C:\sonji\SW 자료\VC++ 팁.docx 에 하기 내용 저장

여기에 없는 사항은 C:\sonji\SW 자료\MFC 자주쓰이는 함수들.docx 참고해 볼것.

**< VC++ 단축키 >**

Ctrl+ E: Match Brace

F12 : Browse go to Definition

Shift +F12: Browse go to Reference

Ctrl+Shift+F8 : Select Column

Ctrl+Shift+F2 : Clear all Bookmark

F2 : Next Bookmark

Shift +F2 : Prev Bookmark

Alt +F2 : Bookmarks

Ctrl +F2 : Bookmark Toggle

**< 워드에서 탭설정 >**

단락설정 -> 탭 버튼 클릭 -> 기본탭 2글자로 설정 -> 확인.

**< MFC에서 클래스 사라졌을 때 >**

프로젝트를 닫고 다음의 파일을 삭제 후 다시 프로젝트를 실행하면 클래스를 찾는다

\*.ncb 또는 \*.positions

**< MFC Help 검색 >**

대소문자 구분없이 입력. 구두점 무시. an, and, as, at 등의 조사 무시, a~z, 0~9 숫자 입력 가능

Help/Search 로 구동

검색창에서 "draw ellipse" : 두개 이상의 단어 문장 검색은 따옴표로 묶는다.

draw\* : 와일드카드 가능.

draw? : 임의의 한 문자 가능

AND, OR, NOT, NEAR 논리연산자 사용 가능

ex) print near paper => print와 near가 8단어 이내에 있고 한문장 안에 둘다 있는 항목이 검색됨.

**< 코드를 작성할 위치 >**

다음의 가상함수들을 재정의...

OnDraw(): 그릴때

OnInitialUpdate: 뷰 초기화

OnUpdateCommandUI : 메뉴 항목 관리

DeleteContent : 문서를 초기화

**< 코드 자동정렬 >**

: Ctrl +A 로 전체 선택후 Alt+F8 또는 Ctrl + K, F

아무튼 VisualStudio 자동 정렬 단축키는 Alt+F8 아니면 Ctrl + K, F

vim 의 경우는 Shift+V 로 전체 선택 후 '=' 키로 자동정렬 가능

**< Preprocessor define 추가>**

Project -> Settings... -> C/C++ 에서 Category를 Preprocessor로 맟추고

Preprocessor definition에 추가

출처: <http://funnytimes.tistory.com/2> [자전거 타는 개발자]

#ifdef \_SON\_PC

#ifdef \_SON\_DBG

#ifndef \_\_NO\_ERR\_RETURN\_\_

\_\_PC\_TEST\_\_

,ICP\_DAS\_CARD

**< 빌드 모드(Debug <-> Release) 변경하기 >**

Build -> Set Active Configuration

**< 콘솔에서 CString 쓰기 >**

Project Setting -> General -> Use MFC in a Shared DLL 설정

Project Setting -> C/C++ -> Category(Code Generation) ,User run-time(Debug Multithreaded, MTD)

**< 코드 프로젝트, 코드 구루 > ????**

[**http://www.codeproject.com**](http://www.codeproject.com)

[**http://www.codeguru.com**](http://www.codeguru.com)

프로그래밍 테크닉.

<http://bab2min.tistory.com/category/%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%98%EB%B0%8D/%ED%85%8C%ED%81%AC%EB%8B%89>

**< 윈도우즈에서의 파일명 대소문자 구분 >**

윈도우는 파일명의 대소문자를 구분하지 않지만, 대소문자를 보존해 주기는 한다고 함. (Case Preserving). 윈도우의 CreateFile()은 대소문자를 구별하지 않음.

#include "stdafx.h"

#include <afx.h>

#include <afxwin.h>

**\* ctags -R .**

**\* cscope -b -R**

**\* 파일의 라인수 구하기**

**cmd> find /v /c \*.cpp \*.h “”**

grep과 wc를 써서 라인수를 세었는데 윈도우에서는 명령어가 없어서 찾아보았다.

정규식을 찾는 findstr과 파일 검색하는 find를 파이프로 묶어 사용하는 방법이 있었다.

findstr /R /N "^" \*.\* | find /C ":"

\*.\* 대신에 \*.cpp 등으로 사용하면 된다.

**\* c:\> netstat -na | findstr "LISTEN" | find /c "8080"**

-> 8080 이라는 숫자를 포함하고 LISTEN 을 포함하는 netstat 결과의 라인 수를 세기

출처: <http://bwoh.tistory.com/56> [Nano Blog - 오병우]

**\* 도스에서 grep 하기 :**

**findstr "nRunFlag1" \*.,cpp**

**findstr /c "**6 Open 10 V" \*.rc

**\*\* OutputDebugString()**

DebugView app(별도 프로그램) 에서 출력을 확인할 수 있다.

주의할 점은, 실행할 때 visual stdio에서 하지 말고 .exe를 수행시켜야 출력이 나온다.

<https://docs.microsoft.com/ko-kr/sysinternals/downloads/debugview>

<http://www.jiniya.net/lecture/maso/hook9.pdf>

**\*\* TRACE()**

visual studio의 debug output 창에 출력이 된다.

주의할 점은 실행할 때 Ctrl+F5로 하면 안 되고 F5(디버그)로 수행해야 한다.

디버그모드로 수행할때는 default로 output 창이 보이진 않으므로, ALT+2 또는 View -> Output을 수행해서 Output 창을 추가로 띄워 준다.

**\* Visual C++ 6.0에는 줄보기 기능이 없음 (line number X )**

**\* Stack size 변경법**

[Project]-[Settings], at [Link] tab, select category "Output" and you can see 'Stack allocations'..

Just write a stack value you want in the edit box, 'Reserve'.

The maximum value is 0xffffffff.. or 0x200000 . 기본은 0x100000 (decimal 1GB)

"0xC00000FD: Stack Overflow" 에러 발생시에 위와 같이 stack size를 수정해서 해결이 가능하다.   
 => 이 방법보다는 stack을 많이 차지하는 함수내의 지역변수를 찾아내서 static을 앞에 추가해 주는게 더 좋은 해결법이다.

static은 전역변수처럼 heap에 올라가지만, 접근은 해당함수내에서만 가능하다.

단, 이전에 동일함수 호출시 사용했던 값이 남아 있다가 오동작을 하지 않도록 함수내에서 사용전에

초기화를 철저히 해주어야 한다.

