style-type: none"> - code(string, codepage) : 문서 변환시 사용되는 코드 페이지를 지정할 수 있으며 code키가 존재할 경우 필터 사용시 사용자 다이얼로그를 띄우지 않는다. <p>[codepage 종류]</p> <p>- ks : 한글 KS 완성형 kssm : 한글 조합형 sjis : 일본 gb : 중국 간체 big5 : 중국 번체 acp : Active Codepage 현재 시스템의 코드 페이지</p>																								
Save(Method)	<p>Description 현재 편집중인 문서를 저장한다.</p> <p>Declaration BOOL Save(VARIANT save_if_dirty)</p> <p>Parameters save_if_dirty : True를 지정하면 문서가 변경된 경우에만 저장한다. False를 지정하면 변경 여부에 관계없이 무조건 저장한다. 생략하면 True가 지정된다.</p> <p>Remark 문서의 경로가 지정되어 있지 않으면 "새이름으로 저장" 대화상자가 떠서 사용자에게 경로를 묻는다.</p>																								
SaveAs(Method)	<p>Description 현재 편집중인 문서를 지정한 이름으로 저장한다.</p> <p>Declaration BOOL Save(BSTR path, VARIANT format, VARIANT arg)</p> <p>Parameters path : 문서 파일의 경로 format : 문서 형식. 별도 설명 참조. 생략하면 "HWP"가 지정된다. arg : 세부 옵션. 의미는 format에 지정한 파일 형식에 따라 다르다. 생략하면 빈 문자열이 지정된다.</p> <p>Remark format, arg의 일반적인 개념에 대해서는 Open 참조. "HWP" 형식으로 파일 저장시 arg에 지정할 수 있는 옵션은 다음과 같다.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #ffffcc;"> <th style="text-align: center;">함수</th> <th style="text-align: center;">인자 타입</th> <th style="text-align: center;">기본 값 (Default)</th> <th style="text-align: center;">설명</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">lock</td> <td style="text-align: center;">boolean</td> <td style="text-align: center;">TRUE</td> <td>저장한 후 해당 파일을 계속 오픈한 상태로 lock을 걸지 여부</td> </tr> <tr> <td style="text-align: center;">backup</td> <td style="text-align: center;">boolean</td> <td style="text-align: center;">FALSE</td> <td>백업 파일 생성 여부</td> </tr> <tr> <td style="text-align: center;">compress</td> <td style="text-align: center;">boolean</td> <td style="text-align: center;">TRUE</td> <td>압축 여부</td> </tr> <tr> <td style="text-align: center;">fullsave</td> <td style="text-align: center;">boolean</td> <td style="text-align: center;">FALSE</td> <td>스토리지 파일을 완전히 새로 생성하여 저장</td> </tr> <tr> <td style="text-align: center;">prvimage</td> <td style="text-align: center;">int</td> <td style="text-align: center;">2</td> <td>미리보기 이미지 (0=off, 1=BMP, 2=GIF)</td> </tr> </tbody> </table>	함수	인자 타입	기본 값 (Default)	설명	lock	boolean	TRUE	저장한 후 해당 파일을 계속 오픈한 상태로 lock을 걸지 여부	backup	boolean	FALSE	백업 파일 생성 여부	compress	boolean	TRUE	압축 여부	fullsave	boolean	FALSE	스토리지 파일을 완전히 새로 생성하여 저장	prvimage	int	2	미리보기 이미지 (0=off, 1=BMP, 2=GIF)
함수	인자 타입	기본 값 (Default)	설명																						
lock	boolean	TRUE	저장한 후 해당 파일을 계속 오픈한 상태로 lock을 걸지 여부																						
backup	boolean	FALSE	백업 파일 생성 여부																						
compress	boolean	TRUE	압축 여부																						
fullsave	boolean	FALSE	스토리지 파일을 완전히 새로 생성하여 저장																						
prvimage	int	2	미리보기 이미지 (0=off, 1=BMP, 2=GIF)																						

	prvtext	int	1	미리보기 텍스트(0=off, 1=on)
Insert(Method)		<p>Description 현재 캐럿 위치에 문서 파일을 삽입한다.</p> <p>Declaration <code>Insert(BSTR path, VARIANT format, VARIANT arg)</code></p> <p>Parameters</p> <p><code>path</code> : 문서 파일의 경로, URL 사용 가능 <code>format</code> : 문서 형식. 별도 설명 참조. 빈 문자열을 지정하면 자동으로 디택트한다. 생략하면 빈 문자열이 지정된다. <code>arg</code> : 세부 옵션. 의미는 <code>format</code>에 지정한 파일 형식에 따라 다르다. 생략하면 빈 문자열이 지정된다.</p> <p>Remark <code>format</code>, <code>arg</code>에 대해서는 Open 참조</p>		
SelectText(Method)		<p>Description 블록을 설정 한다.</p> <p>Declaration <code>BOOL SelectText(long spara, long spos, long epara, long epos)</code></p> <p>Parameters</p> <p><code>spara</code> : 블록 시작 위치의 문단 번호. <code>spos</code> : 블록 시작 위치의 문단 중에서 문자의 위치. <code>epara</code> : 블록 끝 위치의 문단 번호. <code>epos</code> : 블록 끝 위치의 문단 중에서 문자의 위치.</p> <p>Remark <code>epos</code>가 가리키는 문자는 포함되지 않는다.</p>		
CreateField(Method)		<p>Description 캐럿의 현재 위치에 누름틀을 생성한다.</p> <p>Declaration <code>BOOL CreateField(BSTR direction, VARIANT memo, VARIANT name)</code></p> <p>Parameters</p> <p><code>direction</code> : 누름틀에 입력이 안된 상태에서 보여지는 안내문/지시문. <code>memo</code> : 누름틀에 대한 설명/도움말 <code>name</code> : 누름틀 필드에 대한 필드 이름</p>		
MoveToField(Method)		<p>Description 지정한 필드로 캐럿을 이동한다.</p> <p>Declaration <code>BOOL MoveToField(BSTR field, VARIANT text, VARIANT start, VARIANT select)</code></p> <p>Parameters</p> <p><code>field</code> : 필드 이름. <code>GetFieldText/PutFieldText</code>과 같은 형식으로 이름 뒤에 '{{#}}'로 번호를 지정할 수 있다. <code>text</code> : 필드가 누름틀일 경우 누름틀 내부의 텍스트로 이동할지(<code>True</code>) 누름틀 코드로 이동할지(<code>False</code>)를 지정한다. 누름틀이 아닌 필드일 경우 무시된다. 생략하면 <code>True</code>가 지정된다. <code>start</code> : 필드의 처음(<code>True</code>)으로 이동할지 끝(<code>False</code>)으로 이동할지 지정 한다. <code>select</code>를 <code>True</code>로 지정하면 무시된다. 생략하면 <code>True</code>가 지정된다. <code>select</code> : 필드 내용을 블록으로 선택할지(<code>True</code>), 캐럿만 이동할지(<code>False</code>) 지정한다. 생략하면 <code>False</code>가 지정된다.</p> <p>Remark 누름틀은 한글97 메뉴 중 입력 메뉴에 문서마당 정보를 선택하면 누름틀을 만드실 수 있습니다.</p>		
FieldExist(Method)		<p>Description 문서 중에 지정한 데이터 필드가 존재하는지 검사한다.</p> <p>Declaration</p>		

	<p>BOOL FieldExist(BSTR field)</p> <p>Parameters</p> <p>field : 필드 이름</p>
GetFieldText(Method)	<p>Description 지정한 필드에서 문자열을 구한다.</p> <p>Declaration BSTR GetFieldText(BSTR fieldlist)</p> <p>Parameters</p> <p>fieldlist : 텍스트를 구할 필드 이름의 리스트. 다음과 같이 필드 사 이를 문자 코드 0x02로 구분하여 한 번에 여러 개의 필드를 지정할 수 있다. “필드이름#1Wx2필드이름#2Wx2...필드이름#n”</p> <p>지정한 필드 이름이 문서 중에 두 개 이상 존재할 때의 표현 방식은 다 음과 같다.</p> <p style="padding-left: 2em;">필드이름 이름의 필드 중 첫 번째</p> <p style="padding-left: 2em;">필드이름{{n}} 지정한 이름의 필드 중 n 번째</p> <p>예를 들어 “제목{{1}}Wx2본문Wx2이름{{0}}” 과 같이 지정하면 ‘제목’ 이라는 이름의 필드 중 두 번째, ‘본문’이라는 이름의 필드 중 첫 번 째, ‘이름’이라는 이름의 필드 중 첫 번째를 각각 지정한다. 즉, ‘필드이름’과 ‘필드이름0’은 동일한 의미로 해석된다.</p> <p>return : 텍스트 데이터가 돌아온다. 텍스트에서 템은 'Wt'(0x9), 문단 바꿈은 CR/LF(0x0D/0x0A)로 표현되며, 이외의 특수 코드는 포함되지 않 는다. 필드 텍스트의 끝은 0x02로 표현되며, 그 이후 다음 필드의 텍 스트가 연속해서 지정한 필드 리스트의 개수만큼 위치한다. 지정한 이름 의 필드가 없거나 사용자가 해당 필드에 아무 텍스트도 입력하지 않았 으면 해당 텍스트에는 빈 문자열이 돌아온다.</p>
PutFieldText(Method)	<p>Description 지정한 필드의 내용을 채운다.</p> <p>Declaration void PutFieldText(BSTR fieldlist, BSTR textlist)</p> <p>Parameters</p> <p>fieldlist : 내용을 채울 필드 이름의 리스트. 한 번에 여러 개의 필드 를 지정할 수 있으며, 형식은 GetFieldText와 동일하다. 다만 필드 이 름 뒤에 '#'로 번호를 지정하지 않으면 해당 이름을 가진 모든 필드 에, 동일한 텍스트를 채워 넣는다. 즉, PutFieldText에서는 ‘필드이 름’과 ‘필드이름0’의 의미가 다르다.</p> <p>textlist : 필드에 채워 넣을 문자열의 리스트. 형식은 필드 리스트와 동일하게 필드의 개수만큼 텍스트를 0x02로 구분하여 지정한다.</p> <p>Remark 현재 필드에 입력되어 있는 내용은 지워진다. 채워진 내용의 글자모양 은 필드에 지정해 넣은 글자모양을 따라간다.</p> <p>fieldlist의 필드 개수와, textlist의 텍스트 개수는 동일해야 한다. 존재하지 않는 필드에 대해서는 무시한다.</p>
RenameField(Method)	<p>Description 지정한 필드의 이름을 바꾼다.</p> <p>Declaration void RenameField(BSTR oldname, BSTR newname)</p> <p>Parameters</p> <p>oldname : 이름을 바꿀 필드 이름의 리스트. 형식은 PutFieldText과 동 일하다.</p> <p>newname : 새로운 필드 이름의 리스트. oldname과 동일한 개수의 필드 이름을 0x02로 구분하여 지정한다.</p> <p>Remark 예를 들어 oldname에 “title{{0}}Wx2title{{1}}”, newname에 “tt1Wx2tt2” 로 지정하면 첫 번째 title은 tt1로, 두 번째 title은 tt2로 변경된다. oldname의 필드 개수와, newname의 필드 개수는 동일해야 한다. 존재하지 않는 필드에 대해서는 무시한다.</p>
GetCurFieldName(Method)	Description

	<p>현재 캐럿 위치의 데이터 필드 이름을 구한다.</p> <p>Declaration BSTR GetCurFieldName([HwpFieldOption option])</p> <p>Parameters option : 다음과 같은 옵션을 지정할 수 있다. 0을 지정하면 모두 off 이다. 생략하면 0이 지정된다. (HwpFieldOption : short type)</p> <table border="1"> <thead> <tr> <th>ID</th><th>값</th><th>설명</th></tr> </thead> <tbody> <tr> <td>hwpFieldCell</td><td>1</td><td>셀에 부여된 필드 리스트만을 구한다. hwpFieldClickHere와는 함께 지정할 수 없다.</td></tr> <tr> <td>hwpFieldClickHere</td><td>2</td><td>누름틀에 부여된 필드 리스트만을 구한다. hwpFieldCell과는 함께 지정할 수 없다.</td></tr> </tbody> </table> <p>return : 필드 이름이 돌아온다. 필드 이름이 없는 경우 빈 문자열이 돌아온다.</p> <p>Remark GetFieldList()의 option 중에 hwpFieldSelection (= 4) 옵션은 사용하지 않는다.</p>	ID	값	설명	hwpFieldCell	1	셀에 부여된 필드 리스트만을 구한다. hwpFieldClickHere와는 함께 지정할 수 없다.	hwpFieldClickHere	2	누름틀에 부여된 필드 리스트만을 구한다. hwpFieldCell과는 함께 지정할 수 없다.			
ID	값	설명											
hwpFieldCell	1	셀에 부여된 필드 리스트만을 구한다. hwpFieldClickHere와는 함께 지정할 수 없다.											
hwpFieldClickHere	2	누름틀에 부여된 필드 리스트만을 구한다. hwpFieldCell과는 함께 지정할 수 없다.											
SetCurFieldName(Method)	<p>Description 현재 캐럿 위치의 데이터 필드 이름을 설정한다.</p> <p>Declaration BOOL SetCurFieldName(BSTR fieldname, [HwpFieldOption option], [BSTR direction], [BSTR memo])</p> <p>Parameters fieldname : 데이터 필드 이름. option : 다음과 같은 옵션을 지정할 수 있다. 0을 지정하면 모두 off 이다. 생략하면 0이 지정된다. (HwpFieldOption : short type) direction : 누름틀 필드의 안내문. 누름클 필드일 때만 유효하다. memo : 누름틀 필드의 메모. 누름클 필드일 때만 유효하다.</p> <p>Remark GetFieldList()의 option 중에 hwpFieldSelection (= 4) 옵션은 사용하지 않는다.</p>												
ModifyFieldProperties(Method)	<p>Description 지정한 필드의 속성을 바꾼다.</p> <p>Declaration long ModifyFieldProperties(LPCTSTR field, long remove, long add)</p> <p>Parameters field : 속성을 바꿀 필드 이름의 리스트. 형식은 PutFieldText과 동일. remove : 제거될 속성 add : 추가될 속성 return : 음수가 리턴되면 에러임을 나타낸다.</p> <p>Remark 속성의 값은 아래와 같다.</p> <table border="1"> <thead> <tr> <th>long value</th><th>설명</th></tr> </thead> <tbody> <tr> <td>0x00000001</td><td>양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)</td></tr> </tbody> </table> <p>return 값의 bit field는 다음과 같다.</p> <table border="1"> <thead> <tr> <th>long value bit mask</th><th>설명</th></tr> </thead> <tbody> <tr> <td>0x00000001</td><td>양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)</td></tr> <tr> <td>0x80000000</td><td>에러</td></tr> <tr> <td>0x40000000</td><td>필드를 찾을 수 없음</td></tr> </tbody> </table>	long value	설명	0x00000001	양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)	long value bit mask	설명	0x00000001	양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)	0x80000000	에러	0x40000000	필드를 찾을 수 없음
long value	설명												
0x00000001	양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)												
long value bit mask	설명												
0x00000001	양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)												
0x80000000	에러												
0x40000000	필드를 찾을 수 없음												

	<p>remove와 add에 둘다 0이 입력되면 현재 속성을 돌려준다.</p> <p>리턴값이 음수인지 확인하여 쉽게 에러임을 판별할 수 있으며 자세한 에러내용은 bit mask로 and 연산하여 알아 낼 수 있다.</p> <p>리턴값은 여러 가지 추가 정보가 같이 올 수 있으므로 반드시 bit mask를 사용하여 비교해야 한다.</p>																											
SetFieldViewOption(Method)	<p>Description 양식 모드와 읽기 전용모드일때 현재 열린 문서의 필드의 결보기 속성(『』 표시)을 바꾼다.</p> <p>Declaration <code>long SetFieldViewOption(long option)</code></p> <p>Parameters option : 결보기 속성 bit</p> <table border="1"> <thead> <tr> <th>option</th><th>느름틀</th><th>개인정보/문서오약/날짜시간/파일경로</th><th>비고</th></tr> </thead> <tbody> <tr> <td>1</td><td>『』을 표시하지 않음</td><td>『』을 표시하지 않음</td><td></td></tr> <tr> <td>2</td><td>『』을 빨간색으로 표시</td><td>『』을 흰색으로 표시</td><td>설정하지 않았을 때 기본 값</td></tr> <tr> <td>3</td><td>『』을 흰색으로 표시</td><td>『』을 흰색으로 표시</td><td></td></tr> </tbody> </table> <p>return : 설정된 속성이 리턴 된다. 예러일 경우 0 이 리턴 된다.</p> <p>Remark EditMode와 비슷하게 현재 열려있는 문서에 대한 속성이다. 따라서 저장되지 않는다.</p>	option	느름틀	개인정보/문서오약/날짜시간/파일경로	비고	1	『』을 표시하지 않음	『』을 표시하지 않음		2	『』을 빨간색으로 표시	『』을 흰색으로 표시	설정하지 않았을 때 기본 값	3	『』을 흰색으로 표시	『』을 흰색으로 표시												
option	느름틀	개인정보/문서오약/날짜시간/파일경로	비고																									
1	『』을 표시하지 않음	『』을 표시하지 않음																										
2	『』을 빨간색으로 표시	『』을 흰색으로 표시	설정하지 않았을 때 기본 값																									
3	『』을 흰색으로 표시	『』을 흰색으로 표시																										
GetFieldList(Method)	<p>Description 문서 중의 필드 리스트를 구한다.</p> <p>Declaration <code>BSTR GetFieldList([HwpFieldNumber number], [HwpFieldOption option])</code></p> <p>Parameters number : 문서 중에서 동일한 이름의 필드가 여러 개 존재할 때 이를 구별하기 위한 식별 방법을 지정한다. 생략하면 hwpFieldPlain이 지정된다. (HwpFieldNumber : short type)</p> <table border="1"> <thead> <tr> <th>ID</th><th>값</th><th>설명</th></tr> </thead> <tbody> <tr> <td>hwpFieldPlain</td><td>0</td><td>아무 기호 없이 순서대로 필드 이름이 나열된다.</td></tr> <tr> <td>hwpFieldNumber</td><td>1</td><td>필드 이름 뒤에 일련번호가 {{#}}와 같은 형식으로 붙는다.</td></tr> <tr> <td>hwpFieldCount</td><td>2</td><td>필드 이름뒤에 그 이름의 필드가 몇 개 있는지 {{#}}와 같은 형식으로 붙는다.</td></tr> </tbody> </table> <p>option : 다음과 같은 옵션을 조합할 수 있다. 0을 지정하면 모두 off이다. 생략하면 0이 지정된다. (HwpFieldOption : unsigned short type)</p> <table border="1"> <thead> <tr> <th>ID</th><th>값</th><th>설명</th></tr> </thead> <tbody> <tr> <td>hwpFieldCell</td><td>1</td><td>셀에 부여된 필드 리스트만을 구한다. hwpFieldClickHere와는 함께 지정할 수 없다.</td></tr> <tr> <td>hwpFieldClickHere</td><td>2</td><td>느름틀에 부여된 필드 리스트만을 구한다. hwpFieldCell과는 함께 지정할 수 없다.</td></tr> <tr> <td>hwpFieldSelection</td><td>4</td><td>셀렉션 내에 존재하는 필드 리스트를 구한다.</td></tr> <tr> <td>hwpFieldTextbox</td><td>8</td><td>글상자에 부여된 필드 리스트만을 구한다. hwpFieldCell, hwpFieldClickHere와는 함께 지정할 수 없다.</td></tr> </tbody> </table> <p>return : 각 필드 사이를 문자코드 0x02로 구분하여 다음과 같은 형식으로 리턴 한다. (가장 마지막 필드에는 0x02가 붙지 않는다.) "필드이름#1Wx2필드이름#2Wx...필드이름#n"</p> <p>Remark 문서 중에 동일한 이름의 필드가 여러 개 존재할 때는 number에 지정한</p>	ID	값	설명	hwpFieldPlain	0	아무 기호 없이 순서대로 필드 이름이 나열된다.	hwpFieldNumber	1	필드 이름 뒤에 일련번호가 {{#}}와 같은 형식으로 붙는다.	hwpFieldCount	2	필드 이름뒤에 그 이름의 필드가 몇 개 있는지 {{#}}와 같은 형식으로 붙는다.	ID	값	설명	hwpFieldCell	1	셀에 부여된 필드 리스트만을 구한다. hwpFieldClickHere와는 함께 지정할 수 없다.	hwpFieldClickHere	2	느름틀에 부여된 필드 리스트만을 구한다. hwpFieldCell과는 함께 지정할 수 없다.	hwpFieldSelection	4	셀렉션 내에 존재하는 필드 리스트를 구한다.	hwpFieldTextbox	8	글상자에 부여된 필드 리스트만을 구한다. hwpFieldCell, hwpFieldClickHere와는 함께 지정할 수 없다.
ID	값	설명																										
hwpFieldPlain	0	아무 기호 없이 순서대로 필드 이름이 나열된다.																										
hwpFieldNumber	1	필드 이름 뒤에 일련번호가 {{#}}와 같은 형식으로 붙는다.																										
hwpFieldCount	2	필드 이름뒤에 그 이름의 필드가 몇 개 있는지 {{#}}와 같은 형식으로 붙는다.																										
ID	값	설명																										
hwpFieldCell	1	셀에 부여된 필드 리스트만을 구한다. hwpFieldClickHere와는 함께 지정할 수 없다.																										
hwpFieldClickHere	2	느름틀에 부여된 필드 리스트만을 구한다. hwpFieldCell과는 함께 지정할 수 없다.																										
hwpFieldSelection	4	셀렉션 내에 존재하는 필드 리스트를 구한다.																										
hwpFieldTextbox	8	글상자에 부여된 필드 리스트만을 구한다. hwpFieldCell, hwpFieldClickHere와는 함께 지정할 수 없다.																										

