

# **Programming Methodology Practice Session #1**

## **Debugging in Visual C++ 2005**

# Debugging

## ■ Debugging

- 작성한 프로그램을 실행시키면서 **오류(bug)**를 찾아내어 바로 잡는 과정
- 규모가 큰 프로그램을 작성할 때에 있어서 **가장 많은 시간이 소요**되는 과정이다.

## ■ Debugger

- Source code를 **step-by-step**으로 실행시키면서 **프로그램이 실행되는 순서와 각 변수들에 저장되는 값을 실시간**으로 확인함으로써 프로그램의 잘못된 부분을 찾아주는 도구.
- `printf()` 함수를 이용하여 debugging하는 방법은 바람직하지 못하다.

# Setting Breakpoints

```
#include <stdio.h>

/* function main begins */
int main()
{
    int integer1; /* first integer */
    int integer2; /* second integer */
    int sum; /* sum of integers */

    printf("Enter first integer: ");
    scanf("%d", &integer1); /* read an integer */

    printf("Enter second integer#\n"); /* prompt */
    scanf("%d", &integer2); /* read an integer */

    sum = integer1 + integer2; /* assign total to sum */

    printf("Sum is %d#\n", sum); /* print sum */

    return 0; /* indicate that program ended successfully */
} /* end function main */
```

**F9** 키를 누르면 해당 line에 breakpoint가 설정된다.

Debug mode로 프로그램을 실행시키면 이 위치에서 멈추게 된다.

# Setting Breakpoints – Advanced

```
printf("Enter first integer\n"); /* prompt */
scanf("%d", &integer1); /* read an integer */

printf("Enter second integer\n"); /* prompt */
scanf("%d", &integer2); /* read an integer */

sum = integer1 + integer2; /* assign sum */

printf("Sum is %d\n", sum); /* print sum */

return 0; /* indicate that program ended */
} /* end function main */
```

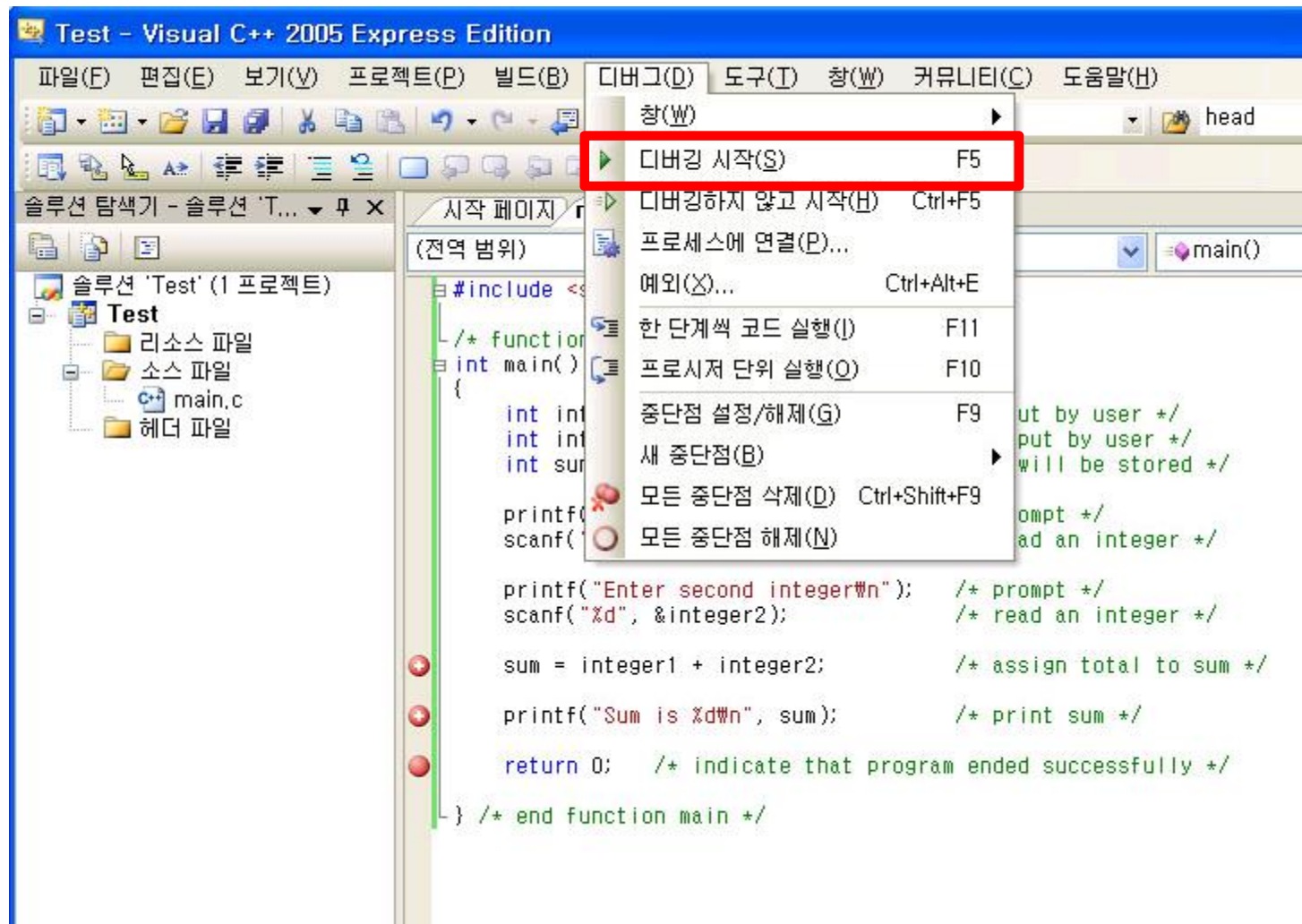
**Alt+F9** 키를 누르면 설정된 breakpoint 들의 목록이 나타나며, 각 breakpoint마다 중단 조건을 설정해 줄 수 있다.

