

# Python 변환을 지원하는, 한글 프로그래밍 언어 KoBasic

## KoBasic이 필요한 이유

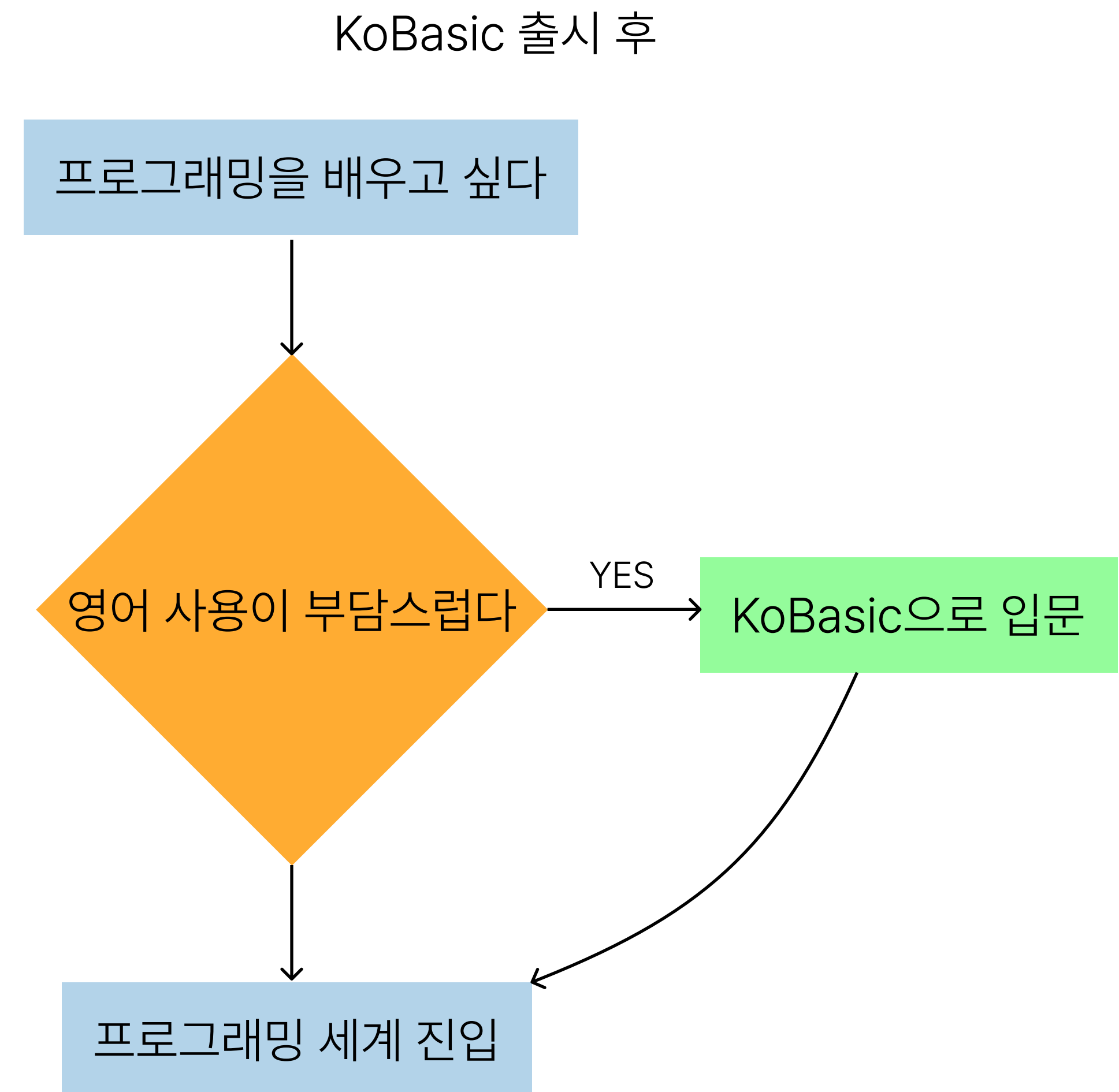
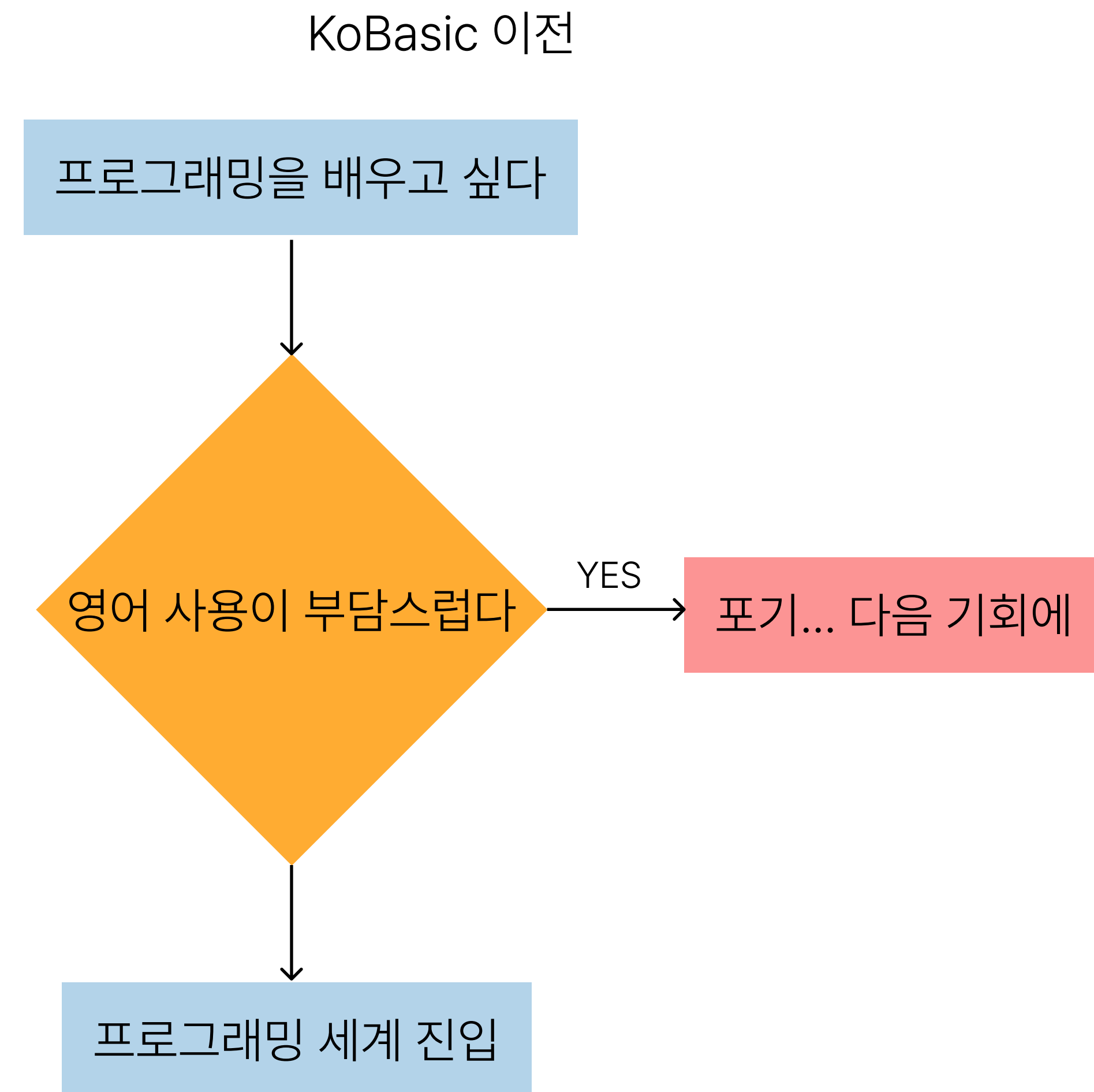