**\* 파일로그 reference**

1. <http://ahm507.blogspot.com/2006/12/simplest-logging-class-in-c.html>

2. <http://www.infernodevelopment.com/c-log-file-class-forget-debuggers>

3. <http://kukuta.tistory.com/10>

**\* 별도 콘솔 로그**

C++ 윈도 응용 프로그램에서 별도의 콘솔 창에 텍스트를 출력하는 방법

Posted on 2014년 3월 4일

콘솔 프로그램이 아니지만 콘솔 창을 띄워서 어떤 값을 실시간으로 확인하고 싶을 때가 있습니다. 이럴 때엔 다음처럼 처리하면 됩니다.

// 콘솔 창을 생성합니다.

AllocConsole();

// 콘솔 창에 텍스트를 출력합니다.

string output\_text;

DWORD number\_of\_characters\_written;

WriteConsole(GetStdHandle(STD\_OUTPUT\_HANDLE), output\_text.c\_str(),

output\_text.size(), &number\_of\_characters\_written, nullptr);

// 콘솔 창을 해제합니다.

FreeConsole();

**\* vi 에서 ^M 없애기**

<http://mwultong.blogspot.com/2007/08/vim-vi-m-m.html>

:%s/^M$//g

^M 은 Ctrl+q, Ctrl+M 을 연속으로 눌러서 간접입력해야 함.

**\* vi에서 여러개 키워드 한번에 찾기**

/Kcount\|goto => Kcount와 goto를 한번에 찾기.

**\* ACE400 암호**

ACE400

**< Visual C++ 헝가리안 표기법 >**

b BOOL형 변수(BOOL) byte형 변수 (BYTE)

d double형 변수 (double)

h 핸들(HANDLE) 변수

n int 형 변수 (INT)

p 또는 lp 포인터 변수

sz 널(NULL) 문자로 끝나는 문자열

u unsigned int 형 변수 (UINT)

w WORD(unsigned short) 형 변수

dw DWORD(unsigned long)형 변수

str CString 형 변수

clr COLORREF

m\_ 클래스 멤버변수

sizeof(int) 4

sizeof(unsigned long) 4

sizeof(DWORD) 4

sizeof(bool) 1

sizeof(WORD) 2

sizeof(double) 8

byte alignment는 4 byte 단위인 것으로 보임

64bit 머신에서는 8byte 단위로 byte alignment가 이루어짐.

int 하나짜리 structure가 있다면 size가 4가 아니라 8이 나올 수 있음.

**<Warning: skipping non-radio button in group>**

Button 을 추가 했을 때 "Warning: skipping non-radio button in group" 이 나오면

방금 추가한 버튼에 Poerperty에서 Group 속성을 추가해 준다.

'Ctrl+D' 를 눌러서 탭순서 를 확인해 보고, 현재 버튼 바로 이전의 컨트롤이

radio 버튼인지 확인한다. 라디오버튼의 range가 다음번 Group이 나올때까지인데 새로 추가한 컨트롤이 radio도 아니고 Group 체크도 안 돼 있다면 나오는 warning임

**< TEXT(" ") 와 T(" " ), L" " >**

<http://plming.tistory.com/64> => 각 타입별 요약표도 있음.

<http://x108zero.blogspot.com/2013/12/text-t-l.html>

<http://yeobi27.tistory.com/entry/MFC-%EB%AC%B8%EC%9E%90%EC%97%B4-T-%EC%99%80-L>

**< MFC :: MainFrame,Doc,View의 포인터 얻기 >**

<http://egloos.zum.com/printf/v/1940633>

**< Active X 컨트롤 개발하기 >**

<http://greenfishblog.tistory.com/233>

<http://egloos.zum.com/potato1004/v/9993178>

**< Active X 컨트롤 등록하기 >**

1. 시작 - 검색상자에서 cmd를 입력하여 나타나는 목록에 오른쪽 마우스를 클릭하여 관리자 계정으로 실행

2. 해당 xxxxx.ocx 파일이 있는 위치에서 "regsvr32 xxxxx.ocx" 수행

3. Active X 컨트롤 등록을 해제하고 싶다면 "regsvr32 /u xxxxx.ocx" 수행

**< Active X 컨트롤 삽입하기 >**

1. 추가할 대화상자 리소스을 열고 대화상자 위에서 마우스 오른쪽 버튼을 눌러

컨텍스트 메뉴가 나타나면 Insert Active X Control 메뉴를 선택. 여기에서 추가.

**< MSFlexGrid 컨트롤 >** : 이전의 GridControl을 확장시킨것

안녕하십니까? 의선농원 님

Microsoft TechNet의 Forum 사이트를 방문해 주셔서 감사합니다.

문의 하신 “Microsoft FlexGrid Control 6.0 구하기”에 대한 답변을 드리겠습니다.

제 생각에는 예상 해결 방안이 한 가지 있습니다.

1. Msflxgrd.ocx 파일을 설치되어 있는 PC or 인터넷에서 검색을 해서 파일을 다운 받습니다.

2. Active X 컨트롤을 수동으로 등록하는 방법을 참조해서 레지스트리에 등록을 합니다.

3. Visual Studio에서 사용하기 위해 도구 상자 항목 선택 대화 상자.. 를 참조해서 Control을 도구 박스에 추가를 해서 사용을 합니다.

참고 자료

ActiveX 컨트롤(.ocx)을 수동으로 등록하는 방법

<http://www.microsoft.com/en-gb/default.aspx>

<https://support.microsoft.com/ko-kr/help/146219/how-to-register-an-activex-control-ocx-manually>

Regsvr32 xxxxxx.ocx

도구 상자 항목 선택 대화 상자, .NET Framework 구성 요소 탭

<http://msdn.microsoft.com/query/dev10.query?appId=Dev10IDEF1&l=ko-kr&k=k(VS.CHOOSEITEMS.FRAMEWORKCOMPONENTS)&rd=true>

Visual Basic 6.0에서 제공되는 컨트롤

<http://support.microsoft.com/kb/194784/ko>

사용법

<http://www.weezzle.net/3285>

**< ActiveX를 프로그램에 포함시키기 >**

[**http://blog.naver.com/mirageld/150010011773**](http://blog.naver.com/mirageld/150010011773)

1. AppWizard로 프로그램 프로젝트를 만들 때 Step 3에서 ActiveX를 체크해 줌으로 써

프로그램에서 ActiveX 컨트롤 가능해짐

2. 어떤 ocx를 사용할 것인가는 Project 메뉴에서 Add to Project를 선택하고

Components and Controls 항목을 선택. 여기에서 Registered ActiveX Controls

항목을 선택하면 현재 등록되어 있는 ActiveX 리스트들이 화면에 나옴.

3. FlexGridContrl5.0을 선택하면 Grid Contrl을 컨트롤하는 소스(\*.cpp, \*.h)가 선택되고

그외 ocx용 폰트 컨트롤 클래스 COLeFont와 이미지 관리 클래스 CPicture가 함께

Project에 삽입된다. 이후 대화상자 에디터에 Grid Control이 포함된 것이 확인 가능

4.

