



즐거운 상상 유쾌한







### • 성영훈(진주교육대학교, yhsung@cue.ac.kr)

센서보드 코딩으로 여러분의 상상을 신나게 표현해 볼 수 있게 구성하였습니다. 영감을 주시는 하나님께 감사드리고 늘 좋은 아이디어를 전해주는 사랑하는 아내와 현, 우, 은에게 감사의 마음을 전하고 아울러 이 책이 나오기까지 애 써주신 한국과학 조원득 대표님에게 깊은 감사의 말씀을 전합니다.

#### • **유승한**(상리초 교사. tito22@hanmail.net)

터치블럭과 센서 블럭으로 여러분들이 상상한 것을 SW로 표현해 보세요. 즐겁고 흥미로운 경험이 여러분들 미래를 밝혀줄 수 있어요.

- 생각쑥쑥 소프트웨어, 뚝딱뚝딱 코딩공작소(한국과학창의재단)교재 집필, SW선도학교 주임 교사

#### • 김미승(호계초 교사, somathak@gmail.com)

SW와 함께하는 즐거운 세상! 여러분이 그 중심이 될 수 있어요. 세상을 바꾸는 SW를 만들어 봐요.

- 삼성 주니어 소프트웨어 아카데미 운영 및 강사, SW선도 및 전문교원, 경남로봇SW교육 연구 회원

### • **손창익**(경상남도과학교육원 교사, netsci@empas.com)

피지컬 컴퓨팅, 생활속의 문제를 해결하고 세상을 살기좋게 바꾸어 봐요.

- 2014 올해의 과학교사상, 로봇창의교실 운영, 삼성주니어 소프트웨어 아카데미 운영, 경남로봇SW교육 연구회운영, STEAM 교사연구회 책임연구원

#### 정동규(마산중앙초 교사, non3801@hanmail)

생각을 키우고 미래와 만나는 SW세상, 여러분도 진정한 메이커(maker)가 될 수 있습니다.

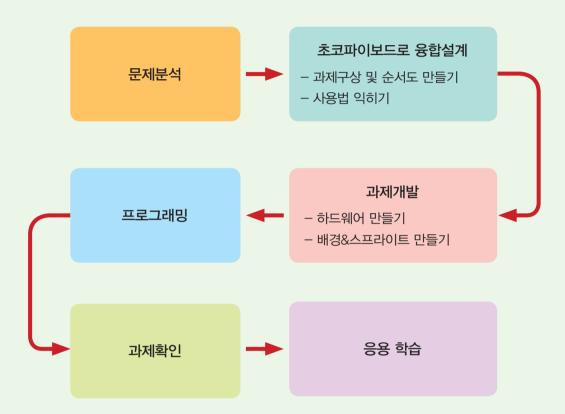
- 2012 올해의 과학교사상, SW선도 및 전문교원, 로봇활용 SW선도교원, 삼성주니어소프트웨어 아카데미 운영 및 강사, 로봇창의교실 운영, SW교구활용교사연구회 운영(현), 경남로봇SW교육연구회 운영(현)

#### • 전광욱(한국과학, woogi7942@naver.com)

시각장애인을 대상으로 프로그래밍 교육을 몇년간 해오면서 코딩의 중요성에 대해 알게 되었습니다. 단기간 교육이 아닌 장기간 교육을 통해 사고의 폭이 자란다는 것을 느꼈습니다. 코딩을 처음 시작하는 학생들에게 사고의 폭이 자라는 즐거운 상상 유쾌한 코딩이 되는 책이었으면 합니다.



이 책은 챕터별로 문제분석, 초코파이보드 융합설계, 과제개발, 프로그래밍, 과제확인, 응용학습 총6단계로 구성된학습체계를 가지고 있습니다. 각 단계별 세부적인 내용은 다음과 같습니다.





### 1단계. 문제분석

챕터의 주제는 먼저 실생활이나 주위에서 일어나는 문제들에 대해서 아이디어를 얻었으며 이것을 해결하기 위해 필요한 이론적인 지식이나 내용을 설명하였습니다.

#### 2단계. 초코파이보드로 융합설계

문제분석 단계에서 수집한 자료와 분석 내용을 기초로 만들고자 하는 과제를 구상하며 세부적인 절차에 대한 알고리즘을 표현하기 위해 순서도 작성하고 기초적인 센서보드의 사용법과 코딩하는 기초에 대해서 학습하게 됩니다.

### 3단계. 과제개발

순서도로 표현된 알고리즘에서 제시하고 있는 단계에 따라 만들고자 하는 과제에 필요한 센서보드, 각종 자료들을 탐색하고 실제 스크래치에서 구현에 필요한 스프라이트, 배경 등에 관련된 디자인을 하게 됩니다.

#### 4단계, 프로그래밍

과제개발 단계에서 구상하고 설계한 내용대로 스크래치에서 블록들을 활용하여 프로그래밍하는 단계입니다.

### 5단계. 과제확인

여러분이 직접 만들고 있는 과제(프로젝트, 프로그램)이 정상적으로 동작하는지 실행하는 과정에서 오류는 없는지 알아보고 더 나은 개선점을 찾아 발전시켜가는 단계입니다.

#### 6단계, 응용학습

여러분이 과제확인 단계에서 발견한 문제점을 기초로 개선하고자 하는 방법을 찾아보고 만들어진 내용을 바탕으로 추가적인 기능을 구현하거나 새로운 아이디어를 덧붙여 더 나은 과제(프로젝트, 프로그램)로 업그레이드 하는 단계 입니다.

