

# 1. C 언어

---

# 1-1 C 언어의 개론

---

## ● 프로그래밍 언어란 무엇인가?

- 사람과 컴퓨터가 이해 할 수 있는 약속된 형태의 언어
- C 언어도 프로그래밍 언어 중 하나이다.

## ● 컴파일이란 무엇인가?

- 프로그래밍 언어로 작성된 프로그램을 컴퓨터가 이해할 수 있도록 기계어로 번역해 주는 역할을 하는 번역기

# 1-1 C 언어의 개론

---

## ● C 언어를 공부한다는 것은?

- 문법을 이해하는 것.
- 프로그래밍 표현 능력을 향상시키는 것.
- 많이 사용할수록 표현에 능숙해진다.
- 다른 이의 표현을 참조할수록 표현이 부드러워진다.

# 1-1 C 언어의 개론(역사와 특징)

---

## ● C 언어의 역사

- 1969~1973년경 UNIX라는 운영체제의 개발을 위해 Dennis Ritchie 와 Ken Thompson이 함께 설계한 범용적인 고급(high-level)언어.

\* 근원: ALGOL 60(1960) ► CPL(1963) ► BCPL(1969) ► B언어(1970)

## ● 고급언어(High Level)? 저급언어(Low Level)?

- 사람이 이해하기 쉬운 언어는 고급언어, 기계어에 가까울수록 저급언어.
- C언어는 고급언어이면서 메모리에 직접 접근이 가능하기 때문에 저급언어의 특성도 함께 지닌다고 이야기한다.

# 1-1 C 언어의 개론

---

## ● C 언어의 장점

- 익숙해 지는데 오랜 시간이 걸리지 않는다. (High-Level)
- 이식성이 좋다.
- 효율성이 높다. (High-Level이지만 Low-Level의 특징을 가지고 있음)

## ● C 언어의 단점

- 프로그래밍 하는데 많은 주의를 요한다.
- 완전한 고급언어(예:Visual-BASIC)에 비해 상대적으로 어렵다.

