



Excel Automation Tutorial : *Using MFC*

ISL / 강한솔

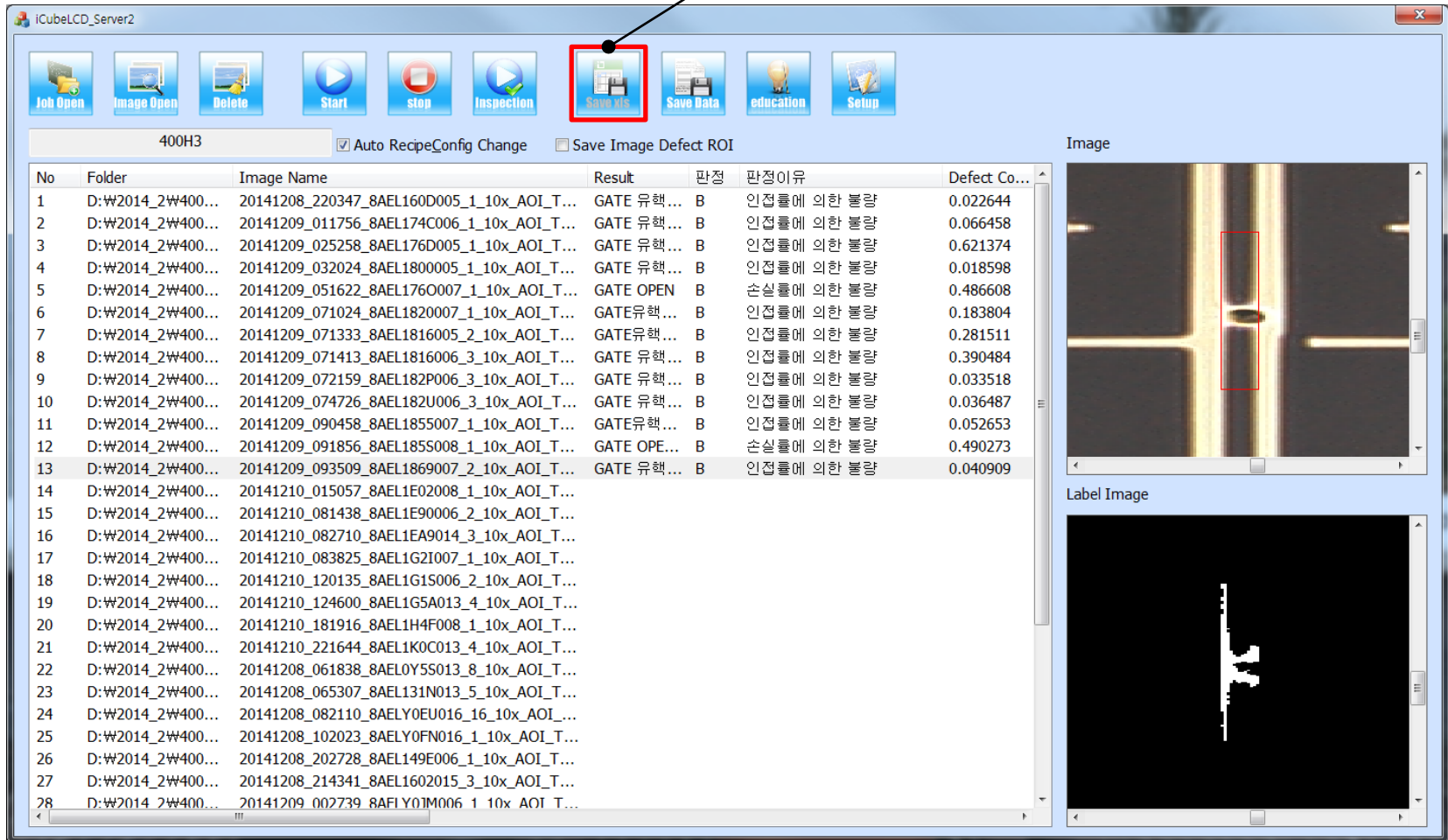
Index

- ✓ Motivation of Study
- ✓ Excel Automation
- ✓ Set Up
- ✓ Class
- ✓ Demo

Motivation of Study

❖ Intek Plus Server Program

엑셀 저장 기능



Motivation of Study

❖ Intek Plus Server Program

Excel interface showing a data table with 10 rows and 26 columns (A to AB). The table contains numerical data and text descriptions. The last two columns (S and T) contain images of defects.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB
1	Defect Co	Pattern In	Corner Pa	Normalize	Intenisty	RGB ratio	RGB ratio	RGB ratio	Area	Defect Ty	Defect Bri	Line Thick	Defect Da	Result	판정	판정이유												
2	0.353846	0.904613	0.907465	192.89953	131.63465	93.673691	80.939018	67.118279	1128.0000	Bright	1.000000	0.000000	0.000000	GATE 잔류B		인접률에 의한 불량												
3	0.531199	0.827637	0.917698	193.43589	129.13720	88.061745	77.450150	66.090950	235.00000	Bright	1.000000	0.000000	0.000000	GATE 잔류B		인접률에 의한 불량												
4	0.523994	0.722003	0.798682	140.94339	90.335594	79.041946	70.757004	59.756763	79.000000	Bright	1.000000	0.000000	0.000000	GATE 잔류B		인접률에 의한 불량												