	<p>타입에 따라 3 가지의 서로 다른 방식을 종에서 선택할 수 있다. 예를 들어 문서 중 title, body, title, body, footer 순으로 5 개의 필드가 존재할 때, hwpFieldPlain, hwpFieldNumber, HwpFieldCount 세 가지 형식에 따라 다음과 같은 내용이 돌아온다.</p> <pre> hwpFieldPlain "title\0x2body\0x2title\0x2body\0x2footer" hwpFieldNumber "title{0}\0x2body{0}\0x2title{1}\0x2body{1}\0x2footer{0}" hwpFieldCount "title{2}\0x2body{2}\0x2footer{1}" </pre>																																																						
MovePos(Method)	<p>Description 캐럿의 위치를 옮긴다.</p> <p>Declaration BOOL MovePos([unsigned int moveID], [unsigned int para], [unsigned int pos])</p> <p>Parameters moveID : 다음과 같은 값을 지정할 수 있다. 생략하면 moveCurList가 지정된다.</p> <table border="1"> <thead> <tr> <th>ID</th><th>값</th><th>설명</th></tr> </thead> <tbody> <tr> <td>moveMain</td><td>0</td><td>루트 리스트의 특정 위치.(para pos로 위치 지정)</td></tr> <tr> <td>moveCurList</td><td>1</td><td>현재 리스트의 특정 위치.(para pos로 위치 지정)</td></tr> <tr> <td>moveTopOfFile</td><td>2</td><td>문서의 시작으로 이동.</td></tr> <tr> <td>moveBottomOfFile</td><td>3</td><td>문서의 끝으로 이동.</td></tr> <tr> <td>moveTopOfList</td><td>4</td><td>현재 리스트의 시작으로 이동</td></tr> <tr> <td>moveBottomOfList</td><td>5</td><td>현재 리스트의 끝으로 이동</td></tr> <tr> <td>moveStartOfPara</td><td>6</td><td>현재 위치한 문단의 시작으로 이동</td></tr> <tr> <td>moveEndOfPara</td><td>7</td><td>현재 위치한 문단의 끝으로 이동</td></tr> <tr> <td>moveStartOfWord</td><td>8</td><td>현재 위치한 단어의 시작으로 이동. (현재 리스트만을 대상으로 동작한다.)</td></tr> <tr> <td>moveEndOfWord</td><td>9</td><td>현재 위치한 단어의 끝으로 이동. (현재 리스트만을 대상으로 동작한다.)</td></tr> <tr> <td>moveNextPara</td><td>10</td><td>다음 문단의 시작으로 이동. (현재 리스트만을 대상으로 동작한다.)</td></tr> <tr> <td>movePrevPara</td><td>11</td><td>앞 문단의 끝으로 이동. (현재 리스트만을 대상으로 동작한다.)</td></tr> <tr> <td>moveNextPos</td><td>12</td><td>한 글자 앞으로 이동. (서브 리스트를 옮겨다닐 수 있다.)</td></tr> <tr> <td>movePrevPos</td><td>13</td><td>한 글자 뒤로 이동. (서브 리스트를 옮겨다닐 수 있다.)</td></tr> <tr> <td>moveNextPosEx</td><td>14</td><td>한 글자 앞으로 이동. (서브 리스트를 옮겨다닐 수 있다.) (머리말/꼬리말, 각주/미주, 글상자 포함.)</td></tr> <tr> <td>movePrevPosEx</td><td>15</td><td>한 글자 뒤로 이동. (서브 리스트를 옮겨다닐 수 있다.) (머리말/꼬리말, 각주/미주, 글상자 포함.)</td></tr> <tr> <td>moveNextChar</td><td>16</td><td>한 글자 앞으로 이동. (현재 리스트만을 대상으로 동작한다.)</td></tr> </tbody> </table>	ID	값	설명	moveMain	0	루트 리스트의 특정 위치.(para pos로 위치 지정)	moveCurList	1	현재 리스트의 특정 위치.(para pos로 위치 지정)	moveTopOfFile	2	문서의 시작으로 이동.	moveBottomOfFile	3	문서의 끝으로 이동.	moveTopOfList	4	현재 리스트의 시작으로 이동	moveBottomOfList	5	현재 리스트의 끝으로 이동	moveStartOfPara	6	현재 위치한 문단의 시작으로 이동	moveEndOfPara	7	현재 위치한 문단의 끝으로 이동	moveStartOfWord	8	현재 위치한 단어의 시작으로 이동. (현재 리스트만을 대상으로 동작한다.)	moveEndOfWord	9	현재 위치한 단어의 끝으로 이동. (현재 리스트만을 대상으로 동작한다.)	moveNextPara	10	다음 문단의 시작으로 이동. (현재 리스트만을 대상으로 동작한다.)	movePrevPara	11	앞 문단의 끝으로 이동. (현재 리스트만을 대상으로 동작한다.)	moveNextPos	12	한 글자 앞으로 이동. (서브 리스트를 옮겨다닐 수 있다.)	movePrevPos	13	한 글자 뒤로 이동. (서브 리스트를 옮겨다닐 수 있다.)	moveNextPosEx	14	한 글자 앞으로 이동. (서브 리스트를 옮겨다닐 수 있다.) (머리말/꼬리말, 각주/미주, 글상자 포함.)	movePrevPosEx	15	한 글자 뒤로 이동. (서브 리스트를 옮겨다닐 수 있다.) (머리말/꼬리말, 각주/미주, 글상자 포함.)	moveNextChar	16	한 글자 앞으로 이동. (현재 리스트만을 대상으로 동작한다.)
ID	값	설명																																																					
moveMain	0	루트 리스트의 특정 위치.(para pos로 위치 지정)																																																					
moveCurList	1	현재 리스트의 특정 위치.(para pos로 위치 지정)																																																					
moveTopOfFile	2	문서의 시작으로 이동.																																																					
moveBottomOfFile	3	문서의 끝으로 이동.																																																					
moveTopOfList	4	현재 리스트의 시작으로 이동																																																					
moveBottomOfList	5	현재 리스트의 끝으로 이동																																																					
moveStartOfPara	6	현재 위치한 문단의 시작으로 이동																																																					
moveEndOfPara	7	현재 위치한 문단의 끝으로 이동																																																					
moveStartOfWord	8	현재 위치한 단어의 시작으로 이동. (현재 리스트만을 대상으로 동작한다.)																																																					
moveEndOfWord	9	현재 위치한 단어의 끝으로 이동. (현재 리스트만을 대상으로 동작한다.)																																																					
moveNextPara	10	다음 문단의 시작으로 이동. (현재 리스트만을 대상으로 동작한다.)																																																					
movePrevPara	11	앞 문단의 끝으로 이동. (현재 리스트만을 대상으로 동작한다.)																																																					
moveNextPos	12	한 글자 앞으로 이동. (서브 리스트를 옮겨다닐 수 있다.)																																																					
movePrevPos	13	한 글자 뒤로 이동. (서브 리스트를 옮겨다닐 수 있다.)																																																					
moveNextPosEx	14	한 글자 앞으로 이동. (서브 리스트를 옮겨다닐 수 있다.) (머리말/꼬리말, 각주/미주, 글상자 포함.)																																																					
movePrevPosEx	15	한 글자 뒤로 이동. (서브 리스트를 옮겨다닐 수 있다.) (머리말/꼬리말, 각주/미주, 글상자 포함.)																																																					
moveNextChar	16	한 글자 앞으로 이동. (현재 리스트만을 대상으로 동작한다.)																																																					

	ID	값	설명
	movePrevChar	17	한 글자 뒤로 이동. (현재 리스트만을 대상으로 동작한다.)
	moveNextWord	18	한 단어 앞으로 이동. (현재 리스트만을 대상으로 동작한다.)
	movePrevWord	19	한 단어 뒤로 이동. (현재 리스트만을 대상으로 동작한다.)
	moveNextLine	20	한 줄 위로 이동.
	movePrevLine	21	한 줄 아래로 이동.
	moveStartOfLine	22	현재 위치한 줄의 시작으로 이동.
	moveEndOfLine	23	현재 위치한 줄의 끝으로 이동.
	moveParentList	24	한 레벨 상위로 이동한다.
	moveTopLevelList	25	탑레벨 리스트로 이동한다.
	moveRootList	26	루트 리스트로 이동한다. 추가 설명 : 현재 루트 리스트에 위치해 있어 더 이상 상위 리스트가 없을 때는 위치 이동 없이 리턴한다. 이동한 후의 위치는 상위 리스트에서 서브리스트가 속한 컨트롤 코드가 위치한 곳이다. 위치 이동시 셀렉션은 무조건 풀린다.
	moveLeftOfCell	100	현재 캐럿이 위치한 셀의 왼쪽
	moveRightOfCell	101	현재 캐럿이 위치한 셀의 오른쪽
	moveUpOfCell	102	현재 캐럿이 위치한 셀의 위쪽
	moveDownOfCell	103	현재 캐럿이 위치한 셀의 아래쪽
	moveStartOfCell	104	현재 캐럿이 위치한 셀에서 행(row)의 시작
	moveEndOfCell	105	현재 캐럿이 위치한 셀에서 행(row)의 끝
	moveTopOfCell	106	현재 캐럿이 위치한 셀에서 열(column)의 시작
	moveBottomOfCell	107	현재 캐럿이 위치한 셀에서 열(column)의 끝
	moveScrPos	200	한/글 문서장에서의 screen 좌표로서 위치를 설정 한다.
	moveScanPos	201	GetText() 실행 후 위치로 이동한다.
	para : 이동할 문단의 번호. moveMain 또는 moveCurList가 지정되었을 때만 사용된다. moveScrPos가 지정되었을 때는 문단번호가 아닌 스크린 좌표로 해석된다. (스크린 좌표 : LOWORD = x 좌표, HIWORD = y 좌표)		
	pos : 이동할 문단 중에서 문자의 위치. moveMain 또는 moveCurList가 지정되었을 때만 사용된다.		
	Remark		
	moveScrPos가 지정되었을 때는 스크린 좌표는 마우스 커서의 (x,y) 좌표를 그대로 넘겨 주면 된다.		
	moveScanPos는 문서를 검색하는 중에 캐럿을 이동 시키려 할 경우에만 사용이 가능하다.		
InitScan(Method)	Description		

문서 검색을 위한 준비 작업을 한다.

Declaration

```
BOOL InitScan([unsigned int option], [unsigned int rang], [unsigned
int spara], [unsigned int spos], [unsigned int epara], [unsigned
int epos])
```

Parameters

option : 찾을 대상을 다음과 같은 옵션을 조합하여 지정할 수 있다. 생략하면 모든 컨트롤을 찾을 대상으로 한다.

ID	값	설명
maskNormal	0x00	본문을 대상으로 검색한다. (서브리스트를 검색하지 않는다.)
maskChar	0x01	char 타입 컨트롤 마스크를 대상으로 한다. (강제줄나눔, 문단끝, 하이픈, 띄움빈칸, 고정폭빈칸, 등...)
maskInline	0x02	inline 타입 컨트롤 마스크를 대상으로 한다. (누름틀 필드 끝, 등...)
maskCtrl	0x04	extende 타입 컨트롤 마스크를 대상으로 한다. (바탕쪽, 프리젠테이션, 다단, 누름틀 필드 시작, Shape Object, 머리말, 꼬리말, 각주, 미주, 번호관련 컨트롤, 새번호 관련 컨트롤, 감추기, 찾아보기, 글자겹침, 등...)

range : 검색의 범위를 다음과 같은 옵션을 조합하여 지정할 수 있다.
생략하면 “문서 시작부터 - 문서의 끝까지” 검색 범위가 지정된다.

ID	값	설명
scanSposCurrent	0x0000	캐럿 위치부터. (시작 위치)
scanSposSpecified	0x0010	특정 위치부터. (시작 위치)
scanSposLine	0x0020	줄의 시작부터. (시작 위치)
scanSposParagraph	0x0030	문단의 시작부터. (시작 위치)
scanSposSection	0x0040	구역의 시작부터. (시작 위치)
scanSposList	0x0050	리스트의 시작부터. (시작 위치)
scanSposControl	0x0060	컨트롤의 시작부터. (시작 위치)
scanSposDocument	0x0070	문서의 시작부터. (시작 위치)
scanEposCurrent	0x0000	캐럿 위치까지. (끝 위치)
scanEposSpecified	0x0001	특정 위치까지. (끝 위치)
scanEposLine	0x0002	줄의 끝까지. (끝 위치)
scanEposParagraph	0x0003	문단의 끝까지. (끝 위치)
scanEposSection	0x0004	구역의 끝까지. (끝 위치)
scanEposList	0x0005	리스트의 끝까지. (끝 위치)
scanEposControl	0x0006	컨트롤의 끝까지. (끝 위치)
scanEposDocument	0x0007	문서의 끝까지. (끝 위치)
scanWithinSelection	0x00ff	검색의 범위를 블록으로 제한.
scanForward	0x0000	정방향. (검색 방향)
scanBackward	0x0100	역방향. (검색 방향)

	<p>spara : 검색 시작 위치의 문단 번호. scanSposSpecified 옵션이 지정되었을 때만 유효하다.</p> <p>spos : 검색 시작 위치의 문단 중에서 문자의 위치. scanSposSpecified 옵션이 지정되었을 때만 유효하다.</p> <p>epara : 검색 끝 위치의 문단 번호. scanEposSpecified 옵션이 지정되었을 때만 유효하다.</p> <p>epos : 검색 끝 위치의 문단 중에서 문자의 위치. scanEposSpecified 옵션이 지정되었을 때만 유효하다.</p> <p>Remark 문서의 검색 과정은 InitScan()으로 검색위한 준비 작업을 하고 GetText()를 호출하여 본문의 텍스트를 얻어온다. GetText()를 반복 호출하면 연속하여 본문의 텍스트를 얻어올 수 있다. 검색이 끝나면 ReleaseScan()을 호출하여 관련 정보를 초기화 해 주면 된다.</p>
ReleaseScan(Method)	<p>Description InitScan()으로 설정된 정보를 초기화 한다.</p> <p>Declaration void ReleaseScan()</p>
GetText(Method)	<p>Description 문서 중에서 텍스트를 얻는다.</p> <p>Declaration long GetText(BSTR FAR* text)</p> <p>Parameters text : 텍스트 데이터가 돌아온다. 텍스트에서 탭은 'Wt'(0x9), 문단 바꿈은 CR/LF(0x0D/0x0A)로 표현되며, 이외 특수 코드는 포함되지 않는다.</p> <p>return : 다음과 같은 결과값을 리턴한다.</p> <ul style="list-style-type: none"> 0 = 텍스트 정보 없음. 1 = 리스트의 끝. 2 = 일반 텍스트. 3 = 다음 문단. 4 = 제어문자 내부로 들어감. 5 = 제어 문자를 빠져 나옴. 101 = 초기화 안됨. (InitScan() 실패 또는 InitScan()을 실행하지 않음) 102 = 텍스트 변환 실패. <p>Remark GetText()의 사용이 끝나면 ReleaseScan()을 반드시 호출하여 관련 정보를 초기화 해주어야 한다. 텍스트가 있는 문단으로 캐럿을 이동 시키려면 moveScanPos를 주고 MovePos()를 호출하면 된다.</p>
GetPos(Method)	<p>Description 현재 캐럿의 위치 정보를 얻어온다.</p> <p>Declaration void GetPos(long FAR* list, long FAR* para, long FAR* pos)</p> <p>Parameters list : 캐럿이 위치한 문서 내 리스트 아이디. para : 캐럿이 위치한 문단 아이디. pos : 캐럿이 위치한 문단 내 글자 단위 위치.</p> <p>Remark 위의 리스트란, 문단과 컨트롤들이 연결된 한/글 문서 내 구조를 뜻한다. 리스트 아이디는 문서내 위치 정보 중 하나로서 SelectText에 넘겨줄 때 사용한다.</p>
SetPos(Method)	<p>Description 캐럿을 문서 내 특정 위치로 위치시킨다.</p> <p>Declaration</p>

	<p><code>void SetPos(long list, long para, long pos)</code></p> <p>Parameters</p> <p>list : 캐럿이 위치할 문서 내 리스트 아이디. para : 캐럿이 위치할 문단 아이디. pos : 캐럿이 위치할 문단 내 글자 단위 위치.</p>																								
KeyIndicator(Method)	<p>Description 상태바에 나타날 정보를 알아낸다.</p> <p>Declaration <code>BOOL KeyIndicator(long FAR* seccnt, long FAR* secno, long FAR* prnpageno, long FAR* colno, long FAR* line, long FAR* pos, short FAR* over, BSTR FAR* ctrlname)</code></p> <p>Parameters</p> <p>seccnt : 총 구역 secno : 현재 구역 prnpageno : 쪽 colno : 단 line : 줄 pos : 칸 over : (true:수정, false:삽입)</p> <p>Remark 컨트롤 바깥쪽에서 상태바를 만들어서 상태바에 표시할 정보들의 내용을 알아낼 때 유용하다. 주의: 이 함수는 빠른 속도가 요구되므로 parameter로 포인터를 받는다. 따라서 포인터를 사용할 수 없는 언어에서는 사용이 불가능하다.</p>																								
GetTextFile(Method)	<p>Description 현재 열린 문서를 문자열로 넘겨준다.</p> <p>Declaration <code>VARIANT GetTextFile(BSTR format, BSTR option)</code></p> <p>Parameters</p> <p>format : 파일의 형식</p> <table border="1"> <thead> <tr> <th>format</th> <th>설명</th> <th>비고</th> </tr> </thead> <tbody> <tr> <td>HWP</td> <td>HWP native format</td> <td>BASE64로 인코딩되어 있다. 저장된 내용을 다른 곳에서 보여줄 필요가 없다면 이 포맷을 사용하기를 권장합니다.</td> </tr> <tr> <td>HWPML2X</td> <td>HWP 형식과 호환</td> <td>문서의 모든 정보를 유지</td> </tr> <tr> <td>HTML</td> <td>인터넷 문서 HTML 형식</td> <td>한글 고유의 서식은 순실된다.</td> </tr> <tr> <td>UNICODE</td> <td>유니코드 텍스트</td> <td>서식정보가 없는 텍스트만 저장</td> </tr> <tr> <td>TEXT</td> <td>일반 텍스트</td> <td>유니코드에만 있는 정보(한자, 고어, 특수문자 등)는 모두 손실된다.</td> </tr> </tbody> </table> <p>option</p> <table border="1"> <thead> <tr> <th>option</th> <th>설명</th> <th>비고</th> </tr> </thead> <tbody> <tr> <td>saveblock</td> <td>선택된 블록만 저장</td> <td>개체 선택 상태에서는 동작하지 않는다.</td> </tr> </tbody> </table> <p>return : 저장된 텍스트로 파일을 문자열로 바꿔서 리턴한다.</p> <p>Remark 이 함수는 JScript나 VBScript와 같이 직접적으로 local disk를 접근하기 힘든 언어를 위해 만들어졌으므로 disk를 접근할 수 있는 언어에서는 사용하지 않기를 권장합니다. disk를 접근할 수 있다면, Save나 SaveBlockAction을 사용하십시오. 이 함수 역시 내부적으로는 save나 SaveBlockAction을 호출하도록 되어있고 텍스트로 저장된 파일이 메모리에서 3~4번 복사되기 때문에 느리고, 메모리를 낭비합니다.</p>	format	설명	비고	HWP	HWP native format	BASE64로 인코딩되어 있다. 저장된 내용을 다른 곳에서 보여줄 필요가 없다면 이 포맷을 사용하기를 권장합니다.	HWPML2X	HWP 형식과 호환	문서의 모든 정보를 유지	HTML	인터넷 문서 HTML 형식	한글 고유의 서식은 순실된다.	UNICODE	유니코드 텍스트	서식정보가 없는 텍스트만 저장	TEXT	일반 텍스트	유니코드에만 있는 정보(한자, 고어, 특수문자 등)는 모두 손실된다.	option	설명	비고	saveblock	선택된 블록만 저장	개체 선택 상태에서는 동작하지 않는다.
format	설명	비고																							
HWP	HWP native format	BASE64로 인코딩되어 있다. 저장된 내용을 다른 곳에서 보여줄 필요가 없다면 이 포맷을 사용하기를 권장합니다.																							
HWPML2X	HWP 형식과 호환	문서의 모든 정보를 유지																							
HTML	인터넷 문서 HTML 형식	한글 고유의 서식은 순실된다.																							
UNICODE	유니코드 텍스트	서식정보가 없는 텍스트만 저장																							
TEXT	일반 텍스트	유니코드에만 있는 정보(한자, 고어, 특수문자 등)는 모두 손실된다.																							
option	설명	비고																							
saveblock	선택된 블록만 저장	개체 선택 상태에서는 동작하지 않는다.																							
SetTextFile(Method)	Description																								

	<p>문서를 문자열로 지정한다.</p> <p>Declaration long SetTextFile(VARIANT data, BSTR format, BSTR option)</p> <p>Parameters</p> <p>data : 문자열로 변경된 text 파일 format : 파일의 형식</p> <table border="1"> <thead> <tr> <th>format</th><th>설명</th><th>비고</th></tr> </thead> <tbody> <tr> <td>HWP</td><td>HWP native format</td><td>BASE64로 인코딩되어 있어야 한다. 저장된 내용을 다른 곳에서 보여줄 필요가 없다면 이 포맷을 사용하기를 권장합니다.</td></tr> <tr> <td>HWPML2X</td><td>HWP 형식과 호환</td><td>문서의 모든 정보를 유지</td></tr> <tr> <td>HTML</td><td>인터넷 문서 HTML 형식</td><td>한글 고유의 서식은 손실된다.</td></tr> <tr> <td>UNICODE</td><td>유니코드 텍스트</td><td>서식정보가 없는 텍스트만 저장</td></tr> <tr> <td>TEXT</td><td>일반 텍스트</td><td>유니코드에만 있는 정보(한자, 고어, 특수문자 등)는 모두 손실된다.</td></tr> </tbody> </table> <p>option</p> <table border="1"> <thead> <tr> <th>option</th><th>설명</th><th>비고</th></tr> </thead> <tbody> <tr> <td>insertfile</td><td>현재커서 이후에 삽입</td><td></td></tr> </tbody> </table> <p>Remark 이 함수는 JScript나 VBScript와 같이 직접적으로 local disk를 접근하기 힘든 언어를 위해 만들어졌으므로 disk를 접근할 수 있는 언어에서는 사용하지 않기를 권장합니다. disk를 접근할 수 있다면, Open이나 Insert를 사용하십시오. 이 함수 역시 내부적으로는 Open이나 Insert를 호출하도록 되어있고 텍스트로 저장된 파일이 메모리에서 3~4번 복사되기 때문에 느리고, 메모리를 낭비합니다.</p>	format	설명	비고	HWP	HWP native format	BASE64로 인코딩되어 있어야 한다. 저장된 내용을 다른 곳에서 보여줄 필요가 없다면 이 포맷을 사용하기를 권장합니다.	HWPML2X	HWP 형식과 호환	문서의 모든 정보를 유지	HTML	인터넷 문서 HTML 형식	한글 고유의 서식은 손실된다.	UNICODE	유니코드 텍스트	서식정보가 없는 텍스트만 저장	TEXT	일반 텍스트	유니코드에만 있는 정보(한자, 고어, 특수문자 등)는 모두 손실된다.	option	설명	비고	insertfile	현재커서 이후에 삽입	
format	설명	비고																							
HWP	HWP native format	BASE64로 인코딩되어 있어야 한다. 저장된 내용을 다른 곳에서 보여줄 필요가 없다면 이 포맷을 사용하기를 권장합니다.																							
HWPML2X	HWP 형식과 호환	문서의 모든 정보를 유지																							
HTML	인터넷 문서 HTML 형식	한글 고유의 서식은 손실된다.																							
UNICODE	유니코드 텍스트	서식정보가 없는 텍스트만 저장																							
TEXT	일반 텍스트	유니코드에만 있는 정보(한자, 고어, 특수문자 등)는 모두 손실된다.																							
option	설명	비고																							
insertfile	현재커서 이후에 삽입																								
CreatePageImage(Method)	<p>Description 지정한 페이지의 이미지를 파일로 생성한다.</p> <p>Declaration boolean CreatePageImage(BSTR path, [long pgno], [short resolution], [short depth], [BSTR format])</p> <p>Parameters</p> <p>path : 생성할 이미지 파일의 경로 pgno : 페이지 번호. 0부터 PageCount - 1까지. 생략하면 0이 사용된다. resolution : 이미지 해상도. DPI 단위(96, 300, 1200)로 지정한다. 생략하면 960이 사용된다. depth : 이미지 파일의 color depth(1, 4, 8, 24)를 지정한다. format : 이미지 파일의 포맷. "bmp", "gif" 중의 하나. 생략하면 "bmp"이 사용된다.</p>																								
Run(Method)	<p>Description 액션을 실행한다.</p> <p>Declaration void Run(BSTR action)</p> <p>Parameters</p> <p>action : 액션 ID (별도 문서 참조)</p>																								
LockCommand(Method)	<p>Description 특정 액션이 실행되지 않도록 잠근다.</p> <p>Declaration void LockCommand(BSTR actionID, boolean lock)</p> <p>Parameters</p>																								

