

## 챕터2. 나의 첫 번째 C 프로그램

### 1절. Hello, World!

핵심 키워드	GCC, 소스 파일, 실행 파일
--------	-------------------

여기서는 무얼 배울까



세상에서 가장 많은 사람이 만들어본 프로그램은 무슨 프로그램일까? 진지하게 그런 프로그램이 어떤 것인지 조사한 사람은 없겠지만, 단언컨대 지금의 모든 개발자는 이 질문에 Hello, World! 프로그램이라고 대답할 것이다. 그들은 모두 그 프로그램을 만들어봤고, 한 번이 아닌 수십, 수백 번도 만들어봤을 것이며, 그들이 본 다른 모든 개발자도 그 프로그램을 만들었을 것이기 때문이다. 이번 챕터에서는 C 언어 프로그래밍을 위한 컴파일러를 설치하고, 우리의 첫 번째 프로그램인 Hello, World! 프로그램 작성하고 실행하는 시간을 갖는다.

#### 1. GCC 설치하기

이 책에서는 Ubuntu를 비롯한 리눅스에서 사용할 수 있는 컴파일러인 GCC를 사용한다. GCC는 GNU 프로젝트의 오픈 소스 컴파일러로 C, C++, Fortran 등 여러 언어의 프로그램 개발을 지원한다. 대부분의 리눅스 컴퓨터는 기본적으로 GCC가 설치되어 있지만, 그렇지 않은 경우도 있으므로 먼저 GCC를 설치하는 방법을 설명한다.

```
woolyung@DESKTOP-JVP38VM:~$ gcc --version
gcc (Ubuntu 9.4.0-1ubuntu1~20.04.1) 9.4.0
Copyright (C) 2019 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

① `gcc --version`으로 GCC 설치 여부를 확인한다. 만약 설치가 되어있다면 다음의 설치 과정이 필요하지 않다.

```
sudo apt update
sudo apt install build-essential
```

② 다음의 명령어로 업데이트 및 프로그램들을 설치한다. 여기에는 GCC와 함께 make, G++ 등 개발을 위한 여러 프로그램이 포함된다.

## 2. Hello, World! 프로그램 작성하기

GCC는 C 언어로 작성된 파일을 컴파일하여 실행할 수 있는 파일로 역할을 한다. 그러므로 우리가 처음으로 할 일은 텍스트 에디터를 통해 소스 파일을 만드는 것이다.

```
woolyung@DESKTOP-JVP38VM:~/C-programming$ touch main.c
woolyung@DESKTOP-JVP38VM:~/C-programming$ ls
main.c
```

① touch main.c로 main.c라는 이름의 C 언어 소스 파일을 디렉토리에 생성한다.

```
#include <stdio.h>

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

② vi main.c로 main.c 파일에 다음의 코드를 작성한다.

DELL 2023/09/15 20:41

**코멘트**  
이 책에서는 리눅스의 기본 텍스트 에디터인 vi의 사용법을 알고 있음을 가정합니다.

손으로 익히는 코딩
<pre>#include &lt;stdio.h&gt;  int main(void) {     printf("Hello, World!");     return 0; }</pre>

실행 결과
Hello, World!

③ :wq 또는 :x로 저장 및 vi 프로그램을 종료한다.

## 3. 프로그램 실행하기

앞서 코드를 작성한 main.c 파일은 C 언어 파일, 즉 사람이 이해하기 쉽도록 만들어진 고급 프로그래밍 언어로 작성된 파일이다. 그렇기에 이 파일에 담긴 내용을 컴퓨터가 이해하고 실행하기 위해서는 컴파일 과정이 필요하다. 이처럼 고급 프로그래밍 언어로 작성되어 컴파

일되기 전의 파일을 **소스 파일**이라고 한다. 그리고 소스 파일을 컴파일하여 실제 컴퓨터가 실행할 수 있도록 기계어로 구성된 파일을 **실행 파일**이라고 한다. 앞서 만든 소스 파일을 실행시키기 위해선 컴파일 과정을 통해 실행 파일로 만드는 것이 선행되어야 한다.

#### (1) 컴파일

```
woolyung@DESKTOP-JVP38VM:~/C-programming$ gcc main.c -o main
woolyung@DESKTOP-JVP38VM:~/C-programming$ ls
main  main.c
```

컴파일은 GCC 명령어를 이용한다. `gcc main.c`는 GCC 컴파일러로 `main.c` 파일을 컴파일한다는 뜻이다. `-o` 옵션과 바로 뒤의 `main`은 컴파일 이후 실행 파일의 이름을 `main`으로 지정한다는 뜻이다.

#### (2) 실행

```
woolyung@DESKTOP-JVP38VM:~/C-programming$ ./main
Hello, World!woolyung@DESKTOP-JVP38VM:~/C-programming$
```

실행 파일은 마치 명령어를 입력하듯 실행하고자 하는 프로그램의 경로로 실행한다. 여기서 컴파일된 `main` 프로그램은 `Hello, World!`를 출력하는 기능을 하므로 `main`을 실행했을 때 그 메시지가 화면에 보이는 것을 확인할 수 있다.