[a] [b] [c] [d] [e]  
[f] [g] [h] [i] [j] [k]  
[l] [m] [n] [o] [p]  
[q] [r] [s] [t] [u] [v]  
[w] [x] [y] [z]

✓ 프로그래밍을 배우고 싶지만 영어에 익숙지 않아 영어 키워드 위주의 기존 프로그래밍 언어에 어려움을 느끼는 어린 학생들이 한글 키워드로 구성되어 있는 KoBasic을 통해 프로그래밍에 수월하게 입문할 수 있음.

✓ 또한 파이썬 변환 기능을 통해 한글 프로그래밍 언어로 진입한 사용자가 이후 파이썬 세계 및 본격적인 프로그램 세계로 쉽게 진입할 수 있도록 돕고 있음.

## KoBasic 출시 이후 기대효과



## KoBasic의 실행 구조

그림 1. 전체 구조

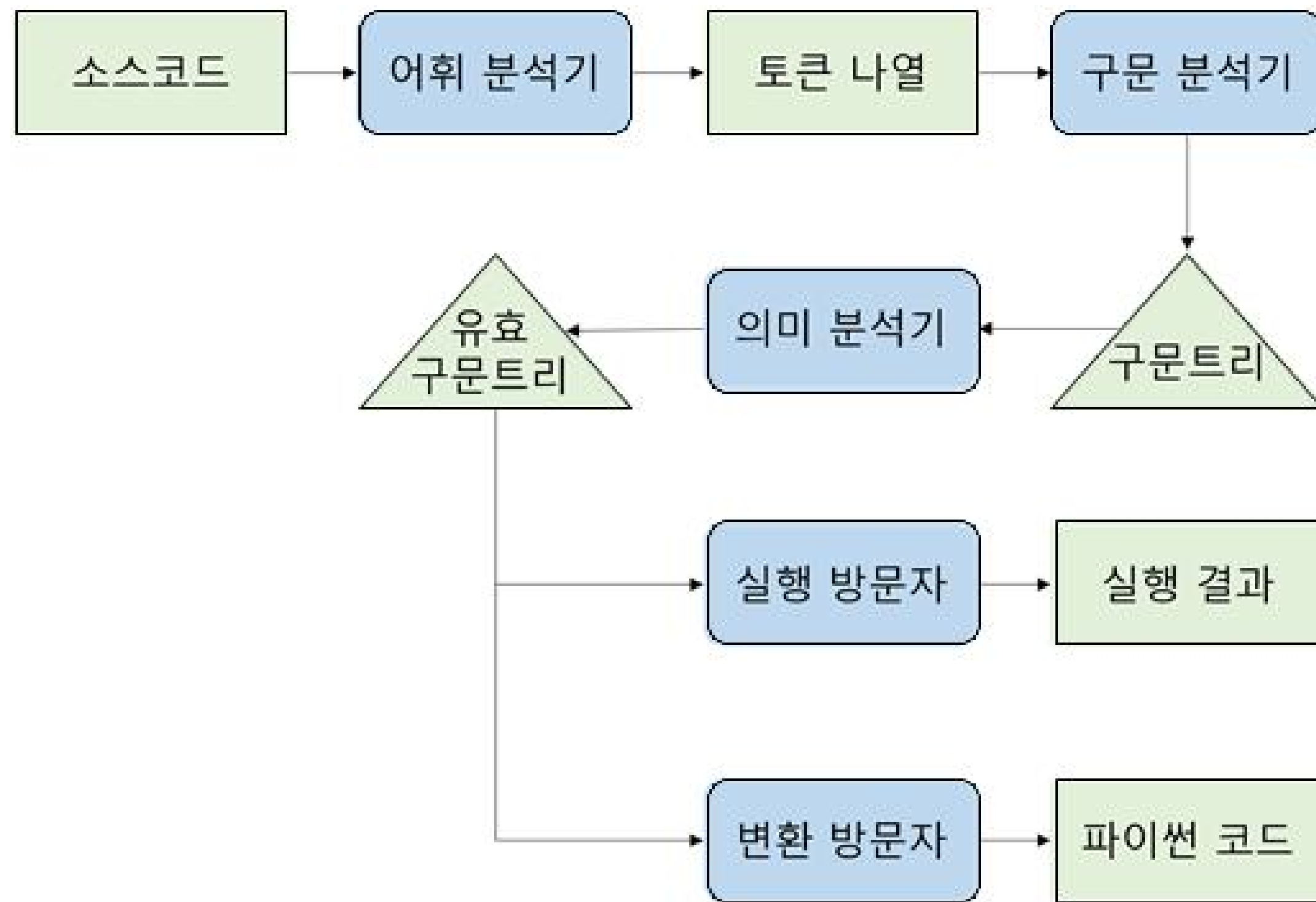
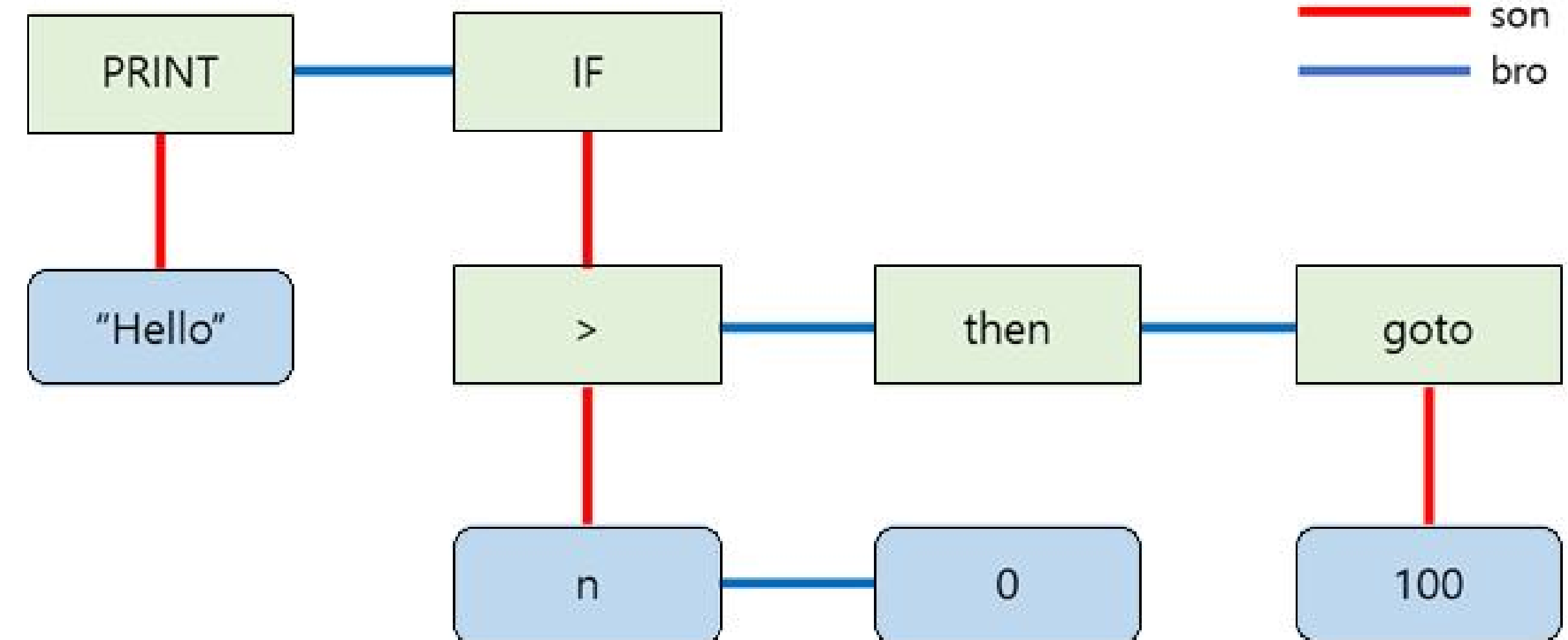


