
스파이크™ 프라임

기존 EV3 사용자, 스파이크™ 프라임 초보자를 위한

무작정 따라하기



CONTENTS



01 스파이크™ 프라임, 넌 누구?

02 스트리트 로봇 파이터 | 모터

03 스트리트 로봇 파이터 | 주행

04 스트리트 로봇 파이터 | 라이트, 사운드

05 반려 로봇 | 거리 센서

06 반려 로봇 | 컬러 센서

07 반려 로봇 | 힘 센서

08 반려 로봇 | 자이로 센서

09 도전 | 라인트레이싱

10 도전 | 사탕가져오기

11 인공지능 1 | 퓨너스 A.I.

12 인공지능 2 | 퓨너스 A.I.

04

[스트리트 로봇 파이터] 라이트, 사운드

- 01. 시작하기
- 02. 생각 다듬기
- 03. 개념 다지기
- 04. 프로그래밍
- 05. 검토하기
- 06. 더 나아가기

04

스트리트 로봇 파이터 | 라이트
사운드

01. 시작하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

05. 검토하기

06. 더 나아가기

시작하기

학습목표

라이트, 사운드 효과와 함께 주행하기

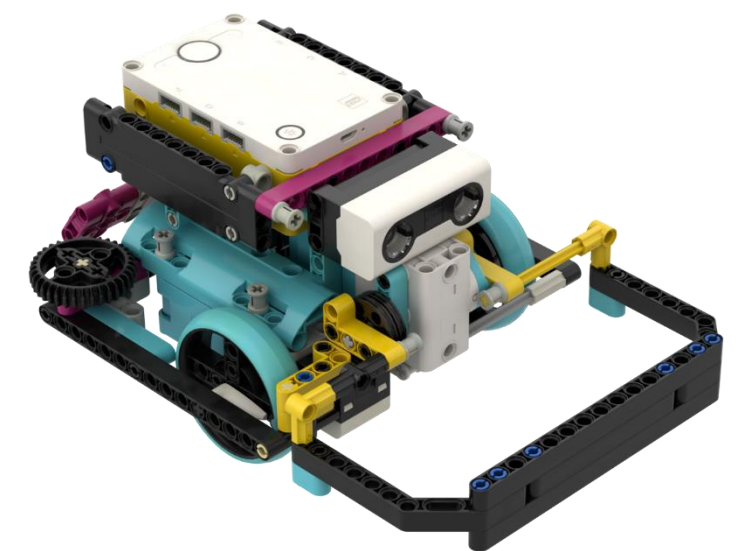
[프로그래밍 개념](#) ▶ 초기 설정, 병렬 처리

[새로 등장하는 블록 팔레트](#) ▶ 라이트, 사운드, 음악, 이벤트

멀티 로버가 멋진 회전 회오리 춤으로 EV-3를 이겼습니다. 엄청 화제가 된 회전 회오리 춤!

이것이 신인의 저력인 것인가...? 이 기세로 1차 미션 배틀 결과, SPIKE 크루가 1등을 했습니다. 와우!

이제 배틀은 잠시 뒤로! 루프탑에서 맞이하는 크루들의 꿀 같은 휴식시간! 이에 멀티 로버는 라이트와 사운드로 크루들과 즐거운 시간을 보내려 합니다. '노는게 제일 좋은' 멀티 로버. 라이트 블록과 사운드 블록으로 휴식시간의 인사가 될 수 있도록 여러분들이 도와주세요!