CHAF	PTER	
	4	
U		

# 초코파이보드와 스크래치

1.1 초코파이보드를 위한 스크래치X 설치하기 2
1.2 Scratch4 Chocopi! 살펴보기 5
1.3 무대 배경 생성 8
1.4 스프라이트 생성 9
1.5 언어 바꾸기 10
1.6 프로젝트 저장하기 / 불러오가 11
1.7 초코파이보드와 연결하기 13
1.8 과제 확인 17
1.9 배운 블록에 대해 이야기하기 18

# CHAPTER 02

## 스크래치 블록 이해하기

2.1 스크래치 블록 사용하기 2.1	20
2.2 프로그램 동작 확인하기	38
9.2 배우 브루에 대해 이아기하기.	20

# CHAPTER 03

### LED색상환

3.1 문제 분석	-42
3.2 초코파이보드 융합 설계	-43
3.3 과제 개발	-46
3.4 프로그래밍	-48
3.5 과제 확인	-50
3.6 응용 학습	-50



СНА	PTER
0	4

# 터치 음악

4.1 문제 분석
4.2 초코파이보드 융합 설계 5.4
4.3 과제 개발
4.4 프로그래밍 60
4.5 과제 확인 62
4.6 응용 학습 62

# CHAPTER 05

## 안전 유도등 만들기

5.1 문제 분석 60
5.2 초코파이보드 융합 설계 6
5.3 과제 개발
5.4 프로그래밍
5.5 과제 확인
5.6 응용 학습

# CHAPTER 06

## 마법 지팡이 만들기

6.1 문제 분석	82
6.2 초코파이보드 융합 설계	83
6.3 과제 개발····	-86
6.4 프로그래밍	89
6.5 과제 확인	95
6.6 응용 한습	.96

CHAPTER 07

### 빛으로 신호 보내기

7.1 문제 분석 100
<b>7.2</b> 초코파이보드 융합 설계·······10··
7.3 과제 개발 106
7.4 프로그래밍
7.5 과제 확인 113
7.6 응용 학습

CHAPTER 08

## 크리스마스 트리 만들기

8.1 문제 분석 116
8.2 초코파이보드 융합 설계 117
8.3 과제 개발 124
8.4 프로그래밍
8.5 과제 확인 133
8.6 응용 한습

CHAPTER 09

## 스마트 LED조명 만들기

9.1 문제 분석	136
9.2 초코파이보드 융합 설계	·137
9.3 과제 개발	··141
9.4 프로그래밍	146
9.5 과제 확인	··153
9.6 응용 학습	154



СН	APTER
4	
	V

# 사운드 LED

<b>10.1</b> 문제 분석
10.2 초코파이보드 융합 설계 157
<b>10.3</b> 과제 개발······162
10.4 프로그래밍
<b>10.5</b> 과제 확인······171
10.6 응용 학습 173

# CHAPTER 1

# <u>5 IN 1 악기</u>

11.1 문제 분석
<b>11.2</b> 초코파이보드 융합 설계······177
<b>11.3</b> 과제 개발·····182
11.4 프로그래밍
<b>11.5</b> 과제 확인 197
11.6 응용 학습 198

# CHAPTER 12

# CHAPTER LED 연주

<b>12.1</b> 문제 분석······200
<b>12.2</b> 초코파이보드 융합 설계······ 201
<b>12.3</b> 과제 개발······206
12.4 프로그래밍 213
<b>12.5</b> 과제 확인 218
12.6 응용 하습

# 차 례

부록	
01	

# 누가 과일을 가져갈까?

부록 1.1 문제 분석 220
부록 1.2 초코파이보드 융합 설계         220
부록 1.3 과제 개발 221
<b>부록 1.4</b> 프로그래밍 224
부록 1.5 과제 확인 225
부록 1.6 응용 학습 225

# <sup>片록</sup> **02**

# LED피아노 만들기

<b>부록 2.1</b> 문제 분석·····	-228
부록 2.2 초코파이보드 융합 설계	-228
부록 2.3 과제 개발	-230
부록 2.4 프로그래밍	234
부록 2.5 과제 확인	-236
부록 2.6 응용 학습	-236





초코파이보드와 스크래치









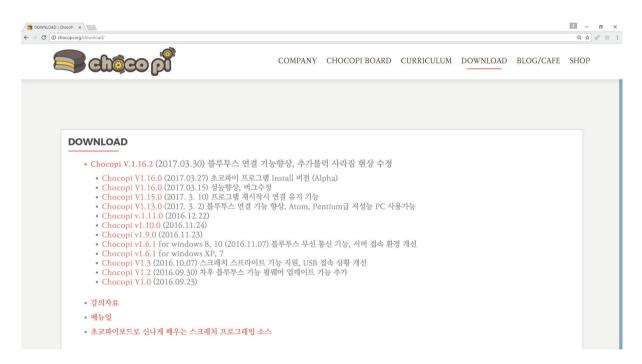


스크래치 프로그램은 누구나 손쉽게 이야기, 게임, 애니메이션을 직접 만들 수 있고 다른 사람들과 온라 인에서 공유할 수 있습니다. 또한 센서 보드와 결합하여 PC 화면에서뿐만 아니라 실제 생활에 필요한 문 제들을 해결할 수 있는 기능을 구현해보고 작품들을 만들 수 있습니다.

### 1.1 초코파이보드를 위한 스크래치X 설치하기

초코파이보드를 사용하기 위해서는 필요한 프로그램을 다운로드 받아야 합니다.

- ① 웹 주소 창에 'http://chocopi.org/download'를 입력하여 초코파이보드 홈페이지 다운로드 사이트 에 접속합니다.
- ② 'DOWNLOAD' 메뉴에서 최신 파일을 다운로드 받습니다.



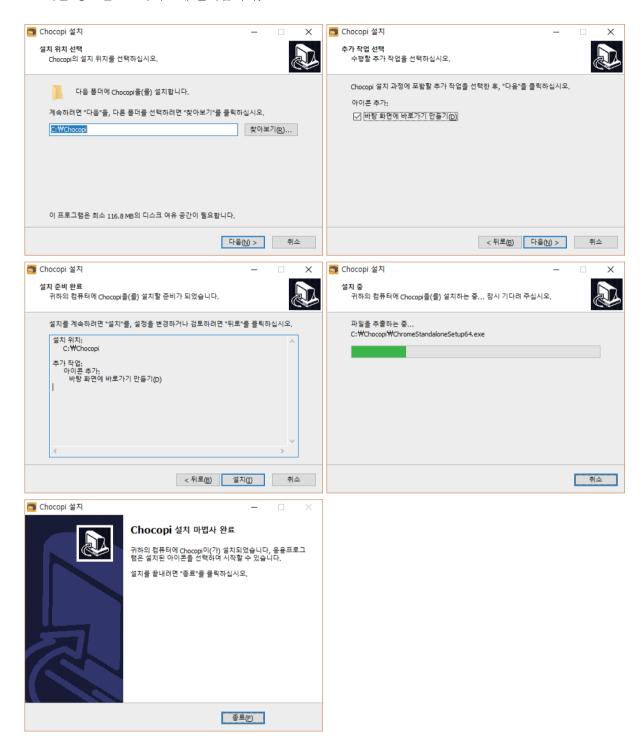








③ 다운로드 받은 'Chocopi\_setup\_V1.16.2.exe' 파일을 더블 클릭하여 설치를 진행합니다. 설치 경로는 기본 경로인 C드라이브에 설치합니다.



