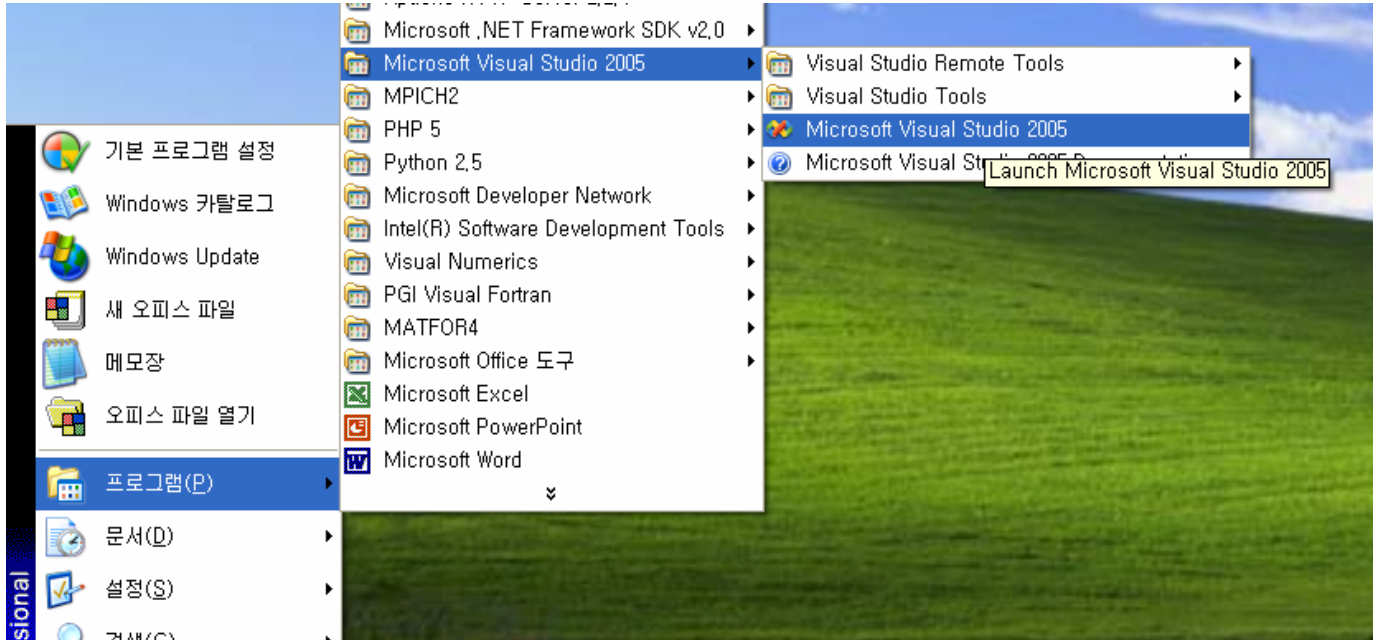


# 1. 영문 Visual Studio 2005

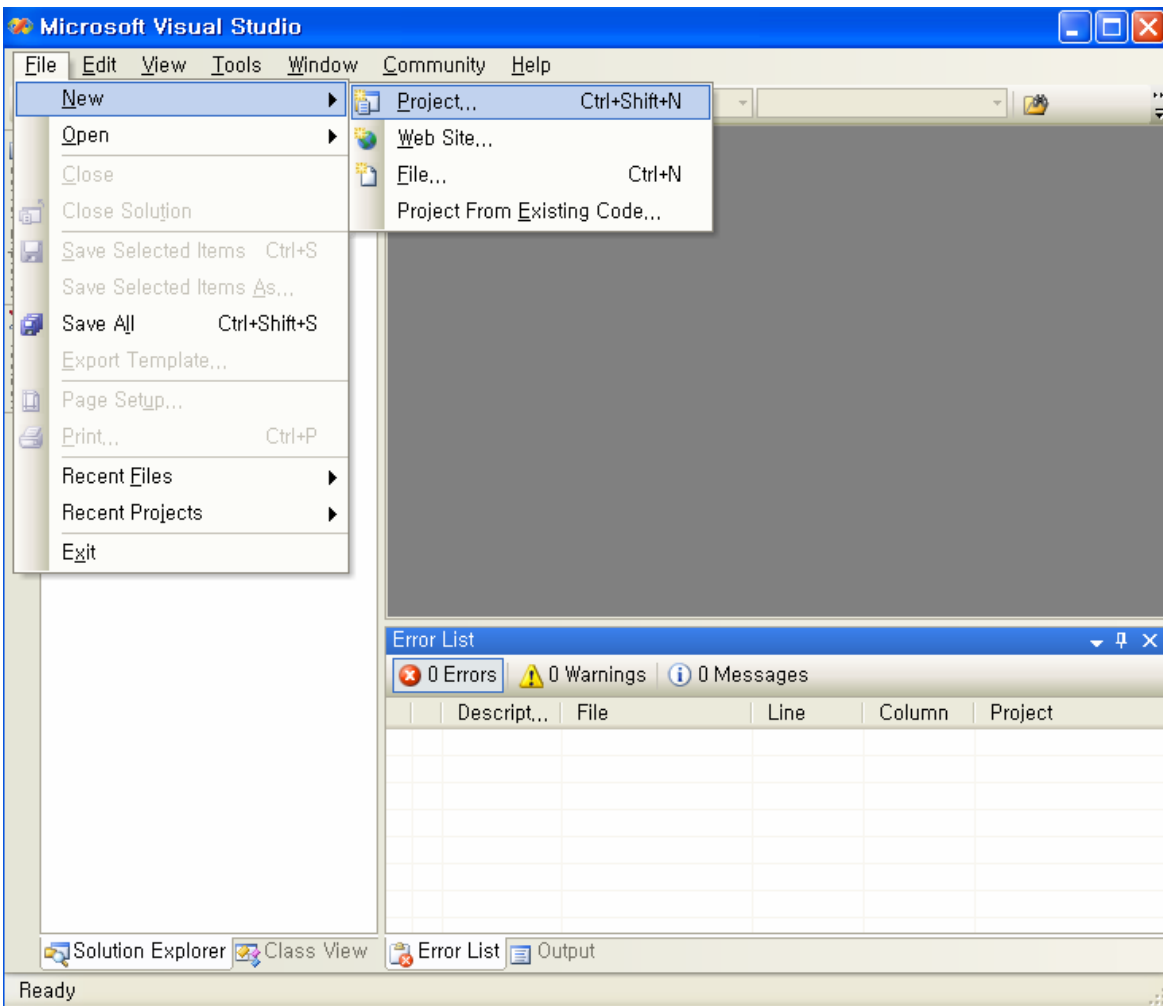
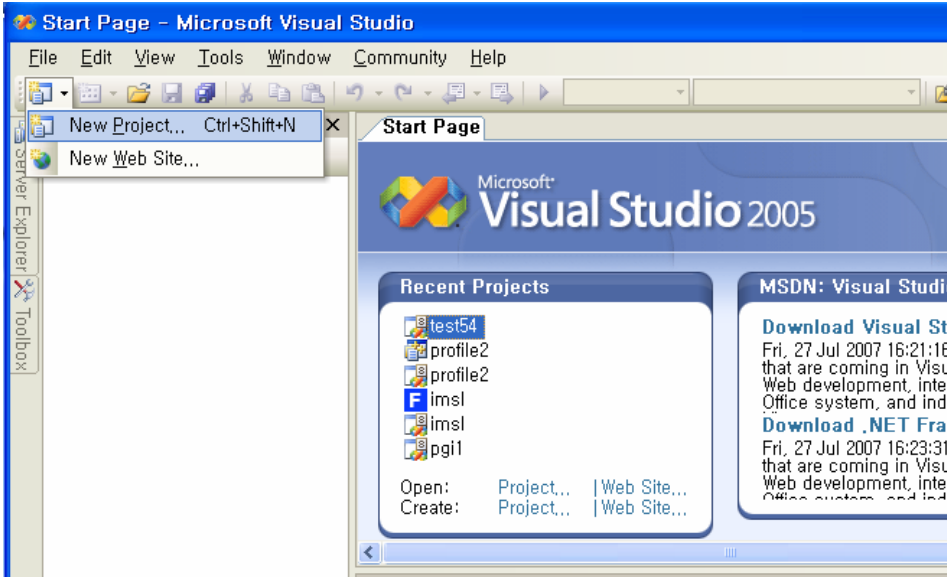
Visual Studio 2005 설치

Visual Studio 2005 시작하기

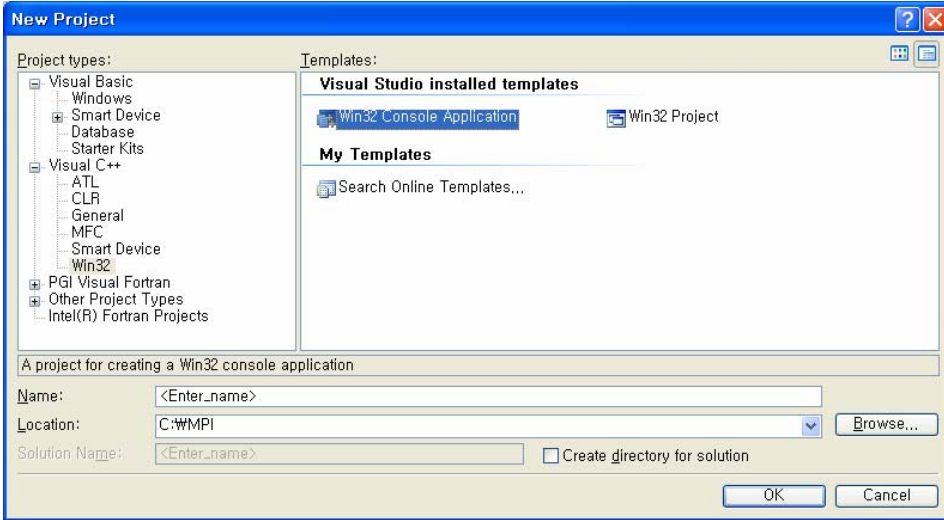


## new project

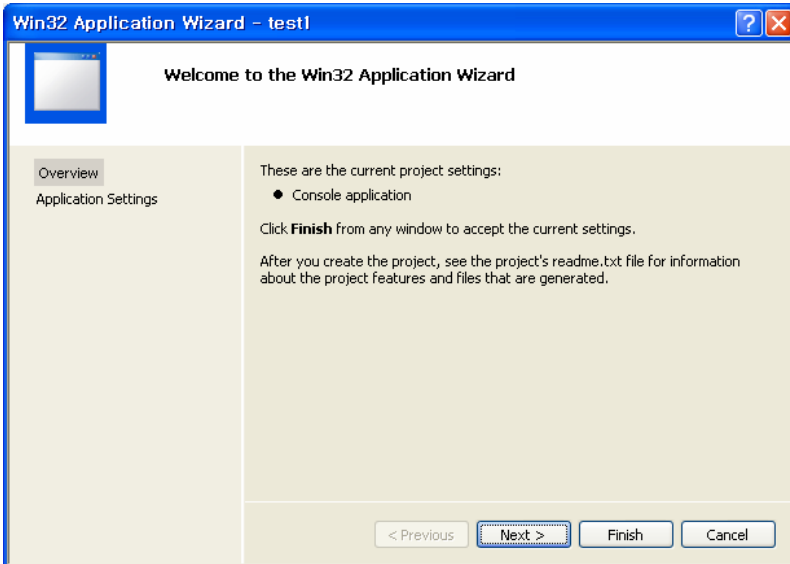
파일 > 새로만들기 > 새로운 프로젝트 > Visual C/C++ Project > Console Application  
File > New > New Project > Visual C/C++ project > Console Application



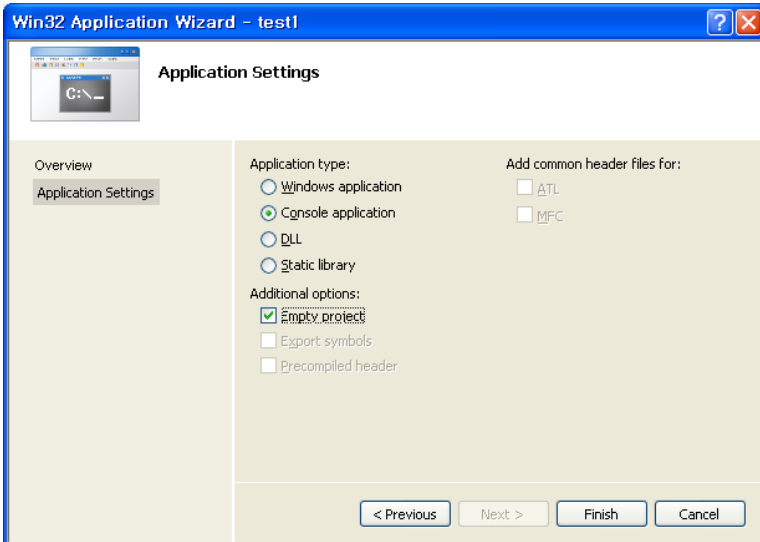
왼쪽에서 Visual C++의 Win32 를 선택한다. 이후 Win32 Consol Application을 선택하고 파일이름을 입력한다.