5	0.186380	0.820431	0.842155	127.21052	114.95352	87.510399	77.245178	65.737595	363.00000	Bright	1.000000	0.000000	0.000000	GATE 잔류B		인접률에 의한 불량												
6	0.394711	0.835500	0.898424	106.94736	106.66626	87.799072	78.039215	65.313950	178.00000	Bright	1.000000	0.000000	0.000000	GATE 잔류B		인접률에 의한 불량												
7	0.424448	0.924650	0.938689	141.29878	125.96257	94.785767	84.902390	70.520737	459.00000	Bright	1.000000	0.000000	0.000000	GATE 잔류B		인접률에 의한 불량												
8	0.568230	0.803906	0.881998	145.48979	120.34981	86.466232	76.675377	66.601303	90.000000	Bright	1.000000	0.000000	0.000000	GATE 잔류B		인접률에 의한 불량												
9	0.444897	0.896163	0.825329	175.95655	136.08625	93.988609	82.324829	68.887688	1088.0000	Bright	1.000000	0.000000	0.000000	GATE 잔류B		인접률에 의한 불량												
10	0.263719	0.896117	0.765264	179.55723	138.11198	93.138115	81.255722	68.342476	2499.0000	Bright	1.000000	0.000000	0.000000	GATE 잔류B		인접률에 의한 불량												

Excel Automation

❖ Automation

: 응용 프로그램 외부에서 응용프로그램 객체를 조작할 수 있는 방법. 표준 소프트웨어 아키텍처인 COM에 기반을 둠.

➡ **Excel Automation** : 엑셀이 아닌 다른 프로그램(MFC)를 이용하여 엑셀을 조작하는 것.

❖ COM(Component Object Model)

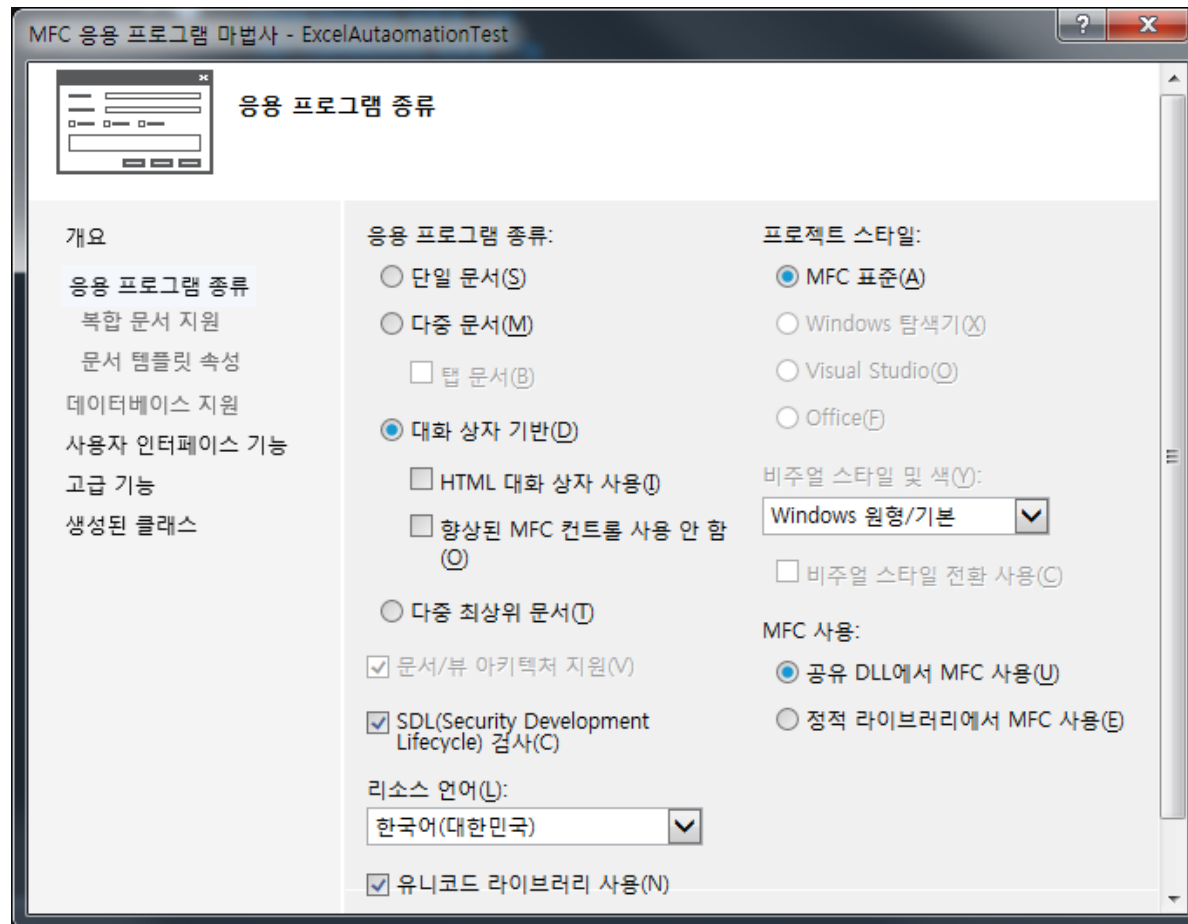
: 다른 소프트웨어 제작사에 의해 만들어진 컴포넌트를 자신의 소프트웨어에서 사용할 수 있게 하는 소프트웨어 아키텍처.