04

스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

05. 검토하기

06. 더 나아가기

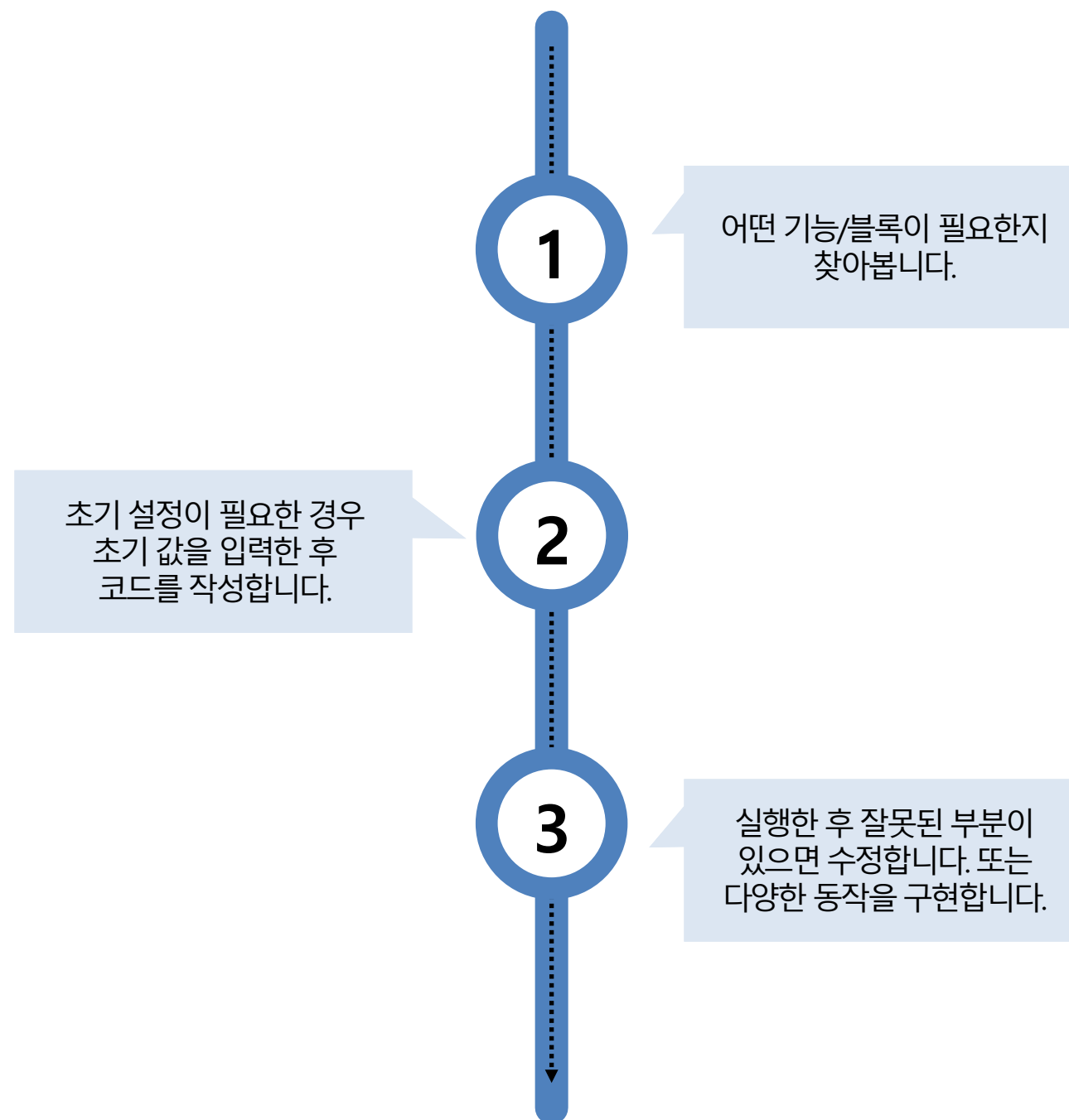
생각 다듬기

노는게 제일 좋아!

새로운 블록에 대한 개념을 익히고, 지금까지 배운 주행과 회전으로 신나게 노는 멀티 로버를 구현 할 거예요. 특히 이번엔 여러가지 프로그램을 동시에 실행하는 방법을 배울 거예요. 개념을 잘 익히고 천천히 코드를 따라 작성하세요.

알고리즘 생각하기

각 기능을 구현하기 위해 처음에 해야 할 설정이 무엇인지 생각한 후 순서대로 코딩을 하면 됩니다.