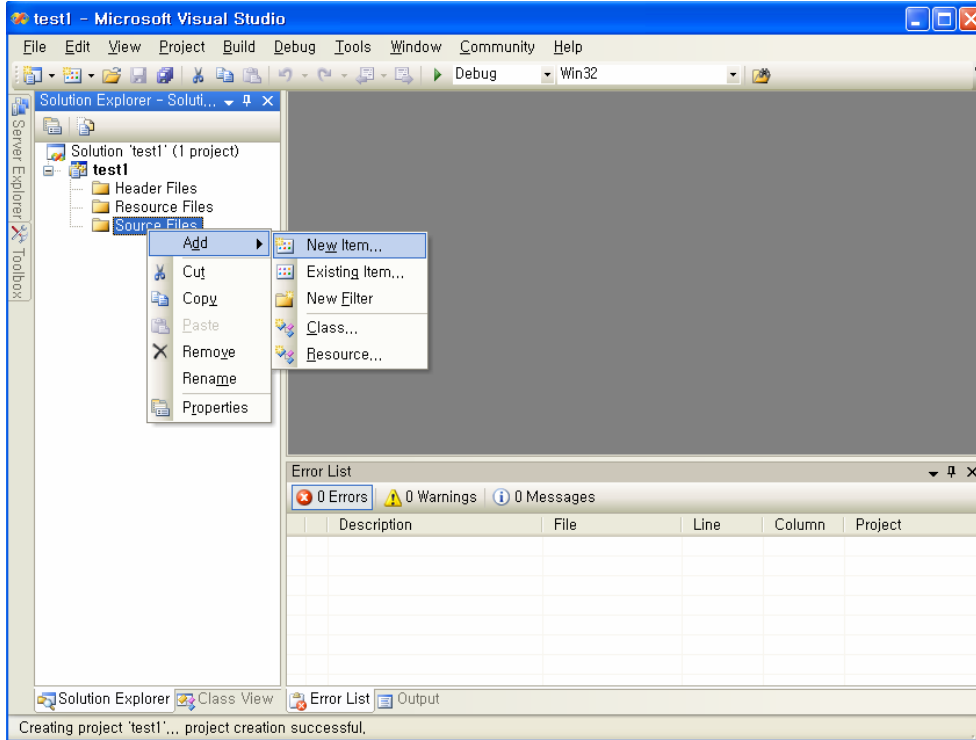
Next를 누르면 프로젝트에 대한 속성을 지정할 수 있다.



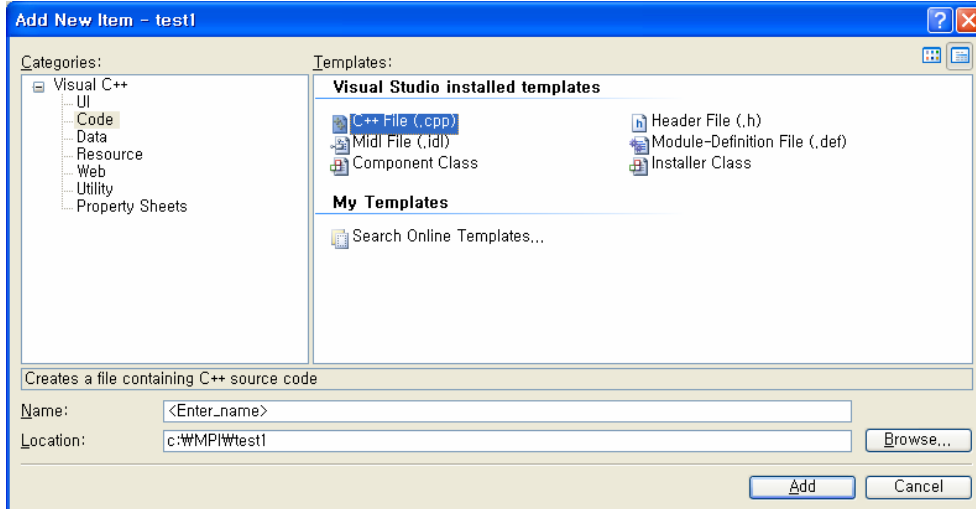
Console application과 Empty project를 선택하자. 윈도우즈 창에 결과가 나타나는 프로그램을 만들고 싶을 때에는 Windows application을 선택하면 된다. 그러면 프로젝트가 하나 생성된다.



프로젝트 탐색기에서 Source Files에서 마우스 오른쪽 클릭을 하여 Add를 선택하고 New Item을 선택한다.



파일의 종류를 Code로 선택하고 C++ Files(.cpp)를 선택한다. 이름에 파일이름을 지정한다. 이때 C언어 형식으로 작업하고 싶을 때는 확장자를 filename.c 파일로, C++언어로 프로그래밍 하고 싶을 때에는 filename.cpp로 지정해주는 것이 편리하다. 확장자를 입력하지 않으면 자동으로 .cpp가 붙게된다.

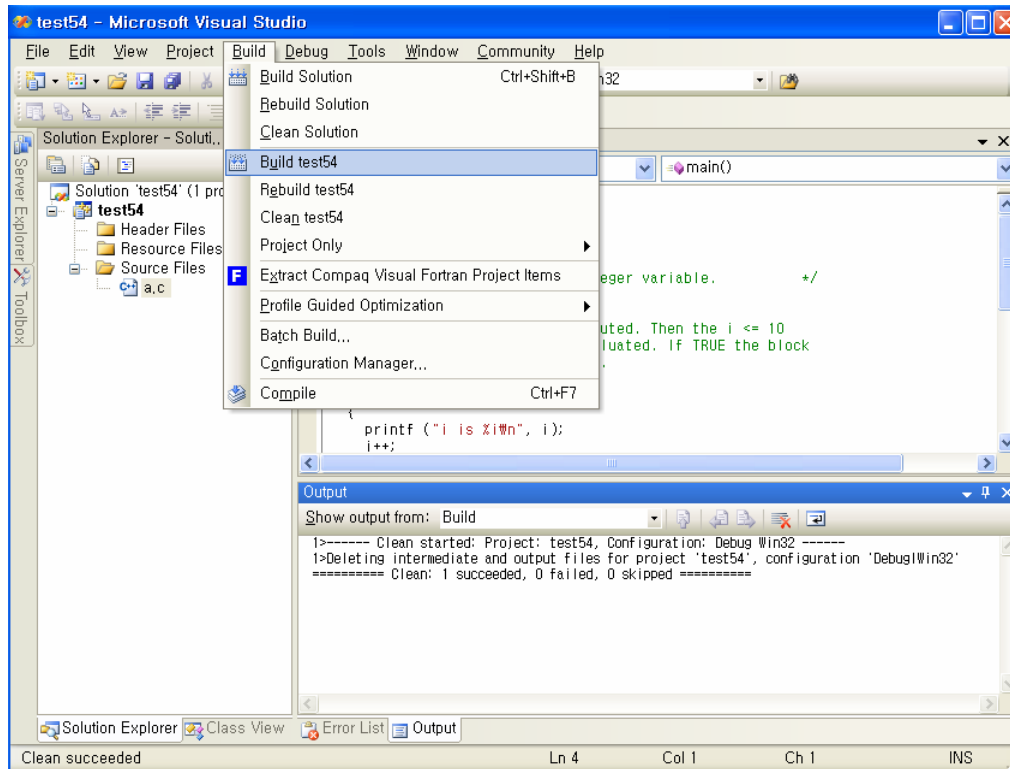


edit

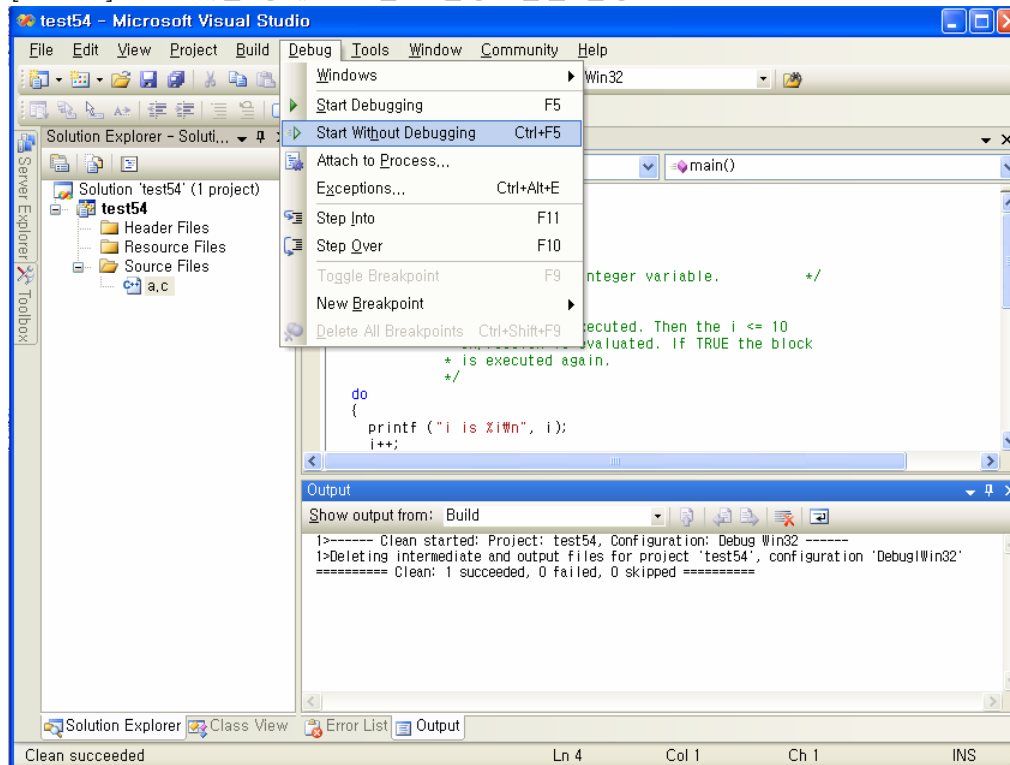


## compile

Build 메뉴의 Build 명령을 통해 컴파일을 실행한다.