**< MSFlex Grid 사용법 2 >**

[**http://blog.naver.com/mirageld/150010011773**](http://blog.naver.com/mirageld/150010011773)

long GetRowSel(); // 현재 선택되어 있는 세로 줄번호

long GetColSel(); // 현재 선택되어 있는 컬럼 번호

CString GetText(); // 현재 컬럼에 데이터 얻기

CString SetText(LPCTSTR) // 현재 컬럼에 데이터 쓰기

long GetRows(); // 라인의 수를 얻는다.

void SetRows(long); // 라인의 수를 설정한다.

long GetCols(); // 컬럼의 수를 얻는다.

void SetCols(long); // 컬럼의 수를 설정한다.

long GetRow(); // 현재 설정된 세로 열 번호를 얻는다. ??? GetRowSel()과 무슨차이?

void SetRow(long); // 세로 열 번호로 커서를 이동한다.

long GetCol(); // 컬럼 번호를 얻는다. ??? GetColSel과 무슨 차이??

void SetCol(long); // 컬럼 번호로 이동한다.

<http://codes.xenotech.net/207> => FlexGrid 에서 입력받기

<http://forums.codeguru.com/showthread.php?58241-How-to-edit-data-in-a-MSFlexGrid-Control>

<http://gogo.da.to/?mid=hds21&sort_index=last_update&order_type=asc&cpage=1&listStyle=gallery&document_srl=16664&category=16517>

Using the MsFlexGrid

<https://www.codeguru.com/cpp/controls/controls/gridcontrol/article.php/c5225/Using-the-Microsoft-Flexgrid-in-Visual-C.htm>

Editable MsFlexGrid

<https://www.codeguru.com/cpp/controls/controls/gridcontrol/article.php/c5291/Editable-MSFLEXGRID-ControlDerived-Class.htm>

**< 다이얼로그에 다이얼로그 붙이기 1>**

1. 부모 다이얼로그 생성

2. 자식 다이얼로그 생성

- 자식 다이얼로그 속성창에서 style은 child, border는 NONE 으로 설정 변경

3. 부모 다이얼로그의 헤더파일에서 해당 자식 다이얼로그 변수 포인터로 생성

- 부모다이얼로그 헤더 에 CChildDlg.h 를 인클루드 한다.

- CChildDlg \*m\_pChildDlg;

4. 부모 다이얼로그의 OnInitDialog에서

m\_pChildDlg = new CChildDlg;

m\_pChildDlg->Create( CChildDlg::IDD, this );

m\_pChildDlg->ShowWindow( SW\_SHOW );

m\_pChildDlg->MoveWindow( rcDlg ); // 크기는 알아서 맞게 지정

이렇게 하면 자식 다이얼로그가 부모 다이얼로그에 붙는다.

-단 프로그램을 끝낼 때 반드시 자식 윈도우를 DestroyWindow() 해 줘야 함.

출처: <http://bigmark.tistory.com/7> [마크의 맥시멈 라이프]

6. 모달리스가 아니라 모달 다이얼로그를 붙이려면

- CChildDlg m\_ChildDlg; 를 부모 다이얼로그의 멤버 변수로 추가

- 부모 다이얼로그의 버튼 function 같은 곳에서 다음을 수행

m\_ChildDlg.DoModal();

**< MFC 프로젝트명 바꾸기 >**

<http://letsfl.tistory.com/21>

1. 디렉토리는 자동 생성해 주지 않으므로 새로운 프로젝트 디렉토리와 서브디렉토리를 생성해 줘야 한다.

ex) C:\sonji\Bioptro\new 디렉토리 밑에 원본 디렉토리를 하나 갖다 놓고, 복사본 디렉토리 (ACE400\_4wGage)를 빈 디렉토리로 하나 생성

2. c:\vsrename A B c:\A

ex) vsrename ACE400Statistics ACE400Gage C:\sonji\Bioptro\new

3. 바꾼 이후에 잘못 convert 된 이름들은 파일을 직접 열어서 수작업을 해준다.

xxxxApp.cpp, xxxxApp.h xxxx.rc xxxx.clw xxxx.dsw xxxx.dsp

ex) StatisticsDialog => GageDialog .cpp, .h 파일 이름도 바꾸고 파일 내부도 들어가서 전부 키워드를 바꿔준다.

ACE400MainDlg => MainDialog

STATISTICSDIALOG => GAGEDIALOG

Statistics => Gage

STATISTAICS => GAGE

4. Resource 뷰와 파일뷰, 클래스뷰에서 이상이 없는지 확인하고 클래스위자드에도 이상이 없는지 확인한다. 클래스위자드에 문제가 있으면 xxxx.clw를 지우고 xxxx.dsw 파일을 더블클릭해서 프로젝트 올린 다음, ctrl + w 로 클래스위자드 다시 수행.

**< VC 6.0 vector 값 보기 >**

vector<int> g;

g.push\_back(123);

g.push\_back(150);

g.push\_back(300);

( VC 6.0 의 경우 )

watch 창에 g.\_First[x] 하면 됩니다.

- 3개의 값을 모두 보고싶으면

g.\_First, 3

- 사이즈를 보고 싶으면 g.size() 를 watch 창에 입력

- vector 안의 vector 값

m\_vNetList.\_First[1].vPinNo.\_First, 2

**< 각종 컨트롤 캡션에 줄바꾸기 >**

Property -> Style -> MultiLine 을 체크한다.

\* Edit Control인 경우는 MiltiLine만 체크해서는 안된다. 다음의 3가지를 모두 만족해야 한다.

- AutoHScroll: False, Horizontal Scroll: False, Multiline : True

\* Statis Control의 경우는 No Wrap = False로 설정해야 Multiline이 된다.

Center Image" 가 True이면 \n이 무시되므로 False로 한다.

관련해서 '컨트롤 자동 줄바꿈' 으로 구글검색해 보면 나옴.

**< combo 박스 >**

콤보박스를 추가한 뒤, 오른쪽 화살표를 클릭하면 콤보박스 down시의 사이즈를 조절할 수 있다.

사용자의 키보드 입력을 막고 싶으면 styles -> Sort 항목 체크 풀고, Type 하목을 Drop List로 설정한다.

**< edit 박스에 유효성검사 추가 등 >**

edit 박스를 ctrl+더블클릭해서 value형 변수를 추가한 다음, 클래스 위자드에서 해당 변수를 찾아 보면

아래쪽에 Minimu, Maximu value를 설정할 수 있다.

**< edit 박스 변경시 관련 코드 추가 >**

=> 클래스 위자드에서 해당 다이얼로그의 IDD\_EDIT\_XXX를 선택하여 EN\_CHANGE notification 관련 코드를 추가한다.