❖ Type Libraries

: COM Object의 기능에 관한 정보를 제공하는 파일.

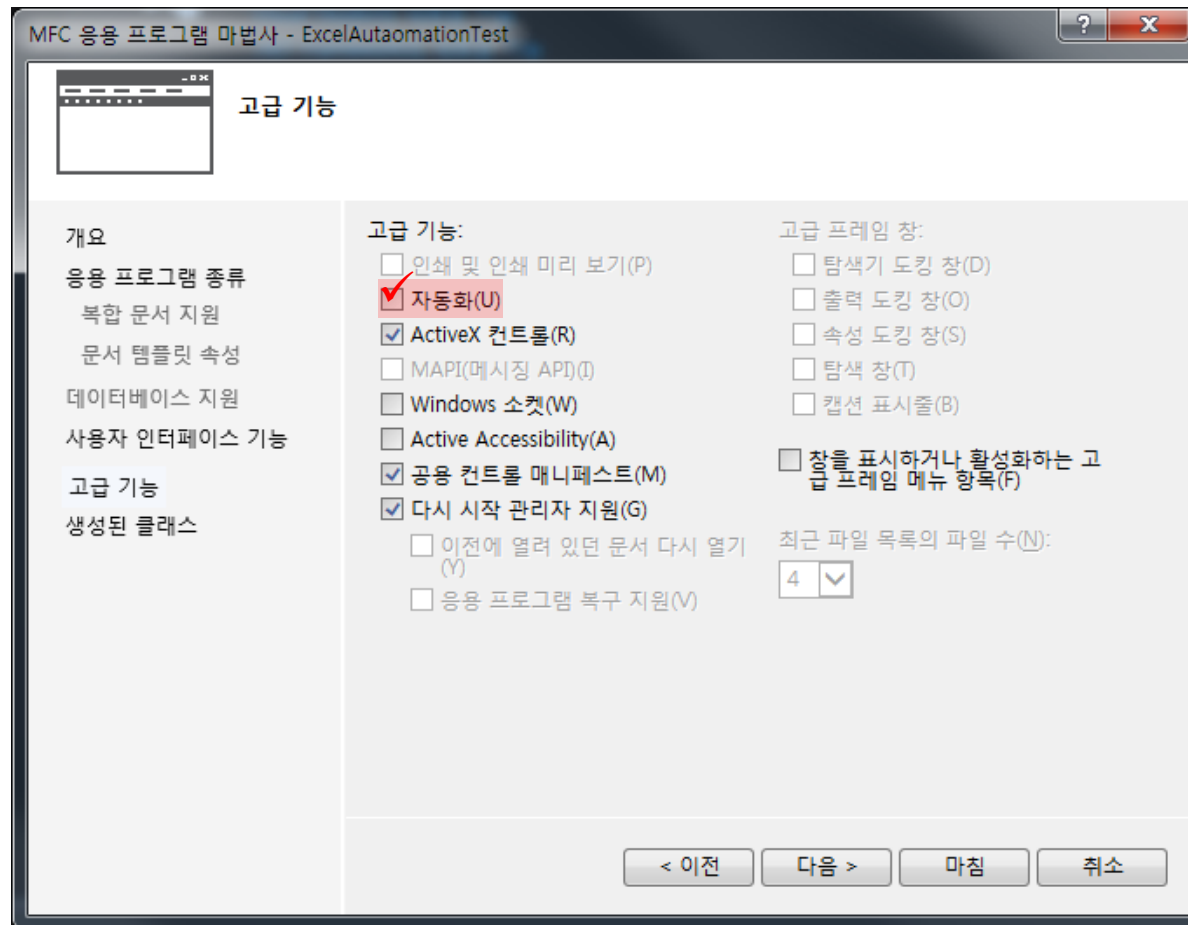