특정 조건(**boolean statement**)이 참일 때만 중단하게 하거나, 특정 변수의 값이 바뀌었을 때 중단하게 할 수 있다.

이름	조건	적용 횟수
<input checked="" type="checkbox"/> main.c, 줄 16	조건: 'sum' 이(가) 변경된 경우	항상 중단
<input checked="" type="checkbox"/> main.c, 줄 18	조건: 'sum == 5' 이(가) 참인 경우	항상 중단
<input checked="" type="checkbox"/> main.c, 줄 20	(조건 없음)	항상 중단

코드 정의 창 | 호출 브라우저 | 출력 | 중단점

# Starting the Debugger



# Debugger Windows

현재 실행 지점

```
int integer1; /* first number to be input by user */
int integer2; /* second number to be input by user */
int sum;      /* variable in which sum will be stored */

printf("Enter first integer\n"); /* prompt */
scanf("%d", &integer1);        /* read an integer */

printf("Enter second integer\n"); /* prompt */
scanf("%d", &integer2);        /* read an integer */

sum = integer1 + integer2;      /* assign total to sum */

printf("Sum is %d\n", sum);

return 0; /* indicate that pro

-} /* end function main */
```

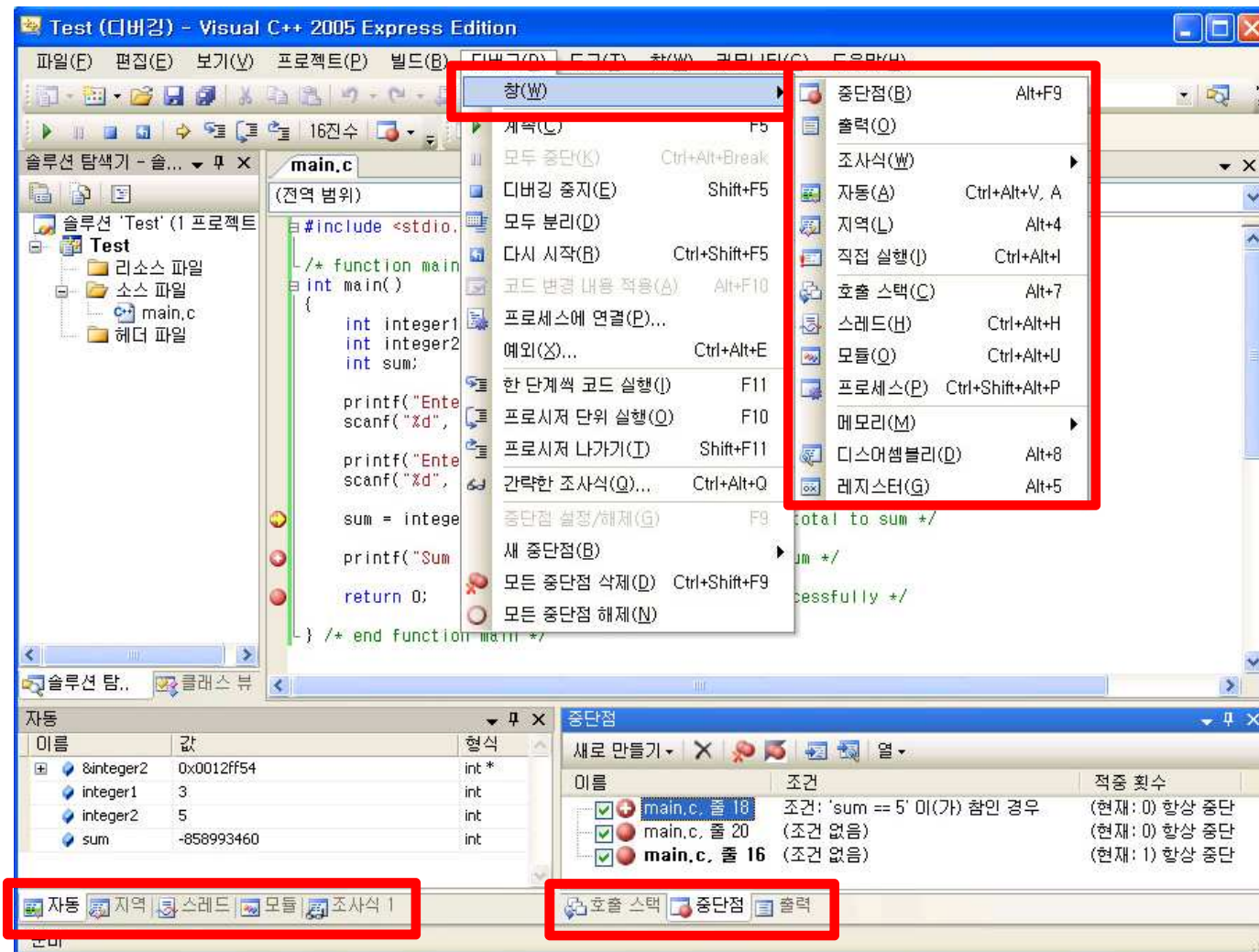
현재 지점에서 각 변수들에 저장된 값

이름	값	형식
&integer2	0x0012ff54	int *
integer1	3	int
integer2	5	int
sum	-858993460	int