# 1-2 프로그램의 완성 과정

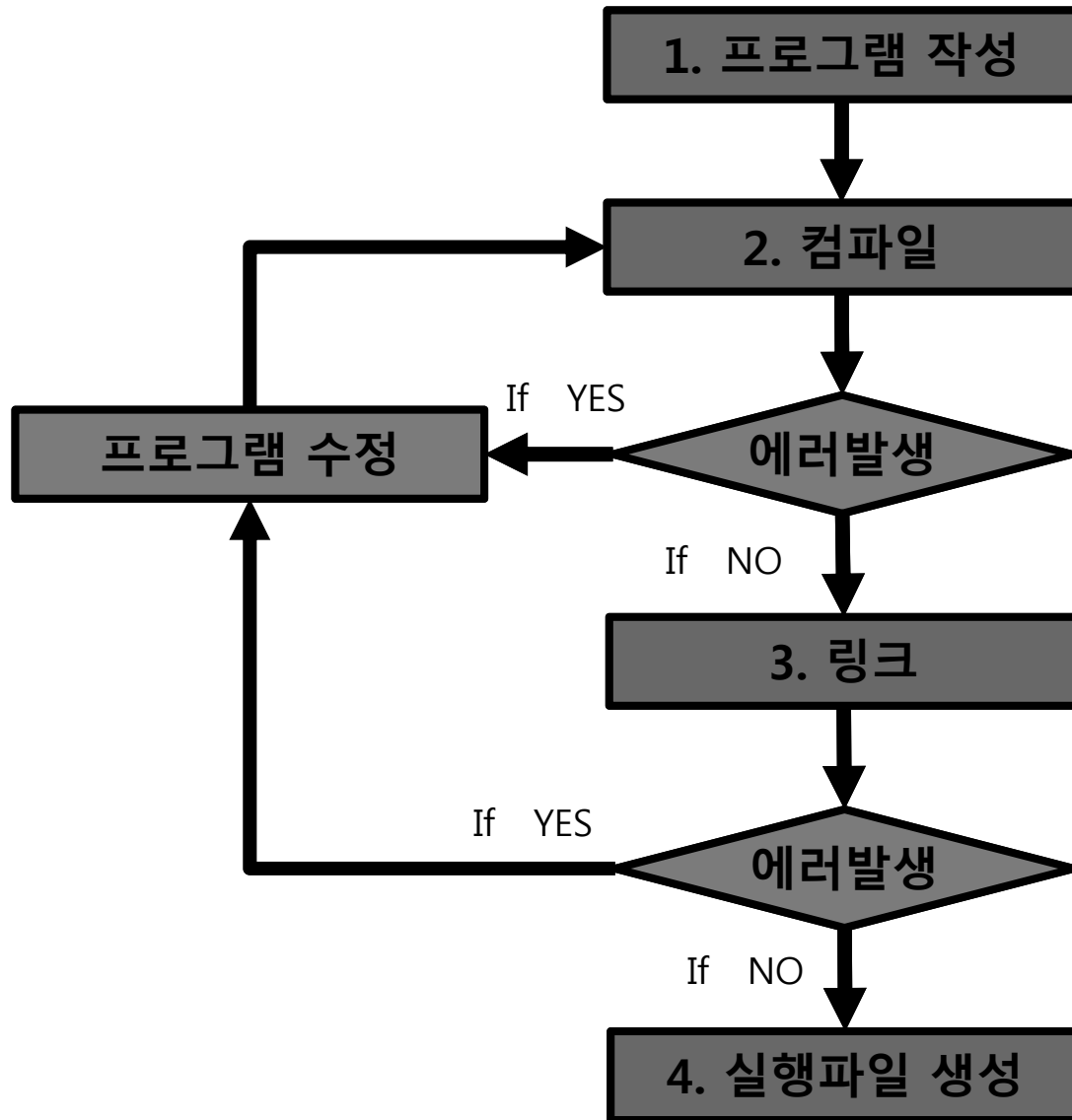
---

## ● 프로그램 작성 및 실행 순서

1. 프로그램 작성 ( C 언어 )
2. 컴파일
3. 링크
4. 실행파일 생성 ( `***.exe` - 응용프로그램 )