	<p>actionID : 액션 ID lock : True이면 액션의 실행을 lock시키고, False이면 unlock시킨다.</p>
IsCommandLock(Method)	<p>Description 특정 액션이 잠금 상태인지 여부를 조사한다. Declaration boolean IsCommandLock(BSTR actionID) Parameters actionID : 액션 ID</p>
InsertPicture(Method)	<p>Description 현재 캐럿의 위치에 그림을 삽입한다. Declaration Ctrl InsertPicture(BSTR path, [boolean embeded], [short sizeoption], [boolean reverse], [boolean watermark], [short effect], [long width], [long height]) Parameters path : 삽입할 이미지 파일, URL 사용 가능 embeded : 이미지 파일을 문서내에 포함할지 여부 (True/False). 생략하면 True sizeoption : 삽입할 그림의 크기를 지정하는 옵션 typedef enum { // 이미지 원래의 크기로 삽입한다. // width와 height를 지정할 필요없다. realSize = 0, // width와 height에 지정한 크기로 그림을 삽입한다. specificSize = 1, // 현재 캐럿이 표의 셀안에 있을 경우, // 셀의 크기에 맞게 자동 조절하여 삽입한다. // width는 셀의 width만큼, // height는 셀의 height만큼 확대/축소된다. // 캐럿이 셀안에 있지 않으면 // 이미지의 원래 크기대로 삽입된다. cellSize = 2, // 현재 캐럿이 표의 셀안에 있을 경우, // 셀의 크기에 맞추어 원본 이미지의 가로 세로의 // 비율이 동일하게 확대/축소하여 삽입한다. cellSizeWithSameRatio = 3 } HwpSizeOption; reverse : 이미지의 반전 유무 (True/False) watermark : watermark효과 유무 (True/False) effect : 그림 효과 typedef enum { RealPicture = 0, // 실제 이미지 그대로 GrayScale = 1, // 그레이스케일 BlackWhite = 2, // 흑백효과 } HwpPictureEffect; width : 그림의 가로 크기 지정. 단위는 mm height : 그림의 높이 크기 지정. 단위는 mm</p>
InsertBackgroundPicture(Method)	<p>Description 배경이미지를 넣는다. Declaration VARIANT InsertBackgroundPicture(BSTR border type, BSTR path, [bool embedded], [long filloption], [bool watermark], [long effect], [long brightness], [long contrast]); Parameters</p>

	<p>border type : 배경 유형을 지정</p> <table border="1"> <thead> <tr> <th>bordertype</th><th>설명</th><th>비고</th></tr> </thead> <tbody> <tr> <td>"SelectedCell"</td><td>현재 선택된 표의 셀의 배경을 변경한다.</td><td></td></tr> <tr> <td>"SelectedCellDelete"</td><td>현재 선택된 표의 셀의 배경을 지운다.</td><td>반드시 셀이 선택되어 있어야 함. 커서가 위치하는 것만으로는 동작하지 않음.</td></tr> </tbody> </table> <p>path : 삽입할 이미지 파일, URL 사용 가능 embeded : 이미지 파일을 문서내에 포함할지 여부 (True/False). 생략하면 True filloption : 삽입할 그림의 크기를 지정하는 옵션</p> <table border="1"> <thead> <tr> <th>filloption</th><th>설명</th><th>비고</th></tr> </thead> <tbody> <tr> <td>0</td><td>바둑판식으로 - 모두</td><td></td></tr> <tr> <td>1</td><td>바둑판식으로 - 가로/위</td><td></td></tr> <tr> <td>2</td><td>바둑판식으로 - 가로/아로</td><td></td></tr> <tr> <td>3</td><td>바둑판식으로 - 세로/왼쪽</td><td></td></tr> <tr> <td>4</td><td>바둑판식으로 - 세로/오른쪽</td><td></td></tr> <tr> <td>5</td><td>크기에 맞추어</td><td>설정하지 않았을 때 기본값</td></tr> <tr> <td>6</td><td>가운데로</td><td></td></tr> <tr> <td>7</td><td>가운데 위로</td><td></td></tr> <tr> <td>8</td><td>가운데 아래로</td><td></td></tr> <tr> <td>9</td><td>왼쪽 가운데로</td><td></td></tr> <tr> <td>10</td><td>왼쪽 위로</td><td></td></tr> <tr> <td>11</td><td>왼쪽 아래로</td><td></td></tr> <tr> <td>12</td><td>오른쪽 가운데로</td><td></td></tr> <tr> <td>13</td><td>오른쪽 위로</td><td></td></tr> <tr> <td>14</td><td>오른쪽 아래로</td><td></td></tr> </tbody> </table> <p>effect : 이미지 효과 watermark : watermark 효과 유무 (True/False) 이 옵션이 true이면 brightness 와 contrast 옵션이 무시된다. effect</p> <table border="1"> <thead> <tr> <th>effect</th><th>설명</th><th>비고</th></tr> </thead> <tbody> <tr> <td>0</td><td>원래 그림</td><td>설정하지 않았을 때 기본값</td></tr> <tr> <td>1</td><td>그래이 스케일</td><td></td></tr> <tr> <td>2</td><td>흑백으로</td><td></td></tr> </tbody> </table> <p>brightness : 밝기 지정 (-100 ~ 100), 기본값 : 0 contrast : 선명도 지정 (-100 ~ 100), 기본값 : 0</p> <p>Remark CellBorderFill의 SetItem 중 FillAttr 의 SetItem FileName에 이미지의 binary data를 지정해 줄 수가 없어서 만든 함수다. 기타 배경에 대한 다른 조정은 Action과 ParameterSet의 조합으로 가능하다.</p>	bordertype	설명	비고	"SelectedCell"	현재 선택된 표의 셀의 배경을 변경한다.		"SelectedCellDelete"	현재 선택된 표의 셀의 배경을 지운다.	반드시 셀이 선택되어 있어야 함. 커서가 위치하는 것만으로는 동작하지 않음.	filloption	설명	비고	0	바둑판식으로 - 모두		1	바둑판식으로 - 가로/위		2	바둑판식으로 - 가로/아로		3	바둑판식으로 - 세로/왼쪽		4	바둑판식으로 - 세로/오른쪽		5	크기에 맞추어	설정하지 않았을 때 기본값	6	가운데로		7	가운데 위로		8	가운데 아래로		9	왼쪽 가운데로		10	왼쪽 위로		11	왼쪽 아래로		12	오른쪽 가운데로		13	오른쪽 위로		14	오른쪽 아래로		effect	설명	비고	0	원래 그림	설정하지 않았을 때 기본값	1	그래이 스케일		2	흑백으로	
bordertype	설명	비고																																																																				
"SelectedCell"	현재 선택된 표의 셀의 배경을 변경한다.																																																																					
"SelectedCellDelete"	현재 선택된 표의 셀의 배경을 지운다.	반드시 셀이 선택되어 있어야 함. 커서가 위치하는 것만으로는 동작하지 않음.																																																																				
filloption	설명	비고																																																																				
0	바둑판식으로 - 모두																																																																					
1	바둑판식으로 - 가로/위																																																																					
2	바둑판식으로 - 가로/아로																																																																					
3	바둑판식으로 - 세로/왼쪽																																																																					
4	바둑판식으로 - 세로/오른쪽																																																																					
5	크기에 맞추어	설정하지 않았을 때 기본값																																																																				
6	가운데로																																																																					
7	가운데 위로																																																																					
8	가운데 아래로																																																																					
9	왼쪽 가운데로																																																																					
10	왼쪽 위로																																																																					
11	왼쪽 아래로																																																																					
12	오른쪽 가운데로																																																																					
13	오른쪽 위로																																																																					
14	오른쪽 아래로																																																																					
effect	설명	비고																																																																				
0	원래 그림	설정하지 않았을 때 기본값																																																																				
1	그래이 스케일																																																																					
2	흑백으로																																																																					
CreateAction(Method)	<p>Description 오브젝트를 생성한다.</p> <p>Declaration Action CreateAction(BSTR action)</p> <p>Parameters action : 액션 ID (별도 문서 참조)</p> <p>Remark 액션에 대한 세부적인 제어가 필요할 때 사용한다. 예를 들어 기능을 수행하지 않고 대화상자만을 띄운다든지, 대화상자 없이 지정한 옵션에 따라 기능을 수행하는 등에 사용할 수 있다.</p>																																																																					

InsertCtrl(Method)	<p>Description 현재 캐럿 위치에 컨트롤을 삽입한다.</p> <p>Declaration Ctrl InsertCtrl(BSTR ctrlid, [ParameterSet initparam])</p> <p>Parameters</p> <p>ctrlid : 삽입할 컨트롤의 ID initparam : 컨트롤의 초기 속성. 생략하면 디폴트 속성으로 생성한다.</p> <p>Remark</p> <ul style="list-style-type: none"> ctrlid에 지정할 수 있는 컨트롤 ID는 HwpCtrl.CtrlID가 리턴하는 ID와 동일하다. 자세한 것은 Ctrl 오브젝트 Properties의 CtrlID를 참조. initparam에는 컨트롤의 초기 속성을 지정한다. 대부분의 컨트롤은 Ctrl.Properties와 동일한 포맷의 parameter set을 사용하지만, 컨트롤 생성시에는 다른 포맷을 사용하는 경우도 있다. 예를 들어 표의 경우 Ctrl.Properties에는 "Table" 셋을 사용하지만, 생성시 initparam에 지정하는 값은 "TableCreation" 셋이다. 																																				
DeleteCtrl(Method)	<p>Description 문서내 컨트롤을 삭제한다.</p> <p>Declaration boolean DeleteCtrl(HwpCtrlCode ctrl)</p> <p>Parameters</p> <p>ctrl : 삭제할 문서내 컨트롤</p>																																				
GetMousePos(Method)	<p>Description 마우스의 현재 위치를 구한다</p> <p>Declaration ParameterSet GetMousePos(long Xrelto, long Yrelto)</p> <p>Parameters</p> <p>Xrelto : X좌표계의 기준 위치</p> <table border="1" data-bbox="653 1139 1440 1267"> <thead> <tr> <th>value</th><th>설명</th><th>비고</th></tr> </thead> <tbody> <tr> <td>0</td><td>종이 기준</td><td>종이 기준으로 좌표를 가져온다.</td></tr> <tr> <td>1</td><td>쪽 기준</td><td>쪽 기준으로 좌표를 가져온다.</td></tr> </tbody> </table> <p>Yrelto : Y좌표계의 기준 위치</p> <table border="1" data-bbox="653 1296 1440 1424"> <thead> <tr> <th>value</th><th>설명</th><th>비고</th></tr> </thead> <tbody> <tr> <td>0</td><td>종이 기준</td><td>종이 기준으로 좌표를 가져온다.</td></tr> <tr> <td>1</td><td>쪽 기준</td><td>쪽 기준으로 좌표를 가져온다.</td></tr> </tbody> </table> <p>Return : "MousePos" ParameterSet 이 리턴된다.</p> <p>[Set ID] MousePos</p> <table border="1" data-bbox="653 1498 1440 1834"> <thead> <tr> <th>Item ID</th><th>Type</th><th>Description</th></tr> </thead> <tbody> <tr> <td>XRelTo</td><td>PIT_UI4</td><td>가로 상대적 기준 0 : 종이 1 : 쪽</td></tr> <tr> <td>YRelTo</td><td>PIT_UI4</td><td>세로 상대적 기준 0 : 종이 1 : 쪽</td></tr> <tr> <td>Page</td><td>PIT_UI4</td><td>페이지 번호 (0 based)</td></tr> <tr> <td>X</td><td>PIT_I4</td><td>가로 클릭한 위치 (HWPUNIT)</td></tr> <tr> <td>Y</td><td>PIT_I4</td><td>세로 클릭한 위치 (HWPUNIT)</td></tr> </tbody> </table> <p>Remark 단위가 HWPUNIT임을 주의하십시오.</p>	value	설명	비고	0	종이 기준	종이 기준으로 좌표를 가져온다.	1	쪽 기준	쪽 기준으로 좌표를 가져온다.	value	설명	비고	0	종이 기준	종이 기준으로 좌표를 가져온다.	1	쪽 기준	쪽 기준으로 좌표를 가져온다.	Item ID	Type	Description	XRelTo	PIT_UI4	가로 상대적 기준 0 : 종이 1 : 쪽	YRelTo	PIT_UI4	세로 상대적 기준 0 : 종이 1 : 쪽	Page	PIT_UI4	페이지 번호 (0 based)	X	PIT_I4	가로 클릭한 위치 (HWPUNIT)	Y	PIT_I4	세로 클릭한 위치 (HWPUNIT)
value	설명	비고																																			
0	종이 기준	종이 기준으로 좌표를 가져온다.																																			
1	쪽 기준	쪽 기준으로 좌표를 가져온다.																																			
value	설명	비고																																			
0	종이 기준	종이 기준으로 좌표를 가져온다.																																			
1	쪽 기준	쪽 기준으로 좌표를 가져온다.																																			
Item ID	Type	Description																																			
XRelTo	PIT_UI4	가로 상대적 기준 0 : 종이 1 : 쪽																																			
YRelTo	PIT_UI4	세로 상대적 기준 0 : 종이 1 : 쪽																																			
Page	PIT_UI4	페이지 번호 (0 based)																																			
X	PIT_I4	가로 클릭한 위치 (HWPUNIT)																																			
Y	PIT_I4	세로 클릭한 위치 (HWPUNIT)																																			
Clear(Method)	<p>Description 현재 편집중인 문서의 내용을 닫고 빈문서 편집 상태로 돌아간다.</p>																																				

	<p>Declaration</p> <pre>void Clear([HwpSaveOption option])</pre> <p>Parameters</p> <p>option : 편집중인 문서의 내용에 대한 처리 방법. 생략하면 hwpAskSave가 선택된다. (HwpSaveOption : short type)</p> <table border="1"> <thead> <tr> <th>ID</th><th>값</th><th>설명</th></tr> </thead> <tbody> <tr> <td>hwpAskSave</td><td>0</td><td>문서의 내용이 변경되었을 때 사용자에게 저장 할지 묻는 대화상자를 띄운다.</td></tr> <tr> <td>hwpDiscard</td><td>1</td><td>문서 내용을 버린다.</td></tr> <tr> <td>hwpSaveIfDirty</td><td>2</td><td>문서가 변경된 경우 저장 한다.</td></tr> <tr> <td>hwpSave</td><td>3</td><td>무조건 저장한다.</td></tr> </tbody> </table> <p>Remark</p> <p>format, arg에 대해서는 Open 참조. hwpSaveIfDirty, hwpSave가 지정된 경우 현재 문서 경로가 지정되어 있지 않으면 "새이름으로 저장" 대화상자를 띄워 사용자에게 경로를 묻는다.</p>	ID	값	설명	hwpAskSave	0	문서의 내용이 변경되었을 때 사용자에게 저장 할지 묻는 대화상자를 띄운다.	hwpDiscard	1	문서 내용을 버린다.	hwpSaveIfDirty	2	문서가 변경된 경우 저장 한다.	hwpSave	3	무조건 저장한다.
ID	값	설명														
hwpAskSave	0	문서의 내용이 변경되었을 때 사용자에게 저장 할지 묻는 대화상자를 띄운다.														
hwpDiscard	1	문서 내용을 버린다.														
hwpSaveIfDirty	2	문서가 변경된 경우 저장 한다.														
hwpSave	3	무조건 저장한다.														
RegisterModule(Method)	<p>Remark</p> <p>보안 모듈 관련 문서 참조</p>															
ReplaceAction(Method)	<p>Description</p> <p>특정 Action을 다른 Action으로 대체한다.</p> <p>Declaration</p> <pre>bool ReplaceAction(BSTR OldActionID, BSTR NewActionID);</pre> <p>Parameters</p> <p>OldActionID : 변경될 원본 Action ID. HwpAutomation 사용할 수 있는 Action ID는 ActionObject.hwp(별도문서)를 참고한다.</p> <p>NewActionID : 변경할 대체 Action ID.</p> <p>Remark</p> <ul style="list-style-type: none"> 특정 Action을 다른 Action으로 대체한다. 이는 메뉴나 단축키로 호출되는 Action을 대체할 뿐, CreateAction()이나, Run() 등의 함수를 이용할 때에는 아무런 영향을 주지 않는다. 즉, ReplaceAction("Cut", "Copy")을 호출하여 "Action을 "복사하기" Action으로 교체하면 Ctrl+X 단축키나 오려내기 메뉴/툴바 기능을 수행하더라도 복사하기 기능이 수행되지만, 코드 상에서 Run("Cut")을 실행하면 오려내기 Action이 실행된다. ReplaceAction()을 사용할 때에는 대체되는 Action들이 서로 Loop를 형성하지 않도록 조심해야 한다. 															
InitParameterSet(Method)	<p>Description</p> <p>HParameterSet Object를 초기화한다.</p>															
GetPosBySet(Method)	<p>Description</p> <p>현재 캐럿의 위치 정보를 ParameterSet으로 얻어온다.</p> <p>Remark</p> <p>캐럿의 위치를 ParameterSet으로 얻어온다. 포인터를 사용할 수 없는 언어에서도 사용 가능하다.</p>															
SetPosBySet(Method)	<p>Description</p> <p>캐럿을 ParameterSet으로 얻어지는 위치로 옮긴다.</p> <p>Declaration</p> <pre>boolean SetPosBySet(LPDISPATCH pos);</pre> <p>Parameters</p> <p>pos : 캐럿을 옮길 위치에 대한 ParameterSet 정보</p>															
Application(Property)	<p>Description</p> <p>최상위 오브젝트 (IHwpObject interface)를 얻는다.</p>															
XHwpMessageBox(Property)	<p>Description</p> <p>메시지 박스를 제어하는 XHwpMessageBox Object를 얻는다.</p>															

XHwpDocuments(Property)	<p>Description 도큐먼트를 관리하는 XHwpDocuments Object를 얻는다.</p>																																							
XHwpWindows(Property)	<p>Description 윈도우를 관리하는 XHwpWindows Object를 얻는다.</p>																																							
HParameterSet(Property)	<p>Description 파라미터셋 오브젝트를 관리하는 HParameterSet Object를 얻는다.</p>																																							
HAction(Property)	<p>Description Action을 제어하는 HAction Object를 얻는다.</p>																																							
XHwpODBC(Property)	<p>Description ODBC로 제어할 수 있는 Object를 얻는다.</p>																																							
Version(Property)	<p>Description 한글과 한글 OCX의 버전 정보를 구한다. 읽기 전용.</p> <ul style="list-style-type: none"> • byte 3 = 한글의 major version. • byte 2 = 한글의 minor version. • byte 1 = 한글 OCX의 major version. • byte 0 = 한글 OCX의 minor version. 																																							
EngineProperties(Property)	<p>Description 환경설정 정보를 설정한다.</p> <p>Declaration void EngineProperties(LPDISPATCH param) - set LPDISPATCH* EngineProperties() - get</p> <p>Parameters param : HParameterSet</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #ffffcc;"> <th style="text-align: center; padding: 2px;">Item ID</th> <th style="text-align: center; padding: 2px;">Type</th> <th style="text-align: center; padding: 2px;">Description</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;">DoAnyCursorEdit</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">마우스로 두 번 누르기 한곳에 입력 가능</td> </tr> <tr> <td style="padding: 2px;">DoOutLineStyle</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">개요 번호 삽입 문단에 개요 스타일 자동 적용</td> </tr> <tr> <td style="padding: 2px;">InsertLock</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">삽입 잠금</td> </tr> <tr> <td style="padding: 2px;">TabMoveCell</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">표 안에서 <Tab>으로 셀 이동</td> </tr> <tr> <td style="padding: 2px;">FaxDriver</td> <td style="text-align: center; padding: 2px;">PIT_BSTR</td> <td style="padding: 2px;">팩스 드라이버</td> </tr> <tr> <td style="padding: 2px;">PDFDriver</td> <td style="text-align: center; padding: 2px;">PIT_BSTR</td> <td style="padding: 2px;">PDF 드라이버</td> </tr> <tr> <td style="padding: 2px;">EnableAutoSpell</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">맞춤법 도우미 작동</td> </tr> <tr> <td style="padding: 2px;">OpenNewWindow</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">새창으로 불러오기</td> </tr> <tr> <td style="padding: 2px;">CtrlMaskAs2002</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">한글 2002 방식으로 조판 부호 표시하기 (한글 버전 : 7.5.11.603)</td> </tr> <tr> <td style="padding: 2px;">ShowGuidelines</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">투명선 보이기 (한글 버전 : 7.5.11.603)</td> </tr> <tr> <td style="padding: 2px;">ImageAutoCheck</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">이미지 파일의 경로 검사 대 이얼로그 띄우기 유무.</td> </tr> <tr> <td style="padding: 2px;">OptimizeWebHwpCopy</td> <td style="text-align: center; padding: 2px;">PIT_UI1</td> <td style="padding: 2px;">웹한글로 복사 최적화 끄고/켜기</td> </tr> </tbody> </table> <p>Remark 환경설정에서 지정할 수 있는 옵션 값을 설정할 수 있다.</p>	Item ID	Type	Description	DoAnyCursorEdit	PIT_UI1	마우스로 두 번 누르기 한곳에 입력 가능	DoOutLineStyle	PIT_UI1	개요 번호 삽입 문단에 개요 스타일 자동 적용	InsertLock	PIT_UI1	삽입 잠금	TabMoveCell	PIT_UI1	표 안에서 <Tab>으로 셀 이동	FaxDriver	PIT_BSTR	팩스 드라이버	PDFDriver	PIT_BSTR	PDF 드라이버	EnableAutoSpell	PIT_UI1	맞춤법 도우미 작동	OpenNewWindow	PIT_UI1	새창으로 불러오기	CtrlMaskAs2002	PIT_UI1	한글 2002 방식으로 조판 부호 표시하기 (한글 버전 : 7.5.11.603)	ShowGuidelines	PIT_UI1	투명선 보이기 (한글 버전 : 7.5.11.603)	ImageAutoCheck	PIT_UI1	이미지 파일의 경로 검사 대 이얼로그 띄우기 유무.	OptimizeWebHwpCopy	PIT_UI1	웹한글로 복사 최적화 끄고/켜기
Item ID	Type	Description																																						
DoAnyCursorEdit	PIT_UI1	마우스로 두 번 누르기 한곳에 입력 가능																																						
DoOutLineStyle	PIT_UI1	개요 번호 삽입 문단에 개요 스타일 자동 적용																																						
InsertLock	PIT_UI1	삽입 잠금																																						
TabMoveCell	PIT_UI1	표 안에서 <Tab>으로 셀 이동																																						
FaxDriver	PIT_BSTR	팩스 드라이버																																						
PDFDriver	PIT_BSTR	PDF 드라이버																																						
EnableAutoSpell	PIT_UI1	맞춤법 도우미 작동																																						
OpenNewWindow	PIT_UI1	새창으로 불러오기																																						
CtrlMaskAs2002	PIT_UI1	한글 2002 방식으로 조판 부호 표시하기 (한글 버전 : 7.5.11.603)																																						
ShowGuidelines	PIT_UI1	투명선 보이기 (한글 버전 : 7.5.11.603)																																						
ImageAutoCheck	PIT_UI1	이미지 파일의 경로 검사 대 이얼로그 띄우기 유무.																																						
OptimizeWebHwpCopy	PIT_UI1	웹한글로 복사 최적화 끄고/켜기																																						
ExportStyle(Method)	<p>Description HStyleTemplate 파라미터셋 오브젝트에 지정된 스타일을 Export 한다.</p> <p>Declaration BOOL ExportStyle(LPDISPATCH param)</p> <p>Parameters param : HStyleTemplate</p>																																							
ImportStyle(Method)	<p>Description</p>																																							