그림 2,3. 구문 트리 예시

```

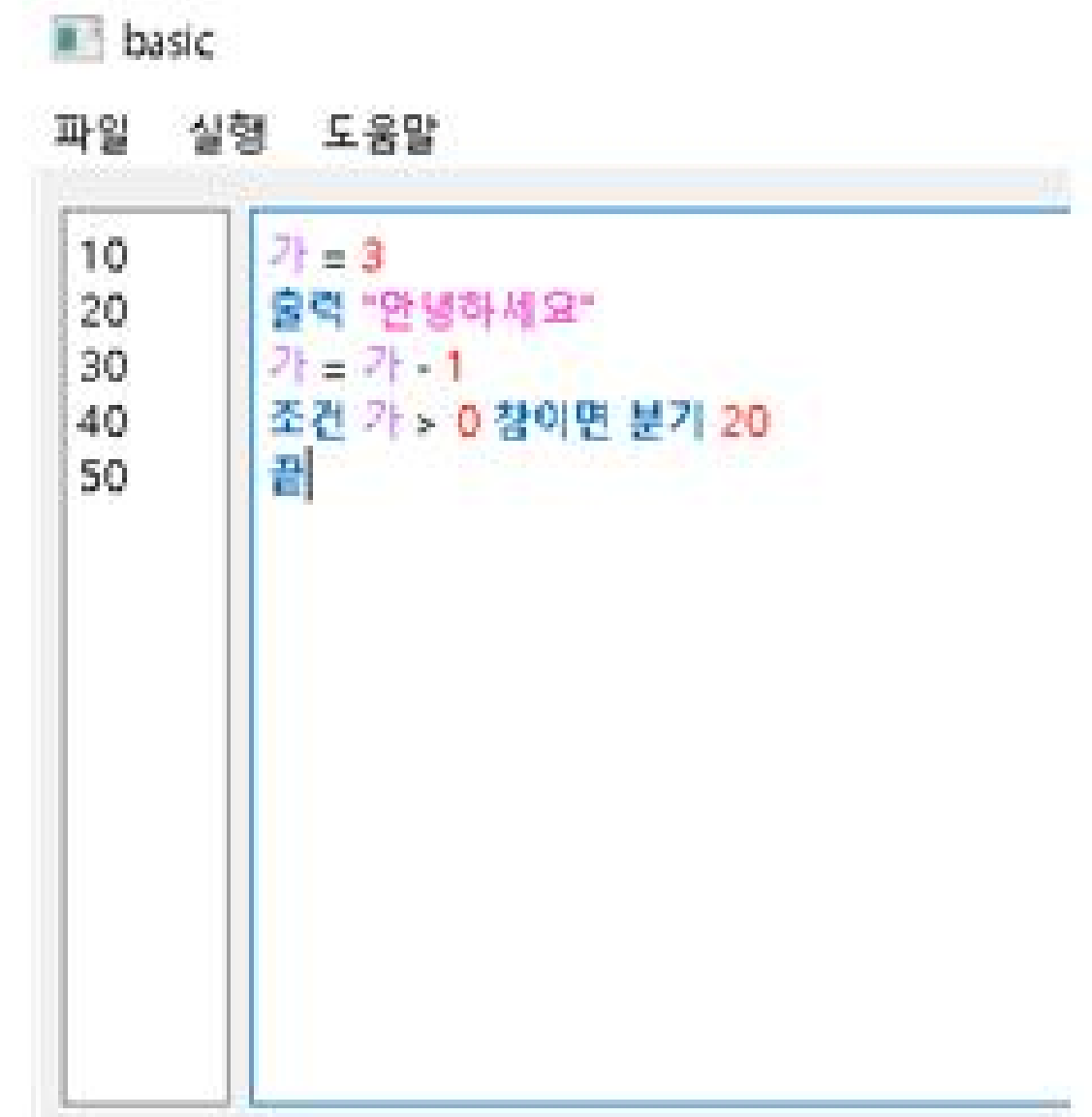