성공적으로 컴파일이 되면 밑의 log창에 성공여부를 나타낸다. Debug > Start Without Debugging 혹은 [Control] + F5 키를 통해 컴파일된 실행파일을 실행한다.

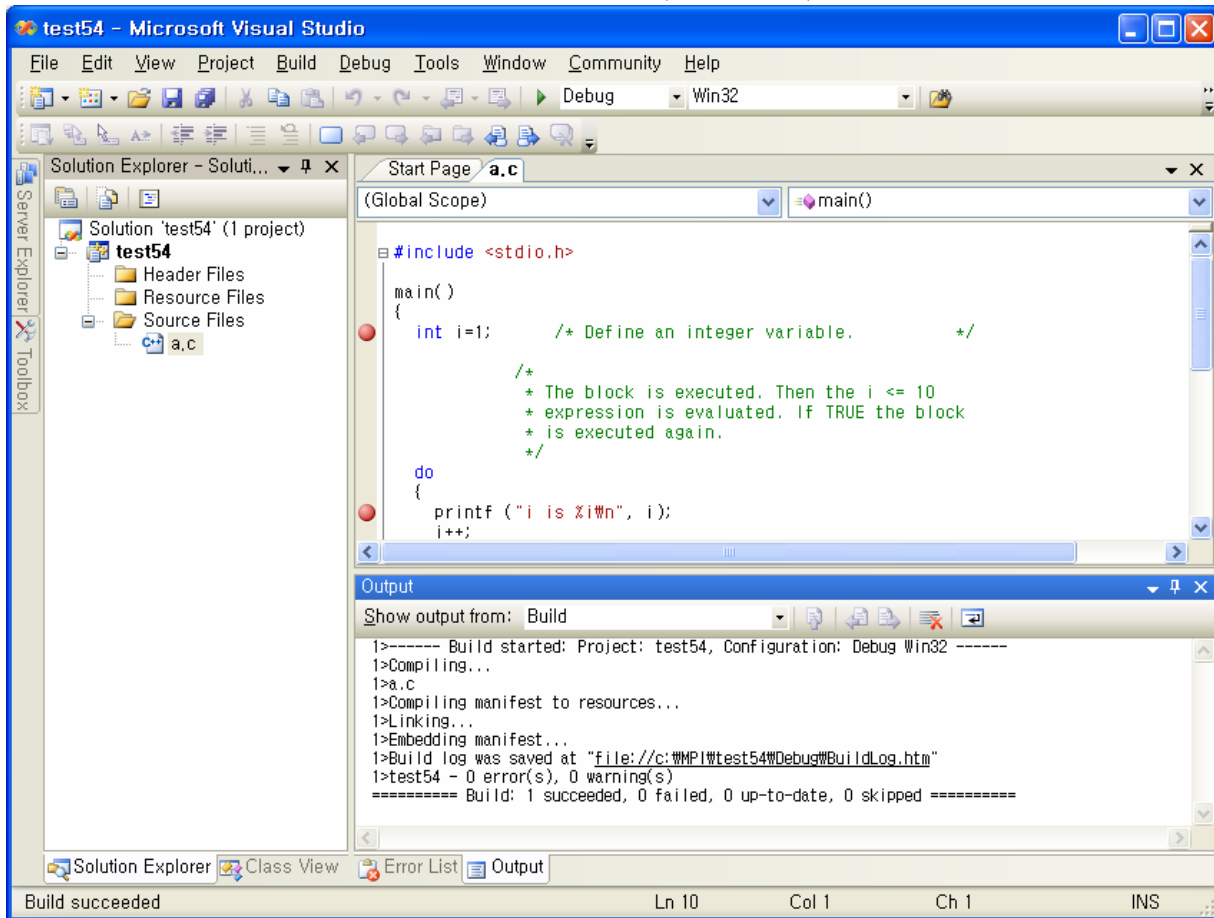


```
CA C:\WINDOWS\system32\cmd.exe
i is 977
i is 978
i is 979
i is 980
i is 981
i is 981
i is 982
i is 983
i is 984
i is 985
i is 986
i is 987
i is 988
i is 989
i is 990
i is 991
i is 992
i is 993
i is 994
i is 995
i is 996
i is 997
i is 998
i is 999
i is 1000
계속하려면 아무 키나 누르십시오 . . .
```

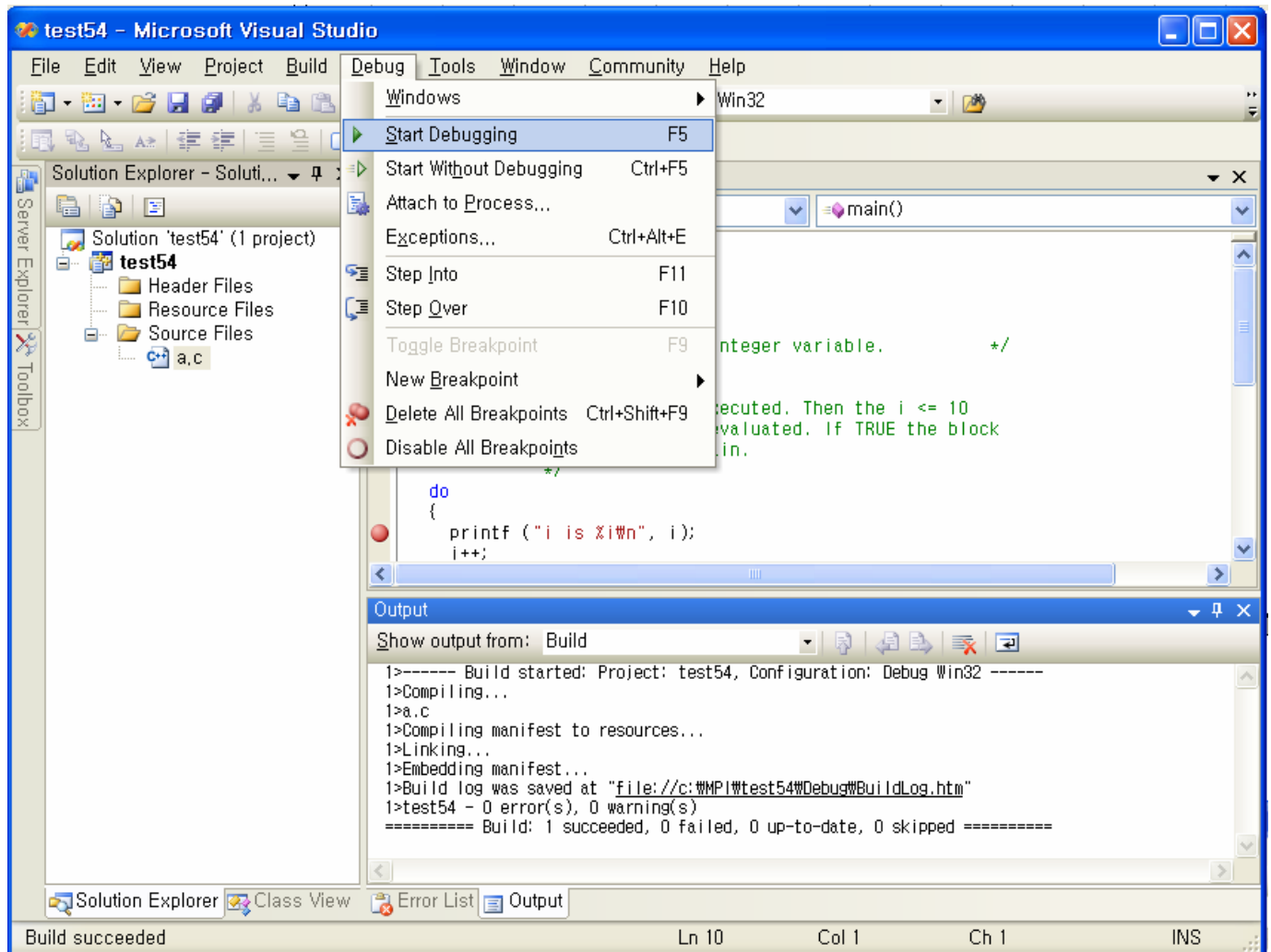
## debugging

프로그램에서 에러의 발생은 크게 두 가지로 생각해볼 수 있다. 첫째, syntax를 지키지 않은 오류로, 이러한 오류는 컴파일을 하다가 에러 메시지를 띄우고, 컴파일이 끝나지 않는다. 이때는 log창을 확인하여 자신의 오류를 수정할 수 있다. 프로그래밍을 처음 배우는 초기에는 {} 를 한쪽만 열고 닫지 않은 경우, 함수이름, 변수이름의 타이핑 오타 등도 간혹 존재한다. 두번째는 컴파일이 제대로 실행되었지만, 실행결과가 엉뚱한 결과값을 나타내는 경우이다. 이러한 경우는 초기값설정을 안하거나 초기화를 안할 때 자주 발생한다. 배열의 참조가 잘못된 경우도 많고, 다양한 오류가 존재할 수 있다. 이러한 오류는 컴파일러에서는 오류메시지가 출력하지 않으므로 프로그래머가 직접 오류를 수정하여야 한다. 이때, 디버깅툴이 유용하게 사용된다.

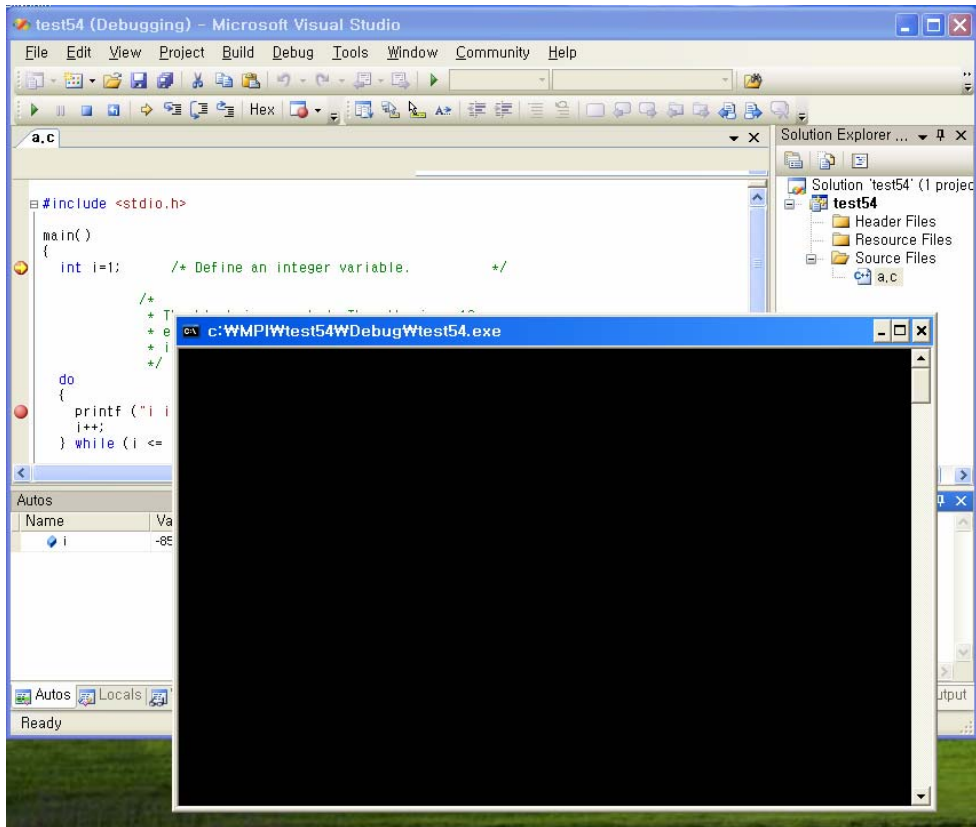
디버깅 전에 브레이크 위치를 표시한다. 대부분 변수쪽, iteration쪽, 중요 함수 호출에 Break Point를 설정한다.