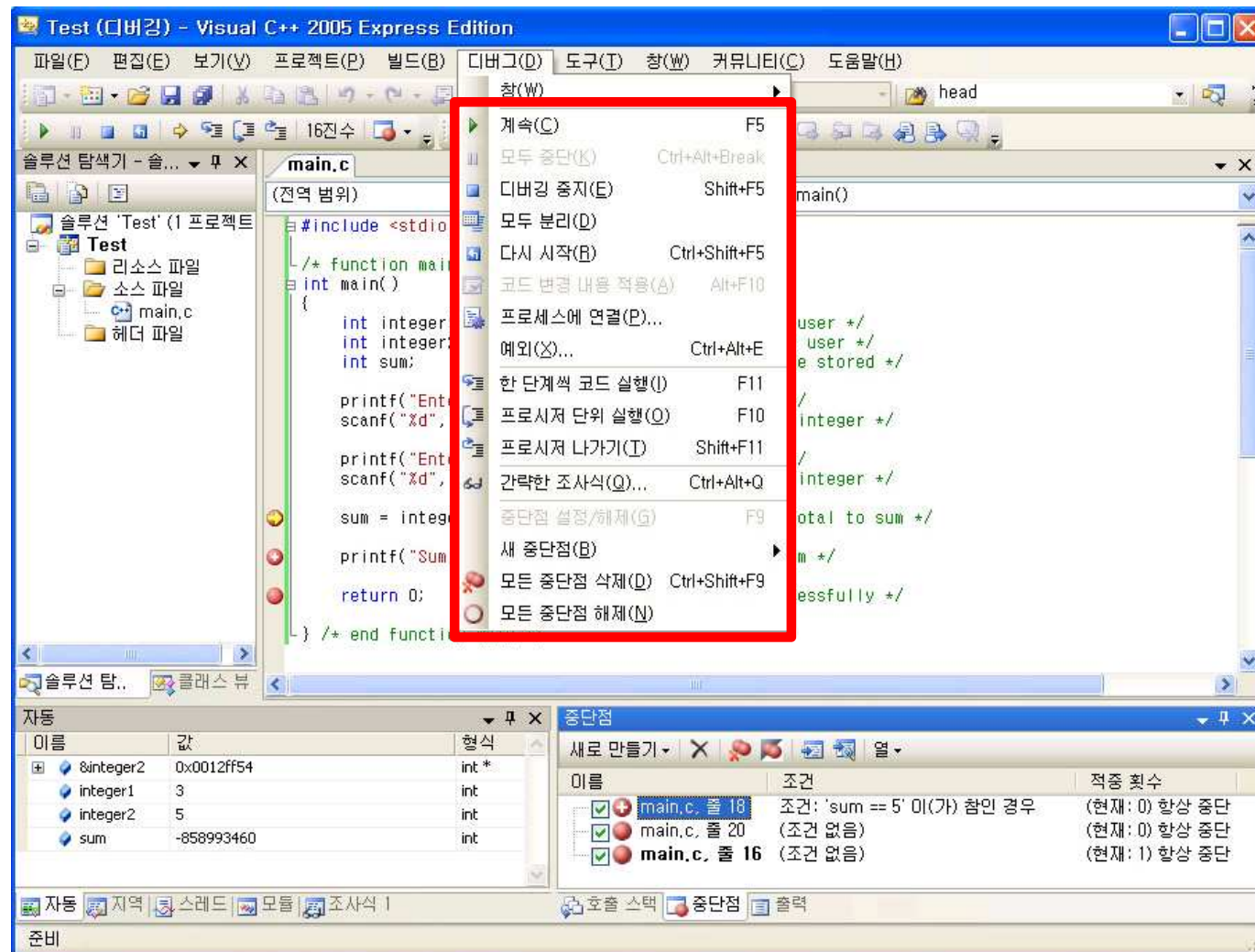
중단점

이름	조건
main.c, 줄 18	조건: 'sum == 5' 이(가
main.c, 줄 20	(조건 없음)
main.c, 줄 16	(조건 없음)

# Debugger Windows – Advanced



# Executing the Program Step-by-Step





# Executing the Program Step-by-Step

'계속' 버튼 혹은 F5 키를 누르면 다음 breakpoint를 만날 때 까지 프로그램을 수행한다.

```
main begins program execution */
integer1; /* first number to be input by user */
integer2; /* second number to be input by user */
sum; /* variable in which sum will be stored */

printf("Enter first integer\n"); /* prompt */
scanf("%d", &integer1); /* read an integer */

printf("Enter second integer\n"); /* prompt */
scanf("%d", &integer2); /* read an integer */

sum = integer1 + integer2; /* assign total to sum */
printf("Sum is %d\n", sum); /* print sum */

return 0; /* indicate that program ended successfully */
} /* end function main */
```

다음 step까지 실행한 뒤 값이 변한 변수는 debugger window에서 붉은색으로 표시된다.

# Sample Practice

- Debugger를 이용해 오른쪽 code를 debugging한다.
  - 주어진 문자열 'PROGRAMMING'을 출력하고, 이를 뒤집어서 ('GNIMMARGORP') 한 번 더 출력하는 프로그램.
- 실행 결과는 아래와 같다.
  - Ctrl + F5 키(디버깅하지 않고 시작)로 실행했을 때의 결과

```
Before : PROGRAMMING
After  : GNIMMARGORP
계속하려면 아무 키나 누르십시오 . . .
```

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

int main()
{
    int l, r;
    char name[] = "PROGRAMMING";

    printf("Before : %s\n", name);

    l = 0;
    r = strlen(name);

    while (l != r) {
        name[r] = name[l];
        name[l] = name[r];
        l++;
        r--;
    }

    printf(" After : %s\n", name);

    return 0;
}
```