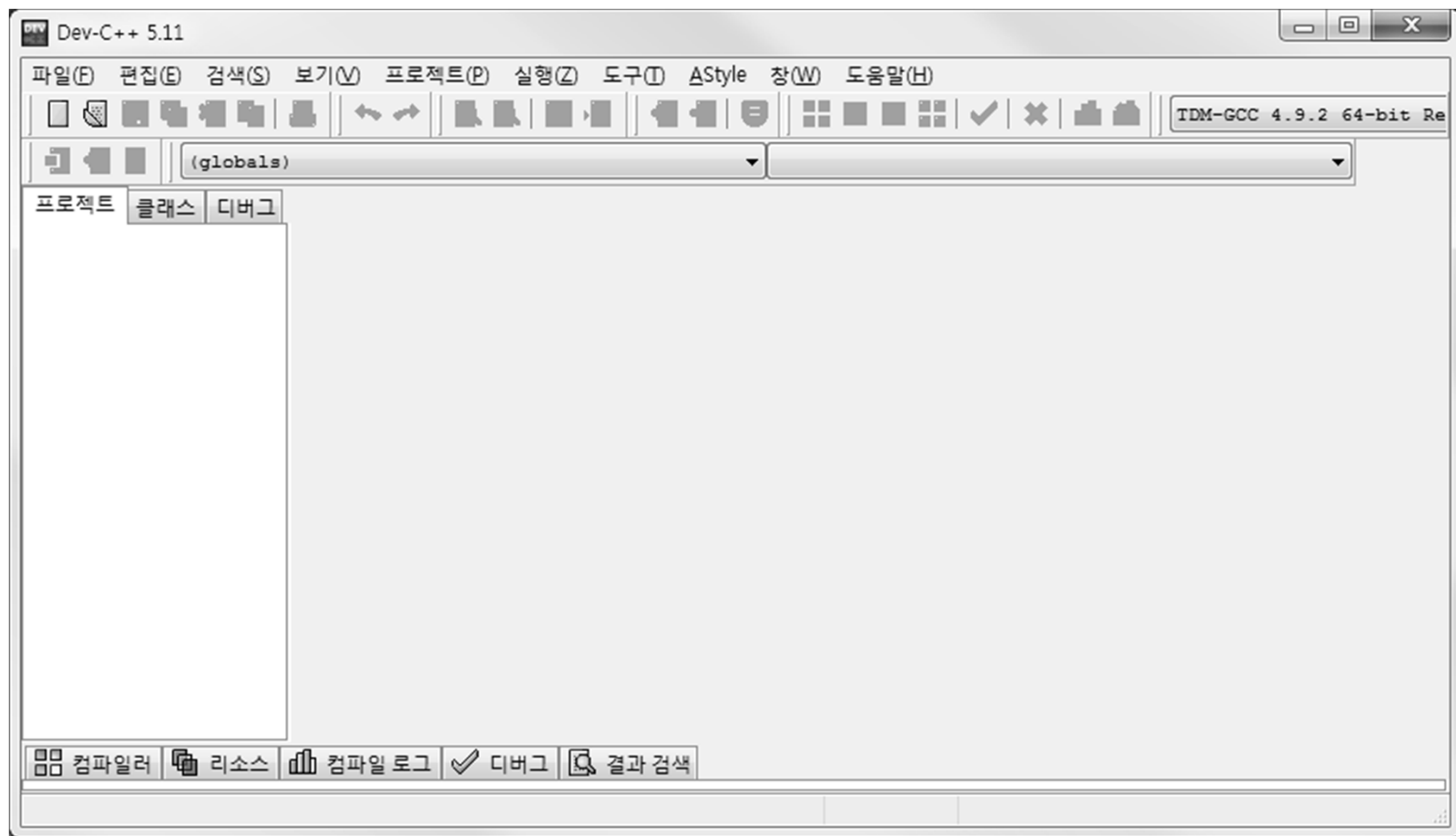
## 1-2 프로그램의 완성 과정

---



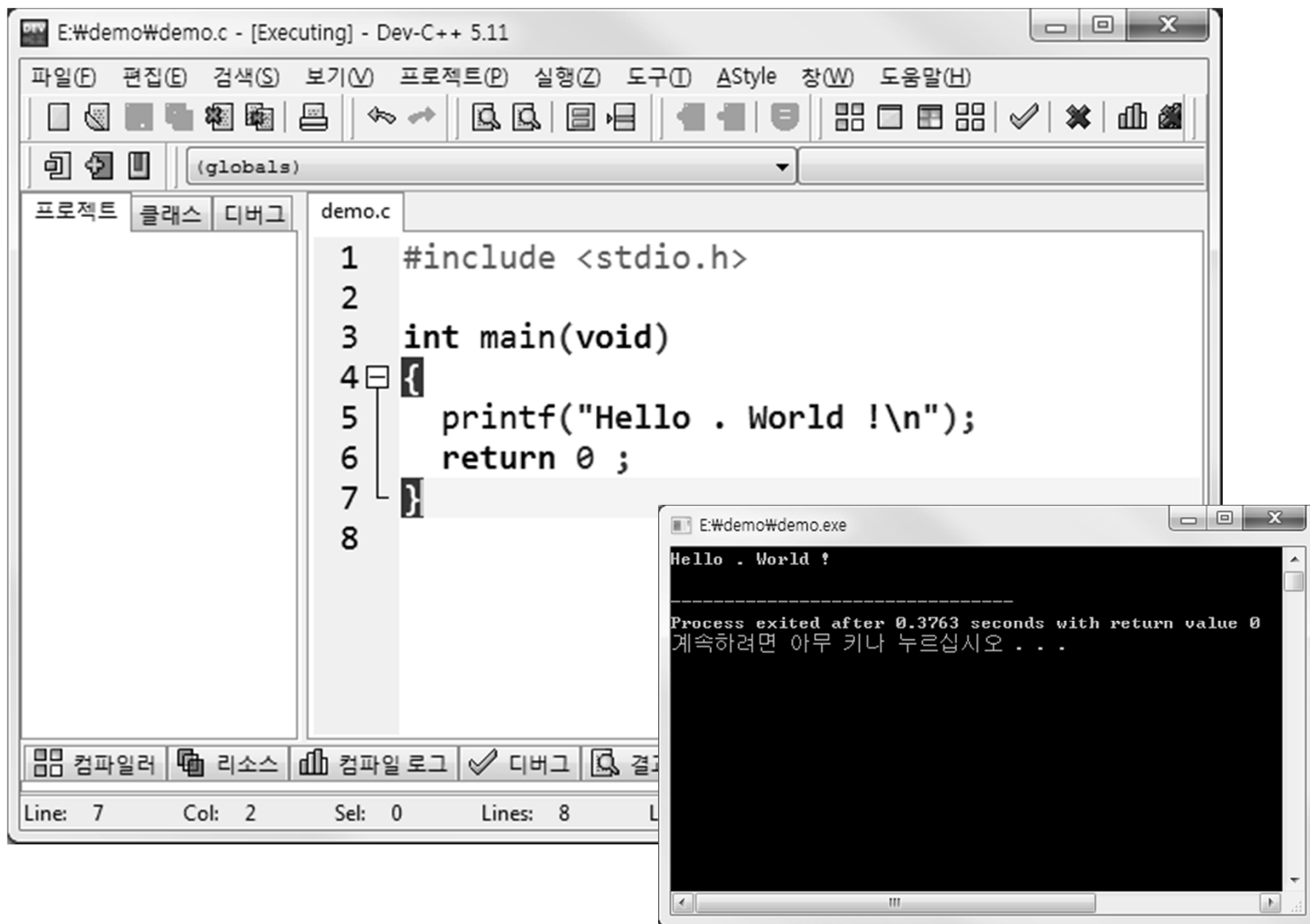
# 1-3 프로그램(DEV C++)의 실행

---





# 1-3 프로그램(Turbo C++)의 실행



## 2. Hello World! 들여다보기

---

## 2-1 기본 단위 “ 함수 ” 의 이해

---

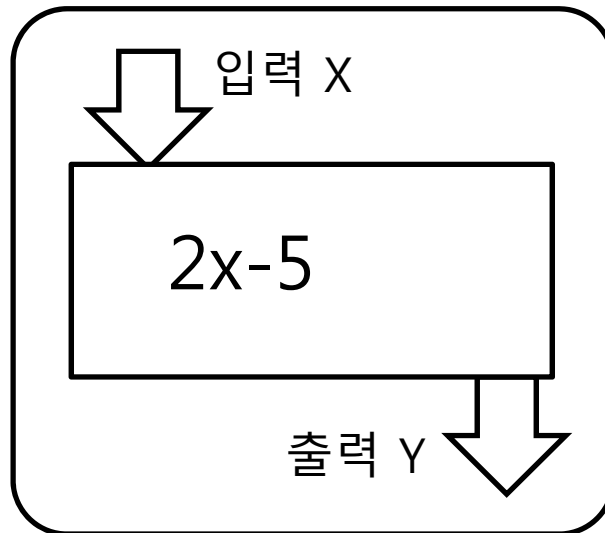
### ● C 언어의 기본 단위

함수를 만들고, 만들어진 함수의 실행 순서를 결정하는 것이 C 언어로 프로그램을 작성하는 것이다.

### ● 수학적 개념의 함수와 C 언어의 함수

[ 수학적 개념 ]