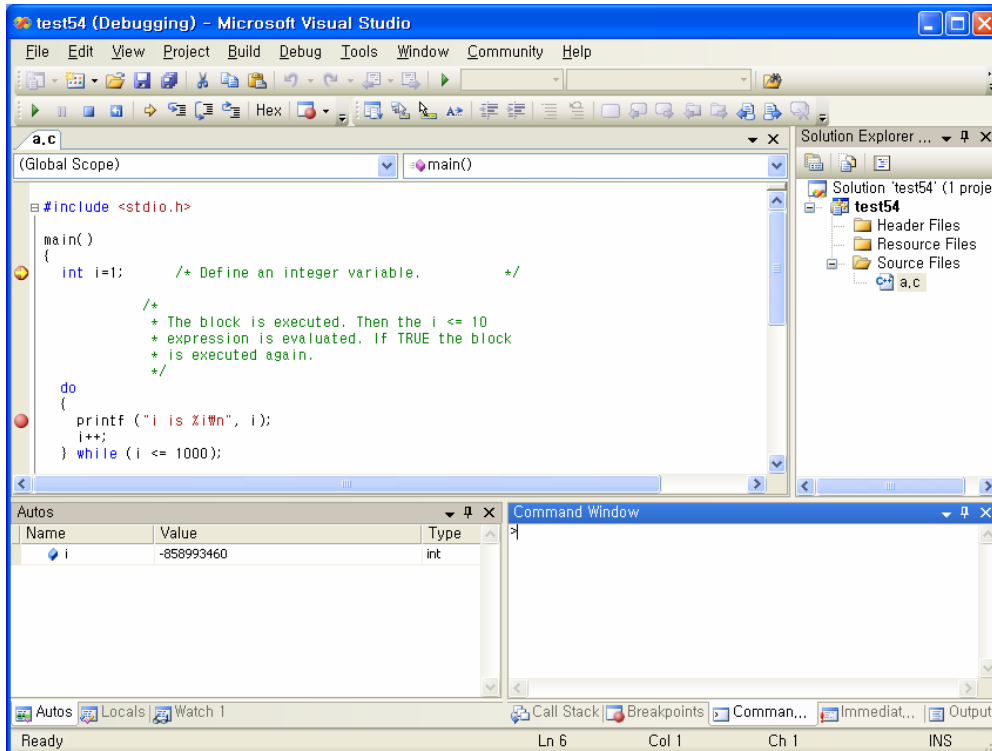
Debug메뉴의 Start Debugging을 선택하거나 F5키를 눌러 디버깅을 시작한다.



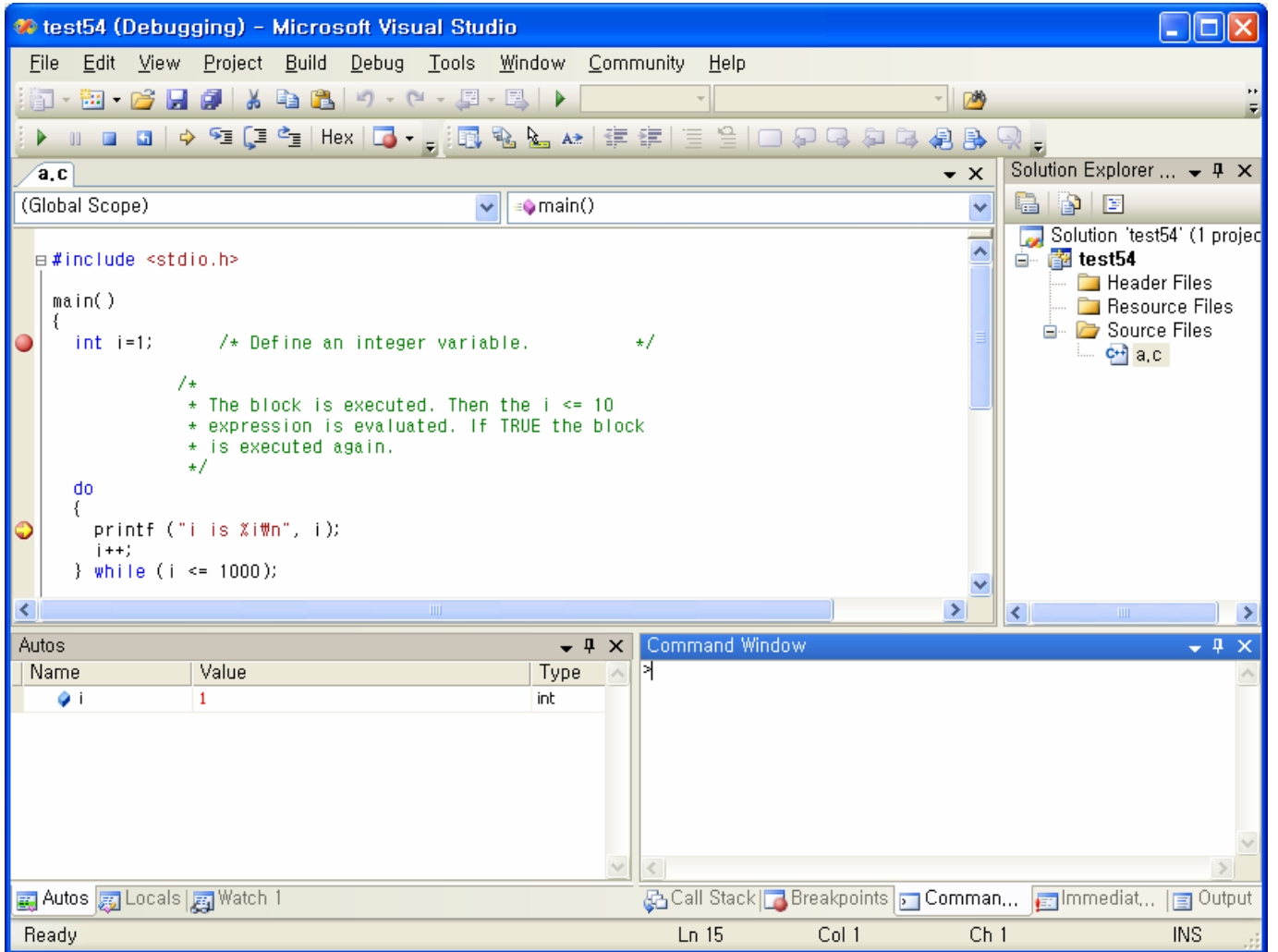
그러면 파일이 실행되고, Break Point에서 실행을 정지하고 기다리고 있다.



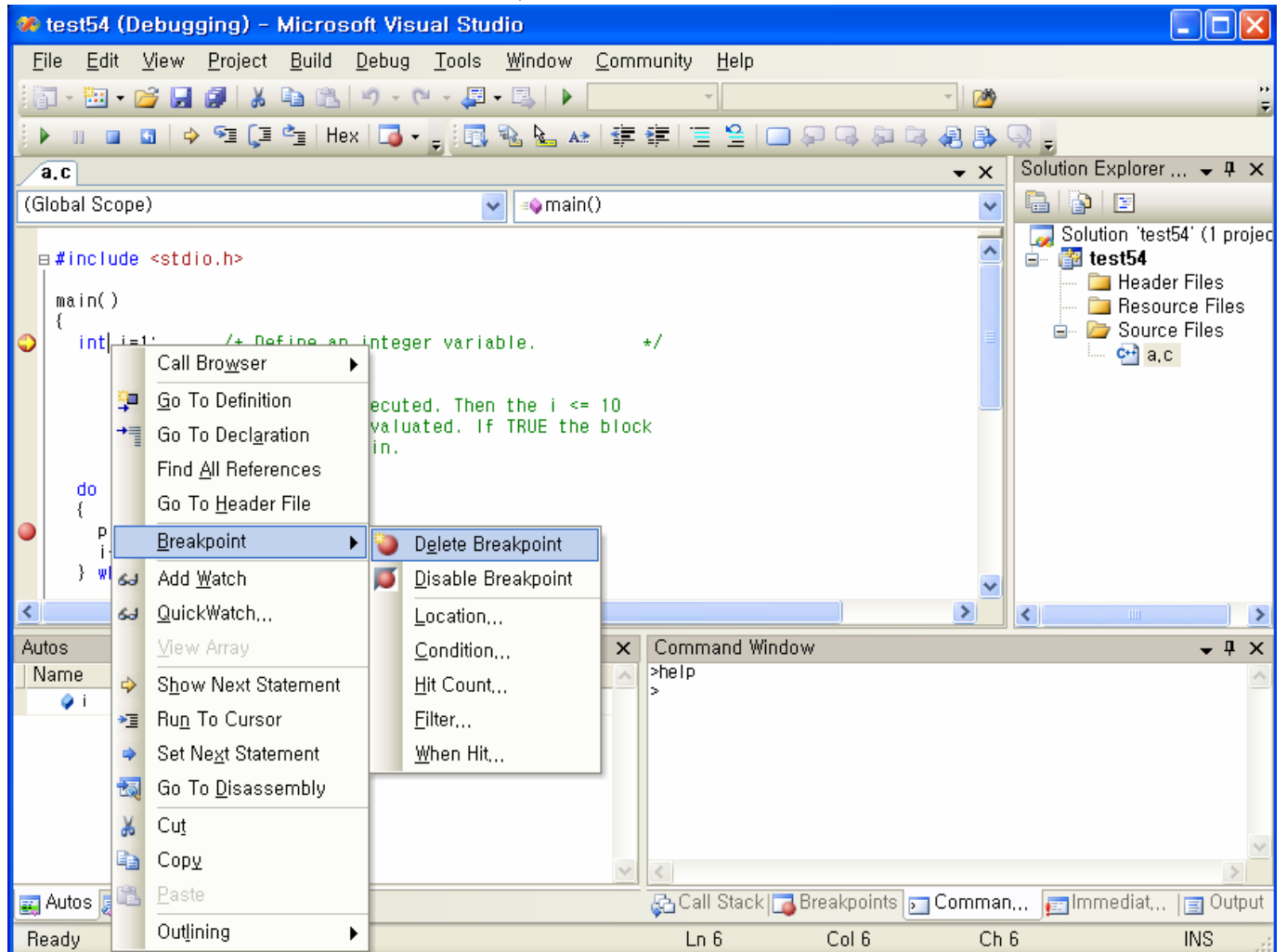
현재 i가 1로 초기화 되기 전에 정지하였기 때문에 value가 1로 나타나지 않고있다. Command 명령창에서 Debug 명령을 실행할 수 있다. 혹은 시작키, 정지키 등의 디버깅 아이콘도 사용할 수 있다.



Next 버튼을 누르면, 화살표가 printf에서 멈추게 되고, i값이 1로 바뀌었음을 알 수 있다.



또한 마우스 오른쪽키를 누르면 다양한 Breakpoint 옵션을 설정할 수 있다.