	<p>HStyleTemplate 파라메터셋 오브젝트에 지정된 스타일을 Import한다.</p> <p>Declaration BOOL ImportStyle(LPDISPATCH param)</p> <p>Parameters param : HStyleTemplate</p>																		
FindCtrl(Method)	<p>Description 현재 캐럿의 위치에서 Ctrl을 찾는다.</p> <p>Declaration BSTR FindCtrl()</p> <p>Parameters return : 컨트롤을 찾은 경우 CtrlID를 return 한다.</p>																		
UnSelectCtrl(Method)	<p>Description 현재 Select된 Ctrl의 Selection을 해제한다.</p>																		
IsActionEnable(Method)	<p>Description 액션이 실행 가능한 상태인지 확인한다.</p> <p>Declaration bool IsActionEnable(BSTR actionID)</p> <p>Parameters actionID: 컨트롤을 찾은 경우 CtrlID를 return 한다. • Action ID는 ActionObject.hwp(별도문서)를 참고한다.</p>																		
GetScriptSource(Method)	<p>Description 한글에서 사용하는 매크로의 소스코드를 가져온다.</p> <p>Declaration BSTR GetScriptSource(BSTR FileName);</p> <p>Parameters FileName : 매크로 소스를 가져올 한글문서</p>																		
GetFileInfo(Method)	<p>Description 파일 정보를 알아낸다.</p> <p>Declaration LPDISPATCH* GetFileInfo(BSTR FileName);</p> <p>Parameters FileName : 정보를 구하고자 하는 파일명 반환값(Return) : "FileInfo" ParameterSet이 반환된다.</p> <table border="1"> <thead> <tr> <th>Item ID</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>Format</td> <td>PIT_BSTR</td> <td>파일 형식</td> </tr> <tr> <td>VersionStr</td> <td>PIT_BSTR</td> <td>파일 버전 문자열 ex) 13.0.0.1</td> </tr> <tr> <td>VersionNum</td> <td>PIT_UI4</td> <td>파일 버전 ex) 0x0d000001</td> </tr> <tr> <td>Encrypted</td> <td>PIT_UI1</td> <td>암호화 여부 -1 : 판단 할 수 없음 0 : 암호가 걸려 있지 않음 양수: 암호가 걸려 있음.</td> </tr> <tr> <td>Compressed</td> <td>PIT_UI1</td> <td>압축 여부</td> </tr> </tbody> </table>	Item ID	Type	Description	Format	PIT_BSTR	파일 형식	VersionStr	PIT_BSTR	파일 버전 문자열 ex) 13.0.0.1	VersionNum	PIT_UI4	파일 버전 ex) 0x0d000001	Encrypted	PIT_UI1	암호화 여부 -1 : 판단 할 수 없음 0 : 암호가 걸려 있지 않음 양수: 암호가 걸려 있음.	Compressed	PIT_UI1	압축 여부
Item ID	Type	Description																	
Format	PIT_BSTR	파일 형식																	
VersionStr	PIT_BSTR	파일 버전 문자열 ex) 13.0.0.1																	
VersionNum	PIT_UI4	파일 버전 ex) 0x0d000001																	
Encrypted	PIT_UI1	암호화 여부 -1 : 판단 할 수 없음 0 : 암호가 걸려 있지 않음 양수: 암호가 걸려 있음.																	
Compressed	PIT_UI1	압축 여부																	
RunScriptMacro(Method)	<p>Description 한글문서 내에 존재하는 매크로를 실행한다.</p> <p>Declaration bool RunScriptMacro(BSTR FunctionName, long uMacroType, long uScriptType)</p> <p>Parameters FunctionName : 실행할 매크로 함수이름</p>																		

	<p>uMacroType : 매크로의 유형. 밑의 값 중 하나이다.</p> <table border="1"> <thead> <tr> <th>ID</th><th>값</th><th>Description</th></tr> </thead> <tbody> <tr> <td>HWP_GLOBAL_MACRO_TYPE</td><td>0</td><td>전역 매크로</td></tr> <tr> <td>HWP_DOCUMENT_MACRO_TYPE</td><td>1</td><td>문서에만 적용되는 매크로</td></tr> </tbody> </table> <p>uScriptType : 스크립트의 유형. 현재는 JScript만을 유일하게 지원한다. reserved.</p> <p>Remark 한글은 매크로 기능을 지원하고 있다. 이렇게 작성된 매크로를 HwpAutomation으로 실행시킬 수 있도록 지원하는 함수이다.</p>	ID	값	Description	HWP_GLOBAL_MACRO_TYPE	0	전역 매크로	HWP_DOCUMENT_MACRO_TYPE	1	문서에만 적용되는 매크로						
ID	값	Description														
HWP_GLOBAL_MACRO_TYPE	0	전역 매크로														
HWP_DOCUMENT_MACRO_TYPE	1	문서에만 적용되는 매크로														
GetPageText(Method)	<p>Description 페이지 단위의 텍스트 추출</p> <p>Declaration BSTR GetPageText(long pgno, VARIANT option);</p> <p>Parameters pgno : 텍스트를 추출 할 페이지의 번호(0부터 시작) option : 추출 대상을 다음과 같은 옵션을 조합하여 지정할 수 있다. 생략(또는 0xffffffff)하면 모든 텍스트를 추출한다.</p> <table border="1"> <thead> <tr> <th>ID</th><th>값</th><th>Description</th></tr> </thead> <tbody> <tr> <td>maskNormal</td><td>0x00</td><td>본문 텍스트를 추출한다.(항상 기본)</td></tr> <tr> <td>maskTable</td><td>0x01</td><td>표에 대한 텍스트를 추출한다.</td></tr> <tr> <td>maskTextbox</td><td>0x02</td><td>글상자 텍스트를 추출한다.</td></tr> <tr> <td>maskCaption</td><td>0x04</td><td>캡션 텍스트를 추출한다. (표, ShapeObject)</td></tr> </tbody> </table> <p>Remark 일반 텍스트(글자처럼 취급 도형 포함)를 우선적으로 추출하고, 도형(표, 글상자) 내의 텍스트를 추출한다.</p>	ID	값	Description	maskNormal	0x00	본문 텍스트를 추출한다.(항상 기본)	maskTable	0x01	표에 대한 텍스트를 추출한다.	maskTextbox	0x02	글상자 텍스트를 추출한다.	maskCaption	0x04	캡션 텍스트를 추출한다. (표, ShapeObject)
ID	값	Description														
maskNormal	0x00	본문 텍스트를 추출한다.(항상 기본)														
maskTable	0x01	표에 대한 텍스트를 추출한다.														
maskTextbox	0x02	글상자 텍스트를 추출한다.														
maskCaption	0x04	캡션 텍스트를 추출한다. (표, ShapeObject)														
SetBarcodeImage(Method)	<p>Description 바코드 이미지를 삽입한다.</p> <p>Declaration bool SetBarcodeImage(BSTR imagePath, long pgno, long index, long x, long y, long width, long height)</p> <p>Parameters imagePath : 바코드 이미지 경로 pgno : 바코드 이미지를 삽입 할 페이지 번호(0부터 시작) index : 페이지에 삽입될 바코드 이미지의 번호(0부터 시작) <ul style="list-style-type: none"> 같은 번호를 지정하면 이전에 삽입된 바코드의 이미지가 수정된다. 바코드 이미지를 공백으로 지정하면 해당 번호의 바코드 이미지가 삭제된다. x : 바코드 이미지가 삽입 될 위치의 x좌표 (mm단위) <ul style="list-style-type: none"> 바코드 이미지의 좌측 상단 기준으로 페이지(종이 기준) 좌측 상단에서의 거리이다. y : 바코드 이미지가 삽입 될 위치의 y좌표 (mm단위) <ul style="list-style-type: none"> 바코드 이미지의 좌측 상단 기준으로 페이지(종이 기준) 좌측 상단에서의 거리이다. width : 바코드 이미지의 너비. 생략하면 원본 이미지의 너비가 적용된다. height : 바코드 이미지의 높이. 생략하면 원본 이미지의 높이가 적용된다.</p>															
GetMessageBoxMode(Method)	<p>Description 현재 메시지 박스의 Mode를 long으로 얻어온다.</p> <p>Parameters return : <i>// 메시지 박스의 종류</i> <i>#define MB_MASK</i> 0x00FFFFFF</p>															

	<pre> // 1. 확인(MB_OK) : IDOK(1) #define MB_OK_IDOK 0x00000001 #define MB_OK_MASK 0x0000000F // 2. 확인/취소(MB_OKCANCEL) : IDOK(1), IDCANCEL(2) #define MB_OKCANCEL_IDOK 0x00000010 #define MB_OKCANCEL_IDCANCEL 0x00000020 #define MB_OKCANCEL_MASK 0x000000F0 // 3. 종료/재시도/무시(MB_ABORTRETRYIGNORE) : IDABORT(3), IDRETRY(4), IDIGNORE(5) #define MB_ABORTRETRYIGNORE_IDABORT 0x00000100 #define MB_ABORTRETRYIGNORE_IDRETRY 0x00000200 #define MB_ABORTRETRYIGNORE_IDIGNORE 0x00000400 #define MB_ABORTRETRYIGNORE_MASK 0x0000F000 // 4. 예/아니오/취소(MB_YESNOCANCEL) : IDYES(6), IDNO(7), IDCANCEL(2) #define MB_YESNOCANCEL_IDYES 0x00010000 #define MB_YESNOCANCEL_IDNO 0x00020000 #define MB_YESNOCANCEL_IDCANCEL 0x00004000 #define MB_YESNOCANCEL_MASK 0x0000F000 // 5. 예/아니오(MB_YESNO) : IDYES(6), IDNO(7) #define MB_YESNO_IDYES 0x00010000 #define MB_YESNO_IDNO 0x00020000 #define MB_YESNO_MASK 0x000F0000 // 6. 재시도/취소(MB_RETRYCANCEL) : IDRETRY(4), IDCANCEL(2) #define MB_RETRYCANCEL_IDRETRY 0x00100000 #define MB_RETRYCANCEL_IDCANCEL 0x00200000 #define MB_RETRYCANCEL_MASK 0x00F00000 </pre>						
SetMessageBoxMode(Method)	<p>Description 한글에서 쓰는 다양한 메시지박스가 뜨지 않고, 자동으로 특정 버튼을 클릭한 효과를 주기 위해 사용한다.</p> <p>Declaration long SetMessageBoxMode(long mode);</p> <p>Parameters mode : GetMessageBoxMode 참고 return : 기존에 적용되어 있던 oldMode 값</p>						
GetBinDataPath(Method)	<p>Description Binary Data의 경로를 가져온다.</p> <p>Declaration BSTR GetBinDataPath(long binid);</p> <p>Parameters 바이너리 데이터의 ID 값 (1부터 시작)</p>						
SetDRMAuthority(Method)	<p>Description DRM모듈에 권한을 설정한다.</p> <p>Declaration bool SetDRMAuthority(long authority);</p> <p>Parameters authority :</p> <table border="1"> <thead> <tr> <th>값</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0x0001</td> <td>읽기</td> </tr> <tr> <td>0x0002</td> <td>편집</td> </tr> </tbody> </table>	값	Description	0x0001	읽기	0x0002	편집
값	Description						
0x0001	읽기						
0x0002	편집						

	<table border="1"> <tr><td>0x0004</td><td>인쇄</td></tr> <tr><td>0x0008</td><td>클립보드</td></tr> <tr><td>0x0010</td><td>같은 프로세스 내에서도 시스템 클립보드를 사용하기 위한 권한</td></tr> <tr><td>0x0020</td><td>클립보드 히스토리 사용하지 않도록 하기 위한 권한</td></tr> </table>	0x0004	인쇄	0x0008	클립보드	0x0010	같은 프로세스 내에서도 시스템 클립보드를 사용하기 위한 권한	0x0020	클립보드 히스토리 사용하지 않도록 하기 위한 권한
0x0004	인쇄								
0x0008	클립보드								
0x0010	같은 프로세스 내에서도 시스템 클립보드를 사용하기 위한 권한								
0x0020	클립보드 히스토리 사용하지 않도록 하기 위한 권한								
CreateSet(Method)	<p>Description ParameterSet을 생성한다.</p> <p>Declaration LPDISPATCH* CreateSet(BSTR setID);</p> <p>Parameters setID : 생성할 ParameterSet의 ID</p>								
GetHeadingString(Method)	<p>Description 글머리표/문단번호/개요번호를 추출한다.</p> <p>Parameters return : 문자열이 반환된다.</p> <ul style="list-style-type: none"> • 글머리표/문단번호/개요번호가 있다면 해당 문자열이 반환된다. <p>Remark</p> <ul style="list-style-type: none"> • 글머리표/문단번호/개요번호가 있는 경우, 해당 문자열을 얻어올 수 있다. • 문단에 글머리표/문단번호/개요번호가 없는 경우, 빈 문자열이 추출된다. 								
SetTitleName(Method)	<p>Description 사용자가 임의로 Title을 설정한다.</p> <p>Declaration void SetTitleName(BSTR title);</p> <p>Parameters title : 변경할 사용자 Title문자열</p>								
GetSelectedPos(Method)	<p>Description 현재 설정된 블록의 위치정보를 얻어온다.</p> <p>Declaration bool GetSelectedPos(long* slist, long* spara, long* spos, long* elist, long* epara, long* epos);</p> <p>Parameters</p> <ul style="list-style-type: none"> slist : 설정된 블록의 시작 리스트 아이디. spara : 설정된 블록의 시작 문단 아이디. spos : 설정된 블록의 문단 내 시작 글자 단위 위치. elist : 설정된 블록의 끝 리스트 아이디. epara : 설정된 블록의 끝 문단 아이디. epos : 설정된 블록의 문단 내 끝 글자 단위 위치. <p>Remark</p> <ul style="list-style-type: none"> • 위의 리스트란, 문단과 컨트롤들이 연결된 한글문서 내 구조를 뜻한다. • 리스트 아이디는 문서 내 위치정보중 하나로서 SelectText에 넘겨줄 때 사용한다. • 매개변수로 포인터를 사용하므로, 포인터를 사용할 수 없는 언어에서는 사용이 불가능하다. • 포인터를 사용하지 않는 언어를 지원하기 위해서 ParameterSet을 사용하는 GetSelectedPosBySet()이 존재한다. 								
GetSelectedPosBySet(Method)	<p>Description 현재 설정된 블록의 위치정보를 얻어온다. (GetSelectedPos의 ParameterSet버전)</p> <p>Declaration</p>								

	<pre>bool GetSelectedPosBySet(LPDISPATCH sset, LPDISPATCH eset);</pre> <p>Parameters</p> <p>sset : 설정된 블록의 시작 파라메터셋 (ListParaPos) eset : 설정된 블록의 끝 파라메터셋 (ListParaPos)</p>																								
SetPrivateInfoPassword(Method)	<p>Description 개인정보보호를 위한 암호를 등록한다.</p> <p>Declaration <code>bool SetPrivateInfoPassword(BSTR password);</code></p> <p>Parameters</p> <p><code>password</code> : 새 암호 <code>return</code> : true / false</p> <ul style="list-style-type: none"> 정상적으로 암호가 설정되면 true를 반환한다. 암호설정에 실패하면 false를 반환한다. false를 반환하는 경우는 다음과 같다 암호의 길이가 너무 짧거나 너무 길 때 (영문: 5~44자, 한글: 3~22자) 암호가 이미 설정되었음. 또는 암호가 이미 설정된 문서임 <p>Remark 개인정보 보호를 설정하기 위해서는 우선 개인정보 보호 암호를 먼저 설정해야 한다. 그러므로 개인정보 보호 함수를 실행하기 이전에 반드시 이 함수를 호출해야 한다.</p>																								
RegisterPrivateInfoPattern(Method)	<p>Description 개인정보의 패턴을 등록한다.</p> <p>Declaration <code>bool RegisterPrivateInfoPattern(long PrivateType, BSTR PrivatePattern);</code></p> <p>Parameters</p> <p><code>PrivateType</code> : 등록할 개인정보 유형. 다음의 값 중 하나다.</p> <table border="1"> <thead> <tr> <th>값</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0x0001</td> <td>전화번호</td> </tr> <tr> <td>0x0002</td> <td>주민등록번호</td> </tr> <tr> <td>0x0004</td> <td>외국인등록번호</td> </tr> <tr> <td>0x0008</td> <td>전자우편</td> </tr> <tr> <td>0x0010</td> <td>계좌번호</td> </tr> <tr> <td>0x0020</td> <td>신용카드번호</td> </tr> <tr> <td>0x0040</td> <td>IP 주소</td> </tr> <tr> <td>0x0080</td> <td>생년월일</td> </tr> <tr> <td>0x0100</td> <td>주소</td> </tr> <tr> <td>0x0200</td> <td>사용자 정의</td> </tr> <tr> <td>0x0400</td> <td>기타 - 기타는 등록할 수 없다</td> </tr> </tbody> </table> <p><code>PrivatePatterns</code> : 등록할 개인정보 패턴. 예를 들면 이런 형태로 입력한다.</p> <ul style="list-style-type: none"> (예) 주민등록번호 - "NNNNNNN-NNNNNNN" 한/글이 이미 정의하고 있는 패턴은 정의하면 안 된다. 함수를 여러 번 호출하는 것을 피하기 위해 패턴을 ";" 기호로 구분 반복해서 입력할 수 있도록 한다. <p>Remark</p> <ul style="list-style-type: none"> 찾아낼 개인정보 패턴을 정의한다. (찾아서 보호 기능에서 사용됨) 한/글이 이미 정의하고 있는 패턴은 새로 등록할 수 없다. <code>PrivateType</code>으로 등록할 개인정보 유형을 선택한 뒤 <code>PrivatePattern</code>으로 패턴을 정의하는 식이다. 한/글이 기본적으로 등록한 패턴은 새로 등록할 수 없다. 또한 이미 등록한 패턴 역시 등록이 실패하여 false 반환한다. 한글은 패턴을 등록할 수 있는 단어를 제한해두고 있다. 사용할 수 있는 문자는 'N', '-' , '.' 빙 칸 등이며, IP 주소 유형의 경우에만 	값	Description	0x0001	전화번호	0x0002	주민등록번호	0x0004	외국인등록번호	0x0008	전자우편	0x0010	계좌번호	0x0020	신용카드번호	0x0040	IP 주소	0x0080	생년월일	0x0100	주소	0x0200	사용자 정의	0x0400	기타 - 기타는 등록할 수 없다
값	Description																								
0x0001	전화번호																								
0x0002	주민등록번호																								
0x0004	외국인등록번호																								
0x0008	전자우편																								
0x0010	계좌번호																								
0x0020	신용카드번호																								
0x0040	IP 주소																								
0x0080	생년월일																								
0x0100	주소																								
0x0200	사용자 정의																								
0x0400	기타 - 기타는 등록할 수 없다																								