$$2x - 5 = y$$



[ 언어적인 개념 ]

```
int sum(char x)
{
    y=2x-5;
    return y;
}
```

## 2-1 기본 단위 “ 함수 ” 의 이해

---

### ● C 언어 함수와 관련 된 용어

- 함수의 정의 : 만들어진 함수, 실행이 가능한 함수를 일컬음
- 함수의 호출 : 함수의 실행을 명령하는 행위
- 인자의 전달 : 함수의 실행을 명령할 때 전달하는 입력 값

**C언어는 함수로 시작해서 함수로 끝이 난다.**

## 2-2 함수는 어디에?

---

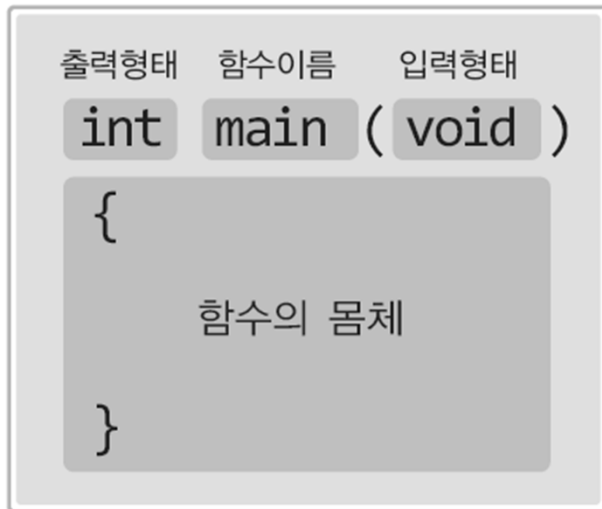
### ● 프로그램의 시작

첫 번째 함수가 호출되면서 프로그램은 시작한다.