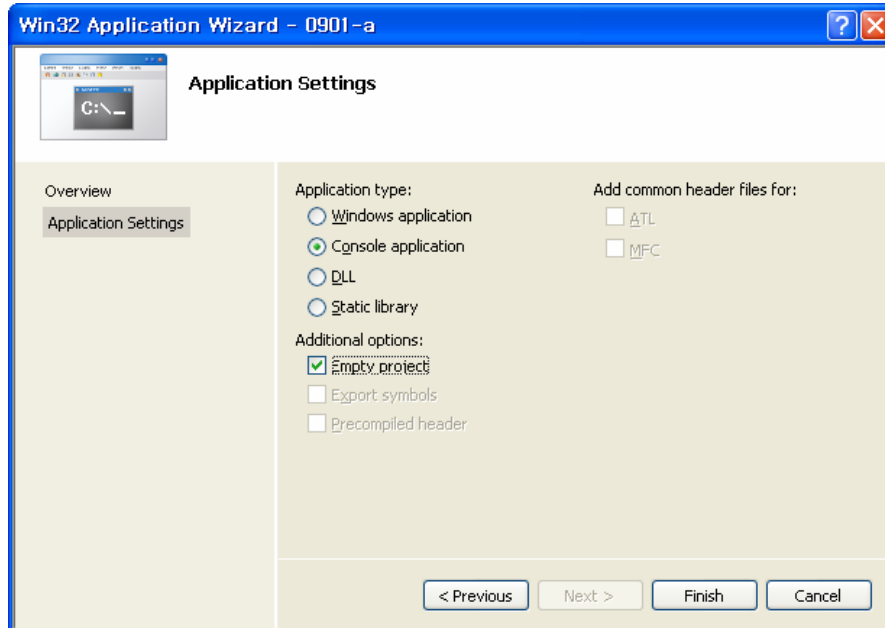
이러한 기능을 이용하여 한줄 한줄 명령을 실행하며 변수값등을 파악하면 쉽게 코드의 잘못된 곳을 발견하여 고칠 수 있다.

## C/C++ OpenMP programming with Visual Studio 2005

### OpenMP environment

OpenMP is attractive parallel programming method. We can easily parallelize our serial codes, but OpenMP is only for SMP machine. Nowadays, it is easy to find out the PC which have 2 or 4 core CPU. In this machine, we can learn and test openmp code in windows PC environment. Various compiler support this parallel programming method. First, I select Visual Studio 2005 to start the journey. We can parallel programming with C/C++ in Visual Studio 2005. IMHO, Its IDE has many benefit for beginners.

Visual Studio2005는 기본 지원하기 때문에 컴파일 옵션 변경만으로 쉽게 OpenMP 프로그래밍할 수 있습니다. 먼저 Visual C/C++ 프로젝트를 생성합니다. 이때, 빈 project를 생성합니다.

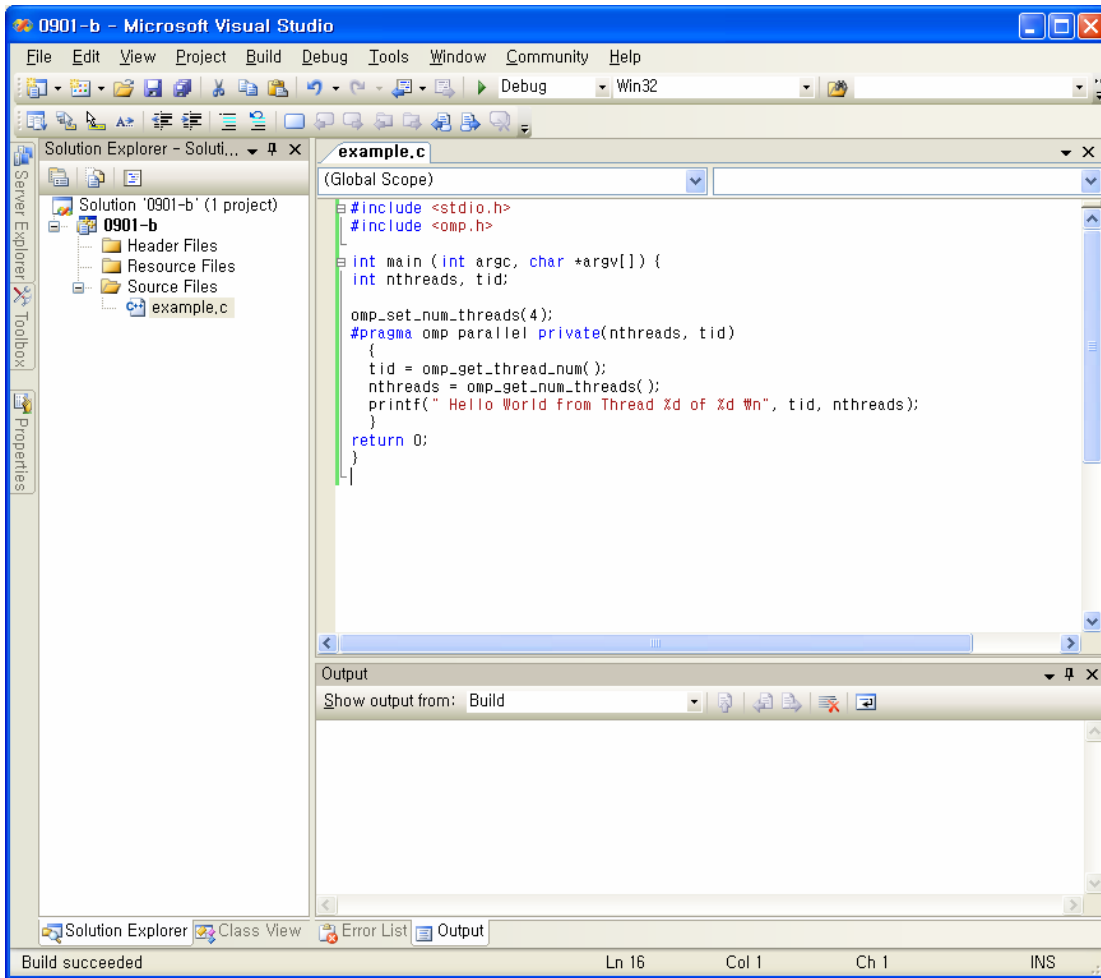


소스파일을 하나 생성하여 코드를 입력합니다. 기초예제는 다음과 같습니다.

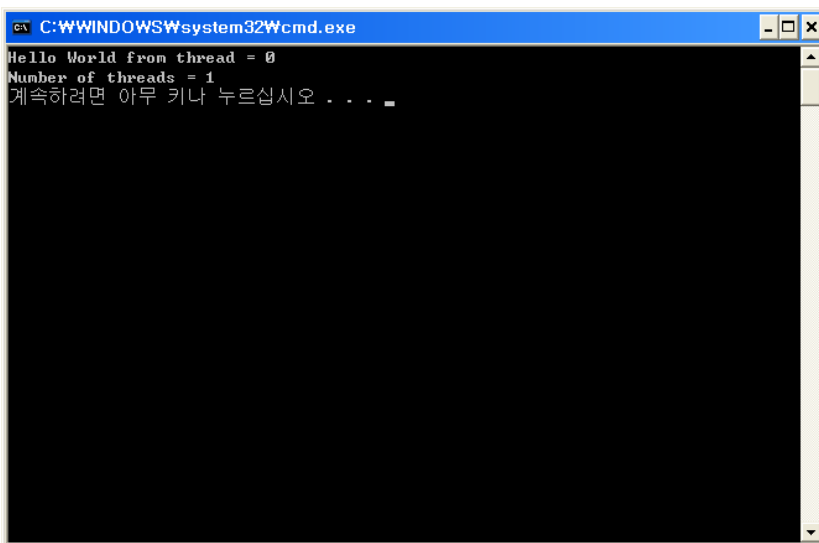
```
#include <stdio.h>
#include <omp.h>

int main (int argc, char *argv[]) {
int nthreads, tid;

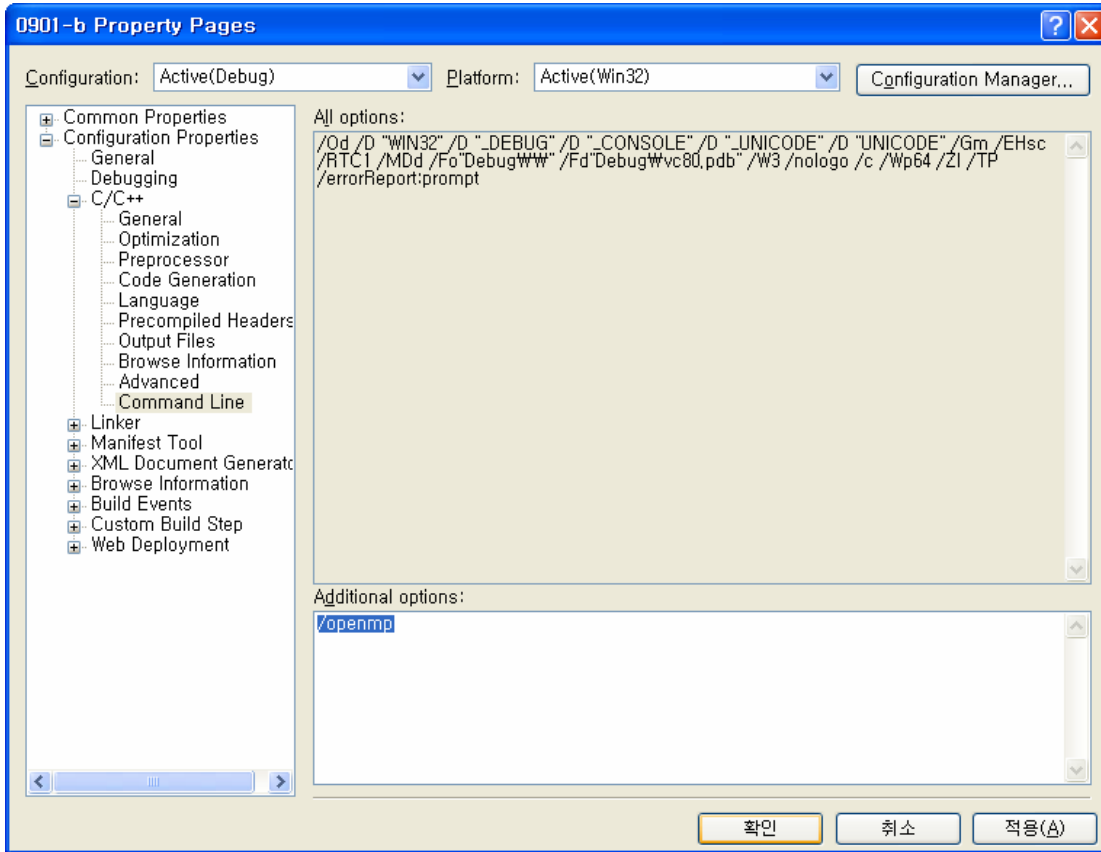
omp_set_num_threads(4);
#pragma omp parallel private(nthreads, tid)
{
tid = omp_get_thread_num();
nthreads = omp_get_num_threads();
printf(" Hello World from Thread %d of %d \n", tid, nthreads);
}
return 0;
}
```



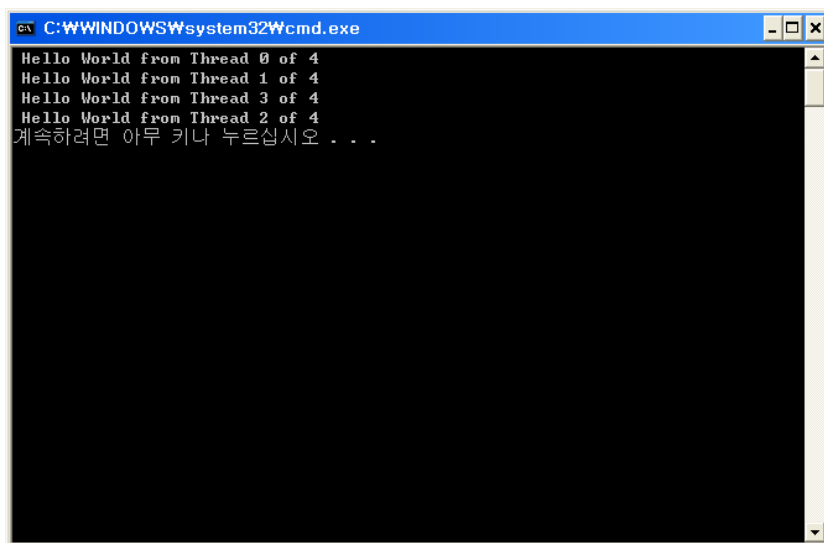
Visual Studio 2005는 OpenMP C/C++을 기본으로 지원합니다. openmp.h 를 헤더파일에 넣어주면 에러없이 정상출력됩니다.