04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

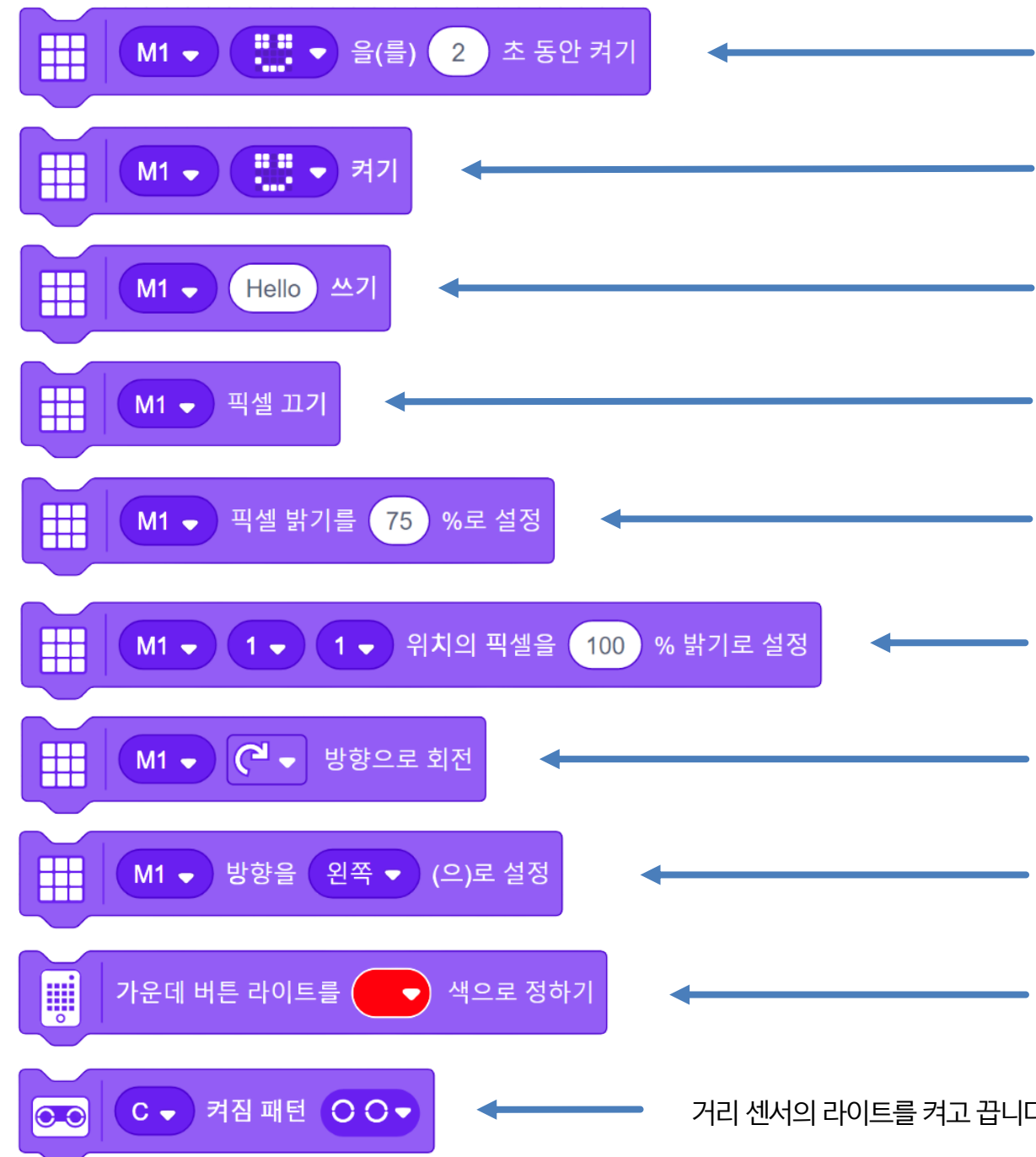
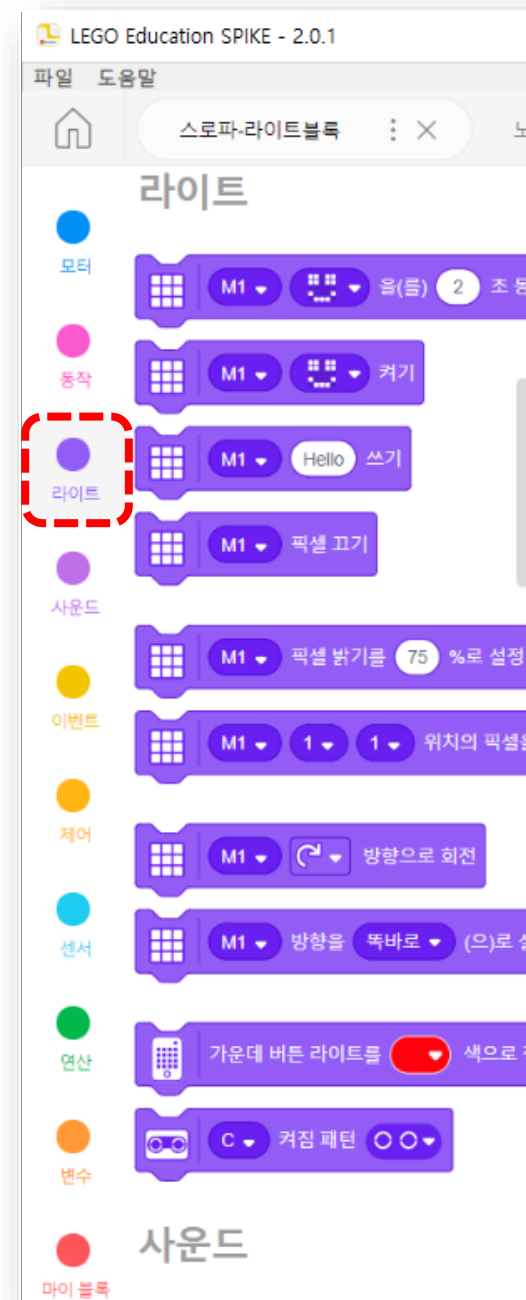
04. 프로그래밍