#### 2절. C 언어의 구성 요소

Windows 버전의 1권 2챕터 3절과 동일

<b>연습문제</b>
<b>OX 퀴즈</b>
C 언어를 개발하기 위해선 GCC를 필수적으로 사용해야 한다.
GCC는 C 언어가 아닌 다른 프로그래밍 언어를 지원한다.
<b>모두 고르기</b>
다음 중 문제가 없는 서술문의 형태를 모두 고르시오.
<code>printf("Hello, World!");</code>
<code>scanf("%f", &amp;height)</code>

DELL 2023/02/12 22:15

기초 용어 정리

소스 파일

고급 프로그래밍 언어로 작성되어 컴파일되기 전의 파일

DELL 2023/02/12 22:15

기초 용어 정리

실행 파일

컴퓨터가 실행할 수 있도록 기계어로 구성된 파일

```
{  
    int a = 5;  
    int b = 10;  
    int c = a + b;  
}
```

```
{  
    { { } } {  
        { }  
    } }  
}
```

```
{  
    printf("Hello, C!");  
}
```

#### 단답형

다음 프로그램은 몇 개의 단일 서술문과 몇 개의 복합 서술문으로 구성되어 있는가?

```
#include <stdio.h>  
  
int main(void)  
{  
    {  
        printf("A");  
        printf("B");  
        {  
            printf("C");  
            {  
                printf("D");  
            }  
        }  
    }  
    printf("E");  
}
```

## 한 장으로 요약 정리

### 키워드로 정리하기

- GCC는 GNU 프로젝트의 오픈 소스 컴파일러로 C, C++, Fortran 등 여러 언어의 프로그램 개발을 지원한다.
- 고급 프로그래밍 언어로 작성된 파일을 **소스 파일**, 소스 파일을 컴파일하여 컴퓨터가 실행할 수 있는 상태의 파일을 **실행 파일**이라고 한다.
- **콘솔**은 화면에 글자들을 출력하고 사용자로부터 키보드 입력을 받을 수 있는 창을 의미한다.
- C 언어의 최소 실행 단위는 **단일 서술문**으로 세미콜론으로 구분하며, 0개 이상의 단일 서술문을 중괄호로 묶어 **복합 서술문**으로 만들 수 있다.

### 예제 훑아보기

```
#include <stdio.h> // (1)

int main(void) // (2)
{
    printf("Hello, World!"); // (3)
    return 0; // (4)
}
```

- (1) 이 프로그램에서 표준 입출력을 사용하겠다는 의미이다.
- (2) 프로그램 본문의 시작이다.
- (3) 콘솔에 Hello, World라는 글자를 출력한다.
- (4) 프로그램이 문제없이 실행되었음을 알린다.

## 연습 문제 해설과 답

### OX 퀴즈

C 언어를 개발하기 위해선 GCC를 필수적으로 사용해야 한다.

- X, C 언어를 개발하기 위한 프로그램은 다양하므로 다른 프로그램을 사용해도 괜찮다.

GCC는 C 언어가 아닌 다른 프로그래밍 언어를 지원한다.

- O, GCC는 C++, Fortran 등의 프로그래밍 언어도 지원한다.

### 모두 고르기

다음 중 문제가 없는 서술문의 형태를 모두 고르시오.

```
printf("Hello, World!");
```

O, 완전한 형태의 단일 서술문이다.

```
scanf("%f", &height)
```

X, 단일 서술문은 세미콜론으로 끝나야 한다.

```
{
    int a = 5;
    int b = 10;
    int c = a + b;
}
```

O, 3개의 단일 서술문을 묶은 복합 서술문이다.

```
{
    { { } } {
        { }
    } }
}
```

X, 중괄호는 한 쌍이 복합 서술문을 이루지만 여기서는 닫는 중괄호 하나가 쌍이 없다.

```
{
    printf("Hello, C!");;
}
```

O, 2개의 단일 서술문을 묶은 복합 서술문입니다. 마지막 세미콜론은 빈 단일 서술문을 의미합니다.

### 단답형

다음 프로그램은 몇 개의 단일 서술문과 몇 개의 복합 서술문으로 구성되어 있는가?

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

```
int main(void)
{
    {
        printf("A");
        printf("B");
        {
            printf("C");
            {
                printf("D");
            }
        }
    }
    printf("E");
}
```

중괄호가 4쌍이므로 복합 서술문 4개, 세미콜론이 5개이므로 단일 서술문 5개이다.

[2쪽] [메모:3] DELL 2023/09/15 20:41

#### 코멘트

이 책에서는 리눅스의 기본 텍스트 에디터인 vi의 사용법을 알고 있음을 가정합니다.

[3쪽] [메모:1] DELL 2023/02/12 22:15

#### 기초 용어 정리

#### 소스 파일

고급 프로그래밍 언어로 작성되어 컴파일되기 전의 파일

[3쪽] [메모:2] DELL 2023/02/12 22:15

#### 기초 용어 정리

#### 실행 파일

컴퓨터가 실행할 수 있도록 기계어로 구성된 파일