

# 3장. 변수 이해

# 3.1 변수(1/14)

## (1) 변수의 의미

- 프로그램 변수는 수학에서와 유사하게 변하는 값을 갖는다.
- 프로그램의 변수는 처리할 데이터를 저장하는 저장소의 역할을 한다.
- C/C++, Java 같은 언어들은 데이터 처리에서 오류 발생을 피하기위해서 사용하는 변수에 대입되는 데이터를 고려해서 변수의 자료형을 규정한다.
  - 변수의 자료형과 동일한 데이터를 처리하도록 강제화
- 스크래치에서 변수에 저장되는 데이터 형에 대한 명시적인 제한이 없음
  - 저장되어야 할 값이 정수인지 실수인지 또는 문자열인지에 대한 구분 없음
  - 데이터를 보다 추상화시켜서 사용자들이 직관적으로 편하게 프로그램을 작성할 수 있도록 하기 위함이다.

## 3.1 변 수(2/14)

- (2) 기본 변수
- 시스템에서 기본적으로 제공하는 변수
- 기본 변수들은 모두 타원과 유사한 블록

〈표 7.1〉 스크래치의 기본 변수 블록

블록 영역	변수 블록	기능
동작	x 좌표	스프라이트의 x 좌표 값을 갖는 변수
	y 좌표	스프라이트의 y 좌표 값을 갖는 변수
	방향	스프라이트가 바라보는 방향에 대한 각도 값을 갖는 변수
모양	모양 번호 ▾	현재 화면에 나타난 스프라이트의 모양 번호 값을 갖는 변수
	배경 번호 ▾	현재 화면에 나타난 배경의 번호 값을 갖는 변수
	크기	스프라이트의 크기 값을 갖는 변수

# 3.1 변 수(3/14)

## 기본 변수(계속)

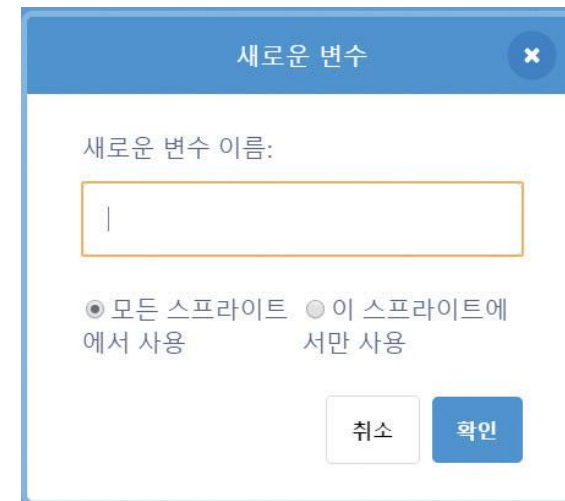
- 기본 변수들은 사용자들이 임의로 생성하거나 제거할 수 없음
- 프로젝트가 실행되면 기본 변수는 상황에 따라 그 값이 변함
- 기본 변수는 값을 변경할 때 사용할 블록이 제공되는데, 변경 블록이 없는 변수들은 그 값을 읽기만 할 수 있음

소리		음량 값을 갖는 변수
감지		묻고 기다리기 블록에 대한 응답을 저장하는 변수
		마우스의 x 좌표 값을 갖는 변수
		마우스의 y 좌표 값을 갖는 변수
		소리의 음량 크기 값을 갖는 변수
		실행 타이머 값을 갖는 변수. 1/1000 초 단위로 측정한다.
		무대에 대한 정보(배경 번호, 배경 이름, 음량, 나의 변수)를 얻거나 스프라이트에 대한 정보(x 좌표, y 좌표, 방향, 모양 번호, 모양 이름, 크기, 음량)를 얻는다.
		현재의 날짜와 시간 정보(년, 월, 일, 요일, 시, 분, 초)를 얻는다.
		2000년 1월 1일부터 지금까지 지난 날짜 수를 얻는다.
		스크래치 사이트에서의 로그인 사용자 이름을 얻는다.
변수		스프라이트에 기본적으로 제공되는 일반 변수