05. 검토하기

06. 더 나아가기

개념 다지기

라이트 파레트



라이트 매트릭스에 원하는 패턴을 원하는 시간(초)만큼 켭니다. 입력한 시간이 지나면 라이트가 꺼집니다.

위 블록과는 다르게 시간이 없죠? 이 블록을 작성한 후 라이트 매트릭스를 제어하는 다른 블록이 이후 프로그램에 없다면 이 블록의 상태를 계속 유지합니다.

입력 칸에 작성한 문자열을 라이트 매트릭스에 스크롤 방식으로 표시합니다.

라이트 매트릭스에 켜진 모든 라이트를 끕니다.

라이트 매트릭스의 밝기를 설정합니다. 밝기 범위는 0-100 입니다.

라이트 매트릭스의 밝기를 개별로 설정합니다. 위치를 지정한 픽셀만 밝기가 변경됩니다.

라이트 매트릭스에 표시되는 모양의 방향을 시계 방향 또는 반시계 방향으로 회전합니다.

위 블록에서 똑바로, 거꾸로 방향이 추가되었습니다.

허브 가운데 동그란 버튼의 라이트 색상을 설정합니다.

거리 센서의 라이트를 켜고 끕니다.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

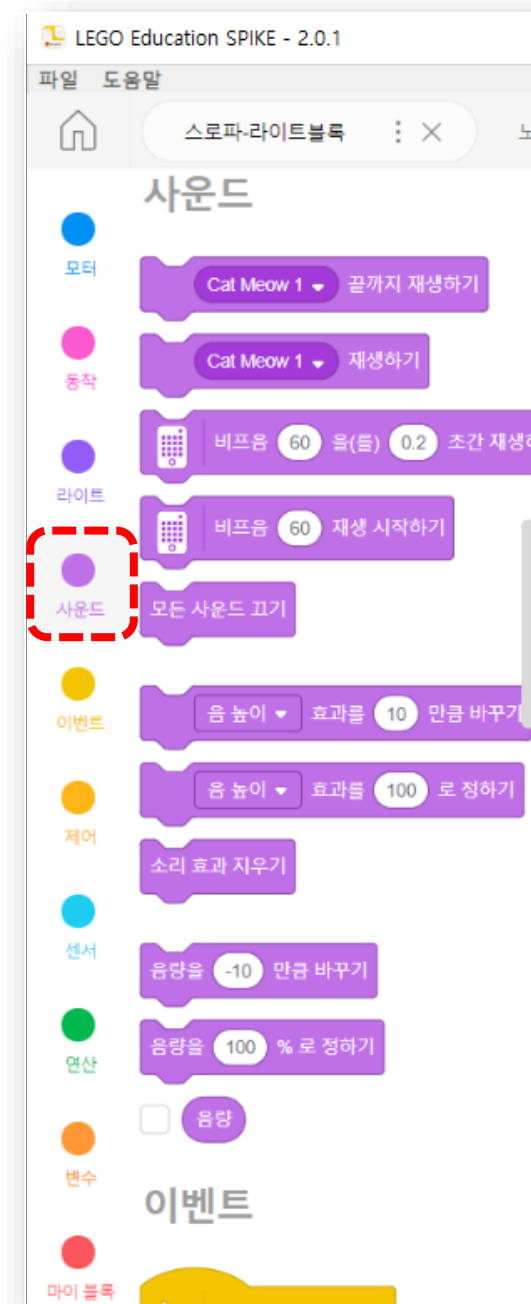
04. 프로그래밍

05. 검토하기

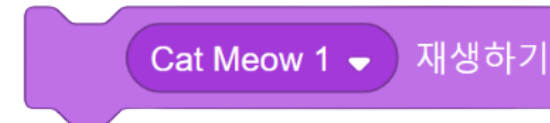
06. 더 나아가기

개념 다지기

사운드 파레트



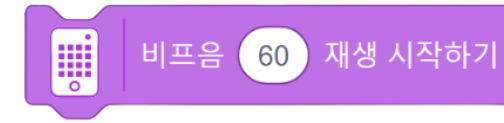
선택한 사운드가 재생되고 완료될 때까지 아래 블록을 실행하지 않습니다.