### ● 제일 먼저 호출 되는 함수는?

main 함수.

### ● 함수의 구조



- **함수의 이름** : 함수를 호출할 때 사용하는 이름
- **출력형태** : 실행의 결과
- **입력형태** : 함수 호출 시, 전달되는 입력 값.

## 2-3 C 언어의 기본 규칙 - 세미콜론

---

### ● 세미콜론

문장, 즉 명령어의 끝을 표현하기 위한 문자.

### ● 프로그램이 10 줄(line)이면 명령도 10개 인가?

```
int main(void)
{
    printf(" Hello World! \n"); return 0;
}
```

```
int main(void)
{
    printf(" Hello World! \n");
    return 0;
}
```

```
int main(void) { printf(" Hello World! \n"); return 0; }
```

## 2-4 Source Code의 세세한 분석

---

```
#include <stdio.h>  헤더파일 선언
int main(void)
{
    printf("Hello world! \n");
    return 0;        함수 호출
}
```

### ● 표준 함수

이미 만들어져 있어 기본적으로  
제공되는 함수, 그리고 이것들을  
모아놓은 것은 표준 라이브러리

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

"stdio.h 파일을 이 위치에 포함하라."  
표준함수 사용을 위해 반드시 필요.

```
printf("Hello world! \n");
```

" "Hello world!" 문자열을 표시하라."

```
return 0;
```

"사용중인 함수를 종료하라."

## 2-5 C 언어의 기본 규칙 - 주석

---

### ● 주석

소스코드에 삽입된 메모를 뜻하는데, 컴파일 할 때 제외대상이 되므로 실행의 결과에 영향을 미치지 않는다.

### ● 주석의 필요성

실제 코드는 책 읽는 것처럼 쉽지는 않음. 따라서 다음 프로그래머를 위해서나 시간이 지난 작성자 스스로를 위해서 메모는 꼭 필요함.

### ● 블록 단위의 주석

```
/* 주석 처리 문장 */
```

```
/* 주석처리 문장 1
```

```
주석처리 문장 2
```

```
주석처리 문장 3
```

```
*/
```

### ● 행 단위의 주석

```
// 주석처리 문장 1
```

```
// 주석처리 문장 2
```

```
// 주석처리 문장 3
```



## 2-5 C 언어의 기본 규칙 - 주석

---

### ● 주석 처리의 예

```
/*
제목 : Hello World 출력하기
기능 : 문자열의 출력
파일이름 : demo.c
수정일 : 2017-11-20
작성자 : 정기철
*/
#include <stdio.h>    // 헤더파일 선언
int main(void)    // main 함수의 시작
{
    /*
    이 함수 내에서는 하나의 문자열을 출력한다. 문자열은 모니터로 출력된다.
    */
    printf("Hello World!\n");    // 문자열의 출력
    return 0;    // 0의 반환 함수의 종료
}    // main 함수의 끝
```

## 2-5 C 언어의 기본 규칙 - 주석

---

### ● 주석 사용의 옳지 않은 예

```
/*  
주석처리 문장 1  
/* 단일 행 주석 처리 */  
주석처리 문장 2  
주석처리 문장 3  
*/
```

컴파일 오류 있음.