하지만, 분명 thread 갯수를 4개로 지정했는데, 1개만 실행되는 것을 확인할 수 있습니다. Visual Studio에서 openMP컴파일을 하기 위해서는 /openmp 옵션을 줘야 합니다.  
 C/C++ > Command Line > Additional Options >



이제 다시 컴파일하고 실행하면 4개의 thread가 생성된 것을 확인할 수 있습니다.



이제 OpenMP컴파일을 위한 환경설정은 끝났습니다. 이제 OpenMP로 실제 병렬프로그래밍을 하면 됩니다.

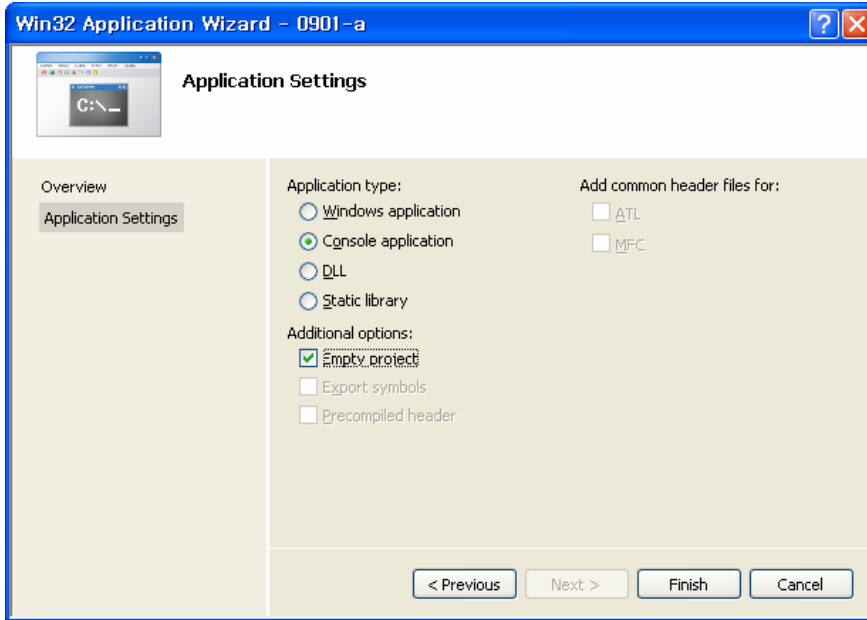
### 3. C/C++ MPI programming with Visual Studio 2005

MPI(Message Passing Interface)는 Cluster Computer 즉, 여러대의 컴퓨터를 병렬로 연결한 시스템에서 병렬프로그래밍을 돌리기 위한 환경이다. OpenMP가 컴파일러 수준에서 지원하는 것과 다르게 MPI는 운영체제 즉 OS 수준에서 지원해야 한다. Linux와 Unix에서는 다양하게 MPI환경을 구축할 수 있는데, Windows 환경은 크게 3가지로 나뉘볼 수 있다.

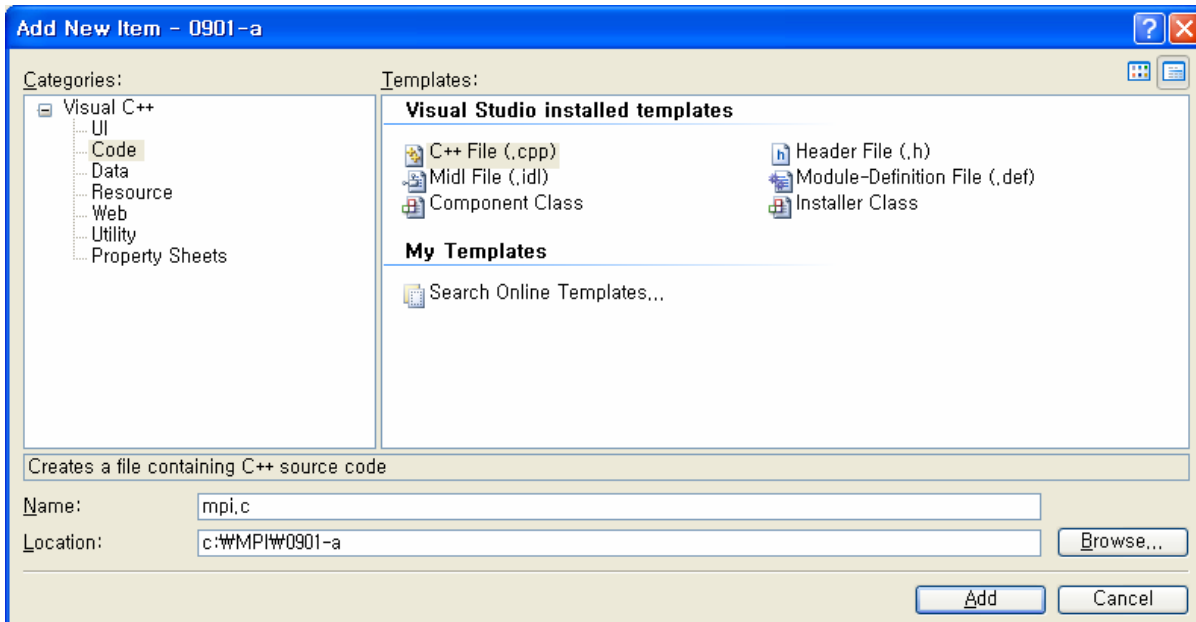
1. Windows 2003 서버제품에 Cluster Comute Pack(CCP)를 설치하거나 Windows 2003 CCP를 설치하는 방법이다. OS차원에서 직접 지원하기 때문에 편리하지만, 각 노드별 라이선스를 받아야 하기 때문에 비용적인 면에서 큰 부담이 된다.
2. Windows에 Cygwin을 설치하고 Windows용 MPI인 LAM을 설치하여 사용할 수 있다. cygwin은 자신의 컴퓨터에 Linux환경을 설치하는 것과 비슷하기 때문에, 다양한 Linux 응용프로그램을 사용할 수 있는 장점이 있고, 무료이다. 처음접하는 초보자에게 약간 어려움이 있을 수 있지만, 최근에는 설치와 사용이 매우 쉽게 만들어져 쉽게 사용할 수 있다.
3. Windows에 Windows용 MPICH2를 설치하는 방법이다. 비용도 없고, 가장 쉽게 사용할 수 있고, Cygwin은 물론 Visual studio, Intel Fortran Compiler등 다양한 환경을 지원한다. 본 글에서도 Windows에 MPICH2를 설치하는 것을 기본으로 하고 있다.

## 2. Select Visual C++ project Windows application & empty project

파일 > 새로만들기 > 새로운 프로젝트 > Visual C/C++ Project > Console Application  
File > New > New Project > Visual C/C++ project > Console Application  
빈 프로젝트를 만듭니다.



이후 프로젝트 탐색기에서 Add New item을 통해 Code > C++ File 을 선택해 소스코드파일을 하나 만듭니다. 만약 C++로 짜고싶다면 filename.cpp, C로 짜고싶다면 filename.c 로 설정합니다. 여기서는 filename.c를 사용합니다. C에서의 사용방법을 알면 똑같은 방법으로 C++에서도 사용할 수 있습니다.



이제, 새로운 창에 소스코드를 입력합니다. 사용할 코드는 다음과 같습니다. 아직은 MPI 프로그래밍의 기초를 배우지 않았기 때문에, 똑같이 입력하고, 컴파일하는 것만 연습해 볼 것입니다. 주의할 것은 C language는 대소문자를 가리기 때문에 주의하셔야 합니다. 특히, MPI 명령어는 MPI\_Axxxx\_xxxx() 의 형태를 띠고 있습니다. 즉, 첫 4자만 대문자이고 나머지는 모두 소문자입니다.

```
#include <stdio.h>
#include "mpi.h"
int main(int argc, char *argv[]) {
    int rank, size;

    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MPI_COMM_WORLD,&rank);
    MPI_Comm_size(MPI_COMM_WORLD,&size);
    printf("Hello, world I am %d rd core of %d core system\n",rank,size);
    MPI_Finalize();
    return 0;
}
```

이 파일을 컴파일 해봅니다.

Check Compile !!

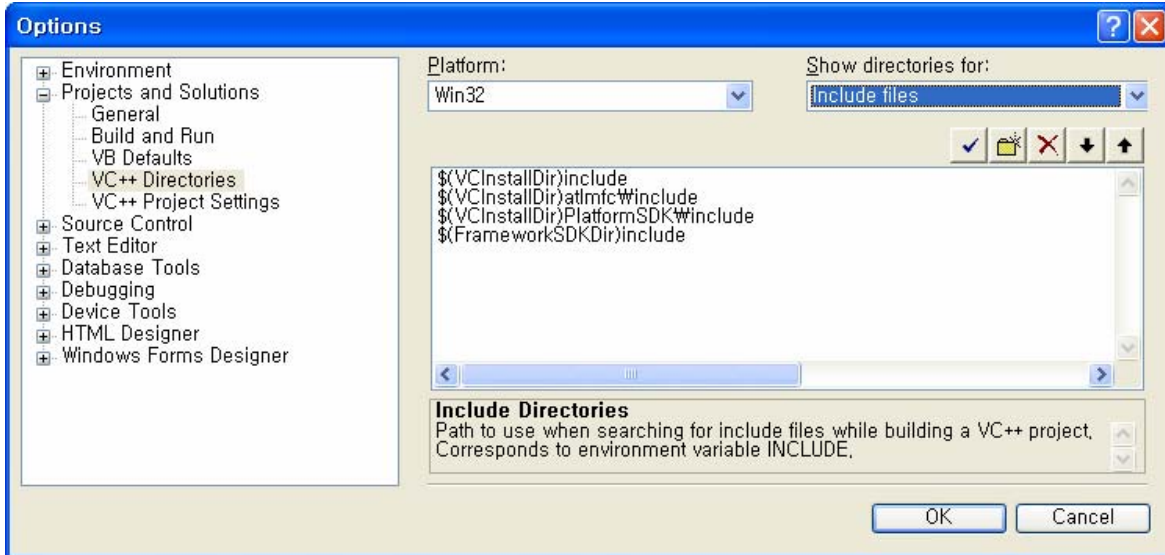
You find Error Message!!. We need to register mpi.h library

include files

도구 > 옵션 > 프로젝트 > Visual C++ 디렉토리 > 추가 파일

Tools > Options... Menu > Projects and Solutions > Visual C++ Directories > Include files