선택한 사운드가 재생되면 완료될 때까지 기다리는 게 아니라 바로 아래 블록을 실행합니다.



입력한 비프음(계이름) 지정한 시간(초)동안 재생합니다.



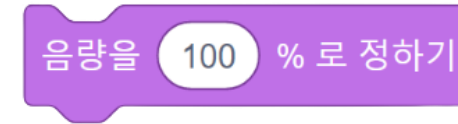
입력한 비프음(계이름)을 무제한으로 재생합니다. 중지하기 위해서는 프로그램을 종료하거나 아래 블록을 작성합니다.



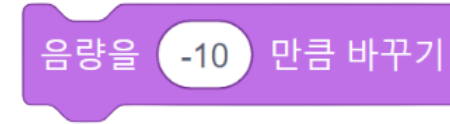
현재 재생중인 모든 사운드를 종료합니다.



사운드의 높낮이 효과를 줍니다. 범위는 -100 부터 100까지이며 단위가 높을수록 고음을 냅니다.



사운드의 기본 음량(volume)을 설정합니다. 0부터 100까지이며, 이 블록을 작성하지 않을 시 기본 음량은 100입니다.



현재 재생중인 사운드 음량을 입력한 양 만큼 변경합니다.



현재 음량을 보고합니다.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

05. 검토하기

06. 더 나아가기

개념 다지기

음악 파레트



(1) 스네어 드럼 ▾ 번 타악기를 0.25 박자로 연주하기

선택한 리듬 사운드를 입력한 박자(비트)동안 연주합니다.

0.25 박자 쉬기

지정된 박자(비트)만큼 쉬표를 연주합니다. 멈춘 같지만 사실 쉬표를 연주하고 있습니다.

60 번 음을 0.25 박자로 연주하기

입력한 박자동안 양의 정수로 입력한 피아노 음을 재생합니다.

악기를 (1) 피아노 ▾ (으)로 정하기

위 블록을 어떤 악기로 음을 낼 건지 정하는 블록입니다.

빠르기를 60 (으)로 정하기

박자의 빠르기를 설정하는 블록입니다. 박자의 빠르기에 대한 설명은 아래 내용을 참고하세요.

빠르기를 20 만큼 바꾸기

박자의 빠르기의 양을 변경하는 블록입니다.

빠르기

현재 진행중인 박자의 빠르기를 알려주는 블록입니다.

* 박자의 빠르기를 템포(tempo)라고 하며, 템포는 분당 비트(bpm; beats per minute)로 표시합니다. 예를 들어 60bpm은 1분에 비트를 60번 나눠서 박자를 진행한다는 뜻입니다. 즉 1초마다 1비트 씩 진행된다는 의미입니다.

* 모터 추가메뉴 파레트를 가져온 것처럼 프로그램 좌측 하단 블록 확장요소 표시 버튼을 눌러 음악 팔레트를 불러오세요.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