### ● 주석 사용의 올바른 예

```
/*  
주석처리 문장 1  
// 단일 행 주석 처리  
주석처리 문장 2  
주석처리 문장 3  
*/
```

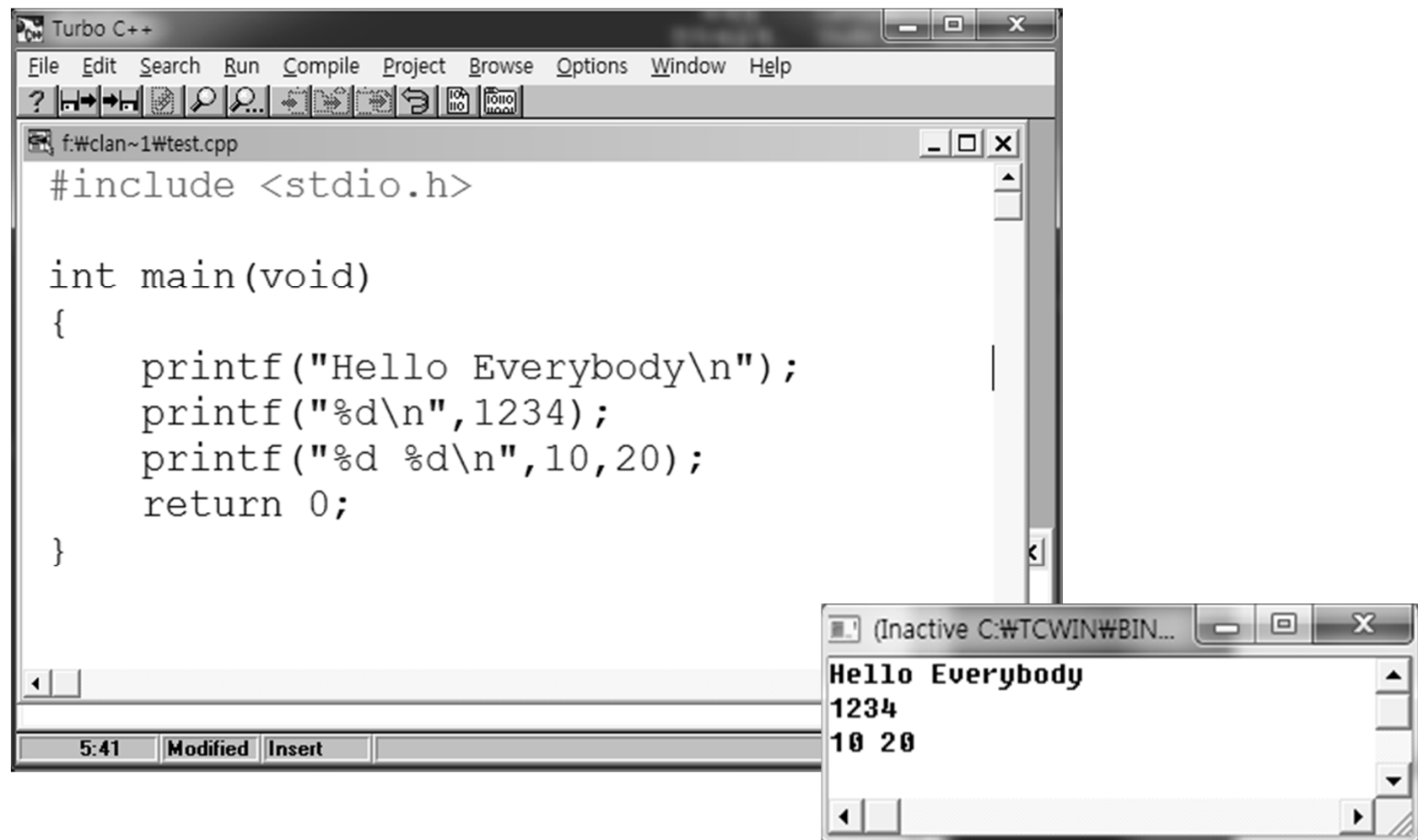
컴파일 오류 없음.

### 3. printf 함수의 활용

---

## 3-1 printf 함수의 다양한 출력

---



The image shows a screenshot of the Turbo C++ integrated development environment. The main window displays a C program file named `f:\wclan~1\test.cpp`. The code includes the `<stdio.h>` header and defines a `main` function that uses `printf` to output three lines of text. A separate window titled `(Inactive C:\TCWIN\BIN...` shows the program's output.

```
#include <stdio.h>

int main(void)
{
    printf("Hello Everybody\n");
    printf("%d\n", 1234);
    printf("%d %d\n", 10, 20);
    return 0;
}
```

Output:

```
Hello Everybody
1234
10 20
```

## 3-1 printf 함수의 다양한 출력

f:\wclan~1\test.cpp

```
#include <stdio.h>

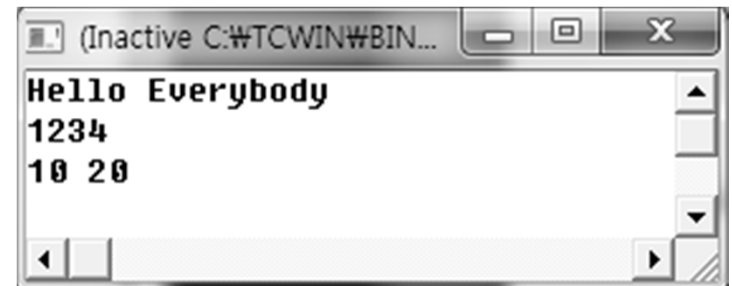
int main(void)
{
    printf("Hello Everybody\n");
    printf("%d\n", 1234);
    printf("%d %d\n", 10, 20);
    return 0;
}
```

### ● \n

특수 문자로 "Enter" 기능,  
즉 행 띄우기 기능을 가지고  
있다.