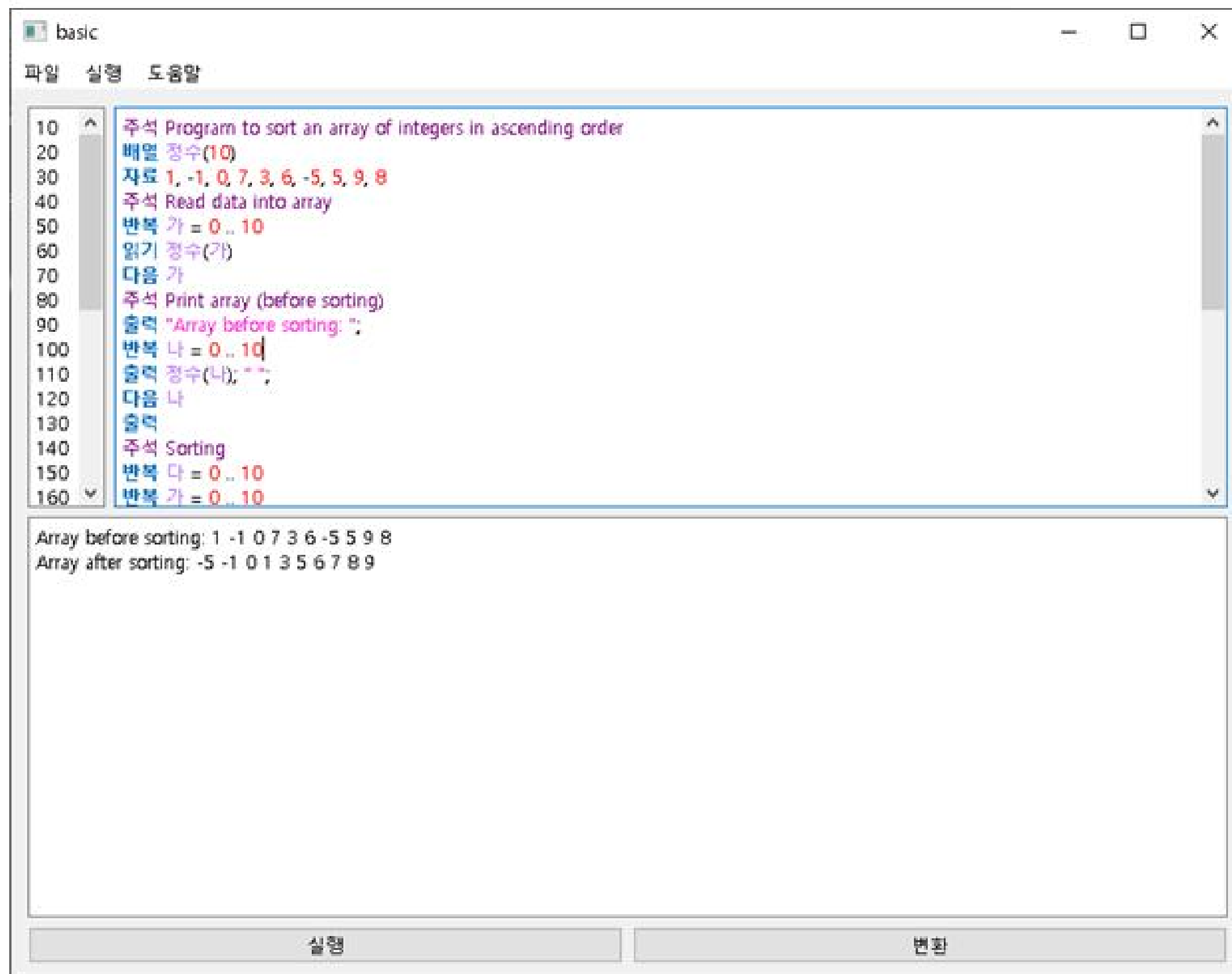
10 print "hello"
20 if n > 0 then goto 100
  