Set Up

❖ 프로젝트 생성



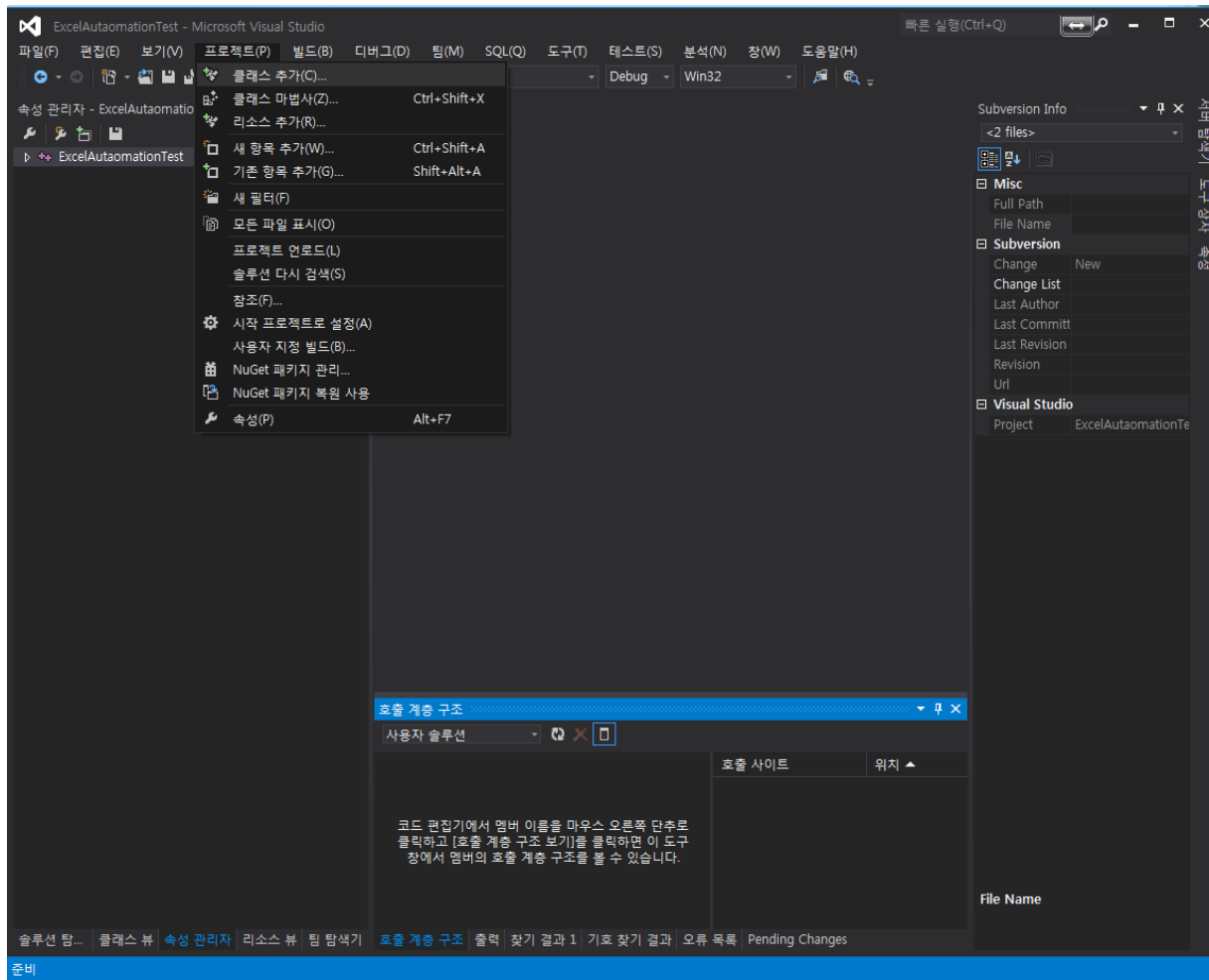
Set Up

❖ 프로젝트 생성