1장, 초코파이보드와 스크래치





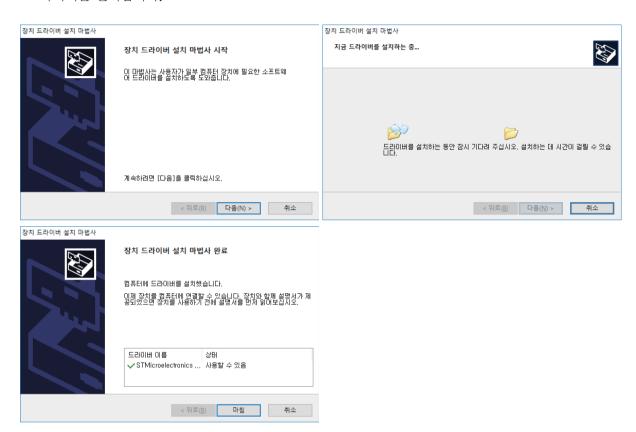






④ Chocopi설치 마법사가 설치 완료되면 '종료' 버튼을 클릭합니다.

프로그램 설치 종료 후 Windows 10버전은 초코파이보드에 대한 드라이버 설치가 필요 없으며 Windows 8 이하 버전은 그림과 같이 드라이버 설치마법사가 나타납니다. '다음' 버튼을 클릭하여 드라이버를 설치합니다.



장치 드라이버 설치 마법사가 완료되면 '마침' 버튼을 클릭합니다.

드라이버 설치 후 바탕화면에 설치된 'Chocopi' 아이콘 🚉 을 클릭하여 실행시킵니다.



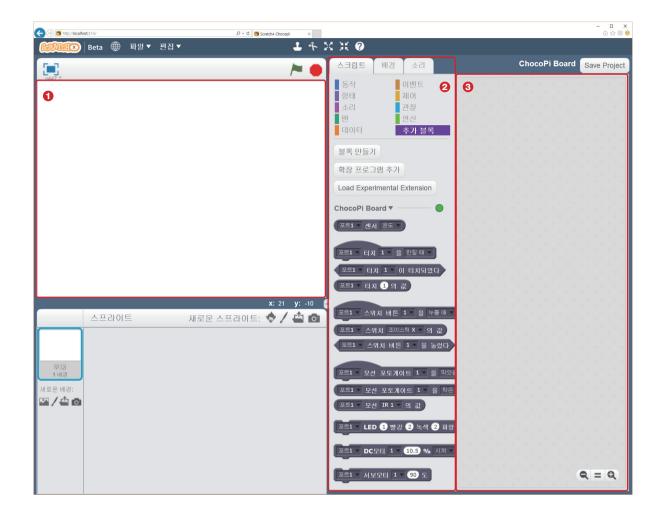








### 1.2 Scratch4 Chocopi! 살펴보기

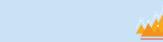


스크래치 프로그램은 전체적으로 다음과 같이 구성되어 있습니다.

- ① 배경: 스프라이트를 배치하여 다양하게 활용할 수 있는 배경입니다.
- ② 스크립트, 배경, 소리 탭: 선택된 스프라이트에 해당하는 블록들, 속성들을 선택할 수 있는 탭입니다.

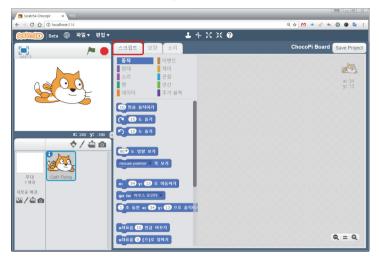








### • 스크립트 탭을 선택했을 경우



스크립트 탭을 클릭하면 동작, 형 태, 소리, 펜, 데이터, 이벤트, 제 어, 관찰, 연산, 추가 블록 팔레트 가 있습니다. \*스프라이트를 생성 하고 각각의 팔레트를 클릭해 보 세요. 각 팔레트는 명령어의 유형 에 따라 모여 있으며 팔레트 내에 는 다음과 같은 모양의 블록이 있 습니다.

### 각 팔레트마다 다양한 모양을 확인해 봅니다.

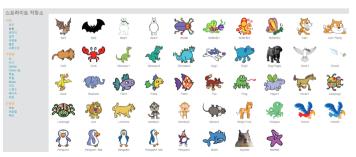
블록 모양	설 명
10 만큼 움직이기	명령 블록 : 필요한 입력 값을 🥥 에 입력하면 입력 값에 따른 명령이 실행됩니다.
스페이스 ▼ 키를 눌렀을 때	이벤트 블록(~할 때): 특정 시점에 발생하는 신호 또는 명령을 받았을 때 실행되는 블록으로서 '키를 눌렸을 때', '깃발을 클릭했을 때' 등의 프로그램 시작점으로 사용됩니다.
1 부터 10 사이의 난수	데이터 블록: 변수 블록으로서 데이터를 저장해 놓는 블록입니다.
	조건문 블록(TRUE,FALSE): 제어문에 사용하는 블록으로서 조건이 일치할 때 true, 조건이 일치하지 않을 때 false 값을 가지고 있는 변수 블록의 한 종류입니다.
x型표	화면 표시 블록

[블록 모양에 따른 명령 구분]

### ₩ 스프라이트 생성



'저장소에서 스프라이트 선택' 아이콘을 클릭합니다.