	<p>‘@’ 를 허용한다. 잘못된 문자를 사용하여 패턴을 등록하면 위와 마찬가지로 false를 반환한다.</p> <ul style="list-style-type: none"> 마지막으로 동일 유형에는 여러 패턴을 반복적으로 입력할 수 있다. 이때 함수를 한번만 호출할 수 있도록 ‘;’ 구분자를 두어 반복해서 등록한다. 																								
FindPrivateInfo(Method)	<p>Description 개인정보를 찾는다.</p> <p>Declaration <code>long FindPrivateInfo(long PrivateType, VARIANT PrivateString);</code></p> <p>Parameters PrivateTypes : 찾을 개인정보 유형. 다음의 값을 하나이상 조합한다.</p> <table border="1"> <thead> <tr> <th>값</th><th>Description</th></tr> </thead> <tbody> <tr><td>0x0001</td><td>전화번호</td></tr> <tr><td>0x0002</td><td>주민등록번호</td></tr> <tr><td>0x0004</td><td>외국인등록번호</td></tr> <tr><td>0x0008</td><td>전자우편</td></tr> <tr><td>0x0010</td><td>계좌번호</td></tr> <tr><td>0x0020</td><td>신용카드번호</td></tr> <tr><td>0x0040</td><td>IP 주소</td></tr> <tr><td>0x0080</td><td>생년월일</td></tr> <tr><td>0x0100</td><td>주소</td></tr> <tr><td>0x0200</td><td>사용자 정의</td></tr> <tr><td>0x0400</td><td>기타</td></tr> </tbody> </table> <p>PrivateString : 기타 문자열. 0x0400 유형이 존재할 경우에만 유효하므로, 생략가능하다</p> <p>return : 찾은 개인정보의 유형 값. 값의 의미는 위 표와 동일하다.</p> <ul style="list-style-type: none"> 개인정보가 없는 경우에는 0을 반환한다. 또한, 검색 중 문서의 끝(end of document)을 만나면 -1을 반환한다. 이는 함수가 무한히 반복하는 것을 막아준다. <p>Remark</p> <ul style="list-style-type: none"> 개인 정보 유형으로 등록된 정보를 찾아 이동한다. 동작방식은 한/글의 찾기와 비슷하다. 함수가 호출되면 현재 캐럿 위치에서부터 개인정보를 검색하기 시작한다. 개인정보가 검색되면 검색된 개인정보를 블록 선택한다. 만약 문서의 끝까지 개인정보가 검색되지 않을 경우에는 함수는 -1을 반환하여 무한으로 검색하는 것을 막는다. PrivateTypes는 검색할 개인정보 유형을 나타낸다. 단 하나의 유형을 검색할 수도 있지만 여러 유형을 bit flag로 조합해서 검색할 수 있다. 예를 들면, 주민등록번호와 외국인등록번호를 한꺼번에 검색 할 수 있다. PrivateTypes에 0(NULL)을 줄 수 있는데 이것은 기타유형을 제외한 모든 유형을 검색하게 한다. (0x03FF 값을 입력한 것과 동일함) PrivateTypes의 값 중 0x400 값은 단어를 직접 검색할 때 사용한다. 0x400 값은 뒤의 PrivateString과 조합되어 사용될 수 있으며, 다음과 같이 사용된다. 예) 신용카드번호 또는 단어 “신한카드”를 직접 검색 FindPrivateInfo(0x20+0x400, “신한카드”); FindPrivateInfo() 함수는 리턴값으로 현재 선택된 개인정보의 유형을 반환한다. 사용자는 리턴값을 확인함으로 선택된 개인정보의 유형을 확인할 수 있으며, 선택적으로 정보보호가 가능하다. 함수는 문서 내 개인정보가 없을 경우에는 0을 반환한다. 그리고, 문서의 끝에 도달한 경우에는 -1을 반환한다. 이 두 값은 이전 리턴 값보다 의미상 중요한데, 이유는 루프문의 탈출코드로 사용되기 때문이다. 	값	Description	0x0001	전화번호	0x0002	주민등록번호	0x0004	외국인등록번호	0x0008	전자우편	0x0010	계좌번호	0x0020	신용카드번호	0x0040	IP 주소	0x0080	생년월일	0x0100	주소	0x0200	사용자 정의	0x0400	기타
값	Description																								
0x0001	전화번호																								
0x0002	주민등록번호																								
0x0004	외국인등록번호																								
0x0008	전자우편																								
0x0010	계좌번호																								
0x0020	신용카드번호																								
0x0040	IP 주소																								
0x0080	생년월일																								
0x0100	주소																								
0x0200	사용자 정의																								
0x0400	기타																								
ProtectPrivateInfo(Method)	<ul style="list-style-type: none"> Description 개인정보를 보호한다. 																								

- **Declaration**
- bool ProtectPrivateInfo(BSTR ProtectingChar, VARIANT PrivatePatternType);
- **Parameters**
- ProtectingChar : 보호문자. 개인정보는 해당문자로 가려진다.
- ProtectingType : 보호유형. 개인정보 유형마다 설정할 수 있는 값이 다르다.
- 0값은 기본 보호유형으로 모든 개인정보를 보호문자로 보호한다.
- 개인정보 유형에 따른 보호유형은 다음과 같다

개인정보 유형	보호 유형	보호유형 형태
전화번호 (0x0001)	0	*****
	1	NNN-****-NNNN
	2	NNN-NNN-***
주민등록번호 (0x0002)	0	*****
	1	NNNNNN-*****
	2	NNNNNN-N*****
	3	NNNNNN-N*****N
외국인등록번호 (0x0004)	0	*****
	1	NNNNNN-*****
	2	NNNNNN-N*****
	3	NNNNNN-N*****N
전자우편 (0x0008)	0	*****
	1	***@TT.TT.TT
	2	T**@TT.TT.TT
	3	TT*@TT.TT.TT
	4	TTT@***.*.*.***
계좌번호 (0x0010)	0	*****
	1	NNN-***-***-NNN
	2	***-NN-NNNN-***
	3	NNN-**-NNNN-***
	4	NNN-NN-****-***
신용카드번호 (0x0020)	0	*****
	1	NNNN-****-****-NNNN
	2	NNNN-****-NNNN-****
	3	****-NNNN-NNNN-****
	4	NNNN-NNNN-****-****
IP 주소 (0x0040)	0	*****
	1	NNN.*.*.N
	2	NNN.*.N.*
	3	***.N.N.*
생년월일 (0x0080)	0	*****
	1	NNNN-**-NN
	2	NNNN-***-**

	3	NNNN-NN-**
	<ul style="list-style-type: none"> • 이 값은 생략 가능하며, 생략할 경우 0(기본 보호유형)값을 설정한 것과 동일하게 동작한다. • 사용자가 선택한 글자를 보호하는 경우(선택 글자 보호 기능) 이 값은 항상 0이다. <p>return : true / false</p> <ul style="list-style-type: none"> • 개인정보를 보호문자로 치환한 경우에 true를 반환한다. • 개인정보를 보호하지 못할 경우 false를 반환한다. • 문자열이 선택되지 않은 상태이거나, 개체가 선택된 상태에서는 실패한다. • 또한, 보호유형이 잘못된 설정된 경우에도 실패한다. <p>Remark</p> <ul style="list-style-type: none"> • 선택된 문자열을 보호한다. • 한/글의 경우 “찾아서 보호” 와 “선택 글자 보호” 를 다른 기능으로 구현되었지만, API에서는 하나의 함수로 구현한다. • ProtectingChar는 개인정보를 보호할 때 개인정보 대신 입력될 문자이다. 어떤 문자라도 상관없다. • ProtectingType은 보호유형으로 개인정보의 전체를 보호할지 부분적으로 보호할지를 정해주는 값이다. 보호유형은 개인정보 유형에 따라 각각 다르다. (단, 기본 보호유형은 모든 보호유형에서 동일한 값(0)을 가진다) • ProtectingType은 생략이 가능한데 생략할 경우에 기본 보호유형이 설정된다. • ProtectPrivateInfo() 함수는 FindPrivateInfo() 함수로 선택된 문자열뿐만 아니라 사용자가 임의로 선택한 문자열도 보호할 수 있다. 예를 들어 “RepeatFind” 액션으로 찾아 선택한 문자열의 경우에도 보호가 가능하다. • 개인정보 보호가 성공할 경우에는 true를 반환한다. • 문자열이 선택된 경우에는 대부분 성공하지만 이전에 보호암호를 설정하지 않은 경우에는 실패하게 된다. 또한, 문자열이 선택되지 않은 경우와 개인정보 유형과 맞지 않는 보호유형을 설정할 경우에도 실패하게 된다. 	
SolarToLunar (Method)	<p>Description 양력을 음력으로 변환</p> <p>Declaration</p> <pre>bool SolarToLunar(long sYear, long sMonth, long sDay, long* lYear, long* lMonth, long* lDay, VARIANT_BOOL* lLeap);</pre> <p>Parameters</p> <ul style="list-style-type: none"> sYear : 양력 년 sMonth : 양력 월 sDay : 양력 일 lYear : 음력으로 반환된 년 lMonth : 음력으로 반환된 월 lDay : 음력으로 반환된 일 lLeap : 윤달 <p>Remark</p> <ul style="list-style-type: none"> • 1841~2043년 사이만 변환 가능 • 1841년 1월 23일 이전 날짜는 변환 불가 	
SolarToLunarBySet (Method)	<p>Description SolarToLunar 의 ParameterSet 버전</p> <p>Declaration</p> <pre>LPDISPATCH* SolarToLunarBySet(long sYear, long sMonth, long sDay);</pre> <p>Parameters</p> <ul style="list-style-type: none"> sYear : 양력 년 sMonth : 양력 월 sDay : 양력 일 	

	<p>return : ParameterSet형식으로 반환</p> <table border="1"> <thead> <tr> <th>Item ID</th><th>Type</th><th>Description</th></tr> </thead> <tbody> <tr> <td>Year</td><td>PIT_UI4</td><td>년</td></tr> <tr> <td>Month</td><td>PIT_UI4</td><td>월</td></tr> <tr> <td>Day</td><td>PIT_UI4</td><td>일</td></tr> <tr> <td>Leap</td><td>PIT_UI1</td><td>윤달인지 아닌지</td></tr> </tbody> </table> <p>Remark</p> <ul style="list-style-type: none"> • 1841~2043년 사이만 변환 가능 • 1841년 1월 23일 이전 날짜는 변환 불가 	Item ID	Type	Description	Year	PIT_UI4	년	Month	PIT_UI4	월	Day	PIT_UI4	일	Leap	PIT_UI1	윤달인지 아닌지
Item ID	Type	Description														
Year	PIT_UI4	년														
Month	PIT_UI4	월														
Day	PIT_UI4	일														
Leap	PIT_UI1	윤달인지 아닌지														
LunarToSolar (Method)	<p>Description 음력을 양력으로 변환</p> <p>Declaration</p> <pre>bool LunarToSolar(long lYear, long lMonth, long lDay, VARIANT_BOOL lLeap, long* sYear, long* sMonth, long* sDay);</pre> <p>Parameters</p> <ul style="list-style-type: none"> lYear : 음력 년 lMonth : 음력 월 lDay : 음력 일 lLeap : 윤달 sYear : 양력으로 반환된 년 sMonth : 양력으로 반환된 월 sDay : 양력으로 반환된 일 <p>Remark</p> <ul style="list-style-type: none"> • 1841~2043년 사이만 변환 가능 • 입력한 날이 그 달에 포함되지 않으면 실패 															
LunarToSolarBySet (Method)	<p>Description LunarToSolar의 ParameterSet버전</p> <p>Declaration</p> <pre>LPDISPATCH* LunarToSolarBySet(long lYear, long lMonth, long lDay, VARIANT_BOOL lLeap);</pre> <p>Parameters</p> <ul style="list-style-type: none"> lYear : 음력 년 lMonth : 음력 월 lDay : 음력 일 lLeap : 윤달 <p>return : ParameterSet형식으로 반환</p> <table border="1"> <thead> <tr> <th>Item ID</th><th>Type</th><th>Description</th></tr> </thead> <tbody> <tr> <td>Year</td><td>PIT_UI4</td><td>년</td></tr> <tr> <td>Month</td><td>PIT_UI4</td><td>월</td></tr> <tr> <td>Day</td><td>PIT_UI4</td><td>일</td></tr> </tbody> </table> <p>Remark</p> <ul style="list-style-type: none"> • 1841~2043년 사이만 변환 가능 • 입력한 날이 그 달에 포함되지 않으면 실패 	Item ID	Type	Description	Year	PIT_UI4	년	Month	PIT_UI4	월	Day	PIT_UI4	일			
Item ID	Type	Description														
Year	PIT_UI4	년														
Month	PIT_UI4	월														
Day	PIT_UI4	일														
ScanFont (Method)	<p>Description GetFontList를 호출하기 이전에 필수적으로 호출해야 한다</p>															
GetFontList (Method)	<p>Description 현재 문서에서 사용된 글꼴 목록을 문자열 형태로 반환</p> <p>Declaration</p> <pre>BSTR GetFontList(VARIANT langid);</pre> <p>Parameters</p> <ul style="list-style-type: none"> langid : 글꼴 언어 															

	<table border="1"> <thead> <tr> <th>LangID</th><th>Description</th></tr> </thead> <tbody> <tr><td>0</td><td>한글</td></tr> <tr><td>1</td><td>영문</td></tr> <tr><td>2</td><td>한자</td></tr> <tr><td>3</td><td>일어</td></tr> <tr><td>4</td><td>외국어</td></tr> <tr><td>5</td><td>기호</td></tr> <tr><td>6</td><td>사용자</td></tr> </tbody> </table> <p>Remark 언어온 글꼴 목록은 Wx02 구분되어 있다.</p>	LangID	Description	0	한글	1	영문	2	한자	3	일어	4	외국어	5	기호	6	사용자
LangID	Description																
0	한글																
1	영문																
2	한자																
3	일어																
4	외국어																
5	기호																
6	사용자																
ReplaceFont(Method)	<p>Description 문서에 적용된 글꼴을 변경한다</p> <p>Declaration <code>bool ReplaceFont(long langid, BSTR desFontName, long desFontType, BSTR newFontName, long newFontType);</code></p> <p>Parameters</p> <ul style="list-style-type: none"> langid : 글꼴 언어(위 GetFontList참고) desFontName : 대상 글꼴 이름 desFontType : 대상 글꼴 타입 (TTF or HFT) newFontName : 변경 할 글꼴 이름 newFontType : 변경 할 글꼴 타입 (TTF or HFT) 																
ReleaseAction(Method)	<p>Description 생성한 액션을 명시적으로 릴리즈할 필요가 있을 경우에 사용.</p> <p>Declaration <code>void ReleaseAction(LPDISPATCH action);</code></p> <p>Parameters</p> <ul style="list-style-type: none"> action : ActionObject.hwp 참고 																
SetUserInfo(Method)	<p>Description 환경설정 - 일반 - 사용자 정보 변경</p> <p>Declaration <code>void SetUserInfo(long userInfold, BSTR value);</code></p> <p>Parameters</p> <ul style="list-style-type: none"> userInfold : 변경할 정보 <table border="1"> <thead> <tr> <th>userInfold</th><th>Description</th></tr> </thead> <tbody> <tr><td>0</td><td>사용자 이름</td></tr> <tr><td>1</td><td>회사 이름</td></tr> <tr><td>2</td><td>직책 이름</td></tr> <tr><td>3</td><td>부서 이름</td></tr> </tbody> </table> <p>value : 입력할 정보</p>	userInfold	Description	0	사용자 이름	1	회사 이름	2	직책 이름	3	부서 이름						
userInfold	Description																
0	사용자 이름																
1	회사 이름																
2	직책 이름																
3	부서 이름																
GetUserInfo(Method)	<p>Description 사용자 정보에 해당하는 값을 얻어온다</p> <p>Declaration <code>BSTR GetUserInfo(long userInfold);</code></p> <p>Parameters</p> <ul style="list-style-type: none"> userInfold : 변경할 정보. 위 SetUserInfo참고 return : userInfold에 해당하는 정보 																
FileTranslate(Method)	<p>Description 문서를 번역해서 적용시켜준다</p> <p>Declaration <code>bool FileTranslate(BSTR curLang, BSTR transLang);</code></p> <p>Parameters</p>																

	<p>curLang : 현재 문서에 적용된 언어</p> <table border="1"> <thead> <tr> <th>curLang</th><th>Description</th></tr> </thead> <tbody> <tr> <td>ko</td><td>한국어(대한민국)</td></tr> <tr> <td>en</td><td>영어(미국)</td></tr> <tr> <td>vi</td><td>베트남어(베트남)</td></tr> <tr> <td>ja</td><td>일본어(일본)</td></tr> <tr> <td>zh-cn</td><td>중국어(간체, PRC)</td></tr> </tbody> </table> <p>transLang : 번역할 대상 언어</p> <p>Remark 한국어에서 다른 언어로 번역은 가능하지만, 현재 다른언어에서의 번역은 한국어밖에 지원하지 않음.</p>	curLang	Description	ko	한국어(대한민국)	en	영어(미국)	vi	베트남어(베트남)	ja	일본어(일본)	zh-cn	중국어(간체, PRC)
curLang	Description												
ko	한국어(대한민국)												
en	영어(미국)												
vi	베트남어(베트남)												
ja	일본어(일본)												
zh-cn	중국어(간체, PRC)												
GetTranslateLangList(Method)	<p>Description 번역 가능한 언어의 목록을 반환한다</p> <p>Declaration BSTR GetTranslateLangList(BSTR curLang);</p> <p>Parameters curLang : 현재 문서에 적용된 언어 return : 현재 언어로 번역 가능한 목록을 문자열로 반환</p>												
SetCurMetatagName(Method)	<p>Description 선택된 개체나 블록, 셀 또는 문서정보에 메타태그 이름을 설정한다</p> <p>Declaration bool SetCurMetatagName(BSTR tag);</p> <p>Parameters tag : 설정할 태그 이름</p> <p>Remark 한/글 2022 부터 지원</p>												
GetCurMetatagName(Method)	<p>Description 현재 메타태그의 이름을 가져온다</p> <p>Declaration BSTR GetCurMetatagName();</p> <p>Remark 한/글 2022 부터 지원</p>												
RenameMetatag(Method)	<p>Description 현재 메타태그의 이름을 변경한다</p> <p>Declaration bool SetCurMetatagName(BSTR tag);</p> <p>Parameters tag : 설정할 태그 이름</p> <p>Remark 한/글 2022 부터 지원</p>												
MetatagExist(Method)	<p>Description 메타태그가 등록되어 있는지 확인한다.</p> <p>Declaration bool MetatagExist(BSTR tag);</p> <p>Parameters tag : 태그 이름</p> <p>Remark 한/글 2022 부터 지원</p>												
MoveToMetatag(Method)	<p>Description 지정한 메타태그로 캐럿을 이동한다.</p> <p>Declaration bool MoveToMetatag(BSTR tag, VARIANT text, VARIANT start, VARIANT</p>												

	<p><code>select);</code></p> <p>Parameters</p> <p><code>tag</code> : 태그 이름. <code>GetMetatagNameText/PutMetatagNameText</code>과 같은 형식으로 이름 뒤에 '{{#}}'로 번호를 지정할 수 있다.</p> <p><code>text</code> : true/false. 메타태그가 내부에 텍스트가 있는 경우 텍스트 안으로 이동할지 지정</p> <p><code>start</code> : true/false. 메타태그의 시작으로 이동할지 끝으로 이동할지 지정</p> <p><code>select</code> : true/false. 메타태그의 내용을 블록으로 선택할지, 캐럿만 이동할지 지정</p> <p>Remark</p> <p>한/글 2022 부터 지원</p>				
<code>GetMetatagNameText</code> (Method)	<p>Description</p> <p>지정한 이름의 메타태그 내용을 가져온다</p> <p>Declaration</p> <p><code>BSTR GetMetatagNameText(BSTR tag);</code></p> <p>Parameters</p> <p><code>tag</code> : 텍스트를 구할 메타태그 이름의 리스트. 다음과 같이 태그 사이를 문자 코드 0x02로 구분하여 한 번에 여러 개의 태그를 지정할 수 있다.</p> <ul style="list-style-type: none"> • “태그이름#1Wx2태그이름#2Wx2...태그이름#n” • 지정한 태그 이름이 문서 중에 두 개 이상 존재할 때의 표현 방식은 다음과 같다. <table border="1"> <tr> <td>태그이름</td><td>이름의 태그 중 첫 번째</td></tr> <tr> <td>태그이름{{n}}</td><td>지정한 이름의 태그 중 n 번째</td></tr> </table> <p>return :</p> <ul style="list-style-type: none"> • 예를 들어 "제목{{1}}Wx2본문Wx2이름{{0}}" 과 같이 지정하면 '제목'이라는 이름의 태그 중 두 번째, '본문'이라는 이름의 태그 중 첫 번째, '이름'이라는 태그 중 첫 번째를 각각 지정한다. 즉, '태그이름'과 '태그이름0'은 동일한 의미로 해석된다. • 텍스트 데이터가 돌아온다. 텍스트에서 탭은 'Wt'(0x9), 문단 바꿈은 CR/LF(0xD/0xA)로 표현되며, 이외의 특수 코드는 포함되지 않는다. • 태그 텍스트의 끝은 0x02로 표현되며, 그 이후 다음 태그의 텍스트가 연속해서 지정한 태그 리스트의 개수만큼 위치한다. 지정한 이름의 태그가 없거나 사용자가 해당 태그에 아무 텍스트도 입력하지 않았으면 해당 텍스트에는 빈 문자열이 돌아온다. <p>Remark</p> <p>한/글 2022 부터 지원</p>	태그이름	이름의 태그 중 첫 번째	태그이름{{n}}	지정한 이름의 태그 중 n 번째
태그이름	이름의 태그 중 첫 번째				
태그이름{{n}}	지정한 이름의 태그 중 n 번째				
<code>PutMetatagNameText</code> (Method)	<p>Description</p> <p>지정한 메타태그의 내용을 채운다.</p> <p>Declaration</p> <p><code>void PutMetatagNameText(BSTR tag, BSTR text);</code></p> <p>Parameters</p> <p><code>tag</code> : 내용을 채울 메타태그 이름의 리스트. 한 번에 여러 개의 태그를 지정할 수 있으며, 형식은 <code>GetMetatagNameText</code>와 동일하다. 다만 태그 이름 뒤에 '#'로 번호를 지정하지 않으면 해당 이름을 가진 모든 태그에 동일한 텍스트를 채워 넣는다. 즉, <code>PutMetatagNameText</code>에서는 '태그이름'과 '태그이름0'의 의미가 다르다.</p> <p><code>text</code> : 메타태그에 채워 넣을 문자열의 리스트. 형식은 <code>tag</code>와 동일하게 태그의 개수만큼 텍스트를 0x02로 구분하여 지정한다.</p> <p>Remark</p> <p>한/글 2022 부터 지원</p>				
<code>GetMetatagList</code> (Method)	<p>Description</p> <p>문서에 적용된 메타태그 목록을 가져온다</p> <p>Declaration</p> <p><code>BSTR GetMetatagList(VARIANT number, VARIANT option);</code></p>				