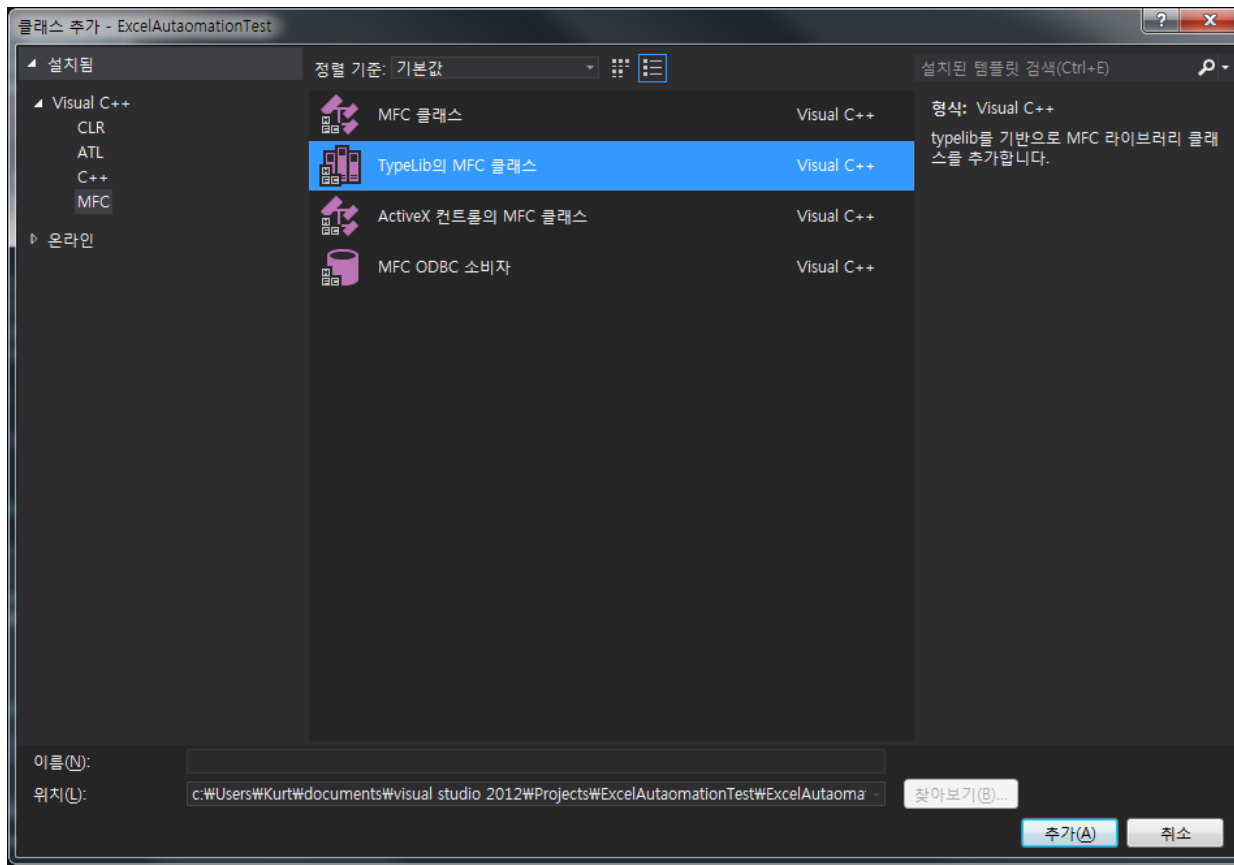
Set Up

❖ 클래스 추가



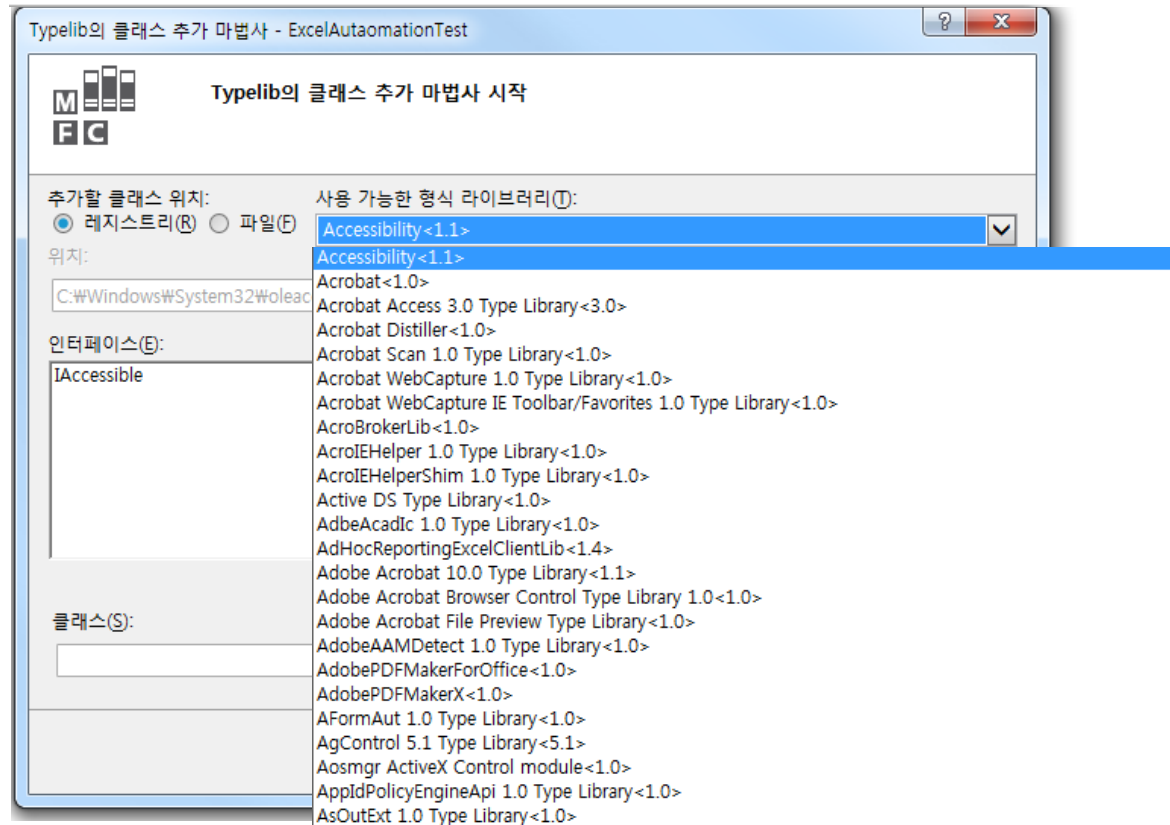
Set Up