C:\WProgram Files\WMPICH2\include

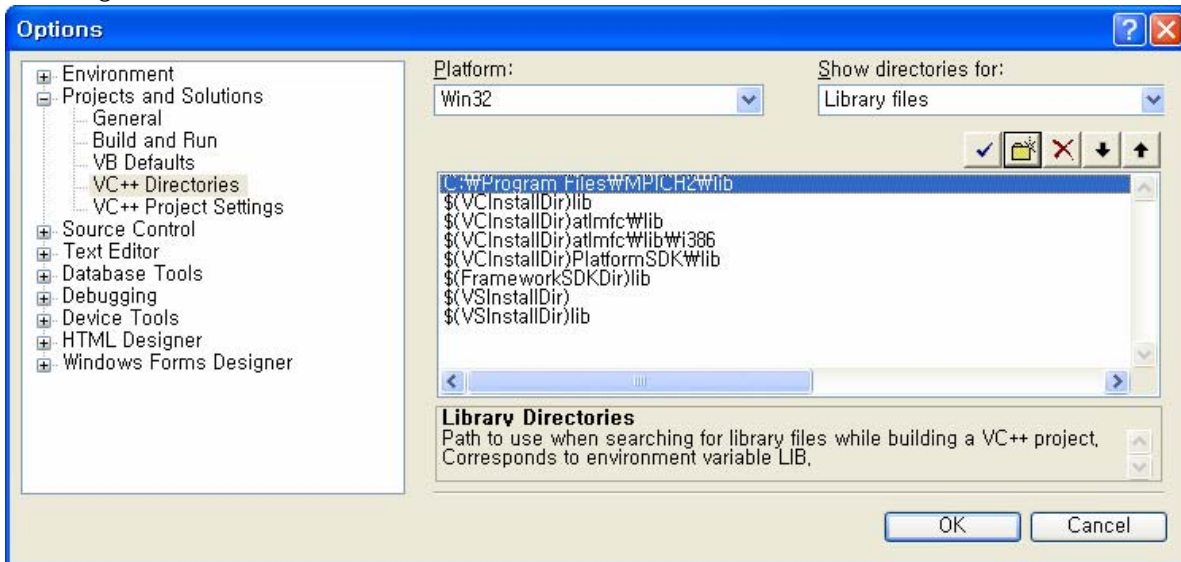


Library Files

도구 > 옵션 > 프로젝트 > Visual C++ 디렉토리 > 라이브러리 파일

Tools > Options... Menu > Projects and Solutions > Visual C++ Directories > Library Files

C:\WProgram Files\WMPICH2\lib

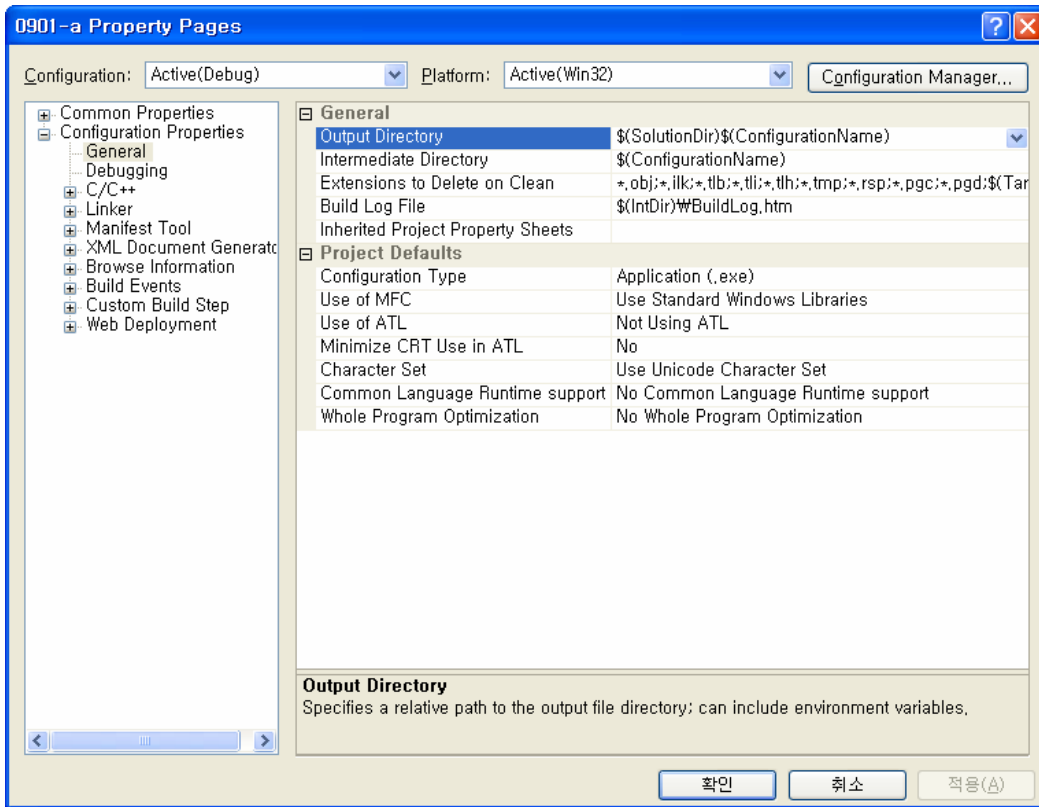


Check Compile. not yet success!!

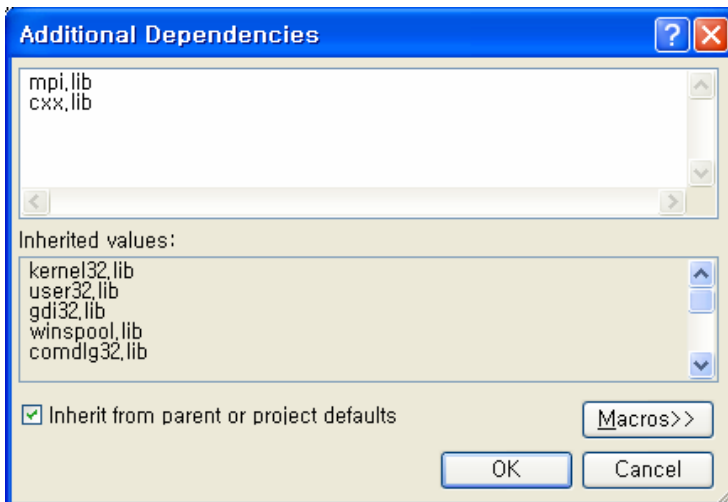
We need to configuration of project property

Click project name and Right Click & select Property.

클릭 프로젝트 이름 > 마우스 오른쪽 클릭 > 환경설정

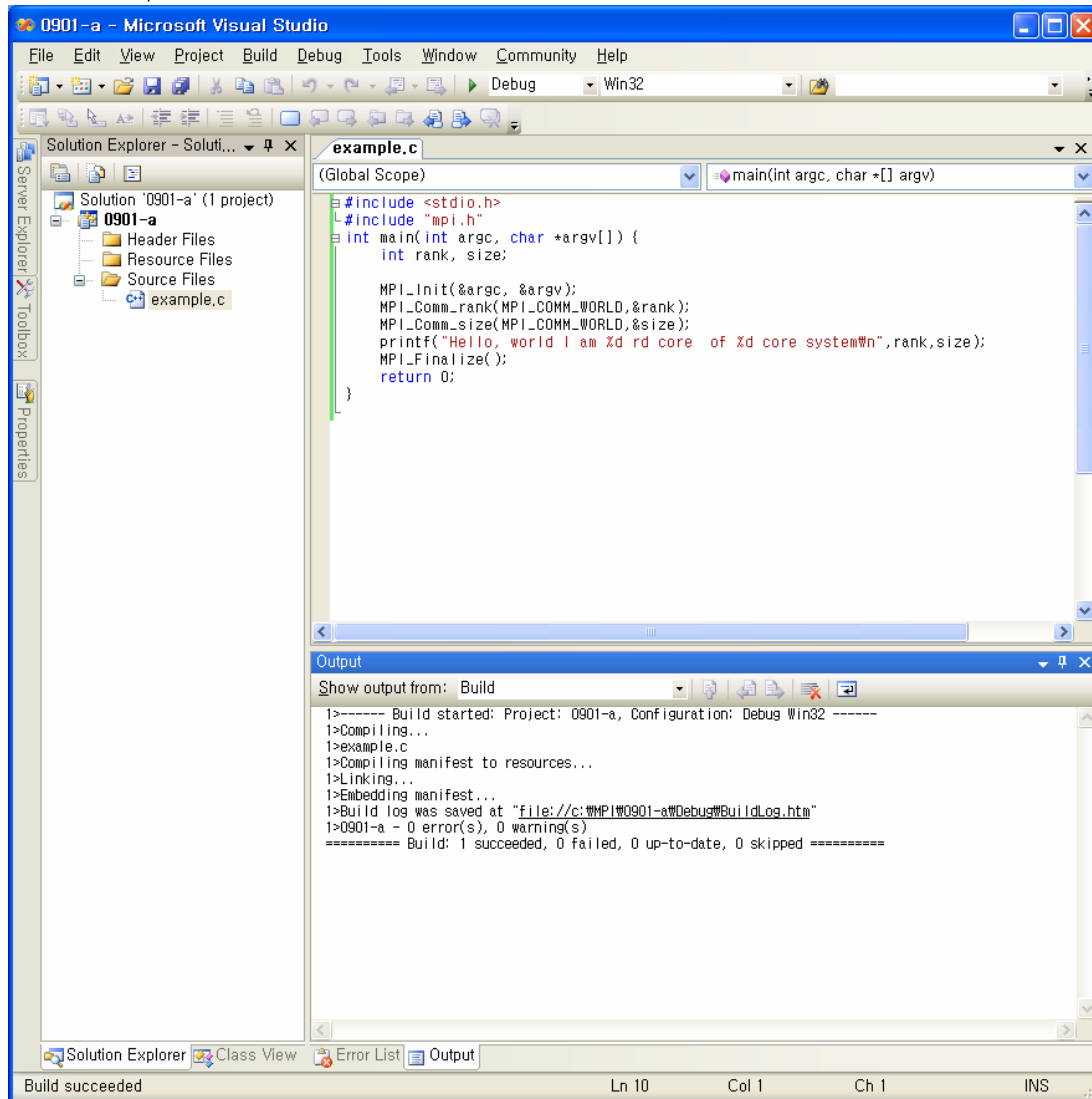


한글 Visual Studio 2005 : 링커 > 입력 > 추가종속성  
 영문 Visual Studio 2005 : Linker > Input > Additional Dependencies  
 에 mpi.lib, cxx.lib 를 추가한다. 이 파일은 MPICH2 의 library파일이다.  
 중요라이브러리는 다음과 같다.  
 mpi.lib C Language용  
 cxx.lib C++용  
 fmpich2.lib Fortran용 이다.  
 따라서, mpi.lib cxx.lib 두개를 추가시킨다.

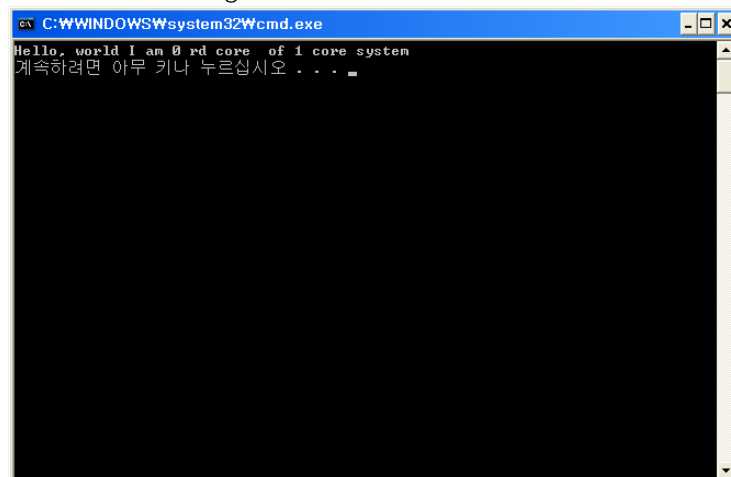


이제 환경설정이 완료되었습니다.