	<p>Parameters</p> <p>number : 문서 중에서 동일한 이름의 태그가 여러 개 존재할 때 이를 구별하기 위한 식별 방법을 지정한다.</p> <table border="1"> <thead> <tr> <th>ID</th><th>값</th><th>설명</th></tr> </thead> <tbody> <tr> <td>hwpMetatagPlain</td><td>0</td><td>아무 기호 없이 순서대로 태그 이름이 나열된다.</td></tr> <tr> <td>hwpMetatagNumber</td><td>1</td><td>태그 이름 뒤에 일련 번호가 {{#}}와 같은 형식으로 붙는다</td></tr> <tr> <td>hwpMetatagCount</td><td>2</td><td>태그 이름 뒤에 그 이름의 태그가 몇 개 있는지 {{#}}와 같은 형식으로 붙는다.</td></tr> </tbody> </table> <p>option : 태그를 가져올 범위. 아래 값을 더해서 범위를 지정할 수 있다.</p> <table border="1"> <thead> <tr> <th>ID</th><th>값</th><th>설명</th></tr> </thead> <tbody> <tr> <td>hwpMetatagPara</td><td>1</td><td>셀렉션 내에 부여된 태그 리스트만을 구한다.</td></tr> <tr> <td>hwpMetatagCell</td><td>2</td><td>셀에 부여된 태그 리스트만을 구한다.</td></tr> <tr> <td>hwpMetatagTable</td><td>4</td><td>표에 부여된 태그 리스트만을 구한다.</td></tr> <tr> <td>hwpMetatagClickHere</td><td>8</td><td>누름틀에 부여된 태그 리스트만을 구한다.</td></tr> <tr> <td>hwpMetatagShapeObject</td><td>16</td><td>도형에 부여된 태그 리스트만을 구한다.</td></tr> <tr> <td>hwpMetatagPicture</td><td>32</td><td>그림에 부여된 태그 리스트만을 구한다.</td></tr> </tbody> </table> <p>return : 각 태그 사이를 문자코드 0x02로 구분하여 다음과 같은 형식으로 리턴 한다. (가장 마지막 태그에는 0x02가 붙지 않는다.) "태그이름#1Wx2태그이름#2Wx2...태그이름#n"</p> <p>Remark 문서 중에 동일한 이름의 태그가 여러 개 존재할 때는 number에 지정한 탑입에 따라 3 가지의 서로 다른 방식을 중에서 선택할 수 있다. 한/글 2022 부터 지원</p>	ID	값	설명	hwpMetatagPlain	0	아무 기호 없이 순서대로 태그 이름이 나열된다.	hwpMetatagNumber	1	태그 이름 뒤에 일련 번호가 {{#}}와 같은 형식으로 붙는다	hwpMetatagCount	2	태그 이름 뒤에 그 이름의 태그가 몇 개 있는지 {{#}}와 같은 형식으로 붙는다.	ID	값	설명	hwpMetatagPara	1	셀렉션 내에 부여된 태그 리스트만을 구한다.	hwpMetatagCell	2	셀에 부여된 태그 리스트만을 구한다.	hwpMetatagTable	4	표에 부여된 태그 리스트만을 구한다.	hwpMetatagClickHere	8	누름틀에 부여된 태그 리스트만을 구한다.	hwpMetatagShapeObject	16	도형에 부여된 태그 리스트만을 구한다.	hwpMetatagPicture	32	그림에 부여된 태그 리스트만을 구한다.
ID	값	설명																																
hwpMetatagPlain	0	아무 기호 없이 순서대로 태그 이름이 나열된다.																																
hwpMetatagNumber	1	태그 이름 뒤에 일련 번호가 {{#}}와 같은 형식으로 붙는다																																
hwpMetatagCount	2	태그 이름 뒤에 그 이름의 태그가 몇 개 있는지 {{#}}와 같은 형식으로 붙는다.																																
ID	값	설명																																
hwpMetatagPara	1	셀렉션 내에 부여된 태그 리스트만을 구한다.																																
hwpMetatagCell	2	셀에 부여된 태그 리스트만을 구한다.																																
hwpMetatagTable	4	표에 부여된 태그 리스트만을 구한다.																																
hwpMetatagClickHere	8	누름틀에 부여된 태그 리스트만을 구한다.																																
hwpMetatagShapeObject	16	도형에 부여된 태그 리스트만을 구한다.																																
hwpMetatagPicture	32	그림에 부여된 태그 리스트만을 구한다.																																
CurMetatagState(Property)	<p>Description 캐럿이 위치한 메타태그의 상태 정보를 구한다.(읽기전용)</p> <p>Declaration <code>long CurMetatagState();</code></p> <p>Parameters</p> <p>return : 얻어지는 내용은 일종의 flag값으로 비트가 set되어 있으면 다음과 같은 의미를 가진다.</p> <table border="1"> <thead> <tr> <th>value</th><th>Description</th></tr> </thead> <tbody> <tr> <td>0x01</td><td>셀</td></tr> <tr> <td>0x02</td><td>누름틀</td></tr> <tr> <td>0x04</td><td>글상자</td></tr> <tr> <td>0x08</td><td>도형</td></tr> <tr> <td>0x10</td><td>표</td></tr> <tr> <td>0x20</td><td>태그명 없음</td></tr> <tr> <td>0x40</td><td>문단</td></tr> </tbody> </table> <p>Remark 한/글 2022 부터 지원</p>	value	Description	0x01	셀	0x02	누름틀	0x04	글상자	0x08	도형	0x10	표	0x20	태그명 없음	0x40	문단																	
value	Description																																	
0x01	셀																																	
0x02	누름틀																																	
0x04	글상자																																	
0x08	도형																																	
0x10	표																																	
0x20	태그명 없음																																	
0x40	문단																																	
ModifyMetatagProperties(Method)	<p>Description 지정한 메타태그의 속성을 바꾼다.</p> <p>Declaration <code>long ModifyMetatagProperties(BSTR tag, long remove, long add)</code></p> <p>Parameters</p>																																	

	<p>tag : 속성을 바꿀 태그 이름의 리스트. 형식은 PutMetatagNameText과 동일.</p> <p>remove : 제거될 속성</p> <p>add : 추가될 속성</p> <p>return : 음수가 리턴되면 에러임을 나타낸다.</p> <p>Remark 속성의 값은 아래와 같다.</p> <table border="1"> <thead> <tr> <th>long value</th><th>설명</th></tr> </thead> <tbody> <tr> <td>0x00000001</td><td>양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)</td></tr> </tbody> </table> <p>return 값의 bit field는 다음과 같다.</p> <table border="1"> <thead> <tr> <th>long value bit mask</th><th>설명</th></tr> </thead> <tbody> <tr> <td>0x00000001</td><td>양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)</td></tr> <tr> <td>0x80000000</td><td>에러</td></tr> <tr> <td>0x40000000</td><td>태그를 찾을 수 없음</td></tr> </tbody> </table> <p>remove와 add에 둘다 0이 입력되면 현재 속성을 돌려준다. 리턴값이 음수인지 확인하여 쉽게 에러임을 판별할 수 있으며 자세한 에러내용은 bit mask로 and 연산하여 알아 낼 수 있다. 한/글 2022 부터 지원</p>	long value	설명	0x00000001	양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)	long value bit mask	설명	0x00000001	양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)	0x80000000	에러	0x40000000	태그를 찾을 수 없음
long value	설명												
0x00000001	양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)												
long value bit mask	설명												
0x00000001	양식모드에서 편집가능 속성 (0: 편집 불가, 1: 편집 가능)												
0x80000000	에러												
0x40000000	태그를 찾을 수 없음												
IsTrackChange(Property)	<p>Description 변경내용추적 상태를 확인하고 설정한다</p> <p>Parameters 0 - 해제 1 - 설정</p>												
IsTrackChangePassword(Property)	<p>Description 변경내용추적 암호가 걸려 있는지 확인한다(읽기전용)</p>												
GetProperty(Method)	<p>Description 문서 정보 - 사용자 지정의 속성을 가져온다.</p> <p>Declaration BSTR GetUserProperty(BSTR name, long option);</p> <p>Parameters name : 속성을 가져올 이름 option : 0으로 고정(사용하지 않음)</p> <p>Remark 가져온 속성은 형식 + 값으로 되어 있으며, 문자코드 0x02로 구분된다.</p> <p>Remark 한/글 2022 부터 지원</p>												
SetProperty(Method)	<p>Description 문서 정보 - 사용자 지정 속성을 설정한다</p> <p>Declaration long SetProperty(BSTR name, long format, BSTR value, long option);</p> <p>Parameters name : 사용자가 지정할 속성의 이름을 직접 입력한다. format : 사용자가 지정할 속성의 데이터 형식을 목록에서 선택한다. 선택한 형식에 따라 입력할 수 있는 값이 달라진다.</p> <table border="1"> <thead> <tr> <th>value</th><th>Description</th></tr> </thead> <tbody> <tr> <td>0</td><td>텍스트+숫자</td></tr> <tr> <td>1</td><td>날짜</td></tr> </tbody> </table>	value	Description	0	텍스트+숫자	1	날짜						
value	Description												
0	텍스트+숫자												
1	날짜												

	<table border="1"> <tr><td>2</td><td>숫자</td></tr> <tr><td>3</td><td>예 또는 아니요</td></tr> </table> <p>value : 사용자가 선택한 형식에 맞는 속성 값을 입력한다. 선택한 형식에 따라 텍스트, 숫자, 또는 날짜를 입력하거나 '예' 또는 '아니요' 값을 할당할 수도 있다.</p> <p>option : 0으로 고정(사용하지 않음)</p> <p>Remark 한/글 2022 부터 지원</p>	2	숫자	3	예 또는 아니요																	
2	숫자																					
3	예 또는 아니요																					
GetUserPropertyNameList(Method)	<p>Description 문서 정보 - 사용자 지정 속성에 등록된 이름 목록을 가져온다</p> <p>Declaration BSTR GetUserPropertyNameList(long option);</p> <p>Parameters option : 0으로 고정(사용하지 않음)</p> <p>Remark 한/글 2022 부터 지원</p>																					
GetCtrlToPicture(Method)	<p>Description 선택한 개체를 이미지로 저장한다</p> <p>Declaration bool GetCtrlToPicture(BSTR path, long format, long fullSave);</p> <p>Parameters path : 이미지가 저장 될 경로 및 파일명(확장자를 포함해서 작성해야 한다) format : 이미지를 저장할 형식</p> <table border="1"> <thead> <tr> <th>Format</th><th>값</th><th>설명</th></tr> </thead> <tbody> <tr><td>EMF</td><td>16</td><td>EMF형식으로 저장한다</td></tr> <tr><td>BMP</td><td>20</td><td>BMP형식으로 저장한다</td></tr> <tr><td>GIF</td><td>22</td><td>GIF형식으로 저장한다 (그림일 경우에만 사용가능)</td></tr> <tr><td>JPG</td><td>25</td><td>JPG형식으로 저장한다 (그림일 경우에만 사용가능)</td></tr> <tr><td>PNG</td><td>27</td><td>PNG형식으로 저장한다 (그림일 경우에만 사용가능)</td></tr> <tr><td>TIF</td><td>29</td><td>TIF형식으로 저장한다 (그림일 경우에만 사용가능)</td></tr> </tbody> </table> <p>fullSave : 표가 2페이지 이상 넘어갔을 경우 모두 저장할지 여부</p> <p>Remark 한/글 2024 부터 지원</p>	Format	값	설명	EMF	16	EMF형식으로 저장한다	BMP	20	BMP형식으로 저장한다	GIF	22	GIF형식으로 저장한다 (그림일 경우에만 사용가능)	JPG	25	JPG형식으로 저장한다 (그림일 경우에만 사용가능)	PNG	27	PNG형식으로 저장한다 (그림일 경우에만 사용가능)	TIF	29	TIF형식으로 저장한다 (그림일 경우에만 사용가능)
Format	값	설명																				
EMF	16	EMF형식으로 저장한다																				
BMP	20	BMP형식으로 저장한다																				
GIF	22	GIF형식으로 저장한다 (그림일 경우에만 사용가능)																				
JPG	25	JPG형식으로 저장한다 (그림일 경우에만 사용가능)																				
PNG	27	PNG형식으로 저장한다 (그림일 경우에만 사용가능)																				
TIF	29	TIF형식으로 저장한다 (그림일 경우에만 사용가능)																				
SelectCtrl(Method)	<p>Description 캐럿이 위치한 메타태그의 상태 정보를 구한다.(읽기전용)</p> <p>Declaration bool SelectCtrl(BSTR ctrlList, long option);</p> <p>Parameters ctrlList: 선택할 컨트롤 ID. 문자코드 0x02를 이용해서 여러 개의 컨트롤 ID를 지정할 수 있다. <ul style="list-style-type: none"> 컨트롤 ID는 HwpCtrl.CtrlID가 리턴하는 ID와 동일하다. 자세한 것은 Ctrl 오브젝트 Properties인 CtrlID를 참조. option : 현재 선택된 컨트롤에서 추가로 선택을 할지, 취소 후 다시 선택할지에 대한 옵션. <ul style="list-style-type: none"> 0 = 추가로 선택 1 = 취소 후 ctrlList에 등록된 컨트롤만 선택 <p>Remark</p> </p>																					

IHwpObjectEvents(_DIHwpObjectEvents) : 한/글에서부터 발생되는 이벤트

Item Name	Return	Description
Quit	없음	한/글을 종료할 때 발생
CreateXHwpWindow	없음	한/글에서 새 문서 창을 열었을 때 발생
CloseXHwpWindow	없음	한/글에서 문서 창을 닫았을 때 발생
NewDocument	long	새 문서를 생성할 경우 발생(Document ID를 반환)
DocumentBeforeClose	long	문서를 닫기 직전에 발생(Document ID를 반환)
DocumentBeforeOpen	long	문서를 열기 직전에 발생(Document ID를 반환)
DocumentAfterOpen	long	문서를 열고 난 후에 발생(Document ID를 반환)
DocumentBeforeSave	long	문서를 저장하기 직전에 발생(Document ID를 반환)
DocumentAfterSave	long	문서를 저장한 후에 발생(Document ID를 반환)
DocumentAfterClose	long	문서를 닫고 난 후에 발생(Document ID를 반환)
DocumentChange	long	문서가 변경됐을 경우에 발생(Document ID를 반환)
DocumentBeforePrint	long	문서를 인쇄하기 직전에 발생(Document ID를 반환)
DocumentAfterPrint	long	문서를 인쇄하고 난 후에 발생(Document ID를 반환)
DocumentClickedHyperlink		하이퍼링크를 클릭했을 때 발생
DocumentModifiedHyperlink		하이퍼링크를 수정했을 때 발생
BeforeQuit	없음	한/글을 종료하기 직전에 발생

**IXHwpDocuments: IXHwpDocument(도큐먼트) 오브젝트를 관리하는 오브젝트
(Document를 관리하는 Collection 개체 - 사용 편의를 위해 제공됨)**

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Item(Property)	Description 지정한 원소의 도큐먼트 오브젝트를 얻어온다. Declaration LPDISPATCH Item(long index) Parameters index : 원소의 인덱스 return : 원소의 값(IXHwpDocument)
Count(Property)	Description 원소의 총개수
Active_XHwpDocument(Property)	Description 현재 활성화 상태인 도큐먼트 Object를 얻어온다.(IXHwpDocument)
Add(Method)	Description 도큐먼트 오브젝트를 추가한다. Declaration LPDISPATCH Add(BOOL isTab) Parameters isTab : TRUE = 새탭으로 열리는 도큐먼트, FALSE = 새창으로 열리는 도큐먼트 return : 열리게 되는 도큐먼트(IXHwpDocument)

<p>Close(Method)</p>	<p>Description 관리하고 있는 도큐먼트 오브젝트를 삭제한다.</p> <p>Declaration</p> <pre>void Close(BOOL isDirty)</pre> <p>Parameters isDirty : TRUE이면 변경된 문서는 닫지 않는다. FALSE이면 변경된 문서도 닫는다.</p>
<p>FindItem(Method)</p>	<p>Description 도큐먼트 아이디로 지정된 도큐먼트 오브젝트를 얻는다.</p> <p>Declaration</p> <pre>LPDISPATCH FindItem(long Docid)</pre> <p>Parameters Docid : 도큐먼트의 고유 ID return : 도큐먼트 ID에 해당하는 도큐먼트 오브젝트(IXHwpDocument)</p>

IXHwpDocument: 도큐먼트 오브젝트 (Document 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Application(Property)	<p>Description 최상위 오브젝트를 얻어옴(IHwpObject – 읽기 전용)</p>
Path(Property)	<p>Description 도큐먼트의 Path를 얻어옴(읽기 전용)</p>
FullName(Property)	<p>Description 도큐먼트의 전체 경로를 얻어옴(읽기 전용)</p>
EditMode(Property)	<p>Description 도큐먼트의 에디트 모드를 설정하거나/얻어옴</p>
Modified(Property)	<p>Description 도큐먼트의 변경 여부를 설정하거나/얻어옴</p>
Format(Property)	<p>Description 도큐먼트의 저장된 포맷을 얻어옴(읽기 전용)</p>
Password(Property)	<p>Description 도큐먼트의 패스워드를 설정(쓰기 전용)</p>
XHwpSummaryInfo(Property)	<p>Description IXHwpSummaryInfo 문서 요약 정보 오브젝트 (읽기 전용)</p>
XHwpDocumentInfo(Property)	<p>Description IXHwpDocumentInfo 문서 정보 오브젝트 (읽기 전용)</p>
XHwpPrint(Property)	<p>Description IXHwpPrint 프린트 오브젝트 (읽기 전용)</p>
XHwpRange(Property)	<p>Description IXHwpRange Range 오브젝트 (읽기 전용)</p>
XHwpFind(Property)	<p>Description IXHwpFind 찾기 오브젝트 (읽기 전용)</p>
XHwpSelection(Property)	<p>Description IXHwpSelection 블록 선택 오브젝트 (읽기 전용)</p>
XHwpFormPushButtons(Property)	<p>Description</p>

	Description IXHwpFormPushButtons 양식개체 푸쉬버튼을 관리하는 오브젝트(읽기 전용)
XHwpFormCheckButtons(Property)	Description IXHwpFormCheckButtons 양식개체 체크박스를 관리하는 오브젝트(읽기 전용)
XHwpFormRadioButtons(Property)	Description IXHwpFormRadioButtons 양식개체 라디오버튼을 관리하는 오브젝트(읽기 전용)
XHwpFormComboBoxs(Property)	Description IXHwpFormComboBoxs 양식개체 콤보박스를 관리하는 오브젝트(읽기 전용)
XHwpFormEdits(Property)	Description IXHwpFormEdits 양식개체 에디트를 관리하는 오브젝트(읽기 전용)
XHwpCharacterShape(Property)	Description IXHwpCharacterShape 글자 모양 속성 오브젝트(읽기 전용)
XHwpParagraphShape(Property)	Description IXHwpParagraphShape 문단 모양 속성 오브젝트(읽기 전용)
XHwpSendMail(Property)	Description IXHwpSendMail 메일 보내기 오브젝트 (읽기 전용)
DocumentID(Property)	Description 도큐먼트의 고유 ID(읽기 전용)
Close(Method)	Description 문서를 닫는다. Declaration BOOL Close(BOOL isDirty) Parameters isDirty : TRUE = 문서 내용이 변경된 상태면 문서를 닫지 않는다./ FALSE = 문서 내용이 변경되었어도 강제로 문서를 닫는다.
Save(Method)	Description 문서를 저장한다. Declaration BOOL Save(Variant save_if_dirty) Parameters save_if_dirty : True를 지정하면 문서가 변경된 경우에만 저장한다. False를 지정하면 변경 여부에 관계없이 무조건 저장한다. 생략하면 True가 지정된다. Remark 문서의 경로가 지정되어 있지 않으면 "새이름으로 저장" 대화상자가 떠서 사용자에게 경로를 묻는다.
SaveAs(Method)	Description 문서를 지정한 이름으로 저장한다. Declaration BOOL Save(BSTR path, VARIANT format, VARIANT arg) Parameters path : 문서 파일의 경로 format : 문서 형식. 별도 설명 참조. 생략하면 "HWP"가 지정된다. arg : 세부 옵션. 의미는 format에 지정한 파일 형식에 따라 다르다. 생략하면 빈 문자열이 지정된다. Remark format, arg의 일반적인 개념에 대해서는 Open 참조.