❖ 클래스 추가



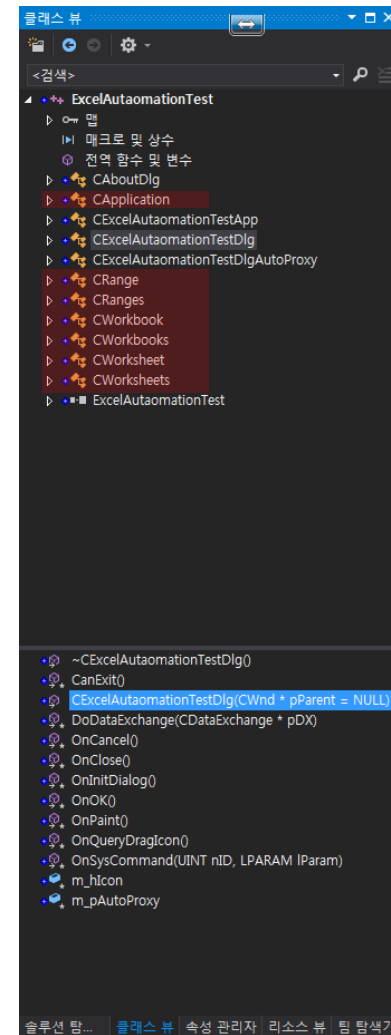
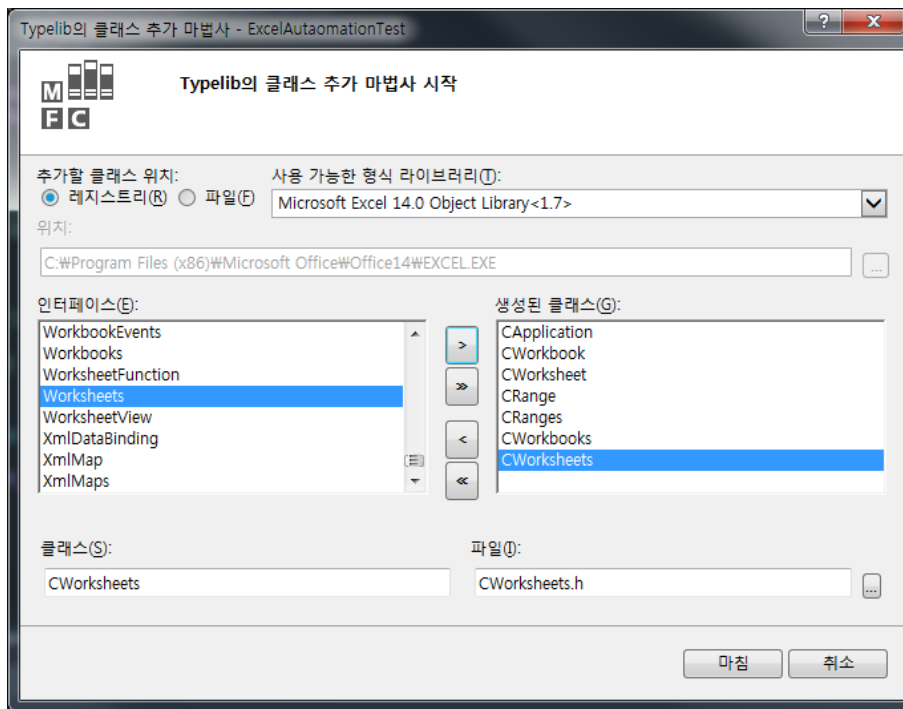
Set Up