스프라이트 저장소에서 'Cat1 Flying' 을 선택합니다.

스프라이트는 단순한 그림이 아닌 코딩에 필요한 속성과 명령을 가지고 있는 객체입니다.









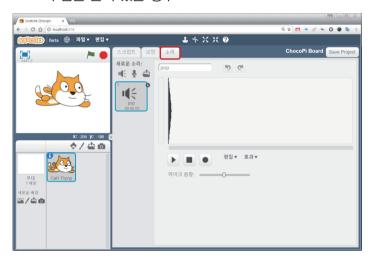


### • 모양 탭을 클릭했을 경우



스프라이트와 유사한 그림이 모여 있 는 저장 장소로서 하나의 스프라이 트에 여러 개의 모양을 만들 수 있으 며 그림 편집도 가능한 작업 영역입니 다.

### • 소리 탭을 클릭 했을 경우



음성을 마이크 또는 외부 파일로부터 입력 받아 저장하고 편집하는 작업 영 역입니다

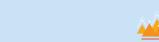
#### ③ 스크립트 작성



블록을 선택하여 다음과 같이 간단하 게 드래그하여 놓으면 됩니다.









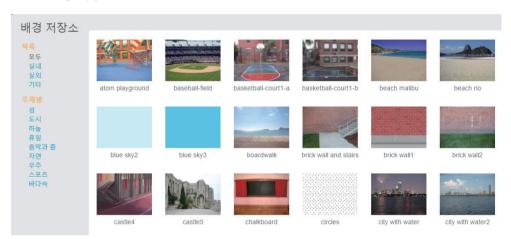
### 1.3 무대 배경 생성

① 새로운 배경에서 🌇을 클릭합니다.

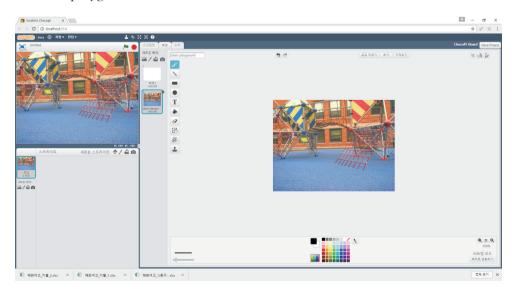


■을 클릭하면 스크래치에서 제공하는 배경 저장소에서 무대 배경을 가져올수있습니다.

② 'atom playground' 배경을 선택하고 확인 버튼을 누릅니다.



③ 'atom playground' 배경을 선택하면 아래와 같이 새로운 배경이 적용됩니다.











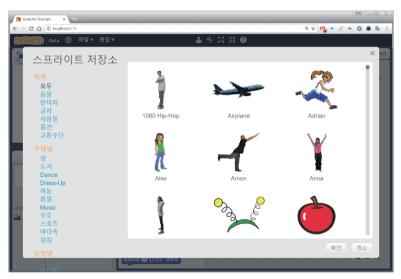
### 1.4 스프라이트 생성

① 새로운 스프라이트에서 🌏 을 클릭합니다.



● 을 클릭하면 스프라이트 장소에서 제공하는 스프라이트를 가져올 수 있습니다.

② 스프라이트를 선택하고 확인 버튼을 누릅니다.





'1080 Hip-Hop'을 선택하면 그림과 같이 무대 위에 선택한 스프라이트를 가져오게 됩니다.













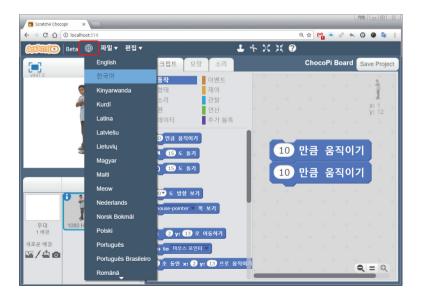
선택한 스프라이트를 10만큼 두 번 움직이기 위해서 그림과 같이 간단하게 블록을 가져다 놓습니다.

④ 블록을 마우스로 왼쪽 클릭하면 실제 스프라이트가 프로그래밍한 내용대로 움직이는 것을 볼 수 있습니다.



### 1.5 언어 바꾸기

① 언어는 그림과 같이 지구본을 클릭하여 설정된 언어를 바꿀 수 있습니다.









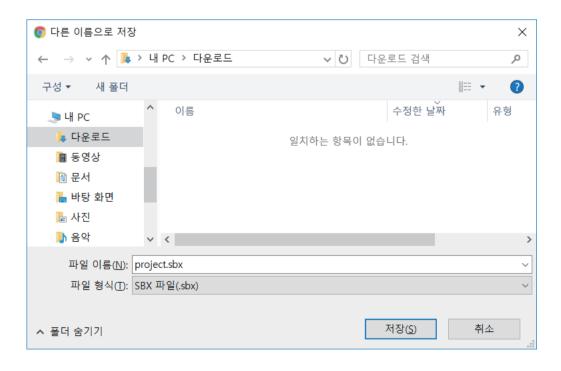


### 1.6 프로젝트 저장하기 / 불러오기

① 프로젝트 저장은 파일 메뉴에서 '프로젝트 저장하기'를 클릭하면 됩니다.

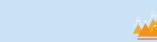


② 파일이름(N) 편집 창에 파일명을 입력하고 저장 버튼을 클릭합니다. 파일은 sbx 확장자로 저장됩니다.









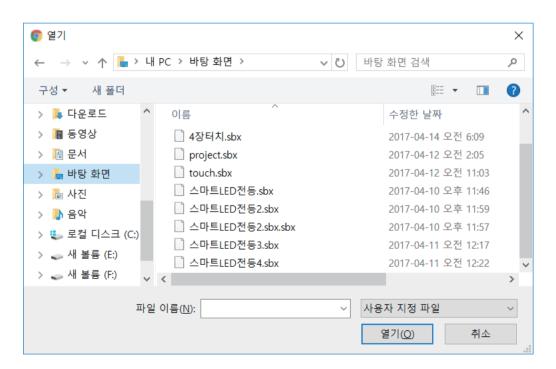




③ 프로젝트로 저장된 파일을 불러오기는 파일 메뉴에서 '프로젝트 불러오기'를 클릭합니다.