### ● %d

부호가 있는 10진수 정수를 표시하라.  
서식 문자로 출력의 형태를 지정한다.



## 3-1 printf 함수의 다양한 출력

---

- `printf("Hello Everybody \n");`



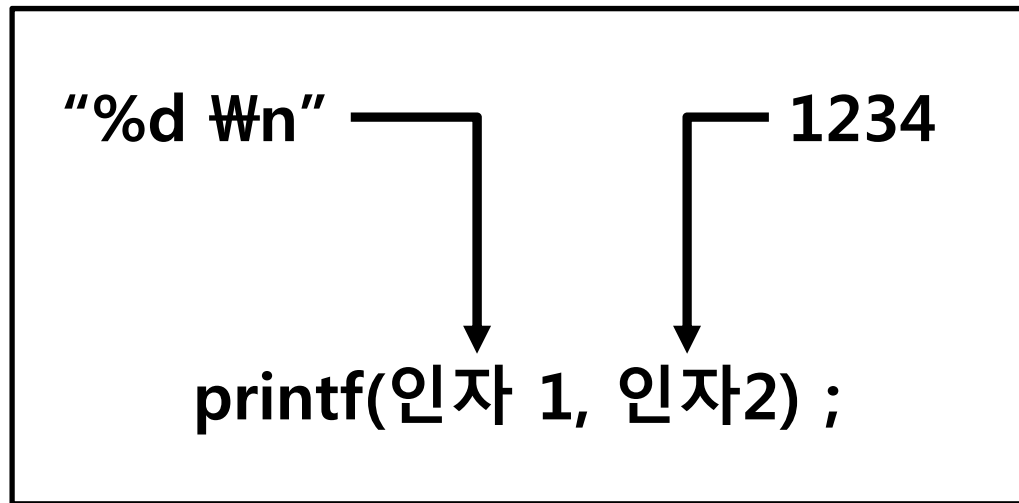
`printf("Hello Everybody \n");`

모니터에 출력

## 3-1 printf 함수의 다양한 출력

---

- `printf("%d \n",1234);`



`printf("%d \n",1234);`



## 3-1 printf 함수의 다양한 출력

---

- `printf("%d %d \n",10,20);`

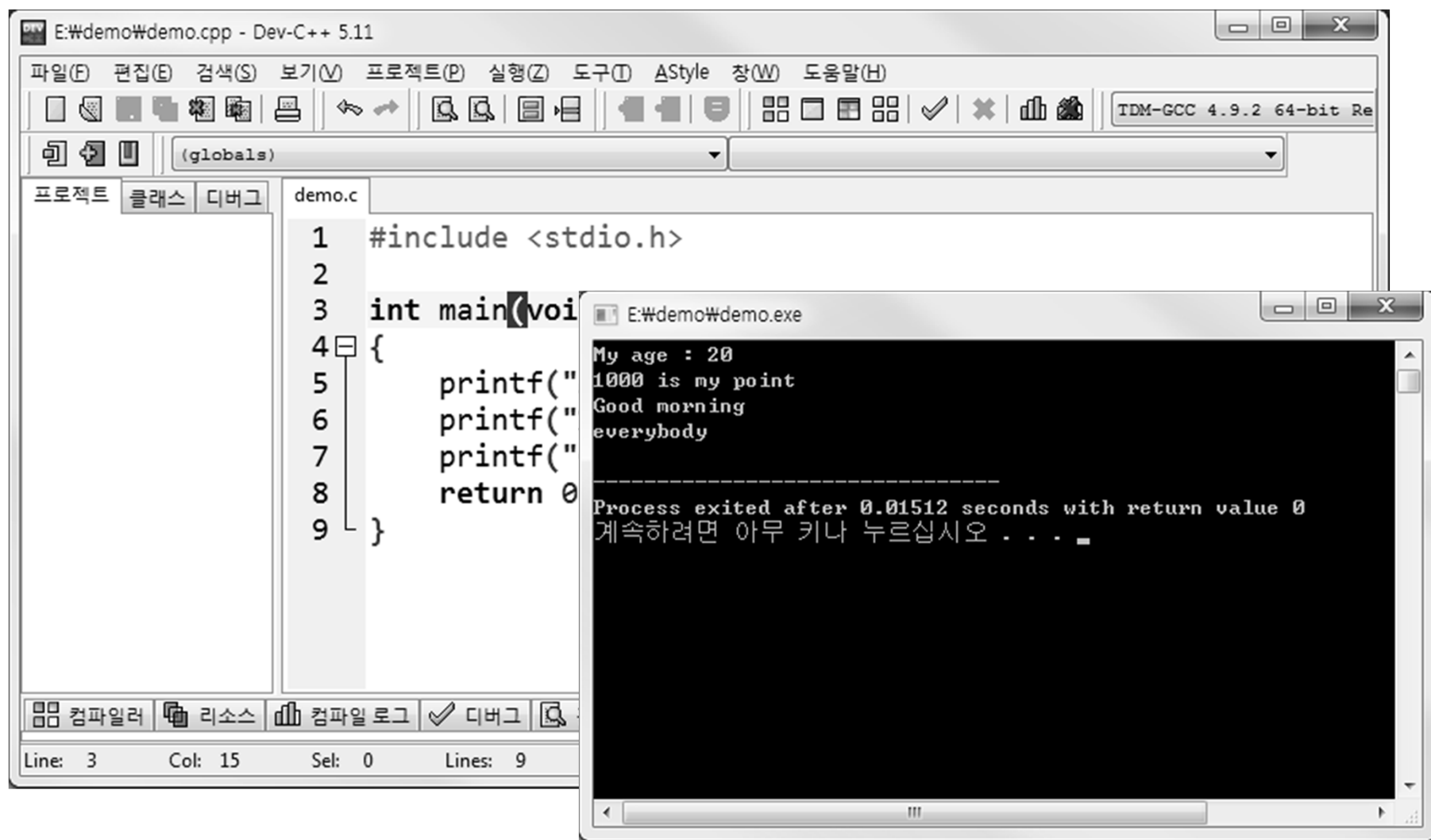
`"%d %d \n"`      `10`      `20`

`printf(인자 1, 인자2, 인자 3) ;`

`printf("%d %d \n",10,20);`



## 3-1 printf 함수의 다양한 출력



The image shows a screenshot of the Dev-C++ 5.11 IDE. The main window displays a C program named `demo.c` with the following code:

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     printf("My age : 20\n");
6     printf("1000 is my point\n");
7     printf("Good morning\n");
8     printf("everybody\n");
9     return 0;
}
```

The output window, titled `E:\demo\demo.exe`, shows the execution results:

```
My age : 20
1000 is my point
Good morning
everybody

-----
Process exited after 0.01512 seconds with return value 0
계속하려면 아무 키나 누르십시오 . . .
```

The status bar at the bottom of the IDE indicates the current position: Line: 3, Col: 15, Sel: 0, Lines: 9.

### 3. 연습문제

---

1. 자신의 이름과 나이, 주소 등을 3줄에 걸쳐서 다음과 같이 출력하는 프로그램을 작성하여보자.

이름 : 정인지

나이 : 24살

주소 : 대전시 유성 553

2. 다음과 같은 형태로 출력하는 프로그램을 작성하여보자.  
단, printf()는 한번만 호출하여야 한다.

Hello

C

Programmers!

### 3. 리포트 표지를 출력하는 프로그램을 작성하여 보자

\*\*\*\*\*

\*            프로그래밍            \*

\*            리   포   트            \*

\*\*\*\*\*

-----

| 전기 전자과 |

-----

201710001

정 재 문

4. 다음과 같이 출력되는 프로그램을 작성하여 보자.

일	월	화	수	목	금	토
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

힌트 :

1. 특수문자 `\t`는 키보드에서 탭키를 누른 것처럼 커서를 이동 시킨다. 따라서 값을 들 정렬시켜서 출력하는데 사용된다. `printf("일\t월\t화\t수\t목\t금\t토\t\n");`
2. `%d`를 사용하여 날짜를 표시하여라.