## 3.1 변수(4/14)

### (3) 변수의 생성

- 기본 변수 블록 이외에 사용자가 필요에 의하여 변수를 생성해서 사용할 수 있음
- 일반 변수는 나의 변수 블록 하나만을 기본적으로 제공
- 일반 변수가 더 필요한 경우에는 변수 만들기를 이용하여 변수를 생성할 수 있음
- 변수 영역에 변수 만들기 버튼이 제공



새로운 변수

새로운 변수 이름:

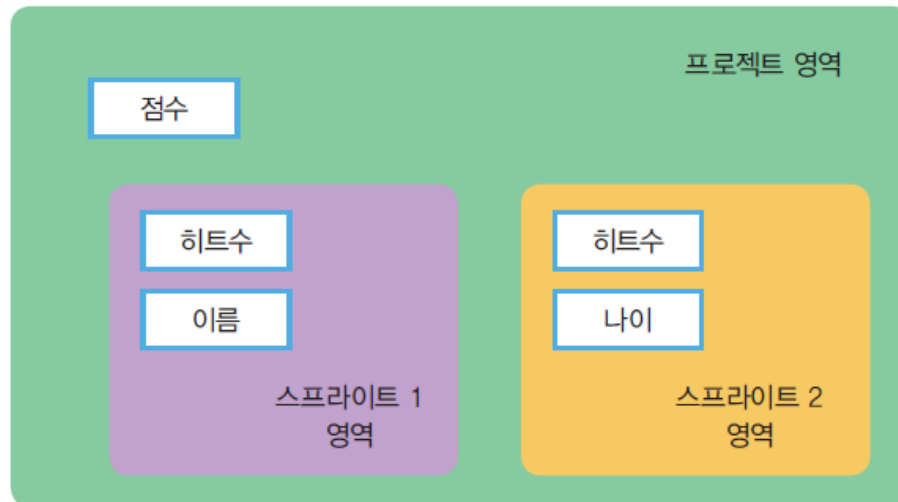
|

☒ 모든 스프라이트에서 사용 ☐ 이 스프라이트에서만 사용

취소 확인

## 3.1 변수(5/14)

- 지역 변수와 전역 변수
- 새로운 변수를 만들 때 이름을 입력하고, 속성으로 '모든 스프라이트에서 사용'과 '이 스프라이트에서만 사용' 중에서 하나를 선택해야 함
- '모든 스프라이트에서 사용'을 선택하면 현재 프로젝트 내에 포함되어 있는 모든 스프라이트들과 무대에서 사용 가능
  - 스프라이트의 지역 변수(local variable)
- '이 스프라이트에서만 사용'을 선택하면 현재 선택되어 있는 스프라이트에서만 사용 가능
  - 스프라이트의 전역 변수(global variable)



점수 : 전역변수  
히트수 : 지역변수  
이름 : 지역변수  
나이 : 지역변수

히트수는 이름이 같아도 서로 다른 변수로 구분함

## 3.1 변수(6/14)

### (5) 변수 처리 블록

- 새로운 변수를 생성하면 그 변수를 처리할 때 사용할 블록들이 생성된다.
- 새로운 변수는 체크되어 있는데, 변수를 체크하면 변수 내용이 실행 화면 창의 왼쪽 위 부분에 나타난다.
  - 프로그램이 실행되면서 변수 값이 변해가는 것을 살펴볼 수 있기 때문에 주로 디버깅(debugging) 목적으로 사용되지만, 게임의 경우 점수나 수명 등을 화면에 보여주기 위해서 사용되기도 함



그림 7.3 새로운 변수 생성과 실행 화면 출력

# 3.1 변 수(7/14)

## (5) 변수 처리 블록

〈표 7.2〉 변수 처리 블록

블록	기능
	변수의 값을 입력한 수로 정한다.
	변수의 값을 입력한 수만큼 증가시킨다.
	변수의 값을 실행 창에 나타낸다.
	변수의 값을 실행 창에서 숨긴다.