④ 불러오기 할 파일을 선택하고 '열기' 버튼을 클릭합니다.











### 1.7 초코파이보드와 연결하기

키보드의 스페이스 바 키를 누르면 초코파이보드에 연결된 LED에 불이 들어오게 해 봅시다.

① 하드웨어를 컴퓨터 – 초코파이보드 – LED블록 – LED모듈 순으로 연결합니다.



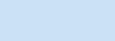
연결 시 주의 사항은 LED블록에서 나오는 3선은 LED 모듈의 IN에 연결해야 합니다. LED의 블록의 화살 표 방향에 유의하여 연결하면 됩니다.

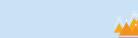


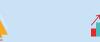
연결 시 주의 사항은 LED블록에서 나오는 3선은 LED 모 듈의 IN에 연결해야 합니다. LED의 블록의 화살표 방향 에 유의하여 연결하면 됩니다.



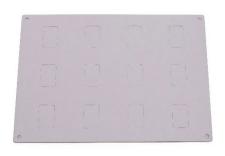






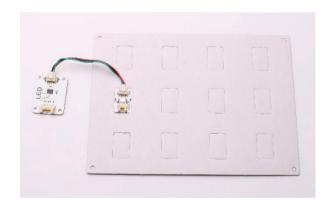


② 연결된 하드웨어를 LED 홀더와 결합합니다.





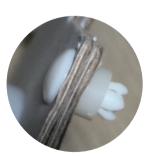




하판에 네모 종이를 제거하고 LED를 그림과 같이 삽입시킵니다.







둥근 구멍이 있는 상판을 위에 놓고 리벳과 워셔를 모서리에 체결합니다.

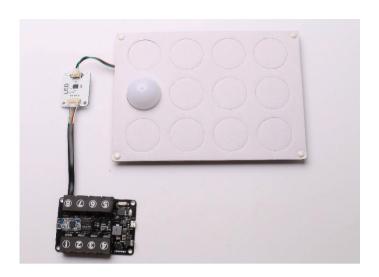












초코파이보드에 LED홀더를 연결하여 하드웨어를 완성합니다.

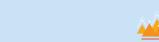
- ③ 바탕화면에 설치된 초코파이 바로가기 실행파일 을 실행합니다.
- ChocoPi Scratch Server V1,16,2 창이 닫혔을 때 초코파이 바로가기 실행 파일을 실행합니다.



프로그램 실행 후 그림과 같이 연결된 상태와 연결된 블록이 화면에 표시 됩니다.









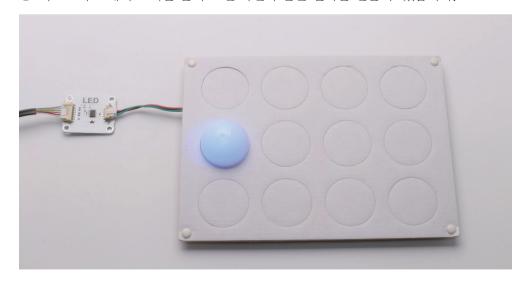
④ 스프라이트 불러오기에서 '1080 hip-pop' 스프라이트를 불러오고 '스크립트' 탭을 클릭 후 이벤트 팔 레트를 선택합니다. 블록에서 '스페이스 키를 눌렀을 때' 블록을 드래그하여 스크립트 영역에 놓습니다.



⑤ 추가블록 팔레트에서 LED 명령 블록을 '스페이스 키를 눌렀을 때' 블록 아래에 가져다 놓고 값을 입력합니다.



⑥ 키보드의 스페이스 키를 눌러 보면 사진과 같은 결과를 얻을 수 있습니다.







W





### 1.8 과제 확인

코딩을 한 프로그램이 동작하는지 확인합니다.

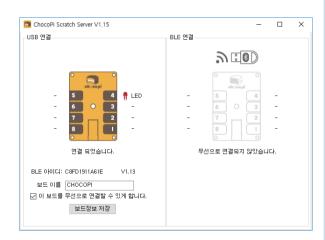
▶ 초코파이보드를 연결했을 때 그림과 같이 실행이 되나요?

예(), 아니오()



▶ LED블록을 연결했을 때 그림과 같이 연결된 LED가 화면에 표시되나요?

예 ( ), 아니오 ( )



- ▶ 블록의 연결 순서는 어떻게 되는지 설명하고 몇 번 반복하여 연결해 보세요.
- ▶ 스페이스 키를 눌렸을 때 LED가 켜지나요?

예( ), 아니오( )

초코파이보드와 터치블록이 정상적으로 연결되어 있는지 확인합니다. 만약 연결되어 있지 않다면 리셋 버튼 또는 프로그램 재실행을 통해 연결을 확인합니다.











# 1.9 배운 블록에 대해 이야기하기

스크래치 블록에 대해 설명하세요.

▶ 스크래치 블록 중 둥근 편집 창에 필요한 입력 값을 입력하고 실행하면 동작이 실행되는 블록은 어떤 블록인가요?
▶ 크기를 비교하여 참, 거짓을 저장하는 블록은 어떤 블록인가요?
▶ 프로그램의 시작점에 주로 사용하는 블록으로 이벤트에 자주 사용하는 블록은 어떤 블록인가요?
▶ 데이터를 저장하는 블록은 어떤 블록인가요?
▶ 초코파이보드에 LED를 켜고 싶을 때 연결하는 순서는 어떻게 되나요?

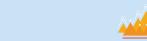




# 스크래치 블록 이해하기











스크래치에서 사용되는 블록들을 살펴보도록 합 시다. 스크래치는 앞서 얘기한 것처럼 스프라이트 를 무대에 배치하고 스프라이트와 관련된 이벤트 를 통해 다양한 동작을 하도록 설계되어 있습니다. 스프라이트를 제어할 수 있는 스크립트 탭에 있는

과 관련된 내용을 살펴봅니다.

### 2.1 스크래치 블록 사용하기

### ① 프로그램의 시작점 블록 가져오기



깃발을 클릭했을 때 동작은 어떻게 되는지 설명해보세요.