	<p>"HWP" 형식으로 파일 저장시 arg에 지정할 수 있는 옵션은 다음과 같다.</p> <table border="1"> <thead> <tr> <th>함수</th><th>인자 타입</th><th>기본 값 (Default)</th><th>설명</th></tr> </thead> <tbody> <tr> <td>lock</td><td>boolean</td><td>TRUE</td><td>저장한 후 해당 파일을 계속 오픈한 상태로 lock을 걸지 여부</td></tr> <tr> <td>backup</td><td>boolean</td><td>FALSE</td><td>백업 파일 생성 여부</td></tr> <tr> <td>compress</td><td>boolean</td><td>TRUE</td><td>압축 여부</td></tr> <tr> <td>fullSave</td><td>boolean</td><td>FALSE</td><td>스토리지 파일을 완전히 새로 생성하여 저장</td></tr> <tr> <td>prvImage</td><td>int</td><td>2</td><td>미리보기 이미지 (0=off, 1=BMP, 2=GIF)</td></tr> <tr> <td>prvText</td><td>int</td><td>1</td><td>미리보기 텍스트(0=off, 1=0)</td></tr> </tbody> </table>				함수	인자 타입	기본 값 (Default)	설명	lock	boolean	TRUE	저장한 후 해당 파일을 계속 오픈한 상태로 lock을 걸지 여부	backup	boolean	FALSE	백업 파일 생성 여부	compress	boolean	TRUE	압축 여부	fullSave	boolean	FALSE	스토리지 파일을 완전히 새로 생성하여 저장	prvImage	int	2	미리보기 이미지 (0=off, 1=BMP, 2=GIF)	prvText	int	1	미리보기 텍스트(0=off, 1=0)
함수	인자 타입	기본 값 (Default)	설명																													
lock	boolean	TRUE	저장한 후 해당 파일을 계속 오픈한 상태로 lock을 걸지 여부																													
backup	boolean	FALSE	백업 파일 생성 여부																													
compress	boolean	TRUE	압축 여부																													
fullSave	boolean	FALSE	스토리지 파일을 완전히 새로 생성하여 저장																													
prvImage	int	2	미리보기 이미지 (0=off, 1=BMP, 2=GIF)																													
prvText	int	1	미리보기 텍스트(0=off, 1=0)																													
Undo(Method)	<p>Description 문서에 기록된 Undo Item을 실행한다. Declaration BOOL Undo(long count) Parameters count : 아이템의 count까지 Undo한다.</p>																															
Redo(Method)	<p>Description 문서에 기록된 Redo Item을 실행한다. Declaration BOOL Redo(long count) Parameters count : 아이템의 count까지 Redo한다.</p>																															
Open(Method)	<p>Description 문서 파일을 연다. Declaration Open(BSTR path, BSTR format, BSTR arg) Parameters path : 문서 파일의 경로(URL 사용 가능) format : 문서 형식. 별도 설명 참조. 빈 문자열을 지정하면 자동으로 인식한다. 생략하면 빈 문자열이 지정된다. arg : 세부 옵션. 의미는 format에 지정한 파일 형식에 따라 다르다. 생략하면 빈 문자열이 지정된다. Remark format에 지정할 수 있는 문서 형식은 현재 시스템에 설치된 문서 필터(*.dft)의 종류에 따라 달라진다. 일반적으로 설치되는 형식들에는 다음과 같은 종류가 있다.</p> <table border="1"> <tr> <td>HWP</td><td>워디안 native format</td></tr> <tr> <td>HWP30</td><td>한글 3.X/96/97</td></tr> <tr> <td>HTML</td><td>인터넷 문서</td></tr> <tr> <td>TEXT</td><td>아스키 텍스트 문서</td></tr> <tr> <td>UNICODE</td><td>유니코드 텍스트 문서</td></tr> <tr> <td>HWP20</td><td>한글 2.0</td></tr> <tr> <td>HWP21</td><td>한글 2.1/2.5</td></tr> <tr> <td>HWP15</td><td>한글 1.X</td></tr> <tr> <td>HWPML1X</td><td>HWPML 1.X 문서 (Open만 가능)</td></tr> <tr> <td>HWPML2X</td><td>HWPML 2.X 문서 (Open / SaveAs 가능)</td></tr> <tr> <td>RTF</td><td>서식있는 텍스트 문서</td></tr> <tr> <td>DBF</td><td>DBASE II/III 문서</td></tr> <tr> <td>HUNMIN</td><td>훈민정음 3.0/2000</td></tr> </table>				HWP	워디안 native format	HWP30	한글 3.X/96/97	HTML	인터넷 문서	TEXT	아스키 텍스트 문서	UNICODE	유니코드 텍스트 문서	HWP20	한글 2.0	HWP21	한글 2.1/2.5	HWP15	한글 1.X	HWPML1X	HWPML 1.X 문서 (Open만 가능)	HWPML2X	HWPML 2.X 문서 (Open / SaveAs 가능)	RTF	서식있는 텍스트 문서	DBF	DBASE II/III 문서	HUNMIN	훈민정음 3.0/2000		
HWP	워디안 native format																															
HWP30	한글 3.X/96/97																															
HTML	인터넷 문서																															
TEXT	아스키 텍스트 문서																															
UNICODE	유니코드 텍스트 문서																															
HWP20	한글 2.0																															
HWP21	한글 2.1/2.5																															
HWP15	한글 1.X																															
HWPML1X	HWPML 1.X 문서 (Open만 가능)																															
HWPML2X	HWPML 2.X 문서 (Open / SaveAs 가능)																															
RTF	서식있는 텍스트 문서																															
DBF	DBASE II/III 문서																															
HUNMIN	훈민정음 3.0/2000																															

	<table border="1"> <tr><td>MSWORD</td><td>マイクロソフト 워드 문서</td></tr> <tr><td>HANA</td><td>하나워드 문서</td></tr> <tr><td>ARIRANG</td><td>아리랑 문서</td></tr> <tr><td>ICHITARO</td><td>一太郎 문서 (일본 워드프로세서)</td></tr> <tr><td>WPS</td><td>WPS 문서</td></tr> <tr><td>DOCIMG</td><td>인터넷 프레젠테이션 문서(SaveAs만 가능)</td></tr> <tr><td>SWF</td><td>Macromedia Flash 문서(SaveAs만 가능)</td></tr> </table>	MSWORD	マイクロソフト 워드 문서	HANA	하나워드 문서	ARIRANG	아리랑 문서	ICHITARO	一太郎 문서 (일본 워드프로세서)	WPS	WPS 문서	DOCIMG	인터넷 프레젠테이션 문서(SaveAs만 가능)	SWF	Macromedia Flash 문서(SaveAs만 가능)
MSWORD	マイクロソフト 워드 문서														
HANA	하나워드 문서														
ARIRANG	아리랑 문서														
ICHITARO	一太郎 문서 (일본 워드프로세서)														
WPS	WPS 문서														
DOCIMG	인터넷 프레젠테이션 문서(SaveAs만 가능)														
SWF	Macromedia Flash 문서(SaveAs만 가능)														
	<p>arg에 지정할 수 있는 옵션의 의미는 필터가 정의하기에 따라 다르지만, 신텍스는 다음과 같이 공통된 형식을 사용한다.</p> <p>key:value;key:value;...</p> <ul style="list-style-type: none"> * key는 A-Z, a-z, 0-9, _ 로 구성된다. * value는 타입에 따라 다음과 같은 3 종류가 있다. <ul style="list-style-type: none"> boolean: ex) fullsave:true (== fullsave) integer: ex) type:20 string: ex) prefix:_This_ * value는 생략 가능하며, 이때는 콜론도 생략한다. * arg에 지정할 수 있는 옵션 														
"모든 파일"	<p>setcurdir(boolean, FALSE) : 로드한 후 해당 파일이 존재하는 폴더로 현재 위치를 변경한다. hyperlink 정보가 상대적인 위치로 되어 있을 때 유용하다.</p>														
"HWP"	<ul style="list-style-type: none"> - lock (boolean, TRUE) : 로드한 후 해당 파일을 계속 오픈한 상태로 lock을 걸지 여부 - notext (boolean, FALSE) : 텍스트 내용을 읽지 않고 헤더 정보만 읽을지 여부. (스타일 로드 등에 사용) - template (boolean, FALSE): 새로운 문서를 생성하기 위해 템플릿 파일을 오픈한다. 이 옵션이 주어지면 lock은 무조건 FALSE로 처리된다. - suspendpassword (boolean, FALSE): TRUE로 지정하면, 암호가 있는 파일일 경우 암호를 묻지 않고 무조건 읽기에 실패한 것으로 처리한다. - forceopen (boolean, FALSE): TRUE로 지정하면, 읽기 전용으로 읽어야 하는 경우 대화상자를 띄우지 않는다. 														
"HTML"	<ul style="list-style-type: none"> - code(string, codepage) : 문서 변환시 사용되는 코드 페이지를 지정할 수 있으며 code키가 존재할 경우 필터 사용시 사용자ダイアル로그를 띄우지 않는다. - textunit(boolean, pixel) : Export될 Text의 크기의 단위 결정(pixel, point, mili 지정 가능.) - formatunit(boolean, pixel) : Export될 문서 포맷 관련 (마진, Object 크기 등) 단위 결정. pixel, point, mili 지정 가능 														
"DOCIMG"	<ul style="list-style-type: none"> - asimg(boolean, FALSE) : 저장할 때 페이지를 image로 저장 - ashtml(boolean, FALSE) : 저장할 때 페이지를 html로 저장 <p>[codepage 종류]</p> <p>ks : 한글 KS 완성형 kssm : 한글 조합형 sjis : 일본 utf8 : UTF8 unicode : 유니코드 gb : 중국 간체 big5 : 중국 번체 acp : Active Codepage 현재 시스템의 코드 페이지</p>														
"TEXT"	<ul style="list-style-type: none"> - code(string, codepage): 문서 변환시 사용되는 코드 페이지를 지정할 수 있으며 code키가 존재할 경우 필터 사용시 사용자ダイアル로그를 띄우지 않는다. 														

	<p>[codepage 종류]</p> <ul style="list-style-type: none"> - ks : 한글 KS 완성형 kssm : 한글 조합형 sjis : 일본 gb : 중국 간체 big5 : 중국 번체 acp : Active Codepage 현재 시스템의 코드 페이지 															
SendBrowser(Method)	<p>Description 문서를 브라우저로 내보내기 기능 Declaration BOOL SendBrowser (void)</p>															
SetActive_XHwpDocument(Method)	<p>Description 문서를 활성화 상태로 하기 Declaration void SetActive_XHwpDocument(void)</p>															
Clear(Method)	<p>Description 문서의 내용을 닫고 빈문서 편집 상태로 돌아간다. Declaration void Clear(Variant option) Parameters option : 편집중인 문서의 내용에 대한 처리 방법. 생략하면 hwpAskSave가 선택된다.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #ffffcc;"> <th>ID</th> <th>값</th> <th>설명</th> </tr> </thead> <tbody> <tr> <td>hwpAskSave</td> <td>0</td> <td>문서의 내용이 변경되었을 때 사용자에게 저장할지 묻는 대화상자를 띄운다.</td> </tr> <tr> <td>hwpDiscard</td> <td>1</td> <td>문서 내용을 버린다.</td> </tr> <tr> <td>hwpSaveIfDirty</td> <td>2</td> <td>문서가 변경된 경우 저장 한다.</td> </tr> <tr> <td>hwpSave</td> <td>3</td> <td>무조건 저장한다.</td> </tr> </tbody> </table> <p>Remark format, arg에 대해서는 Open 참조. hwpSaveIfDirty, hwpSave가 지정된 경우 현재 문서 경로가 지정되어 있지 않으면 "새이름으로 저장" 대화상자를 띄워 사용자에게 경로를 묻는다.</p>	ID	값	설명	hwpAskSave	0	문서의 내용이 변경되었을 때 사용자에게 저장할지 묻는 대화상자를 띄운다.	hwpDiscard	1	문서 내용을 버린다.	hwpSaveIfDirty	2	문서가 변경된 경우 저장 한다.	hwpSave	3	무조건 저장한다.
ID	값	설명														
hwpAskSave	0	문서의 내용이 변경되었을 때 사용자에게 저장할지 묻는 대화상자를 띄운다.														
hwpDiscard	1	문서 내용을 버린다.														
hwpSaveIfDirty	2	문서가 변경된 경우 저장 한다.														
hwpSave	3	무조건 저장한다.														

IXHwpFormPushButtons: IXHwpFormPushButton 오브젝트를 관리하는 오브젝트 (양식개체 PushButton을 관리하는 Collection 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Application(Property)	<p>Description 최상위 오브젝트를 얻어옴(IHwpObject)</p>
Item(Property)	<p>Description 지정한 원소의 양식개체 PushButton 오브젝트를 얻어온다. Declaration LPDISPATCH Item(long index) Parameters index : 원소의 인덱스 return : 원소의 값(IXHwpFormPushButton)</p>
Count(Property)	<p>Description 원소의 총개수</p>
ItemFromName(Property)	<p>Description 양식 개체 PushButton의 이름으로 원소를 찾는다.</p>

	<p>Declaration</p> <p>LPDISPATCH ItemFormName(BSTR name)</p> <p>Parameters</p> <p>name : 양식 개체 PushButton의 이름 return : 양식 개체 PushButton Object(IXHwpFormPushButton)</p>
--	---

IXHwpFormPushButton: 양식 개체 푸쉬 버튼 오브젝트 (PushButton 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Caption(Property)	캡션
Name(Property)	이름
ForeColor(Property)	전경색
BackColor(Property)	배경색
GroupName(Property)	그룹 이름
TabStop(Property)	Tab Stop
TabOrder(Property)	탭 순서
Width(Property)	너비
Height(Property)	높이
Left(Property)	왼쪽 좌표
Top(Property)	위쪽 좌표
CharShapeID(Property)	그리기 개체 Contorl ID
FollowContext(Property)	주위의 글자 속성을 따를지의 여부
AutoSize(Property)	글자 크기에 맞게 개체 크기가 바뀜
BorderType(Property)	테두리 타입
DrawFrame(Property)	틀을 그릴지의 여부
Enabled(Property)	활성, 비활성의 여부
WordWrap(Property)	자동 줄 바꿈

IXHwpFormCheckButtons: IXHwpFormCheckButton 오브젝트를 관리하는 오브젝트 (양식개체 CheckButton을 관리하는 Collection 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Item(Property)	Description 지정한 원소의 양식개체 CheckButton 오브젝트를 얻어온다. Declaration

	LPDISPATCH Item(long index) Parameters index : 원소의 인덱스 return : 원소의 값(IXHwpFormCheckButton)
Count(Property)	Description 원소의 총개수
ItemFromName(Property)	Description 양식 개체 CheckButton의 이름으로 원소를 찾는다. Declaration LPDISPATCH ItemFromName(BSTR name) Parameters name : 양식 개체 CheckButton의 이름 return : 양식 개체 CheckButton Object(IXHwpFormCheckButton)

IXHwpFormCheckButton: 양식 개체 체크 버튼 오브젝트 (CheckButton 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Caption(Property)	캡션
Name(Property)	이름
ForeColor(Property)	전경색
BackColor(Property)	배경색
GroupName(Property)	그룹 이름
TabStop(Property)	Tap Stop
TabOrder(Property)	탭 순서
Width(Property)	너비
Height(Property)	높이
Left(Property)	왼쪽 좌표
Top(Property)	위쪽 좌표
CharShapeID(Property)	그리기 개체 Control ID
FollowContext(Property)	주위의 글자 속성을 따를지의 여부
AutoSize(Property)	글자 크기에 맞게 개체 크기가 바뀜
BackStyle(Property)	배경 투명도
BorderType(Property)	테두리 타입
DrawFrame(Property)	틀을 그릴지의 여부
Enabled(Property)	활성, 비활성의 여부
TriState(Property)	체크 상태 옵션
Value(Property)	값

WordWrap(Property)	자동 줄 바꿈
--------------------	---------

**IXHwpFormRadioButtons: IXHwpFormRadioButton 오브젝트를 관리하는 오브젝트
(양식개체 RadioButton을 관리하는 Collection 개체 - 사용 편의를 위해 제공됨)**

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Item(Property)	Description 지정한 원소의 양식개체 RadioButton 오브젝트를 얻어온다. Declaration LPDISPATCH Item(long index) Parameters index : 원소의 인덱스 return : 원소의 값(IXHwpFormRadioButton)
Count(Property)	Description 원소의 총개수
ItemFromName(Property)	Description 양식 개체 RadioButton의 이름으로 원소를 찾는다. Declaration LPDISPATCH ItemFromName(BSTR name) Parameters name : 양식 개체 RadioButton의 이름 return : 양식 개체 RadioButton Object(IXHwpFormCheckButton)

**HwpFormRadioButton: 양식 개체 라디오 버튼 오브젝트
(RadioButton 개체 - 사용 편의를 위해 제공됨)**

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Name(Property)	이름
ForeColor(Property)	전경색
BackColor(Property)	배경색
GroupName(Property)	그룹 이름
TabStop(Property)	Tap Stop
TabOrder(Property)	탭 순서
Width(Property)	너비
Height(Property)	높이
Left(Property)	왼쪽 좌표
Top(Property)	위쪽 좌표
CharShapeID(Property)	그리기 개체 Control ID
FollowContext(Property)	주위의 글자 속성을 따를지의 여부

AutoSize(Property)	글자 크기에 맞게 개체 크기가 바뀜
BackStyle(Property)	배경 투명도
BorderType(Property)	테두리 타입
Caption(Property)	캡션
DrawFrame(Property)	틀을 그릴지의 여부
Enabled(Property)	활성, 비활성의 여부
RadioGroupName(Property)	라디오 그룹 이름
TriState(Property)	체크 상태 옵션
Value(Property)	값
WordWrap(Property)	자동 줄 바꿈

**IXHwpFormComboBoxes: IXHwpFormComboBox 오브젝트를 관리하는 오브젝트
(양식개체 ComboBox을 관리하는 Collection 개체 - 사용 편의를 위해 제공됨)**

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Item(Property)	Description 지정한 원소의 양식개체 ComboBox 오브젝트를 얻어온다. Declaration LPDISPATCH Item(long index) Parameters index : 원소의 인덱스 return : 원소의 값(IXHwpFormComboBox)
Count(Property)	Description 원소의 총개수
ItemFromName(Property)	Description 양식 개체 ComboBox의 이름으로 원소를 찾는다. Declaration LPDISPATCH ItemFromName(BSTR name) Parameters name : 양식 개체 ComboBox의 이름 return : 양식 개체 ComboBox Object(IXHwpFormComboBox)

**IXHwpFormComboBox: 양식 개체 콤보 박스 오브젝트
(ComboBox 개체 - 사용 편의를 위해 제공됨)**

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Name(Property)	이름
ForeColor(Property)	전경색
BackColor(Property)	배경색

GroupName(Property)	그룹 이름
TabStop(Property)	Tap Stop
TabOrder(Property)	탭 순서
Width(Property)	너비
Height(Property)	높이
Left(Property)	왼쪽 좌표
Top(Property)	위쪽 좌표
CharShapeID(Property)	그리기 개체 Control ID
FollowContext(Property)	주위의 글자 속성을 따를지의 여부
AutoSize(Property)	글자 크기에 맞게 개체 크기가 바뀜
BorderType(Property)	테두리 타입
DrawFrame(Property)	틀을 그릴지의 여부
EditEnable(Property)	에디트 상태 활성, 비활성의 여부
Enabled(Property)	활성, 비활성의 여부
ListBoxRows(Property)	리스트 박스 열
ListBoxWidth(Property)	리스트 박스 너비
Text(Property)	선택된 값
WordWrap(Property)	자동 줄 바꿈
Count(Property)	아이템 개수
CurSel(Property)	현재 선택된 인덱스
LBText(Property)	인덱스에 해당하는 값
InsertString(Method)	<p>Description 양식 개체 ComboBox에 문자열을 채워 넣는다.</p> <p>Declaration <code>void InsertString(BSTR itemvalue, long index)</code></p> <p>Parameters</p> <ul style="list-style-type: none"> itemvalue : 리스트에 채워 넣기 위한 값 index : 리스트의 특정 위치
DeleteString(Method)	<p>Description 리스트의 지정한 위치에 있는 값을 지운다.</p> <p>Declaration <code>void DeleteString(unsigned long index)</code></p> <p>Parameters</p> <ul style="list-style-type: none"> index : 리스트의 특정 위치
FindStringExact(Method)	<p>Description 지정한 문자열이 리스트에 있는지를 찾는다.</p> <p>Declaration <code>long FindStringExact(long index, BSTR itemvalue)</code></p> <p>Parameters</p> <ul style="list-style-type: none"> index : 리스트의 특정 위치 itemvalue : 리스트에서 찾기 위한 값

ResetContent(Method)	Description 리스트 내용을 초기화 한다. Declaration <code>void ResetContent(void)</code>
----------------------	---

**IXHwpFormEdits: IXHwpFormEdit 오브젝트를 관리하는 오브젝트
(양식개체 Edit을 관리하는 Collection 개체 - 사용 편의를 위해 제공됨)**

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Item(Property)	Description 지정한 원소의 양식개체 Edit 오브젝트를 얻어온다. Declaration <code>LPDISPATCH Item(long index)</code> Parameters index : 원소의 인덱스 return : 원소의 값(IXHwpFormEdit)
Count(Property)	Description 원소의 총개수
ItemFromName(Property)	Description 양식 개체 Edit의 이름으로 원소를 찾는다. Declaration <code>LPDISPATCH ItemFromName(BSTR name)</code> Parameters name : 양식 개체 Edit의 이름 return : 양식 개체 Edit Object(IXHwpFormEdit)

**IXHwpFormEdit: 양식 개체 에디트 오브젝트
(Edit 개체 - 사용 편의를 위해 제공됨)**

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject)
Name(Property)	이름
ForeColor(Property)	전경색
BackColor(Property)	배경색
GroupName(Property)	그룹 이름
TabStop(Property)	Tap Stop
TabOrder(Property)	탭 순서
Width(Property)	너비
Height(Property)	높이
Left(Property)	왼쪽 좌표
Top(Property)	위쪽 좌표

CharShapeID(Property)	그리기 개체 Control ID
FollowContext(Property)	주위의 글자 속성을 따를지의 여부
AutoSize(Property)	글자 크기에 맞게 개체 크기가 바뀜
BorderType(Property)	테두리 타입
DrawFrame(Property)	틀을 그릴지의 여부
Enabled(Property)	활성, 비활성의 여부
MaxLength(Property)	에디트 가능한 총 길이
MultiLine(Property)	멀티 라인 지원
Number(Property)	숫자만 입력 가능
PasswordChar(Property)	패스워드 표시에 사용할 글자
ReadOnly(Property)	읽기만 가능
ScrollBars(Property)	스크롤바 표시
TabKeyBehavior(Property)	탭 키를 눌렀을 때 반응
Text(Property)	에디트 텍스트
WordWrap(Property)	자동 줄 바꿈
LineCount(Property)	에디트 라인 줄 수

IXHwpWindows: IXHwpWindow 오브젝트를 관리하는 오브젝트 (Window를 관리하는 Collection 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject – 읽기 전용)
Active_XHwpWindow(Property)	Description 현재 활성화 상태인 윈도우 Object를 얻어온다.(IXHwpWindow)
Item(Property)	Description 지정한 원소의 윈도우 오브젝트를 얻어온다. Declaration LPDISPATCH Item(long index) Parameters index : 원소의 인덱스 return : 원소의 값(IXHwpWindow)
Count(Property)	Description 원소의 총개수
Add(Method)	Description 윈도우를 하나 추가한다.(새창으로 열기 기능과 동일) Declaration LPDISPATCH Add(void) Parameters return : 추가된 윈도우 오브젝트(IXHwpWindow)
Close(Method)	Description 윈도우를 모두 닫는다.

	Declaration BOOL Close(BOOL isDirty) Parameters isDirty : TRUE이면 문서 내용이 변경된 경우 닫지 않는다. FALSE이면 문서 내용이 변경된 것과 상관없이 강제로 닫는다.
--	---

IXHwpWindow: 원도우 오브젝트 (Window 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Application(Property)	Description 최상위 오브젝트를 얻어옴(IHwpObject – 읽기 전용)
XHwpDocuments(Property)	Description 도큐먼트 관리 오브젝트를 얻어옴(IXHwpDocuments – 읽기 전용)
XHwpTabs(Property)	Description 탭 관리 오브젝트를 얻어옴(IXHwpTabs – 읽기 전용)
Left(Property)	Description 원도우의 좌측 위치 좌표를 설정/얻음
Top(Property)	Description 원도우의 맨위 위치 좌표를 설정/얻음
Width(Property)	Description 원도우의 너비를 설정/얻음
Height(Property)	Description 원도우의 높이를 설정/얻음
Visible(Property)	Description 원도우 보이기/보이지 않기 설정/얻음
Close(Method)	Description 원도우를 닫음 Declaration void Close(BOOL isDirty) Parameters isDirty : TRUE이면 문서 내용이 변경된 경우 닫지 않는다. FALSE이면 문서 내용이 변경된 것과 상관없이 강제로 닫는다.