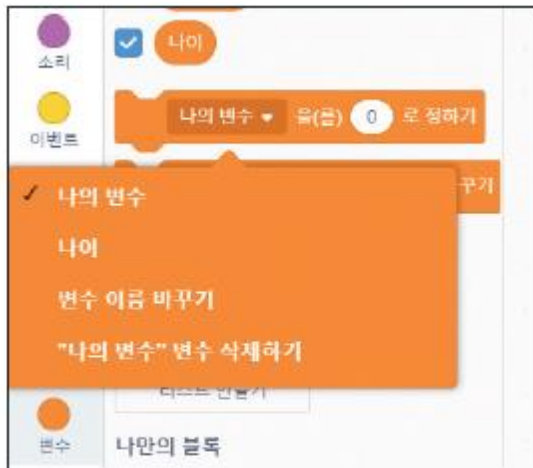


그림 7.4 변수 이름 변경

- 기본적으로 '나의 변수'가 선택되어 있음
- 블록 안에서 이름을 선택하여 다른 변수 처리 블록으로 지정함



## 3.1 변수(8/14)

### 변수를 이용한 반복의 예

- $1+2+3+ \dots + 10$



```
for (나의 변수 = 1; 나의 변수 <= 10; 나의 변수++)  
    합계 = 합계 + 나의 변수;
```

시작하기 버튼을 클릭했을 때  
합계 변수를 0으로 정한다.  
나의 변수를 1로 정한다.  
나의 변수 > 10이 될 때까지(즉 나의 변수가 11이 될 때  
까지) 다음 절차를 반복한다.  
 합계 변수에 나의 변수 값을 더한다.  
 나의 변수를 1만큼 증가시킨다.  
합계 변수를 2초 동안 말한다.

- 만약 1, 3, 5, 7, 9, ...와 같이 증가하는 경우 : 초기 값은 1이며 증가의 폭이 2가 되므로 나의 변수를 2만큼 증가시키면 될 것이다.
- 또한 12, 9, 6, 3, 0, -3, ...과 같은 경우에는 초기 값이 12이고 증가의 폭이 -3이 된다는 것을 알 수 있다.

## 3.1 변 수(9/14)

**[예제 3.1]** 두 정수를 읽어 들여서 각각  $a$ ,  $b$  변수에 저장한 후 두 수 사이의 모든 정수 값을 합하여 말하는 스크립트를 작성하라.

- 준비 단계
  - $n$ ,  $sum$  변수를 만든다.
- 단계 1 : 묻고 기다리기 블록을 두 번 이용하여 읽어 들인 값을 변수  $a$ ,  $b$ 에 저장한다.
- 단계 2 :  $a > b$  인 경우에는  $a$ ,  $b$ 의 값을 서로 맞교환한다.
- 단계 3 : 두 변수  $n$ 과  $sum$ 의 값을  $a$ , 0으로 각각 정한다.
- 단계 4 : 반복 구조를 이용하여  $a$ 에서  $b$ 까지의 모든 정수를  $sum$  변수에 누적시킨다.
- 단계 5 :  $sum$  값을 말한다.
- 단계 2에서 두 값을 서로 맞교환하는 방법은 다음과 같다.
  - 1)  $a$  값을 먼저  $n$ 에 저장한다.
  - 2)  $b$  값을  $a$ 에 저장한다.
  - 3)  $n$  값을  $b$ 에 저장한다.

# 3.1 변 수(10/14)

## [예제] 3.1 스크립트



시작하기 버튼을 클릭했을 때  
묻고 기다리기를 이용하여 첫 번째 정수를 읽어 들  
여 a에 저장한다.  
묻고 기다리기를 이용하여 두 번째 정수를 읽어 들  
여 b에 저장한다.

만약  $a > b$ 이면  
a와 b를 맞교환하여 항상 a가 b보다 작거나 같은  
수가 되도록 한다.

sum은 0으로, n은 a로 초기화한다.

n > b가 될 때까지 다음 절차를 반복한다.  
sum에 n을 더한다.  
n을 1 증가시킨다.