```



## KoBasic GUI

Qt로 개발한 GUI 애플리케이션으로 윈도우 환경에서 설치하여 사용이 가능

- 리넘버링 기능을 추가로 개발하여 편의성을 높임





# KoBasic Web

## KoBasic to Python

KoBasic 코드를 간단하게 Python으로 바꾸고, 실행해보세요!

KoBasic 코드 입력

```
10 반복 I = 1 부터 3 증가분 1
20  반복 J = 1 부터 3 증가분 2
30  출력 I = "I: ", J = "J: ", I * J = "I * J
40  다음 J
50 다음 I
```

변환하기

Python 코드

```
def main():
    try:
        stmt10()
    except End:
        return

vars = {}
```

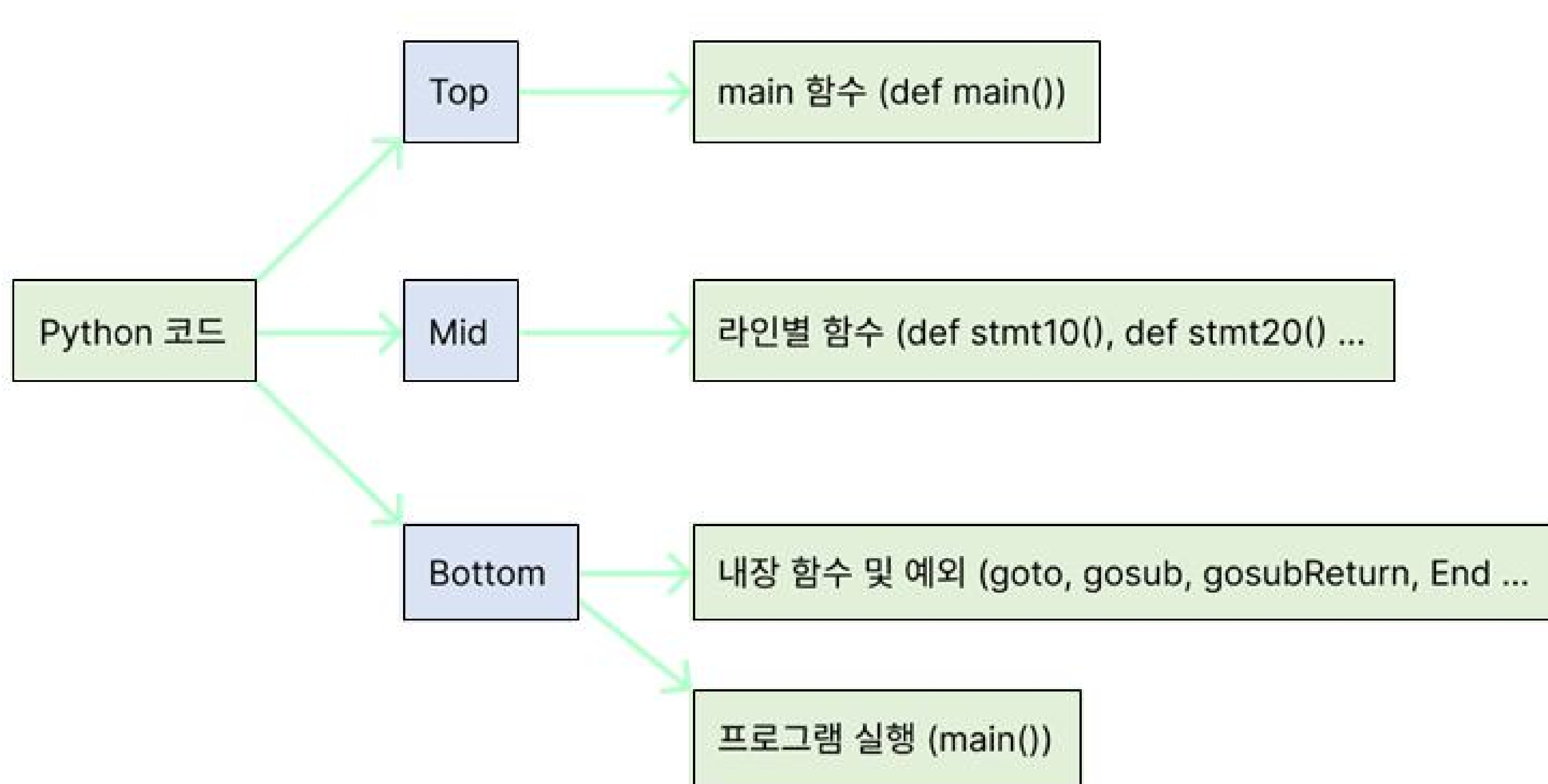
코드 실행 결과

```
I = 1, J = 1, I * J = 1
I = 1, J = 3, I * J = 3
I = 2, J = 1, I * J = 2
I = 2, J = 3, I * J = 6
I = 3, J = 1, I * J = 3
I = 3, J = 3, I * J = 9
```

자바스크립트로 구성한 코드 제너레이터를 리액트 애플리케이션에 포함 시켜, 웹 클라이언트 사이드에서 KoBasic코드를 실행할 수 있도록 함. 휴대폰, PC 등 브라우저를 사용할 수 있는 환경에서 실행 가능

<https://kimjeonghan0625.github.io/kobasic/>

## KoBasic으로부터 변환된 파이썬 코드의 구조



## KoBasic으로부터 변환된 파이썬 코드의 구조



```
1  def main():
2      try:
3          stmt10()
4      except End:
5          return
6
7  vars = {}
8
9  def stmt10():
10     for vars["I"] in range(1, 4, 1):
11         stmt20()
12         stmt50()
13
14 def stmt20():
15     for vars["J"] in range(1, 4, 2):
16         stmt30()
17         stmt40()
18
```

```
18
19 def stmt30():
20     print("I = ", end="")
21     print(vars["I"], end="")
22     print(", J = ", end="")
23     print(vars["J"], end="")
24     print(", I * J = ", end="")
25     print((vars["I"] * vars["J"]), end="")
26     print()
27
28 def stmt40():
29     pass
30
31 def stmt50():
32     pass
33
34 # 내장 함수 및 예외 Inner Function and Exception
35
36 class End(Exception):
37     pass
38
39
40 # 프로그램 실행
41
42 main()
```