프로그램의 시작점이 되는 블록인 '깃발 클릭했을 때' 블록을 그림과 같이 드래그 합니다.

깃발을 클릭해 봅니다.

### ② '1080 HIP-HOP' 스프라이트를 생성합니다.











### ③ 동작 블록 추가하기

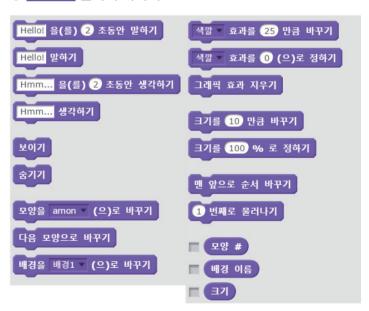


동작 블록은 스프라이트의 동작과 관련된 블록으로 움직임, 위치, (x, y) 좌표. 회전방식 등을 포함하고 있으며 그림과 같이 스크립트 탭에 위치하고 있습니다.

가운데에 있는 동작 블록을 블록 에디 터에 끌어다가 놓으면 됩니다. 🏲 을 클릭하면 10만큼 스프라이트가 움직 이게 됩니다. 또는 10만큼 움직이기를 직 접 클릭해도 실행이 됩니다.

마찬가지로 다른 동작 블록들도 블록 에디터에 끌어다 놓고 실험해 보시기 바랍니다.

### ④ 형태 블록 추가하기



[스크립트 탭에 있는 형태 블록의 다양한 기능들]

형태 블록은 스프라이트의 형태 를 나타내는 블록으로 말하기. 생각 하기, 스프라이트 보이기, 숨기기, 스 프라이트 모양을 바꾸기. 색깔과 크 기 및 배치순서 등을 바꿀 수 있는 기 능을 가지고 있습니다.













또한 형태 블록은 모양 탭과 관련이 있습니다. 그림과 같이 모양에 있는 스프라이트들을 제어할 수 있습니다.



예를 들어 10만큼 이 동하면서 모양을 계속 바꾸게 하려면 그림과 같이 간단하게 블록 프로그래밍을 해 볼 수 있습니다.





ièl







블록에디터에 있는 블록을 더블 클릭하게 되면 그림과 같이 스프라이트 모양 탭에 있는 모양 순서대로 계속해서 바뀌게 됩니다.

[스크립트 모양 탭을 선택한 화면]



[순서대로 계속해서 바뀌게 되는 스크립트 모양]

### ⑤ 소리 블록 추가하기

소리블록은 사운드나 효과음 등을 재생하거나 컴퓨터에 내장되어 있는 미디악기 사운드를 연주하거나 소리의 크기, 소리의 빠르기를 정할 수 있는 기능들이 포함되어 있습니다.











[다양한 소리 블록들]

그러면 이제 클릭할 때마다 소리를 내게 해 봅시다.



물론 이것은 다음의 소리 탭에 있는 내용과 관련이 있습니다.











[소리 탭에 있는 dance celebrate.wav]

다른 소리 블록에 있는 블록들을 이용하여 재미있는 프로그램들을 만들어 보세요.

#### ⑥ 펜 블록 추가하기

펜 블록은 블록의 이름에서 알 수 있듯이 무대배경에서 스프라이트를 활용하여 그림을 그릴 수 있는 블록 입니다. 지우기, 도장 찍기(그린 그림 복사), 펜 내리고 올리는 기능, 펜의 색깔과 명암 및 굵기 등을 변경 할 수 있습니다.

그림을 그리는 순서는 다양한 방법이 있지만 다음과 같은 방법을 사용합니다.

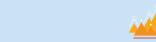
- 펜의 색깔과 굵기를 정하기
- 펨을 내리기
- 동작블록을 사용하여 일정한 방향으로 이동하기
- 펜을 올리기

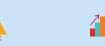
여기서 그린 그림을 지우고 싶다면 지우기 블록을 사용하면 됩니다.

자, 그럼 이제 춤추는 아이가 그림도 같이 그릴 수 있게 블록 프로그래밍을 해 봅시다.











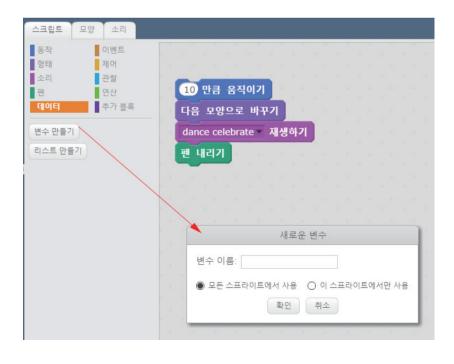
[굵기가 1인 크기의 파란색 선을 10만큼 움직이면서 그리는 화면]

마우스를 클릭하면 위와 같이 그림을 그리면서 이동하는 모습을 볼 수 있습니다.

### ⑦ 데이터 블록 추가하기

데이터 블록은 숫자, 문자를 담을 수 있는 변수를 만들거나 다양한 형태의 값들을 차례로 저장할 수 있는 리스트 만들기 기능을 가지고 있습니다.

우선 펜 블록에서 펜의 굵기가 점점 크게 될 수 있도록 하나의 변수를 만들어 봅시다.



'변수 만들기'를 클릭하면 변수의 이름을 설정하고 만들게 될 변수가 모든 스 프라이트에서 사용할 것인 지(전역변수), 아니면 선택 한 변수에서만 사용할 것 인지 결정하게 됩니다.













변수의 이름을 '펜굵기변수'라 이름짓고 '모든 스프라이트에서 사용'을 선택한 다음 확인 버 튼을 클릭합니다.



그림과 같이 '펜굵기변수'라는 새로운 변수 블록이 만들어지 는 것을 볼 수 있습니다.

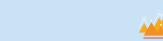
'펜굵기변수'의 값을 정하거나 해당 변수를 무대에서 보이거 나 숨길 수 있는 기능이 있습니다.

자 그러면 이제 '펜굵기변수' 블록을 사용하여 클릭할 때마다 펜의 크기가 1만큼 커지면서 이동하는 블록 을 만들어 봅시다.











