

6강. 소스코드와 명령어

개발자가 작성한 소스 코드

```
#include <stdio.h>

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

예시

고급 언어와 저급 언어

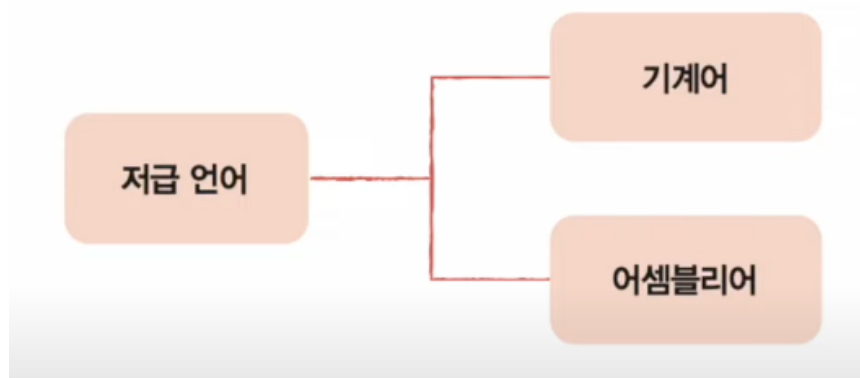
고급 언어와 저급 언어의 관계는 다음 그림과 같습니다.



고급 언어는 개발자가 이해하기 쉽게 만든 언어입니다.

반대로 **저급 언어**는 컴퓨터가 이해하고 실행하는 언어입니다.

저급 언어는 다음과 같이 분류할 수 있습니다.



기계어는 0과 1로 이루어진 명령어로 구성되어 있습니다.

(십육진수로도 표현된 기계어도 있습니다.)

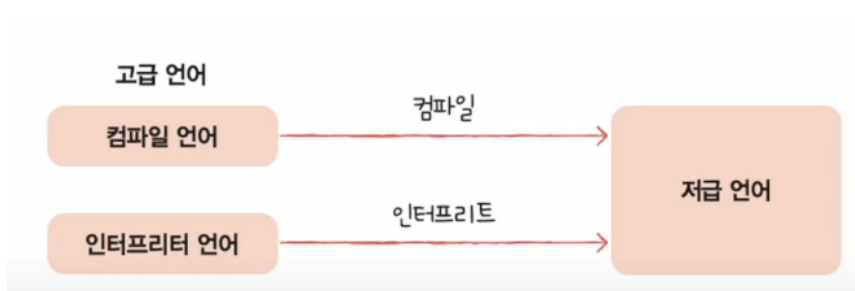
이진수(0과 1)로 표현된 기계어

```
01111111 01000101 01001100 01000110 00000010 00000001
00000001 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000011 00000000
10110111 00000000 00000001 00000000 00000000 00000000
00010000 00000110 00000000 00000000 00000000 00000000
00000000 00000000 01000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 11010000 00011100
00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 01000000 00000000
00111000 00000000 00010001 00000000 01000000 00000000
00011100 00000000 00011011 00000000 00000110 00000000
00000000 00000000 00000100 00000000 00000000 00000000
```

어셈블리어(assembly language)는 0과 1로 이루어진 기계어를 읽기 편한 형태로 번역한 저급 언어입니다.

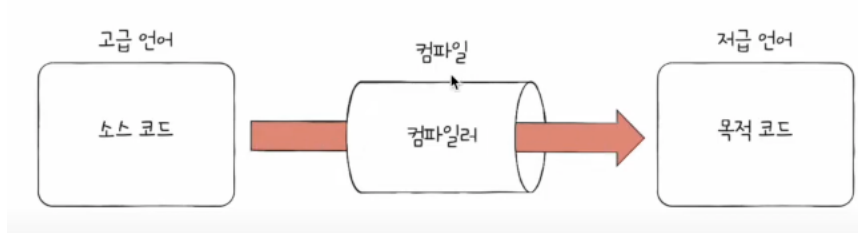
기계어	어셈블리어
0101 0101	→ push rbp
0101 1101	→ pop rbp
1100 0011	→ ret

컴파일 언어와 인터프리터 언어

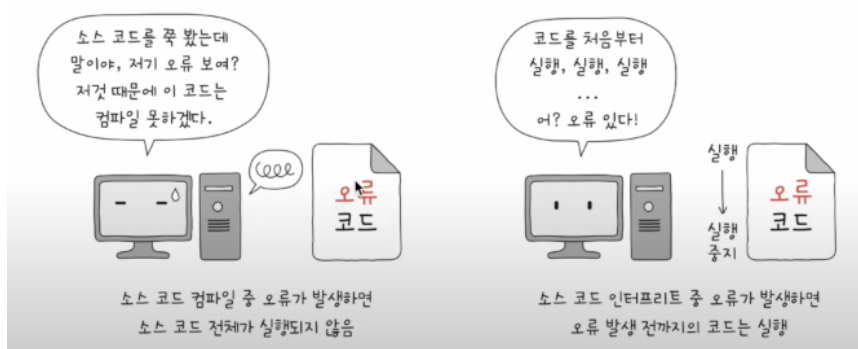


고급 언어는 컴파일 언어와 인터프리터 언어로 나눌 수 있습니다.

컴파일 언어로 작성된 소스 코드는 컴파일러에 의해 저급 언어로 변환됩니다. 이 과정을 컴파일이라고 합니다. 컴파일 결과로 저급 언어인 목적 코드가 생성됩니다.



인터프리트 언어는 소스 코드 전체가 저급 언어로 변환되지 않고, 한 줄씩 실행됩니다.



컴파일 & 인터프리트 과정 알아보기

해당 링크에서는 다양한 언어가 어떻게 목적코드로 변환되는 지 실습해보실 수 있습니다.

- <https://godbolt.org/>