만약, 해당 다이얼로그 클래스에 edit박스가 안 보인다면, xxx.clw를 지우고 비주얼 스튜디오를 다시 시작한다

클래스 위자드를 재 시작하면, 해당 다이얼로그의 Dlg 클래스를 새로 만들것인지 물어보는데

이때 기존의 Dlg 클래스를 재 지정해 주면 된다.

**< MFC 다이얼로그 호출시 실행 메시지 순서 >**

MyApp.cpp 에서 InitInstance() 가 실행

=> InitInstance() : CMyAppDlg 객체 생성, DoModal() 함수 실행

DoModal()

PreSubclassWindow()

OnNcCreate()

OnNcCalcSize()

OnCreate()

OnSize()

OnMove()

OnSetFont()

**OnInitDialog()** 로그인 다이얼로그 CLoginDialog 객체 생성, DoModal() 함수 실행

OnWindowPosChanging()

OnMove()

OnWindowPosChanged()

OnWindowPosChanging()

OnNcActivate()

OnActivate()

**OnShowWindow()** CWnd가 감추어지거나 혹은 보여질 때 호출

OnWindowPosChanging()

OnNcPaint()

OnEraseBkgnd()

OnChildNotify()

OnCtlColor()

OnWindowPosChanged()

OnPaint()

OnCtlColor()

OnCtlColor()

OnNcHitTest()

OnSetCursor()

PreTranslateMessage()

OnMouseMove()

OnNcHitTest()

OnSetCursor()

PreTranslateMessage()

**출처:** [**http://mititch.tistory.com/65**](http://mititch.tistory.com/65) **[미티치]**

**< 참고 사이트 및 자료 >**

[**www.devpia.com**](www.devpia.com) **: 비주얼툴 국내 개발자 천국**

**codeguru.earthweb.com: 각종 컨트롤과 예제가 수준 높고 유용하게 정리됨**

[**www.codeproject.com**](www.codeproject.com) **: 제2의 코드그루**

**winapi.co.kr : Win api 강좌 및 유용하나 자료 잘 정리됨**

**Construct GameSDK OpenSource**

**책**

**- Teach yourshell 21days : 초보가자 쉽게 Visual Studio C++ 사용하도록 설명이 잘 됨**

**- visual C++ 바이블(이상엽) : 방대한 백과사전.**

**- VisualC++/MFC 프로젝트 따라하기 : 프로젝트성 예제를 죻은 설명과 소스를 제공.**

**< CString::GetBuffer 샘플>**

// example for CString::GetBuffer

CString s( "abcd" );

#ifdef \_DEBUG

afxDump << "CString s " << s << "\n";

#endif

LPTSTR p = s.GetBuffer( 10 );

strcpy( p, "Hello" ); // directly access CString buffer

s.ReleaseBuffer( );

#ifdef \_DEBUG

afxDump << "CString s " << s << "\n";

#endif

==> Debug Output 창에 다음과 같이 출력됨.

CString s abcd

CString s Hello

**< First-chance exception in ACE400Cal.exe ... >**

strTemp.Format("%s", g\_saTestCondition[nCalMode].nCC);

==> int를 %s로 출력하려고 한 것이 문제였음.

**< 싱글톤 패턴 설명 >**

<http://egloos.zum.com/sungod0/v/4310980>

싱글톤과 전역변수의 차이점.

싱글톤은 오로지 객체가 하나만이 존재할 수 있지만 전역은 같은 타입이 여러개

존재할 수도 있다. 특정 클래스의 객체가 하나만 존재해야 한다면 싱글톤이 답이다.

class Singleton {

private:

Singleton() {} // 생성자를 private으로 선언

Singleton(const Singleton& other);

public:

static Singleton& getInstance() {

static Singleton ctx;

return ctx;

}

};

**< 신규 다이얼로그 생성 >**

**VC++ 단계별 실습 288 page**

**VC++ 완벽가이드 274 page**

1. Resource View 에서 "Dialog" 폴더에서 오른쪽 마우스 버튼을 누르면 단축메뉴가 나오는데

여기서 Insert Dialog 항목을 선택 => 새로운 다이얼로그 생성됨.

대화상자의 property에서 IDD\_DIALOG\_XXX ID 항목을 설정하고, Caption을 새로 입력한다.

2. 필요한 컨트롤 들을 배치하고 각 컨트롤의 property도 설정한다. (ID, 캡션)

3. 대화상자와 연결할 클래스를 만든다.

- 추가한 대화상자를 클릭하고 클래스위자드를 실행 (Ctrl+w) 하면 'Adding a Class' 대화상자가

나타난다. 'Create a new Class'를 선택하고 OK하면

**- '**New Class' 대화상자가 나타난다. 새 클래스의 이름을 쓰고 기저 Class가 CDialog인지 확인후

OK를 선택하면 클래스가 새로 생성된다.

4. 기존 메인윈도우에 버튼을 하나 만들고, 버튼을 더블 클릭해서 "OnButton해당 다이얼로그" 함수를 생성한다.

5. 기존 메인클래스에 새로운 클래스의 멤버변수를 추가한다.

- ctrl+w 의 'Member Variables' 탭에서 원하는 컨트롤의 ID를 선택하고 'Add Variable' 수행

- 또는 다이얼로그상에서 ctrl 키를 누른 상태에서 control을 더블클릭. 하여 value형 멤버변수를 추가한다. **VC++ 완벽가이드 229 page**

**< VC++ 6.0 에서 include 경로 추가 >**

- 프로젝트에 추가

[Project] -> [Settins..]->[C/C++]-> Category: Preprocessor -> Additional include directories에 추가

- 자체 Visual Studio에 추가

[Tools]-> [Option] ->[Directories] -> Show directoried for : include files -> 아래쪽 Directories: 에 추가

**< INK : fatal error LNK1181: cannot open input file "mfc42u.lib">**

Error executing link.exe.

컴파일 하실때 Release, Debug 를 설정할 수 있는 곳에서, Unicode Compile 옵션으로 되어있는 것 같네요 :)

그 부분을 Unicode 가 아닌 Release, Debug 로 설정해서 Rebuidl All 을 해 보세요

**< 클래스위자드에 클래스가 안 보일 때 >**

일단 VC 를 닫고 프로젝트 폴더로 가서 확장자가 .clw 인 파일을 삭제한다.

그리고 프로젝트를 다시 열어서 클래스 위자즈(Ctrl + W)를 띄우면 메세지 박스가 뜬다. 그럼 확인 를 눌러 주고 나면 아래와 같이 추가할 소스 창이 뜬다. 그럼 딴거 볼 필요없이 그냥 Add All 을 눌러 OK 버튼을 누른다.

클래스 위자드에 전부 추가되어진 클래스 들이 보일것이다.