Check compile!! .. We Succeeded !!.



Run without Debug



cmd 명령을 통해 DOS prompt 로 가 파일을 살펴보면 0901-a.exe 파일이 생긴 것을 확인할 수 있습니다.

```

C:\WINDOWS\system32\cmd.exe

C:\MPI\0901-a\Debug>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: BCA6-C9E0

C:\MPI\0901-a\Debug 디렉터리

2007-09-02 오후 01:14 <DIR> .
2007-09-02 오후 01:14 <DIR> ..
2007-09-02 오후 01:14 40,960 0901-a.exe
2007-09-02 오후 01:14 403 0901-a.exe.embed.manifest
2007-09-02 오후 01:14 468 0901-a.exe.embed.manifest.res
2007-09-02 오후 01:14 385 0901-a.exe.intermediate.manifest
2007-09-02 오후 01:14 310,900 0901-a.ilc
2007-09-02 오후 01:14 297,984 0901-a.pdb
2007-09-02 오후 01:14 8,914 BuildLog.htm
2007-09-02 오후 01:14 4,491 example.obj
2007-09-02 오후 01:14 62 mt.dep
2007-09-02 오후 01:14 27,648 vc80.idb
2007-09-02 오후 01:14 53,248 vc80.pdb
11개 파일 745,463 바이트
2개 디렉터리 21,704,200,192 바이트 남음

C:\MPI\0901-a\Debug>

```

### Run MPI

mpirun -n N 실행파일이름 명령을 통해 실행하면 결과를 확인해 볼 수 있습니다.

```

C:\WINDOWS\system32\cmd.exe

2007-09-02 오후 01:14 385 0901-a.exe.intermediate.manifest
2007-09-02 오후 01:14 310,900 0901-a.ilc
2007-09-02 오후 01:14 297,984 0901-a.pdb
2007-09-02 오후 01:14 8,914 BuildLog.htm
2007-09-02 오후 01:14 4,491 example.obj
2007-09-02 오후 01:14 62 mt.dep
2007-09-02 오후 01:14 27,648 vc80.idb
2007-09-02 오후 01:14 53,248 vc80.pdb
11개 파일 745,463 바이트
2개 디렉터리 21,704,200,192 바이트 남음

C:\MPI\0901-a\Debug>mpirun -n 1 0901-a
Hello, world I am 0 rd core of 1 core system

C:\MPI\0901-a\Debug>mpirun -n 2 0901-a
Hello, world I am 1 rd core of 2 core system
Hello, world I am 0 rd core of 2 core system

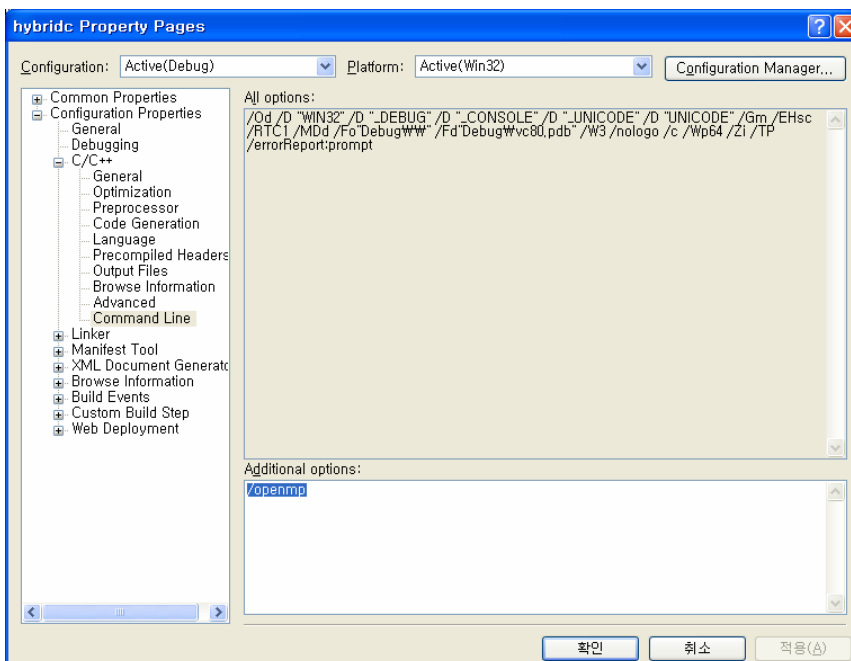
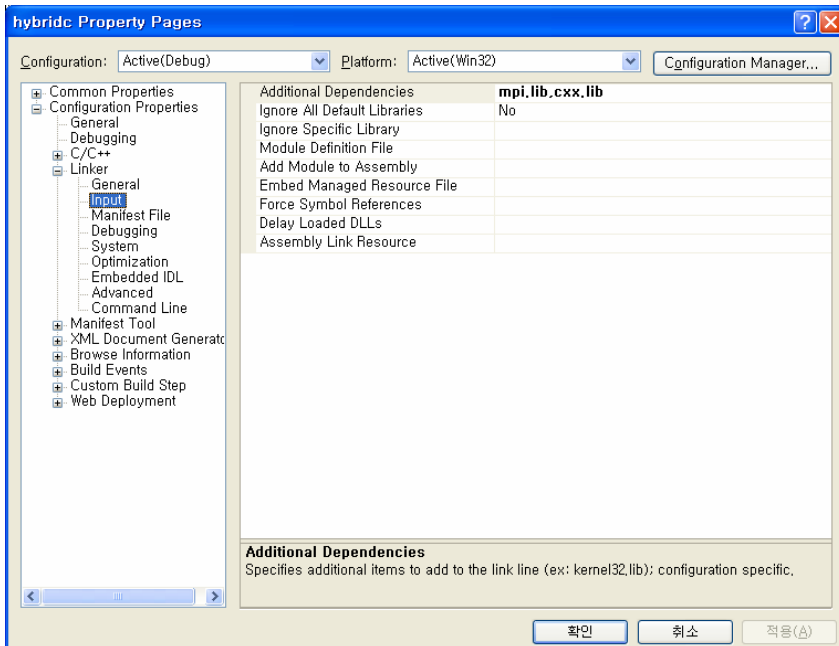
C:\MPI\0901-a\Debug>mpirun -n 4 0901-a
Hello, world I am 3 rd core of 4 core system
Hello, world I am 0 rd core of 4 core system
Hello, world I am 2 rd core of 4 core system
Hello, world I am 1 rd core of 4 core system

C:\MPI\0901-a\Debug>

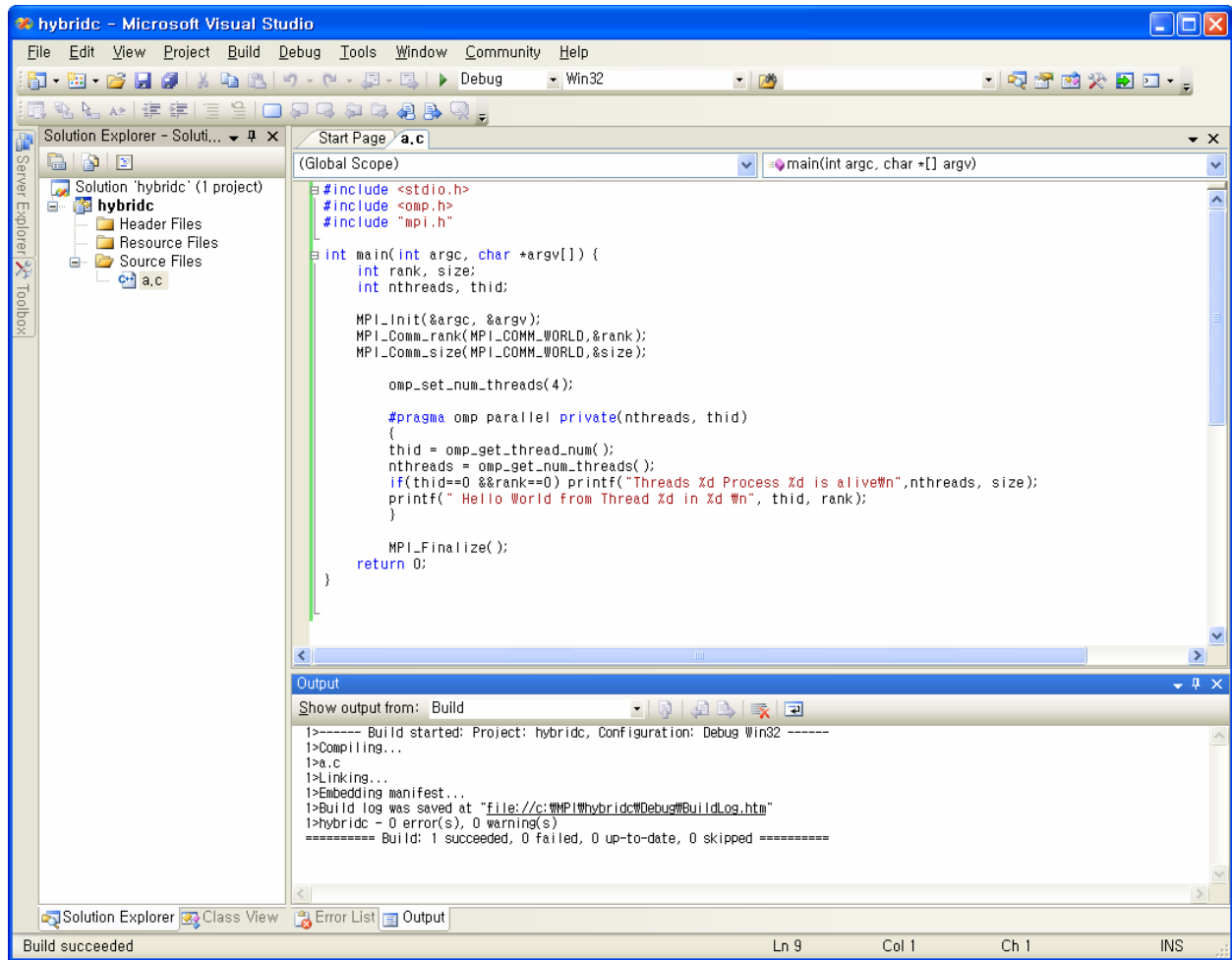
```

## C/C++ Hybrid (OpenMP & MPI) programming with Visual Studio 2005

자신의 컴퓨터가 SMP cluster인 경우는 OpenMP와 MPI를 동시에 사용하는 프로그램을 짤 필요성이 있습니다. MPI는 네트워크로 통신을 하기 때문에 통신부하가 많이 걸리기 때문에, SMP간에는 OpenMP코드로 돌아가게 하고 Node간에는 MPI로 통신을 하게 하면 병렬화를 통한 이득을 많이 볼 수 있습니다. 이를 위해서는 OpenMP와 MPI를 동시사용하게 옵션을 주면 되는데, 이미 MPI directory는 저장되었으므로, 프로젝트 생성시 추가 라이브러리와 openmp 컴파일 옵션을 사용하면 됩니다. 즉



간단한 형태의 Hybrid 코드를 입력합니다.  
 MPI 안에 OpenMP 코드가 다시 작동되는 것을 확인할 수 있습니다.



컴파일후 실행하면 정상실행됩니다. 이때 주의할 점은 노드갯수와 SMP core 갯수를 자신의 시스템에 맞게 설정해 주어야 최적의 실행모드가 됩니다.