❖ 클래스 추가



Set Up

❖ 클래스 추가



Set Up

❖ 헤더파일 주석 처리

```
CApplication
// 컴퓨터에서 형식 라이브러리 마법사의 [클래스 추가]를 사용하여 생성한 IDispatch 래퍼 클래스입니다.

#import "C:\\Program Files (x86)\\Microsoft Office\\Office14\\EXCEL.EXE" no_namespace
// CApplication 래퍼 클래스
```

```
CApplication
// 컴퓨터에서 형식 라이브러리 마법사의 [클래스 추가]를 사용하여 생성한 IDispatch 래퍼 클래스입니다.

// #import "C:\\Program Files (x86)\\Microsoft Office\\Office14\\EXCEL.EXE" no_namespace
// CApplication 래퍼 클래스
```

Set Up

❖ CRange의 DialogBox 주석 처리

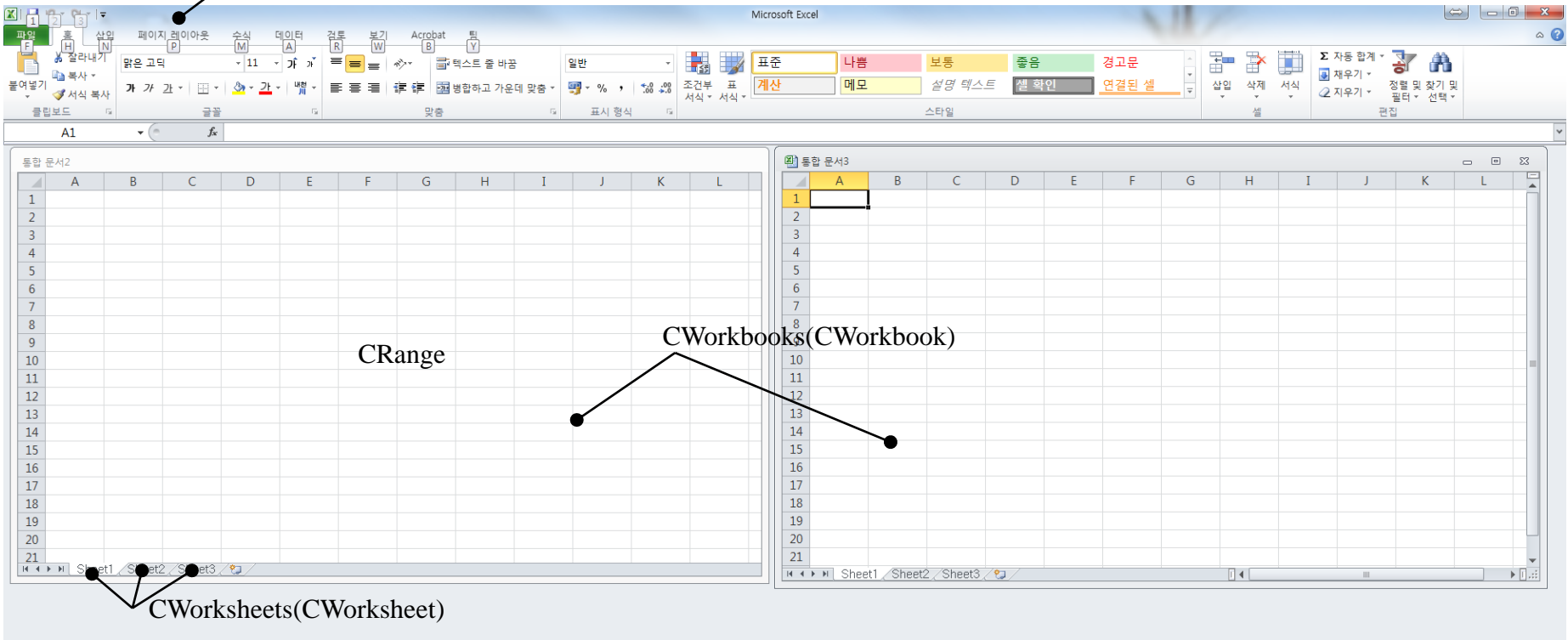
```
CRange
//VARIANT DialogBox()
//{
//  VARIANT result;
//  InvokeHelper(0xf5, DISPATCH_METHOD, VT_VARIANT, (void*)&result, NULL);
//  return result;
//}
```

```
WinUser.h  X  CRange.h  ExcelAutomaationT...T_DIALOG - Dialog
(전역 범위)
#ifdef UNICODE
#define DialogBox DialogBoxW
#else
#define DialogBox DialogBoxA
#endif // !UNICODE
```