**< custom control 사용하기 >**

[**http://www.codeproject.com/KB/miscctrl/customcontrol.aspx**](http://www.codeproject.com/KB/miscctrl/customcontrol.aspx)

[**http://blog.naver.com/PostView.nhn?blogId=linux313&logNo=150030826739&redirect=Dlog&widgetTypeCall=true**](http://blog.naver.com/PostView.nhn?blogId=linux313&logNo=150030826739&redirect=Dlog&widgetTypeCall=true)

**< CString 사용 예제 >**

<http://egloos.zum.com/potato1004/v/7446586>

<http://blog.naver.com/PostView.nhn?blogId=hextrial&logNo=60199306242>

**< 다이얼로그 복사하기 >**

1. VS 에서 열기를 누르고 rc 파일을 선택합니다.

2. 다이얼로그에서 복사할 다이얼로그를 찾습니다. (열지말고 트리컨트롤에서 해당 항목 선택)

3. 복사할 다이얼로그 항목을 선택하고 CTRL 키를 누르면서 현재 작업중인 워크스페이스

리소스 탭에 드래그합니다!!!! (마우스포인터에 + 모양 이미지 떠야할꺼예요)

4. 다이얼로그 항목에 드랍합니다!

5. 옮겨야 할 다이얼로그의 xxxxDialog.h와 xxxxDialog.cpp 도 가져와서 프로젝트에 추가한다.

xxxxDialog.cpp 를 열어서 위에서 두번째 프로젝트 헤더 include 문을 현재 프로젝트의 헤더로 바꾸어 준다.

6. Main dialog를 바꾸는 경우라면 MainApp에 해당하는 파일을 열고

#include "xxxxDialog.h" 를 추가한다. 그리고 pMainApp 의 포인터를 추가한 Dialog의 타입으로 바꾸어 준다.

7. VS를 저장하고 끄고, 프로젝트의 .clw 파일을 지우고, .dws 를 클릭해서 다시 수행,

Ctrl+w 를 눌러서 클래스위자드를 수행하면, clw를 다시 만들거냐는 다이얼로그 생성된다.

'Add All' 을 선택하고 clw를 다시 만든다.

**< GetLastError() 사용법 >**

DWORD dwError;

TCHAR errMsg[1024];

HANDLE m\_hMapping = CreateFileMapping( hFile, // hFile: 파일 매핑 핸들

NULL, // 보안속성

PAGE\_READWRITE, // read/write option flag

0, // 상위 메모리 offset, max size

1500000000, // 실제 사용할 메모리의 max size (2G)

NULL ); // mapping 파일명

if(m\_hMapping == NULL) {

dwError = GetLastError();

FormatMessage(FORMAT\_MESSAGE\_FROM\_SYSTEM, NULL, dwError, 0, errMsg, 1024, NULL);

printf("<ERROR=%d>%s: CreateFileMapping, m\_hMapping=%d\n", dwError, errMsg, m\_hMapping);

getchar();

return 0;

}

**< 공백을 포함한 문자열 입력 방법 >**

[**http://blog.daum.net/\_blog/BlogTypeView.do?blogid=09ehJ&articleno=18230445&categoryId=789155**](http://blog.daum.net/_blog/BlogTypeView.do?blogid=09ehJ&articleno=18230445&categoryId=789155)

[**https://m.blog.naver.com/PostView.nhn?blogId=dks444&logNo=220644628425&proxyReferer=https%3A%2F%2Fwww.google.co.kr%2F**](https://m.blog.naver.com/PostView.nhn?blogId=dks444&logNo=220644628425&proxyReferer=https%3A%2F%2Fwww.google.co.kr%2F)

[**http://egloos.zum.com/mcchae/v/10685164**](http://egloos.zum.com/mcchae/v/10685164)

char str[80];

scanf("%[^\n]s", str); // \n 문자를 만나기 전까지 입력받기

scanf("%79[^]n]s", str); // \n 문자를 만나기 전까지 입력받는데 79자까지만 입력받기

scanf("%[a-zA-Z]s", str); // 영문자만 입력받기

ps: gets()는 입력받을 글자수를 제한할 수 없으므로 제한을 두고 싶을 땐 scanf() 함수를 사용.

**< scanf, fscanf, sscanf 문제점 >**

[**http://wpollock.com/CPlus/PrintfRef.htm**](http://wpollock.com/CPlus/PrintfRef.htm)

- %d, 읽는 데에는 문제가 없으나 %f read가 잘 안된다. %f는 %s로 read한 다음 atof()로 바꾸어 줘야 한다.

- %d, %s => %d와 %s 사이의 , 와 ' ' 은 모두 구분자이다. 만약에 ':'를 구분자로 쓰고 싶다면

%d:%d 이런식으로 써주면 된다.

그런데 %s의 경우에는 구분자를 ' ' 공백 외에는 제대로 쓸 수 없다.

- scanf는 입력 버퍼를 자동으로 청소해 주지 않고 뒤에 '\n' 이 붙어 있을 위험이 있다.

따라서 CString 같은 걸로 수백번 반복해서 read하고 Empty() 함수를 호출해 주지 않으면 buffer overflow가 생긴다.

**< 파일에 구조체를 읽고 쓰기 >**

[**https://dojang.io/mod/page/view.php?id=627**](https://dojang.io/mod/page/view.php?id=627)

**쓰기**

[**https://dojang.io/mod/page/view.php?id=628**](https://dojang.io/mod/page/view.php?id=628)

**읽기**

[**https://dojang.io/mod/page/view.php?id=629**](https://dojang.io/mod/page/view.php?id=629)

**< CFile 사용법 >**

[**http://www.tipssoft.com/bulletin/board.php?bo\_table=FAQ&wr\_id=900**](http://www.tipssoft.com/bulletin/board.php?bo_table=FAQ&wr_id=900)

**< fopen binary 사용법과 CFile 사용법>**

[**http://tipssoft.com/bulletin/board.php?bo\_table=story&wr\_id=12393**](http://tipssoft.com/bulletin/board.php?bo_table=story&wr_id=12393)

CFile이 CStdioFile 보다 상위 class 개념입니다.

CFile 은 Binary 를 위주로, CStdioFile은 Txt 를 위주로 써 한다고 보시면 됩니다.

사실 CFile 에서도 txt 를 사용하는데 지장은 없지만, 디버깅 상의 문제로 CStdioFile 을 써주는것이

더 낫다고 봅니다. ( 좀더 완벽에 가까우려면,,)

**<CFile로 text 저장하기 >**

저장할 데이터를 string으로 만든다음 fwrite하던지 CArchive로 연결해서 text로 저장.

VC++ 완벽가이드 412p. (텍스트 파일로 내보내기) 참조.