"sum = sum값"을 2초 동안 말한다.

## 3.1 변수(11/14)

### (6) 무대에서의 변수 생성

- 무대에서 변수 만들기 : 모든 스프라이트에서 사용, 전역 변수

새로운 변수

✕

새로운 변수 이름:

이 변수는 모든 스프라이트에서 사용할 수 있습니다.

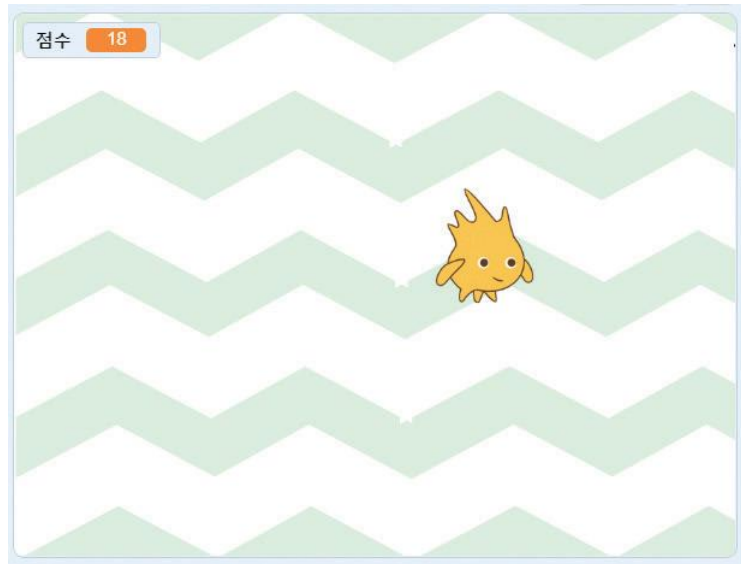
취소

확인

## 3.1 변 수(12/14)

### [예제 3.2] 단순한 유령잡기 게임

- 화면에 유령이 나타났다 사라지기를 반복한다.
- 유령이 화면에 나타났을 때 마우스로 클릭하면 점수가 1점 올라간다.
- 유령이 나타나는 횟수는 총 30회로 한다.
- 화면에 점수가 나타나도록 한다.



- 준비 단계
  - 유령 스프라이트를 스프라이트 고르기에서 Gobo를 선택한다.
  - 배경은 배경 고르기에서 Stripes를 선택한다.
  - 무대에서 점수 변수를 만든다.

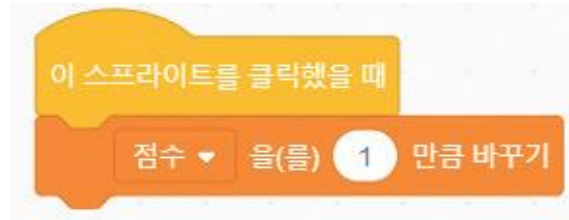
## 3.1 변 수(13/14)

### [예제3.2]스프라이트 동작

- 유령의 초기화와 나타났다 사라지기를 반복하는 동작
  - 단계 1 크기를 70%로 줄인다. 점수를 0으로 정한다.
  - 단계 2 다음 절차를 30번 반복한다.
  - 단계 3 숨기기를 한 다음 1초 동안 기다린다.
  - 단계 4 임의의 위치로 이동한 다음 보이기를 하고 0.8초 기다린다.
- 유령이 마우스로 클릭되었을 때 점수 1 증가시키기
  - 이 스프라이트를 클릭했을 때 블록을 이용하여 이벤트를 잡는다.
  - 점수 변수를 1 증가시킨다.

## 3.1 변 수(14/14)

### [예제3.2] 스크립트



- 이 스프라이트를 클릭했을 때에 대한 독립적인 스크립트를 작성해서 병렬 처리를 하도록 하였다.
- 유령 보이기를 한 상태에서 0.8초간 기다리는 중에 사용자로부터 마우스 클릭이 발생할 경우 병렬처리