Class

❖ Class

CApplication



Class

❖ CApplication

: 엑셀 실행 또는 관리하기 위한 클래스

❖ Member Function

BOOL CreateDispatch(**LPCTSTR** lpszProgID, **ColeException*** pError = **NULL**);

: 엑셀을 실행.

ex) app.CreateDispatch(_T("Excel.Application"));

LPCDISPATCH get_Workbooks()

: 워크북의 정보를 불러옴.

void put_Visible(**BOOL** newValue)

: 엑셀을 가시화 시켜줌.

void put_UserControl(**BOOL** newValue)

: 유저가 엑셀을 제어할 수 있도록 함.

Class

❖ CWorkbooks & CWorkbook

: 엑셀 파일을 열거나 저장할 때 사용하는 클래스

❖ Member Function

[Cworkbooks]

LPDISPATCH Add(**VARIANT**& Template)

: 엑셀파일을 생성 또는 불러옴

```
ex) book = books.Add (COleVariant((long)DISP_E_PARAMNOTFOUND,VT_ERROR));  
    book = books.Add(COleVariant(_T("test.xls")));
```

[Cworkbook]

LPDISPATCH get_Worksheets()

: 워크시트 정보를 불러옴.

Class

❖ CWorksheets & CWorksheet

: 엑셀 워크북 내에 있는 시트를 다루는 클래스.

❖ Member Function

[Cworksheets]

LPDISPATCH Add(**VARIANT**& Before, **VARIANT**& After, **VARIANT**& Count, **VARIANT**& Type)

: 워크 시트 추가.

ex) sheets.Add(covOptional,covOptional,COleVariant((long)5),covOptional);

LPDISPATCH get_Item(**VARIANT**& Index)

: 워크 시트 선택.

[Cworksheet]

LPDISPATCH get_Range(**VARIANT**& Cell1, **VARIANT**& Cell2)

: 시트에서 사용할 범위 지정.

ex) range = sheet.get_Range(COleVariant(_T("A1")),COleVariant(_T("A1")));
range = sheet.get_Range(COleVariant(_T("A1")), COleVariant(_T("D1")));

Class

❖ CRange

: 엑셀 시트 안에서 이용할 셀의 사용범위를 지정하는 클래스(실질적인 텍스트나 데이터 입력).

❖ Member Function

`void put_Value2(VARIANT& newValue)`

: 해당하는 범위에 데이터 입력.

```
ex) range = sheet.get_Range(COleVariant(_T("A1")),COleVariant(_T("A1")));  
    range.put_Value2(COleVariant(_T("First Name")));
```

`LPDISPATCH get_Font()`

: 폰트 정보를 불러옴.

`void put_HorizontalAlignment(VARIANT& newValue), void put_VerticalAlignment(VARIANT& newValue)`

: 해당하는 셀의 정렬을 설정.

```
ex) range.put_HorizontalAlignment(COleVariant((short)-4108));  
    range.put_VerticalAlignment(COleVariant((short)-4108));
```

```
xlVAlignBottom = -4107,  
xlVAlignCenter = -4108,  
xlVAlignDistributed = -4117,  
xlVAlignJustify = -4130,  
xlVAlignTop = -4160
```

Class

❖ CRange

: 엑셀 시트 안에서 이용할 셀의 사용범위를 지정하는 클래스(실질적인 텍스트나 데이터 입력).

❖ Member Function

`void put_Formula(VARIANT& newValue)`

: 해당하는 범위에 수식 입력.

```
ex) range = sheet.get_Range(COleVariant(_T("D2")), COleVariant(_T("D6")));  
    range.put_Formula(COleVariant(_T("=RAND()*100000")));
```

`void put_NumberFormat(VARIANT& newValue)`

: 해당하는 범위에 숫자 표현형태 설정.

```
ex) range.put_NumberFormat(COleVariant(_T("0.0")));  
    range.put_NumberFormat(COleVariant(_T("0.000")));  
    range.put_NumberFormat(COleVariant(_T("$0.00")));
```

`LPDISPATCH get_EntireColumn()`

: 해당하는 범위의 전체 컬럼 정보를 얻음.

`VARIANT AutoFit()`

: 해당하는 범위의 자동 간격 정렬

```
ex) range = sheet.get_Range(COleVariant(_T("A1")), COleVariant(_T("D1")));  
    cols = range.get_EntireColumn();  
    cols.AutoFit();
```

Demo



Q & A