CFile file;

file.Open(dlg.GetPathName(), CFile::modeCreate|CFile::modeWrite);

CArchive ar (&file, CArchive::store);

...

ar.WriteString(str);

ar.WriteString("\r\n");

**< CFileDialog 사용법 >**

<http://www.tipssoft.com/bulletin/board.php?bo_table=FAQ&wr_id=1533>

[**http://blog.saja92.net/entry/CFileDialog%EC%9D%98-%EB%AA%A8%EB%93%A0%EA%B2%83**](http://blog.saja92.net/entry/CFileDialog%EC%9D%98-%EB%AA%A8%EB%93%A0%EA%B2%83)

**< conditional break 넣기 >**

1. 원하는 위치에 커서를 놓고 Ctrl + 'b' 를 누른다.

2. Location tab에서 'Break at:' 의 오른쪽 화살표를 클릭해서 현재 라인을 선택

3. 조건입력 다이얼로그에 원하는 조건을 입력한다 (ex: row==450)

**< memory leak 잡기 >**

<http://egloos.zum.com/tiger5net/v/4743365>

<http://artwook.tistory.com/47>

<http://blog.naver.com/goli81/10006754359>  
<http://blog.naver.com/nfox_79/120029389752>  
<http://blog.naver.com/ahnshy/110008986986>  
<http://blog.naver.com/nfox_79/120029878624>

**< CTreeCtrl 사용법 종합 >**

<http://saack.tistory.com/62>

<http://bleedmin.blogspot.kr/2012/04/ctreectrl_11.html> => 위와 내용 같음.

<https://m.blog.naver.com/PostView.nhn?blogId=dbsauddus&logNo=30084603119&proxyReferer=https%3A%2F%2Fwww.google.co.kr%2F>

**< CArray 사용법 >**

<http://codes.xenotech.net/45>

**< Visual C++ 6.0 팁 >**

<http://dmaxm11.tistory.com/127>

<http://www.sosori.com/2010/03/visual-c-%EC%9C%A0%EC%9A%A9%ED%95%9C-%ED%8C%81-%EC%A0%95%EB%A6%AC.html>

**< RGB 코드값 >**

<http://lahng.tistory.com/entry/%EC%83%89%EC%83%81%ED%91%9C-RGB-%EC%BD%94%EB%93%9C%EA%B0%92>

**< 툴팁 설정하기 >**

<http://www.tipssoft.com/bulletin/board.php?bo_table=FAQ&wr_id=293>

**< 메인 아이콘 바꾸기 >**

1. 아이콘 생성툴로 먼저 바꿔야 할 My\_xxx.ico를 생성한다.

<https://stackoverflow.com/questions/3384112/error-rc2176-old-dib-in-res-icon3-ico-pass-it-through-sdkpaint> 참고

1) My\_xxx.bmp 파일을 확보한 다음, GIMP 툴로 편집을 수행한다.

2) GIMP에서 File-> Export As.. 수행 (<https://www.gimp.org/downloads/> 에서 윈도우용 GIMP 설치 후)

3) 하단의 파일 유형을 'Microsoft Windows 아이콘'으로 선택 후 '내보내기' 버튼 클릭하여 My\_xxx.ico 파일로 저장한다

2. xxx.rc 파일을 열어서 IDR\_MAINFRAME ICON "res\\xxx.ico" 를 "res\\My\_xxx.ico"로 변경한다.

3. 프로젝트를 다시 빌드한다.

**< Release로 컴파일하기 >**

1. Build -> Set Active Configuration 선택 후 Release 모드로 바꾼다.

2. Debug 모드에서 추가했던 include 디렉토리가 있다면 다시 설정한다.

[Project] -> [Setting..] -> [C++ 탭] -> Category: Preprocessor 선택 후

'Additional include derectories: " 에 './gridctrl\_src/' 를 추가하였다.

3. 프로젝트를 다시 빌드

**< CArray 관련 자료 >**

<http://wwwi.tistory.com/60>

<http://www.dreamy.pe.kr/zbxe/CodeClip/3768914>

**< 동적 배열 관련 자료 >**

<https://kldp.org/node/50137>

C++ 프로그래밍책 . new, delete관련 부분 참고. 특히 deep copy 복사생성자에 대한 부분 191p

**< 윈도우에서 메모리 사용량 확인하기 >**

<http://ospace.tistory.com/514>

**< 파일 용량 확인 >**

<https://m.blog.naver.com/PostView.nhn?blogId=esuom&logNo=120072302182&targetKeyword=&targetRecommendationCode=1>

**< 외부 프로그램 실행시키기 >**

<http://igotit.tistory.com/entry/%EC%99%B8%EB%B6%80%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%A8-%EC%8B%A4%ED%96%89%EC%8B%9C%ED%82%A4%EA%B8%B0-CreateProcess-ShellExecute-WinExec>

::ShellExecute() 의 여러가지 사용법

<http://www.tipssoft.com/bulletin/board.php?bo_table=FAQ&wr_id=779>

**<Git 관련 자료 >**

Git 기초 사용법

<http://programmingskills.net/archives/384>

<http://marklodato.github.io/visual-git-guide/index-ko.html>

Git을 시작하기 위한 간편 안내서

<http://rogerdudler.github.io/git-guide/index.ko.html>

HEAD: 현재 브랜치

git help <verb> 또는 git <verb> --help : 도움말보기

git config --list : 현재 git config 보기. 또는 gvim ~/.gitconfig => gvim으로 파일 직접보기

**git init .** : git Repository 만들기

불필요한 파일을 add 하지 않게 먼저 .gitignore를 해당 디렉토리에 포함시키고 add해야 한다.

**git add** <files> :

**git rm <file >**  : git track에서 파일을 삭제. 실제 파일도 삭제

**git rm --cached <file >** : git track에서만 파일을 삭제. stage에서 제외시킴. 실제 파일은 삭제 안함.

**git commit -m "설명"**

git commit -a : add 하고 바로 한번에 commit 하기 (working에서 stage 안 거치고 바로 Repository로 이동)

**git commit -am "설명"** : add하고 바로 설명까지 한번에 commit 하기

단, -am 은 -a나 git add 로 1회 이상 add가 된 파일에 대해서만 수행된다.

즉, untracked 파일을 tracked로 바꾸지는 않는다.

git commit --amend : 다시 commit하기

commit 했는데 빠진 file을 발견했을 때 git add <빠진 file> 하고 바로 git commit --amend하면

아까 commit 에 그대로 추가된다. commit 메시지를 잘못 적었을 때에도 사용 가능하다.