IXHwpTabs: IXHwpTab오브젝트를 관리하는 오브젝트 (Tab을 관리하는 Collection 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Count(Property)	Description 원도우에 열려있는 탭의 개수(읽기 전용)
Item(Property)	Description 지정한 원소의 탭 오브젝트를 얻어온다. Declaration LPDISPATCH Item(long index) Parameters index : 원소의 인덱스 return : 원소의 값(IXHwpTab)

Add(Method)	<p>Description 지정한 원소의 탭 오브젝트를 추가한다.(문서를 새 탭으로 열기)</p> <p>Declaration LPDISPATCH Add(void)</p> <p>Parameters return : 추가된 원소의 값(IXHwpTab)</p>
Close(Method)	<p>Description 탭을 모두 닫는다.</p> <p>Declaration BOOL Close(BOOL isDirty)</p> <p>Parameters isDirty : TRUE이면 문서 내용이 변경된 경우 닫지 않는다. FALSE이면 문서 내용이 변경된 것과 상관없이 강제로 닫는다.</p>

IXHwpTab: 탭 오브젝트 (Tab 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Close(Method)	<p>Description 탭을 닫는다.</p> <p>Declaration void Close(BOOL isDirty)</p> <p>Parameters isDirty : TRUE이면 문서 내용이 변경된 경우 닫지 않는다. FALSE이면 문서 내용이 변경된 것과 상관없이 강제로 닫는다.</p>

HSet : ParameterSet Item 데이터들의 집합

예) HParameterSet.HSecDef.HSet – HSecDef에 저장된 모든 Item 데이터의 집합을 얻음

Item Name	Description
SetItem(Method)	<p>Description 지정된 아이템 이름을 가진 데이터에 VARIANT값을 대입한다.</p> <p>Declaration SetItem(BSTR itemid, VRIANT value)</p> <p>Parameters itemid : 아이템 이름 value : 데이터</p>

HAction : 한/글에서 특정 기능을 수행하기 위한 액션 오브젝트

예) 아래와 같은 형태로 사용하는 것이 HAction을 사용하는 올바른 사용법이다.

`HAction.GetDefault("Print", HParameterSet.HPrint.HSet); // 액션 초기화`

`HParameterSet.HPrint.NumCopy = 3; //인쇄 매수를 3장으로 지정`

`HAction.Execute("Print", HParameterSet.HPrint.HSet); // 액션 수행`

Item Name	Description
GetDefault(Method)	<p>Description 지정된 액션 이름과 LPDISPATCH로 HSet을 받아 액션을 초기화 한다.</p> <p>Declaration BOOL GetDefault(BSTR actname, LPDISPATCH object)</p> <p>Parameters actname : 액션 이름</p>

	object : HSet Object
Execute(Method)	<p>Description 지정된 액션 이름과 LPDISPATCH로 HSet을 받아 액션을 수행 한다.</p> <p>Declaration BOOL Execute(BSTR actname, LPDISPATCH object)</p> <p>Parameters actname : 액션 이름 object : HSet Object</p>
PopupDialog(Method)	<p>Description 지정된 액션 이름과 LPDISPATCH로 HSet을 받아 액션의ダイ얼로그를 생성한다.</p> <p>Declaration BOOL PopupDialog(BSTR actname, LPDISPATCH object)</p> <p>Parameters actname : 액션 이름 object : HSet Object</p>
Run(Method)	<p>Description GetDefault, PopupDialog, Execute를 동시에 수행하도록 한다.</p> <p>Declaration BOOL Run(BSTR actname)</p> <p>Parameters actname : 액션 이름</p>

HArray : ParameterArray의 데이터 집합

Item Name	Description
Count(Property)	<p>Description Array에 크기를 지정하거나 얻을 수 있다.</p>
Item(Property)	<p>Description Array에 지정된 index에 해당하는 값을 VARIANT로 지정하거나 얻을 수 있다.</p>

IXHwpMessageBox: OLE Automation standard object (메세지박스 MessageBox 개체 - 사용 편의를 위해 제공됨)

Item Name	Description
Application(Property)	<p>Description 최상위 오브젝트를 얻어옴(IHwpObject – 읽기 전용)</p>
String(Property)	<p>Description 메시지 박스에 넣을 문자열</p>
Flag(Property)	<p>Description 메시지 박스에 사용할 Flag</p>
Result(Property)	<p>Description 메시지 박스의 리턴값</p>
DoModal(Method)	<p>Description 메시지 박스 보이기</p> <p>Declaration void DoModal(void)</p>

IDHwpAction: 한글에서 특정 기능을 수행하기 위한 액션 오브젝트
(호환 유지를 위해 제공됨 - HwpCtrl에서 사용되는 DHwpAction과 동일)

Item Name	Description												
ActID(Property)	Description 액션 ID를 나타낸다. 읽기 전용.												
SetID(Property)	Description 액션이 사용하는 parameter set ID를 나타낸다. 읽기 전용.												
GetDefault(Method)	Description 현재 상태에 따라 액션 실행에 필요한 인수를 구한다. Declaration void GetDefault(LPDISPATCH param) Parameters param : 인수를 저장할 Parameter Set												
CreateSet(Method)	Description 액션과 대응하는 Parameter Set을 생성한다. Declaration LPIDSPATCH CreateSet(void) Parameters return : ParameterSet Object (IDHwpParameterSet)												
Execute(Method)	Description 지정한 인수로 액션을 실행한다. Declaration void Execute(LPDISPATCH param) Parameters param : 액션의 실행을 제어할 인수, ParameterSet의 종류와 아이템의 의미는 액션이 정의한 바에 따라 다르다.(IDHwpParameterSet)												
PopupDialog(Method)	Description 액션의 대화상자를 띄운다. Declaration long PopupDialog(LPDISPATCH param) Parameters param : 여기에 지정된 아이템의 값에 따라 대화상자의 컨트롤의 초기값이 결정되고, 대화상자가 닫힐 후에는 사용자가 지정한 값들이 담겨 돌아온다. return : 액션이 정의하기에 따라 다르지만, 일반적으로 다음과 같은 modal dialog result를 리턴한다. <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #cccccc;"> <th style="text-align: center; padding: 2px;">ID</th> <th style="text-align: center; padding: 2px;">값</th> <th style="text-align: center; padding: 2px;">설명</th> </tr> </thead> <tbody> <tr> <td style="text-align: center; padding: 2px;">hwpOK</td> <td style="text-align: center; padding: 2px;">IDOK</td> <td style="text-align: center; padding: 2px;">다이얼로그 박스의 확인버튼을 눌렀을 경우 리턴 되는 값</td> </tr> <tr> <td style="text-align: center; padding: 2px;">hwpCancel</td> <td style="text-align: center; padding: 2px;">IDCANCEL</td> <td style="text-align: center; padding: 2px;">다이얼로그 박스의 취소버튼을 눌렀을 경우 리턴 되는 값</td> </tr> <tr> <td style="text-align: center; padding: 2px;">hwpError</td> <td style="text-align: center; padding: 2px;">-1</td> <td style="text-align: center; padding: 2px;">실행시 에러가 발생 하였을 경우 리턴 되는 값</td> </tr> </tbody> </table>	ID	값	설명	hwpOK	IDOK	다이얼로그 박스의 확인버튼을 눌렀을 경우 리턴 되는 값	hwpCancel	IDCANCEL	다이얼로그 박스의 취소버튼을 눌렀을 경우 리턴 되는 값	hwpError	-1	실행시 에러가 발생 하였을 경우 리턴 되는 값
ID	값	설명											
hwpOK	IDOK	다이얼로그 박스의 확인버튼을 눌렀을 경우 리턴 되는 값											
hwpCancel	IDCANCEL	다이얼로그 박스의 취소버튼을 눌렀을 경우 리턴 되는 값											
hwpError	-1	실행시 에러가 발생 하였을 경우 리턴 되는 값											
Run(Method)	Description 액션을 실행한다. Declaration void Run(void) Remarks CreateSet, GetDefault, PopupDialog, Execute를 차례대로 부른 것과 같다.												

**IDHwpParameterSet: 오브젝트 또는 액션의 실행에 필요한 정보를 주고 받을 수 있도록 하기 위한
오브젝트
(호환 유지를 위해 제공됨 - HwpCtrl에서 사용되는 DHwpParameterSet과 동일)**

Item Name	Description
Count(Property)	Description 현재 존재하는 아이템의 개수를 나타낸다. (읽기 전용)
IsSet(Property)	Description Parameter Set인지 여부를 나타낸다. (읽기 전용) Remarks 임의의 IDispatch 포인터로부터 Parameter set / parameter array를 구분하기 위해 동일한 이름의 Property를 가지고 종류에 따라 TRUE/FALSE를 돌려준다. (Parameter set은 TRUE를 리턴한다.)
GetSetID(Property)	Description Parameter Set의 ID를 나타낸다. (읽기 전용)
Clone(Method)	Description 동일한 데이터를 가진 Parameter Set을 복사하여 리턴한다. Declaration <code>LPDISPATCH Clone(void)</code> Parameters <code>return : ParameterSet을 리턴한다. (IDHwpParameterSet)</code>
CreateItemArray(Method)	Description 아이템으로 Parameter Array 타입의 배열을 생성한다. Declaration <code>LPDISPATCH CreateItemArray(BSTR itemid, long count)</code> Parameters <code>itemid : 아이템 ID</code> <code>count : 생성할 배열의 초기 크기</code> <code>return : 생성된 parameter array 오브젝트 (IDHwpParameterArray)</code> Remarks 동일한 ID를 가진 기존의 아이템은 삭제된다.
CreateItemSet(Method)	Description 아이템으로 Parameter Set을 생성한다. Declaration <code>LPDISPATCH CreateItemSet(BSTR itemid, BSTR setid)</code> Parameters <code>itemid : 아이템 ID</code> <code>setid : 생성할 Parameter Set ID</code> <code>return : 생성된 서브 parameter Set 오브젝트 (IDHwpParameterSet)</code> Remarks ParameterSet 내부에 아이템으로 또 다른 Parameter Set을 가지는 서브셋의 개념이다.
GetIntersection(Method)	Description 두 Parameter Set에 공통적으로 존재하고, 값도 동일한 아이템만으로 구성된 intersection Set을 구한다. Declaration <code>void GetIntersection(LPDISPATCH srcset)</code> Parameters <code>srcset : this와 srcset의 intersection이 this에 저장된다.</code>
IsEquivalent(Method)	Description 두 Parameter Set의 내용이 동일한 값을 가지고 있는지 검사한다.

	<p>Declaration BOOL IsEquivalent(LPDISPATCH srcset) Parameters srcset : this와 srcset의 비교한 결과를 리턴한다. return : 동일하면 TRUE, 다르면 FALSE</p>
Item(Method)	<p>Description 지정한 아이템의 값을 리턴한다. Declaration Variant Item(BSTR itemid) Parameters itemid : 아이템 ID return : 아이템의 값 Remarks 만약 지정한 아이템이 존재하지 않으면 아이템의 포맷에 따라 0 또는 빈 문자열을 리턴한다.</p>
ItemExist(Method)	<p>Description 지정한 아이템이 존재하는지 검사한다. Declaration BOOL ItemExist(BSTR itemid) Parameters itemid : 아이템 ID return : 존재하면 TRUE, 존재하지 않으면 FALSE</p>
Merge(Method)	<p>Description 두 Parameter Set의 내용을 병합한다. Declaration void Merge(LPDISPATCH srcset) Parameters srcset : this와 srcset이 병합되어 this에 저장된다. Remarks 결과는 "this의 모든 아이템 + srcset에만 존재하는 아이템"이다.</p>
RemoveAll(Method)	<p>Description Parameter Set을 초기화 한다. Declaration void RemoveAll(BSTR setid) Parameters setid : 새로 적용할 Set ID Remarks 이미 존재하는 Parameter Set 오브젝트를 이용해 새로운 타입의 Parameter Set으로 초기화하여 재사용하는 목적에 사용된다.</p>
RemoveItem(Method)	<p>Description 지정한 아이템을 삭제한다. Declaration void RemoveItem(BSTR itemid) Parameters itemid : 아이템 ID</p>
SetItem(Method)	<p>Description 지정한 아이템의 값을 설정한다. Declaration void SetItem(BSTR itemid, Variant value) Parameters itemid : 아이템 ID value : 설정할 값</p>

	<p>Remarks 이미 동일한 ID의 아이템이 존재하면 지정한 값으로 바뀌고, 존재하지 않으면 아이템이 생성된다.</p>
--	---

IDHwpParameterArray: Parameter Set의 아이템으로 배열을 표현하는데 사용된다. 일반적인 Method의 독립적인 인수로 사용되는 일은 없고, Parameter Set의 아이템으로만 사용된다.
(호환 유지를 위해 제공됨 - HwpCtrl에서 사용되는 DHwpParameterArray와 동일)

Item Name	Description
Count(Property)	<p>Description 배열의 크기를 나타낸다. Remarks 배열의 크기는 runtime에 dynamic하게 조절 할 수 있다.</p>
IsSet(Property)	<p>Description Parameter Set인지 여부를 나타낸다.(읽기 전용) Remarks 임의의 IDispatch 포인터로부터 Parameter Set / Parameter Array를 구분하기 위해 동일한 이름의 Property를 가지고 종류에 따라 TRUE/FALSE를 돌려준다.(ParameterArray는 FALSE를 리턴한다.)</p>
Clone(Method)	<p>Description 동일한 크기와 데이터를 갖는 ParameterArray 객체를 복사하여 돌려준다 Declaration LPDISPATCH Clone(void)</p>
Copy(Method)	<p>Description 배열을 복사한다. Declaration void Copy(LPDISPATCH srcarray) Parameters srcarray : srcarray의 내용이 그대로 this로 복사된다.</p>
Item(Method)	<p>Description 지정한 원소의 값을 리턴한다. Declaration Variant Item(long index) Parameters index : 원소의 인덱스. 1부터 시작한다.</p>
SetItem(Method)	<p>Description 지정한 원소의 값을 설정한다. Declaration void SetItem(long index, Variant value) Parameters index : 원소의 인덱스, 1부터 시작 value : 원소의 값</p>

IDHwpCtrlCode: 문서 내부의 표, 각주 등의 컨트롤(특수 문자 포함)를 나타내는 오브젝트이다.
(호환 유지를 위해 제공됨 - HwpCtrl에서 사용되는 DHwpCtrlCode와 동일)

Item Name	Description
CtrlCh(Property)	<p>Description 컨트롤 문자. (읽기전용)</p>

Remarks	
일반적으로 컨트롤 ID를 사용해 컨트롤의 종류를 판별하지만, 이보다 더 포괄적인 범주를 나타내는 컨트롤 문자로 판별할 수도 있다. 예를 들어 각주와 미주는 ID는 다르지만, 컨트롤 문자는 17로 동일하다. 컨트롤 문자는 1-31 사이의 값을 사용한다.	
Ch	설명
1	예약
2	구역/단 정의
3	필드 시작
4	필드 끝
5	예약
6	예약
7	예약
8	예약
9	탭
10	강제 줄 나눔
11	그리기 개체 / 표
12	예약
13	문단 나누기
14	예약
15	주석
16	머리말 / 꼬리말
17	각주 / 미주
18	자동 번호
19	예약
20	예약
21	쪽바꿈
22	책갈피 / 찾아보기 표시
23	덧말 / 글짜 겹침,,,
24	하이픈
25	예약
26	예약
27	예약
28	예약
29	예약
30	묶음 빙칸

	Ch	설명	
CtrlID(Property)	31	고정 폭 빙칸	
Description			컨트롤 ID. (읽기 전용)
Remarks			컨트롤 ID는 컨트롤의 종류를 나타내기 위해 할당된 ID로서, 최대 4개의 문자로 구성된 문자열이다. 예를 들어 표는 "tbl", 각주는 "fn"이다. 워디안에서 현재까지 지원되는 모든 컨트롤의 ID는 다음 표 참조.
ID	Property Set	Initialization Set	설명
cold	ColDef	ColDef	단
secd	SecDef	SecDef	구역
fn	FootnoteShape	FootnoteShape	각주
en	FootnoteShape	FootnoteShape	미주
tbl	Table	TableCreation	표
eqed	EqEdit	EqEdit	수식
atno	AutoNum	AutoNum	번호 넣기
nwno	AutoNum	AutoNum	새 번호로
pgct	PageNumCtrl	PageNumCtrl	페이지 번호 제어 (97의 흘수 쪽에서 시작)
pghd	PageHiding	PageHiding	감추기
pgnp	PageNumPos	PageNumPos	쪽 번호 위치
head	HeaderFooter	HeaderFooter	머리말
foot	HeaderFooter	HeaderFooter	꼬리말
%dte	FieldCtrl	FieldCtrl	현재의 날짜/시간 필드
%ddt	FieldCtrl	FieldCtrl	파일 작성 날짜/시간 필드
%pat	FieldCtrl	FieldCtrl	문서 경로 필드
%bmk	FieldCtrl	FieldCtrl	블럭 책갈피
%mmg	FieldCtrl	FieldCtrl	메일 머지
%xrf	FieldCtrl	FieldCtrl	상호 참조
%fmu	FieldCtrl	FieldCtrl	계산식
%clk	FieldCtrl	FieldCtrl	누름 템
%smr	FieldCtrl	FieldCtrl	문서 요약 정보 필드
%usr	FieldCtrl	FieldCtrl	사용자 정보 필드
%hlk	FieldCtrl	FieldCtrl	하이퍼링크
%sig	RevisionDef	RevisionDef	교정 부호(띄움표)
%%%d	RevisionDef	RevisionDef	교정 부호(지움표)

ID	Property Set	Initialization Set	설명
%%*a	RevisionDef	RevisionDef	교정부호(불임표)
%%*C	RevisionDef	RevisionDef	교정부호(쌤표)
%%*S	RevisionDef	RevisionDef	교정부호(틈니표)
%%*T	RevisionDef	RevisionDef	교정부호(생각표)
%%*P	RevisionDef	RevisionDef	교정부호(칭찬표)
%%*L	RevisionDef	RevisionDef	교정부호(줄표)
%%*c	RevisionDef	RevisionDef	교정부호(고침표)
%%*h	HyperLink	HyperLink	교정부호(자료연결)
%%*A	RevisionDef	RevisionDef	교정부호(줄붙임표)
%%*i	RevisionDef	RevisionDef	교정부호(줄이음표)
%%*t	RevisionDef	RevisionDef	교정부호(줄서로바꿈표)
%%*r	RevisionDef	RevisionDef	교정부호(오른자리옮김표)
%%*l	RevisionDef	RevisionDef	교정부호(왼자리옮김표)
%%*n	RevisionDef	RevisionDef	교정부호(자리바꿈표)
%spl	RevisionDef	RevisionDef	교정부호(나눔표)
%%mr	RevisionDef	RevisionDef	교정부호(메모고침표)
%%me	RevisionDef	RevisionDef	메모
bokm	TextCtrl	TextCtrl	책갈피
idxm	IndexMark	IndexMark	찾아보기
tdut	Dutmal	Dutmal	덧말
tcmt	없음	없음	주석
\$con	ShapeObject	ShapeObject	여러 개체를 묶은 개체
\$lin	ShapeObject	ShapeObject	직선
\$rec	ShapeObject	ShapeObject	사각형
\$ell	ShapeObject	ShapeObject	원형
\$arc	ShapeObject	ShapeObject	호
\$pol	ShapeObject	ShapeObject	다각형
\$cur	ShapeObject	ShapeObject	곡선
\$pic	ShapeObject	ShapeObject	그림
form	ShapeObject	ShapeObject	양식 개체
+pbt	ShapeObject	ShapeObject	명령 단추
+rbt	ShapeObject	ShapeObject	라디오 단추
+cbt	ShapeObject	ShapeObject	선택 상자
+cob	ShapeObject	ShapeObject	콤보 상자
+edt	ShapeObject	ShapeObject	입력 상자
\$ole	ShapeObject	OleCreation	OLE개체

◆ Property Set : Ctrl.Properties를 통해 액세스할 수 있는 속성
parameter set ID

	<p>◆ Initialization Set : HwpCtrl.InsertCtrl에 지정할 수 있는 initparam의 parameter set ID</p>												
HasList(Property)	<p>Description 글상자를 지원하는지의 여부(읽기 전용)</p>												
Next(Property)	<p>Description 다음 컨트롤.(읽기 전용) Remarks 문서 중의 모든 컨트롤(표, 그림등의 특수 문자들)은 linked list로 서로 연결되어 있는데, list중 다음 컨트롤을 나타낸다.</p>												
Prev(Property)	<p>Description 앞 컨트롤.(읽기 전용) Remarks 문서 중의 모든 컨트롤(표, 그림등의 특수 문자들)은 linked list로 서로 연결되어 있는데, list중 다음 컨트롤을 나타낸다.</p>												
Properties(Property)	<p>Description 컨트롤의 속성을 나타낸다. Parameters 모든 컨트롤은 대응하는 parameter set으로 속성을 읽고 쓸 수 있다.</p>												
UserDesc(Property)	<p>Description 컨트롤의 종류를 사용자에게 보여줄 수 있는 localize된 문자열로 나타낸다. (읽기 전용)</p>												
GetAnchorPos(Method)	<p>Description 컨트롤의 anchor의 위치를 리턴한다. Declaration LPDISPATCH GetAnchorPos(long type) Parameters type : 기준 위치</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #ffffcc;"> <th style="text-align: center;">값</th> <th style="text-align: center;">설명</th> <th style="text-align: center;">비고</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td>바로 상위 리스트에서의 anchor position</td> <td>default</td> </tr> <tr> <td style="text-align: center;">1</td> <td>탑레벨 리스트에서의 anchor position</td> <td></td> </tr> <tr> <td style="text-align: center;">2</td> <td>루트 리스트에서의 anchor position</td> <td></td> </tr> </tbody> </table> <p>value : 성공했을 경우 LispParaPos Parameter Set이 리턴 된다. 실패했을 경우 NULL이 리턴된다.</p>	값	설명	비고	0	바로 상위 리스트에서의 anchor position	default	1	탑레벨 리스트에서의 anchor position		2	루트 리스트에서의 anchor position	
값	설명	비고											
0	바로 상위 리스트에서의 anchor position	default											
1	탑레벨 리스트에서의 anchor position												
2	루트 리스트에서의 anchor position												

- Action Object

Action Object 매뉴얼 참고(ActionObject.hwp)

- ParameterSet Object

ParameterSet Object 매뉴얼 참고(ParameterSetObject.hwp)

- Add-On Object(메뉴, 툴바 제어 기능, 사용자 정의 액션 기능)