

## 1. 장점:

다양한 자료형들에 대해 배우는 학습량이 줄어들어, 언어를 더 배우기 쉽게 만들 수 있다.

또한, 개발 도중에 type의 변경이 생기거나 새로운 기능을 도입할 때 이를 이식하고 프로그램을 확장하기에도 용이하다.  $\Rightarrow$  flexibility  $\uparrow$

## 단점:

Type이 없어서 개발 중 발생한 오류를 쉽게 catch해내기 어렵고, 런타임 때 여러 발생들을 놓칠 수 있다.

코드를 검토함에 있어서도 각 변수의 type이 없어서 변수간의 구분이나 의미 분석이 어려워 코드 가독성을 저하시킬 수 있다. 형식을 함에 있어서도 코드의 선명도가 낮아 유지보수가 힘들어질 수 있다.

## 2.

Body	variable	where it is declared.
Sub 1	a	Sub 1
	y	Sub 1
	z	Sub 1
	x	main
Sub 2	a	Sub 2
	b	Sub 2
	z	Sub 2
	x	main
	y	Sub 1
Sub 3	a	Sub 3
	x	Sub 3
	w	Sub 3
	y	main
	z	main

3.

Body	variable	where it is declared.
sub 1	x	main
	y	sub 1
	z	sub 1
	a	sub 1
sub 2	x	sub 2
	y	main
	z	main
	a	sub 2
	w	sub 2
sub 3	x	sub 2
	y	main
	z	sub 3
	a	sub 3
	w	sub 2
	b	sub 3

4.

a.

d, e, f	fun 3
c	fun 2
b	fun 1
a	main

b.

d, e, f	fun 3
b, c	fun 1
a	main

c.

b, c, d	fun 1
e, f	fun 3
a	main

d.

b, c, d	fun 1
e, f	fun 3
a	main

e.

c, d, e	fun 2
f	fun 3
b	fun 1
a	main

f.

b, c, d	fun 1
e	fun 2
f	fun 3
a	main

5.

a.

a, x, w	sub 3
b, z	sub 2
y	sub 1

b.

a, x, w	sub 3
y, z	sub 1

c.

a, y, z	sub 1
x, w	sub 3
b	sub 2

d.

a, y, z	sub 1
x, w	sub 3

e.

a, b, z	sub 2
x, y	sub 3
y	sub 1

f.

a, y, z	sub 1
b	sub 2
x, w	sub 3