===> push 한 다음에는 이거하면 안된다. 다음에 또 push하려고 할 때 pull하라고 함. 그러면서

소스가 오염되는 문제가 발생함. pull을 할때에는 merge가 일어나므로 반드시 백업을 하고 push 하자.

git commit에 사용되는 기본 에디터 바꾸기 : git config --global core.editor "vim"

**git checkout commit번호** : **특정 commit의 모든 파일상태로 통째로 돌아가기 \*\*\*. commit을 선택하기!!**

add 했던 파일은 없어지고 여러개의 파일 상태가 특정 commit의 상태로 돌아감. <HEAD>가 가리키는 브랜치를 옮기는 것.

이 상태에서 실수로 어떤 파일들을 수정하고 다시 checkout master를 했다면 돌아가지 못할 수 있다.

git stash를 이용하면 원래의 commit 상태로 청소할 수 있고 이후에 checkout master가 가능하다.

**git checkout master** : **가장 최근으로 다시 돌아가는 것. \*\*\*** (<Head>가 가장 최근의 브랜치를 가리키게 한다)

git stash : 작업중인 내용을 보류한다. 일시적인 백업을 하고 워킹 디렉토리를 HEAD 상태로 돌린다. (책갈피..)

git stash list : 보류중인 내용을 보여준다.

git stash drop : 가장 최근에 save한 stash를 버린다.

git stash drop stash@{번호} : 특정 stash를 버린다.

git stash clear : 전체 stash를 지운다.

git stash show stash이름 : 특정 stash의 자세한 내용을 조회

git stash pop : 가장 최근에 save한 stash를 워킹 디렉토리에 다시 적용. 해당 항목은 stash list에서 삭제

git stash apply stash이름 : stash를 워킹디렉토리에 적용하지만 stash list에서 삭제하지는 않는다.

git checkout -- <file> : stage취소 & 작업내용도 취소

git checkout HEAD <file> : stage취소 & commit 취소&작업내용도 취소 (Repository에서 stage 안거치고 바로 working 영역으로)

원래 checkout은 브랜치를 선택하는 명령어임.

**git reset**  : **commit을 삭제하는 방법**. \*\*\*

<https://opentutorials.org/module/3762/22596> 에서 이해할 것.

**git reset --hard commit번호 : 이 commit 버전이 되겠다. HEAD와 master가 모두 이 commit이 된다.**

**이 commit을 날리겠다는 의미가 절대 아님. 주의.**

**reset --hard 는 기존의 stage와 working 파일까지 다 날리고 이 commit으로 바꾸어 버린다.**

**만약 commit만 지우고 수정하던 stage, working은 그냥 두고 싶다면? --soft나, --mixed를 사용.**

**만약 협업중이라면 공유된 버전은 절대 reset하면 안 된다. \*\*\*\*\***

**git reset --mixed :** 최종 결과는 유지하되, 최근 2개의 commit 과정만 지우고 싶을 때 사용.

마지막 commit은 지정한 commit으로 바뀌고 그 뒤에 했던 commit들은 사라진다.

현재 working copy의 상태는 commit 되지 않은 상태로 유지된다.

**git revert : commit(버전)을 되돌리기, reset 보다 세련되고 삭제도 하고, 보존도 가능하다. 단, 바로 직전의 것만 해야한다.**

**R4, R3라는 commit이 있다면 revert R4를 해야 이전 버전인 R3로 돌아간다.**

**reset의 경우는 reset R3를 해야 R3로 돌아감.**

ex) **git revert R4의commit번호**

**=>** git log 해 보면 log에 R4, R3 는 기존에 다 남아 있고, history도 보이며, 내용만 R3로 바뀜.

**R3와 내용만 같은 새로운 commit 번호의 commit이 현재 master로 생긴다.**

ex) R4, R3, R2, R1이 있는데, R1으로 돌아가고 싶다면, 곧바로 revert R2하면 안된다. (충돌 발생!)

revert 2는 R1-> R2로 의 변경만 돌리므로, R3, R4의 변경상황에 대해서는 대책이 없다.

revert R4-> revert R3 -> revert R2를 순서대로 해줘야 R1으로 돌아간다.

git reset HEAD^ : 최근의 commit 자체를 취소하기

git reset <files> : Repository(Commit 된) 에서 file들을 stage로 가져온다. (commit 취소)

git reset --mixed HEAD <files> 와 같다.

git reset HEAD <file > : add의 반대. 수정해서 add된 파일을 stage에서 제거만. 수정상태는 그대로임.

git unstage <files > 로 만들었음.

git reset eb43bf file.txt : 특정 커밋의 특정 file만 stage로 가져오기

git checkout f6f9f2 aaa.cpp : 특정 커밋의 특정 file만 working 디렉토리로 바로 가져오기 (commit도 없어진다)

reset과 checkout의 차이 :

<https://git-scm.com/book/ko/v2/Git-%EB%8F%84%EA%B5%AC-Reset-%EB%AA%85%ED%99%95%ED%9E%88-%EC%95%8C%EA%B3%A0-%EA%B0%80%EA%B8%B0>

git status : 현재 git 상태를 보기 :

**git status -s** : 현재 git 상태를 보기. 보기 쉬움.

**git diff : 수정파일과 add된 stage의 파일을 비교해 준다.**

여기서 수정파일을 버려야 한다면 **git reset --hard** 하면 수정한 내용이 휙 사라지고 이전 stage의 상태로 돌아간다.

여기서 commit 된 파일들의 모든 diff 결과를 한번에 보고 싶다면 **git log -p** **--stat**한다.

git diff --staged : add된 stage의 파일과 commit 된 저장소의 파일을 비교해 준다.

git 파일 지우기 : git rm 실행후 git commit 으로 확정해야 한다. 실제로 file도 지워진다.

gitk : 변경 history UI로 확인.

**git ls-files -t**  : tracking 리스트를 조회한다.

**git log --oneline** : commit history 한줄로 보기

git log -p -2 : 각 commit의 **diff 결과를** commit 2개까지만 보여준다.

**git log --stat**  :  **commit 마다 포함된 파일들의 리스트를 보여준다.**

**git log -p --stat : 이렇게 혼합해서 쓰면 더 좋다.**

git tag -a v1.4 -m "my version 1.4" : 현재 commit에 tag 만들기 (복잡한 commit 이름 대신 사용가능 )

git tag -a v1.2 9fceb02 : git log --oneline 으로 먼저 commit 번호를 확인하고 해당 commit에 tag 달기

git show : tag 상태 보기

git show v1.2 : 특정 tag 상태 보기

**git remote add origin** [**https://github.com/sonji1/4wStatAnalysis.git**](https://github.com/sonji1/4wStatAnalysis.git): origin이라는 이름의 remote 저장소 주소를 add(등록)

이거하기 전에 github에서 오른쪽 상단 '+' 클릭해서 'New Repository' 생성해야.