우선 펜 블록에서 지우기 블록을 블록에디터에 어딘가에 놓습니다. 그 다음 데이터 블록에서 편录기변수▼을(를) 1 만큼 바꾸기를 가져다가 위의 그림과 같이 붙여 넣습니다.

먼저 '지우기' 블록을 클릭해서 그려진 그림을 지웁니다. 다음으로 아래의 블록들을 클릭 해 봅시다.



그러나 실행해도 펜의 굵기가 증가하지 않습니다.왜 그럴까요?

'펜굵기변수'의 값은 계속해서 증가하나 실제 펜의 굵기에 '펜굵기변수' 값을 넣어주지 않아서 그렇습니다.



그림과 같이 프로그램을 수정합니다.









펜 굵기를 정하는 블록을 끌어다가 블록에디터에 놓습니다. 다음으로 '펜굵기변수'를 펜 굵기를 정하는 펜 블록 값에 다음 그림과 같이 넣습니다.

\*



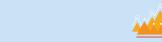
그러면 다음과 같은 블록의 모습으로 변경되었을 것입니다.

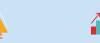
```
지우기
10 만큼 움직이기
다음 모양으로 바꾸기
dance celebrate ▼ 재생하기
펜 내리기
펜굵기변수 ▼ 을(를) 1 만큼 바꾸기
펜 굵기를 펜굵기변수 (으)로 정하기
```

이제 스프라이트를 왼쪽으로 옮기고 지우기를 한 번 실행한 다음, 블록을 실행해 봅시다.











펜의 굵기가 점점 커지는 것을 알 수 있습니다. 그러나 실행할 때마다 '지우기' 블록을 먼저 클릭하고 그 다음 블록들을 클릭해야 하는 불편함이 있습니다. 어떻게 하면 보다 간편하게 자동으로 할 수 있을까요?

#### ⑧ 이벤트 블록 추가하기

이벤트 블록은 사용자가 컴퓨터에게 의사소통하기 위해 입력하게 되는 키보드, 스프라이트의 내용, 배경 이 변경되었을 때, 음량의 변화, 특정한 메시지를 받거나 보낼 때의 경우 사용할 수 있는 기능을 포함하고 있습니다.

일반적으로 '깃발을 클릭했을 때' 다음과 같은 기능을 수행할 수 있도록 프로그램을 변경해 봅시다. 즉 프 로그램이 처음 실행될 때 필요한 초기화 값들을 정해주는 작업입니다.

- 그린 그림을 지운다.
- 스프라이트를 좌표(-180, 0)으로 이동한다.
- '펜굵기변수'의 값을 0으로 정한다.

블록 프로그래밍으로 나타내면 다음과 같습니다.





iĝi







이제 스페이스 키를 누를때마다 이전에 만들었던 블록들이 동작하게 해 봅시다.





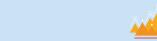
'스페이스 키를 눌렀을 때' 블록을 끌어다가 만들어 놓은 블록들 위에 가져다 놓습니다.

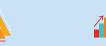
이제 깃발을 클릭해서 실행해 봅시다.

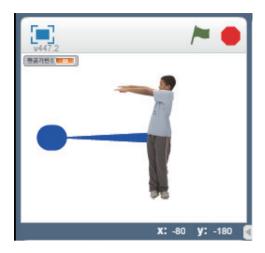
그림이 지워지고 스프라이트가 좌표(-180.0)으로 이동한 다음 화면에서 보이지는 않지만 '펜굵기변수'의 값도 0으로 변경되었 음을 알 수 있습니다.











'스페이스키'를 계속해서 누르게 되면 그림과 같은 화면 이 나타납니다.

처음에 생긴 동그란 점은 이전에 입력되어 있었던 펜굵 기 값으로 펜을 내렸기 때문에 그렇습니다.



자. 이제 다시 초기화 값에 펜의 굵기를 1로 정해서 변경 해 볼까요?



이제 한 번 깃발을 클릭하고 계속해서 스페이스키를 누 르면 다음과 같이 실행됩니다.

### ⑨ 제어 블록 추가하기

제어 블록은 다양한 블록들의 조건과 관련된 블록으로 기다리기, 반복하기, 만약 ~라면, ~까지 기다리 기. ~까지 반복하기. 복제하기 등과 같은 기능들을 가지고 있습니다.

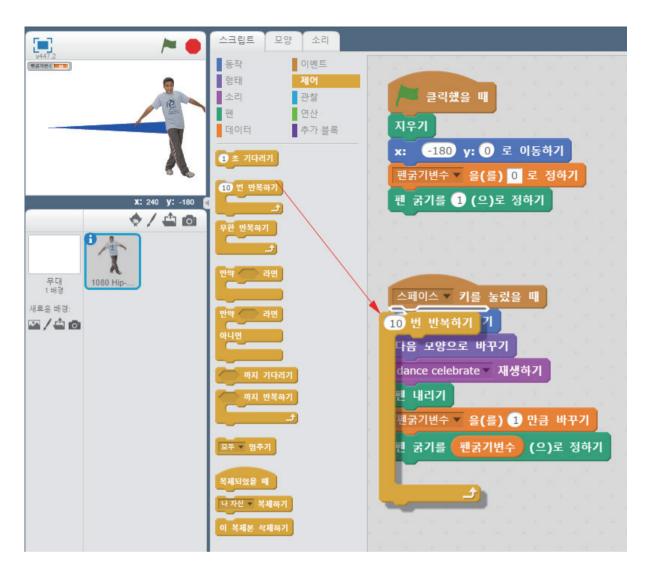
앞서 만들었던 블록을 스페이스 키를 계속 누르는 것이 아니라 10번만 반복하게 하려면 다음과 같이 반복 하기 블록을 추가하여 프로그래밍 할 수 있습니다.











\*



실행하면 스페이스 키를 누를 때마다 자동적으로 10번 씩 반복하여 실행하는 것을 볼 수 있습니다.









#### ⑩ 관찰 블록 추가하기

관찰 블록은 스프라이트가 다른 스프라이트 또는 무대, 색깔, 마우스 등과 같은 이벤트에 대해 감지할 수 있는 닻았을 때, 키를 눌렀을 때, 마우스의 좌표, 비디오, 사용자가 값을 입력했을 때와 같은 기능들을 가 지고 있습니다.