생성이 되면 pc의 원래 소스에서 git init-> git add -> git commit 을 수행하고

github에서 주소를 긁어서 위와 같이 수행해야 한다.

git remote 쳐보면 origin 이라는 이름의 원격저장소에 연결됐다고 나옴.

**git remote -v 하면 원격 저장소 주소도 볼수 있음.**

**git push -u origin master** : origin(remote 저장소)에 업로드 (sonji1/jws92097 입력해야 함)

**git clone**  : 새 디렉토리를 만들고 원격 저장소의 해당 정보를 똑같이 복사해 온다. (복제)

git clone <https://github.com/sonji1/4wStatAnalysis.git> => 이렇게만 하면 이걸 수행한 디렉토리에 4wStatAnalysis 디렉토리가 새로 생성

**git pull**  : 원격저장소의 데이터를 가져온다. push의 반대. git fetch와 git merge를 같이 하는 것.

git checkout -b : 브랜치를 만들고 체크아웃까지 한번에 수행

git branch : 브랜치 만들기

master : default 브랜치 name

git branch -d : 브랜치 지우기

git branch --no-merged : 브랜치가 자주 만들어지고 지워지기 때문에 존재하는 브랜치중에 머지된것과 안된 것

git branch --merged 구분해야 함.

가능하면 merger보다는 rebase를 사용하는게 history관리에 깔끔하다. (rebase는 한 branch를 기준으로 삼고 merge)

**< git tips >**

궁금한 점 찾아보기 좋은 문서

<https://github.com/mingrammer/git-tips>

**< git 단축어 >**

git co : git checkout

git ci : git commit

git br : git branch

**git st** : git status

**git sts** : git status -s

**git unstage** <file > : git reset HEAD <file > : add의 반대

**git last** : git log -1 HEAD : 마지막 수정사항만 확인하기

**< 자주 쓰는 git 명령어 >**

**git init .**  : git Repository 만들기, 이거 하고 바로 .gitignore를 해당 디렉토리에 갖다둔다.

**git add . :** 현재 폴더와 하위폴더의 모든 파일을 저장대상으로 등록 (이거 이전에 .gitignore 파일 만들어둬야 함)

**git commit -m "설명"**

**git status -s** : 현재 git 상태를 보기. 보기 쉬움.

**git ls-files -t** : 현재 tracking 리스트를 조회한다. (주의, ls-files를 다 붙여서 써야 함. )

**git log --oneline** : commit history 한줄로 보기

**git diff**  : 수정된 파일과 add된 기존 파일을 비교해 준다. working과 stage 비교

**git diff --cached** : stage와 commit을 비교

**git log -p --stat**  :  **commit 마다 포함된 파일들의 리스트를 보여준다.**

**git tag -a v1.2 9fceb02**  : git log --oneline 으로 먼저 commit 번호를 확인하고 해당 commit에 tag 달기

**git remote add origin** [**https://github.com/sonji1/ACE400Cal.git**](https://github.com/sonji1/ACE400Cal.git) (github에서 new repository 만든후에 이거 수행)

**git push -u origin master**

**git checkout f6f9f2 <file>** : 특정 커밋의 특정 file만 working, stage로 바로 가져오기

(하기 전에 카피를 떠 놓자. commit도 없어진다,

**git reset eb43bf <file>**  : 특정 커밋의 특정 file만 stage로 가져오기 (이거 하면 git diff 또는 gitk 로 내용비교가 가능하다)

**git rm --cached <file >** : git track에서만 파일을 삭제. stage에서 제외시킴. 실제 파일은 삭제 안함.

git 한글 설정

git config --global i18n.commitEncoding cp949

git config --global i18n.logOutputEncoding cp949

set LESSCHARSET=latin1

git config --global core.quotepath false

git config --local core.quotepath false

한글처리를 위해 8bit 입력을 활성화 해주는 역활

.inputrc 혹은 /etc/inputrc

--------------------------

set output-meta on

set convert-meta off

콘솔에서의 한글 입출력과는 별개로 ls 의 한글 출력이 깨지는 문제

.bashrc

-----------------------------------

alias ls='ls --show-control-chars'

Git-scm: Git 매뉴얼

<https://git-scm.com/book/ko>

<http://git-scm.com/book/ko/v1>

<http://programmingskills.net/archives/384>

[http://marklodato.github.io/visual-git-guide/index-ko.html#conventions](http://marklodato.github.io/visual-git-guide/index-ko.html)

Git 온라인 저장소 만들기: (GITHUB)

<https://www.a-mean-blog.com/ko/blog/MEAN-Stack/%EA%B0%9C%EB%B0%9C-%ED%99%98%EA%B2%BD-%EA%B5%AC%EC%B6%95/Git-GitHub-%EA%B0%84%EB%8B%A8-%EC%82%AC%EC%9A%A9%EB%B2%95>

Learn Git Branching

<http://pcottle.github.io/learnGitBranching/>

Code School - Try git

<https://try.github.io/levels/1/challenges/1>

Atlassian git tutorial (영문)

<https://www.atlassian.com/git>

한글 슬라이드

.버전 관리를 들어본 적 없는 사람들을 위한 DVCS-Git

: <http://www.slideshare.net/ibare/dvcs-git>

.Svn 능력자를 위한 git 개념 가이드

<http://www.slideshare.net/einsub/svn-git-17386752>

.Subversion vs. Git: 엄청난 간략 비교

<http://www.slideshare.net/ienvyou/subversion-vs-git-42605130>

**< Visual studio로 디버깅하기 . dump, pdb >**

<http://ncanis.tistory.com/364>

**< 파이썬 관련자료 >**

<file:///C:/Users/User/Downloads/jump2python.pdf>

**< 자식 다이얼로그에서 다른 자식 다이얼로그로 메시지 보내기 >**

<http://zadd.tistory.com/24>

**< StarUML 사용 >**

<https://kldp.org/node/153082>

1. NEW 프로젝트 생성

2. IMPORT > Frame Work -> Microsoft Foundation Class 6.0 (라이브러리 포인트 얻기)

3. Implemention 선택

4. Model Exploer 에서 Implemention 마우스 우측 버튼 클릭으로 선택

5. 팝업 메뉴에서 C++ - Reverse Engineer 선택

6. 개발중인. 프로젝트 폴더 선택

7. Implemention 선택 -> Reference Data member Creation 을 Association으로 선택

8. 생성된 Implementation 화면을 보는데. 너무 커서 어려우니까. 화면 중앙의 우측 밑에 작은 버튼을 선택하면.

미리보기 화면으로 원하는 좌표를 이동하며 볼 수 있다.

화면이 너무 크면 StarUML상에서 다 볼수가 없으므로, File -> Export Diagram 에서 jpg로 export하면 전체를 다 볼 수 있다.

.chm 설명 문서도 있다.