이제는 '깃발을 클릭했을 때' 입력받은 숫자만큼 반복해서 그림을 그리는 내용으로 블록 프로그래밍을 해 봅시다.





'깃발을 클릭했을 때' 펜을 올려서 더 이상 그림을 그 리지 않게 기존의 프로그램을 개선합니다. 그 다음 관 찰블록에서 '묻고 기다리기'블록을 가져다가 '몇 번 반 복할까요?'라고 바꿉니다. 실제 사용자가 입력한 값은 '대답'이라는 임시 변수에 저장됩니다. 그러면 깃발을 클릭해 봅시다.

What's your name? 묻고 기다리기 대답

몇 번 반복할까요? 라고 아이가 물어보고 있습니다. 입력한 값은 '대답'변수에 저장됩니다.

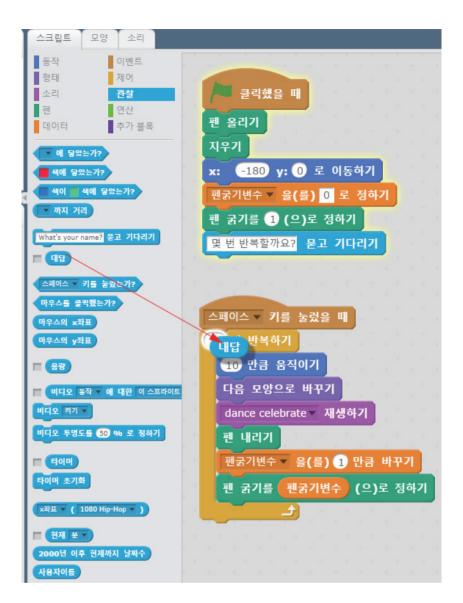






W





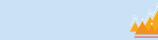
이제 대답 블록을 그림과 같이 반복하기 블록에 있 는 숫자 대신 교체합니다.



깃발을 클릭하고 숫자 5를 입력합니다.











그리고 스페이스 키를 누르면 그림과 같이 실행됩니다.

#### ⑪ 연산 블록 추가하기

연산 블록은 데이터 블록에 있는 변수 블록. '대답' 블록과 같은 다양한 임시 값을 저장할 수 있는 블록들 을 활용하여 계산하거나 비교하거나 결합하는 등 기능을 수행할 수 있는 블록들로 구성되어 있습니다.

연산 블록을 활용하여 펜의 굵기가 10이되면 다시 '펜굵기변수'의 값을 1로 변경시켜 줄 수 있도록 해 봅 시다. 필요한 블록은 제어 블록의 조건 블록과 연산 블록이 필요합니다.



제어블록에서 만약~라면 블록을 가져다 놓습니다.

연산 블록에서 《 = > 을 가져다 놓고 데이터블록에서 '펜굵기변수' 를 끌어다가 연산 블록의 왼쪽 에 가져다 놓고 다음과 같이 변경해 줍니다.



만든 연산 블록 세트를 다시 조건 블록인 '만약~라면'과 결합합니다.







W





```
펜굵기변수 = 10  라면
만약 《
    펜굵기변수 ▼ 을(를) 1 로 정하기
```

```
스페이스 ▼ 키를 눌렀을 때
대답 번 반복하기
 10 만큼 움직이기
 다음 모양으로 바꾸기
 dance celebrate ▼ 재생하기
 펜 내리기
 펜굵기변수▼ 을(를) 1 만큼 바꾸기
 펜굵기변수 ▼ 을(를) 1 로 정하기
```

다음으로 펜굵기변수의 값이 10이라면 '펜굵기 변수' 값을 1로 정해 주기 위해서 '데이터 블록'의 '펜굵기 변수'를 1로 정하기 블록을 가져다가 다 음과 같이 결합합니다.

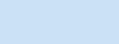
그리고 다음의 위치에 결합된 블록을 추가합니 다.



깃발을 클릭하여 프로그램을 실행한 후 20을 입 력하면 20번 반복하는데 10이 될때마다 펜의 굵 기가 1로 변경되어 그림과 같이 패턴이 생기는 것을 볼 수 있습니다.



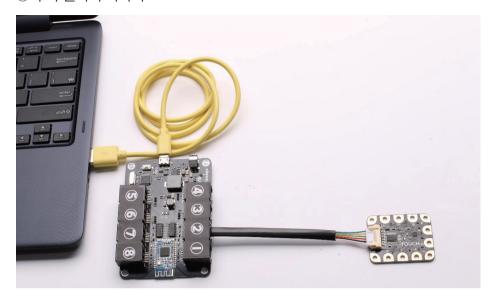








## ⑫ 추가 블록 추가하기



하드웨어를 위의 그림과 같이 연결하고 스페이스▼ 키를 눌렀을 때 이벤트 블록을 추가 블록의





로 블록을 변경해 줍니다.

# 2.2 프로그램 동작 확인하기

코딩한 프로그램이 동작되는지 확인합니다.

- ▶ 깃발을 클릭했을 때 프로그램이 실행이 되나요? 예(), 아니오()
- ▶ "깃발을 클릭했을 때"를 "터치 블록을 만졌을 때"로 변경했을 때 프로그램이 실행되나요? 예(), 아니오()

초코파이보드와 터치블록이 정상적으로 연결되어 있는지 확인합니다. 만약 연결되어 있지 않다면 리셋버 튼 또는 재연결을 통해 연결을 확인합니다.









## 2.3 배운 블록에 대해 이야기하기

스크래치에서 사용한 팔레트가 어떤 기능으로 묶여 있는지 설명해 봅시다.

- ▶ 동작 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?
- ▶ 형태 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?
- ▶ 소리 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?
- ▶ 펜 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?
- ▶ 데이터 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?
- ▶ 이벤트 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?
- ▶ 제어 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?
- ▶ 관찰 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?
- ▶ 연산 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?
- ▶ 추가 블록 팔레트는 어떤 기능들이 모여 있는 팔레트인지 설명해 보세요?