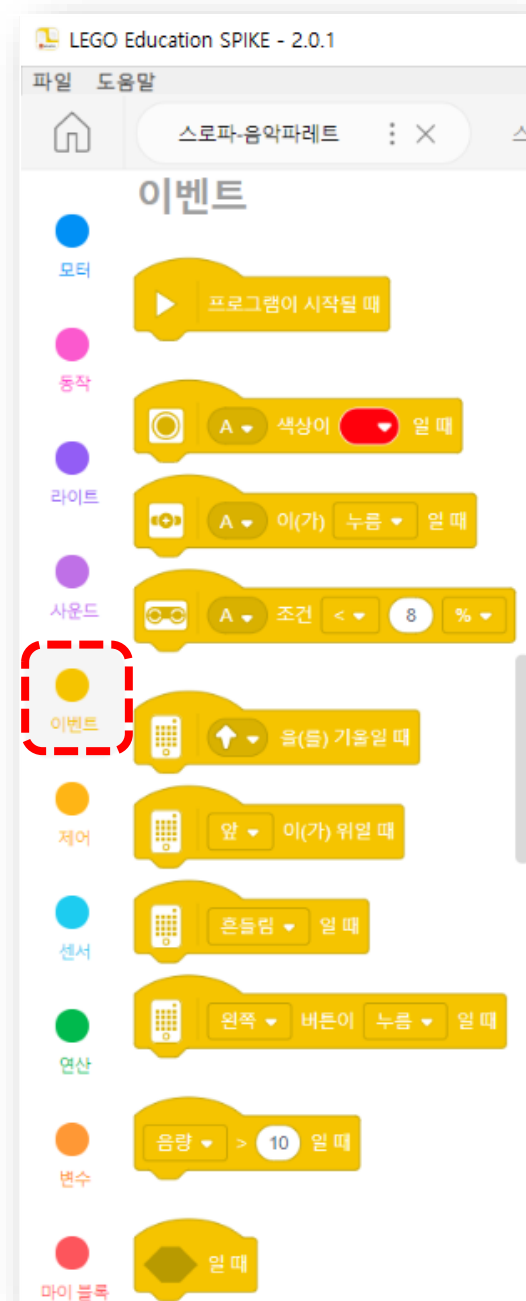
04. 프로그래밍

05. 검토하기

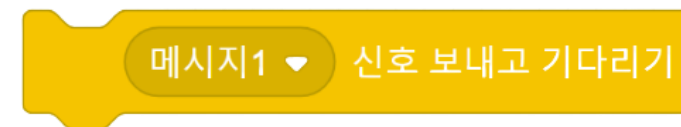
06. 더 나아가기

개념 다지기

이벤트 파레트



이 블록은 *메시지 브로드캐스트*라는 이벤트 신호를 보내는 블록입니다. 이 블록이 실행되면 지정한 메시지 신호를 보내고 바로 아래 블록을 실행합니다.



위 블록처럼 신호를 보내지만 바로 아래 블록을 실행하는 게 아닌, 메시지 신호 받은 블록 꾸러미들이 다 실행된 후 이 블록 아래에 있는 블록을 실행합니다.



위에 두 개의 블록들이 메시지를 보내면 이 블록으로 메시지를 받은 후 이 블록 아래 작성한 블록들을 순차적으로 실행합니다.



어? 이 블록은 맨 처음 시작할 때 시작을 알리는 이벤트 블록이죠? 왜 또 나왔을까요? 이 블록은 한 개가 아닌 여러 개를 둘 수 있으며, 이 블록의 개수만큼 아래 설명한 병렬 처리를 할 수 있어요. 신기하죠?

* *메시지 브로드캐스트* 블록은 프로그램을 동시에 여러 개 실행시키려 할 때 사용할 수 있는 블록입니다. *병렬 처리(Multiprocessing)* 이라고 하죠. 두 가지 이상의 일을 동시에 수행할 수 있게 하는 기능입니다. 참고로 위에 '메시지1'이라고 된 메시지의 명칭을 수정할 수 있습니다.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

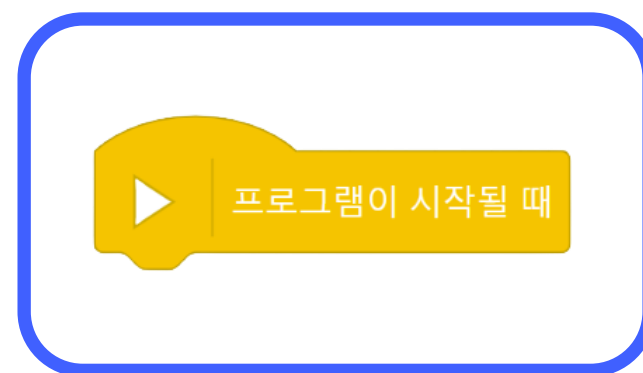
05. 검토하기

06. 더 나아가기

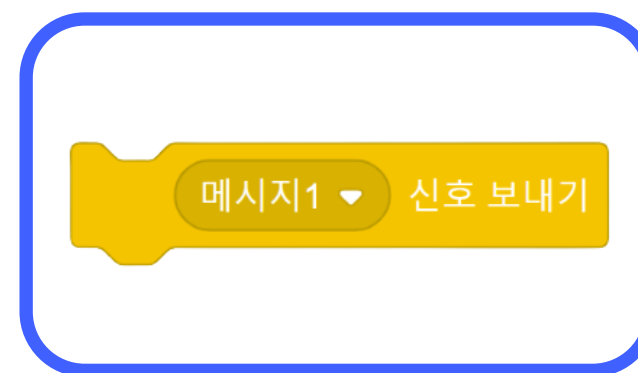
개념 다지기

병렬 처리 | Multi Processing

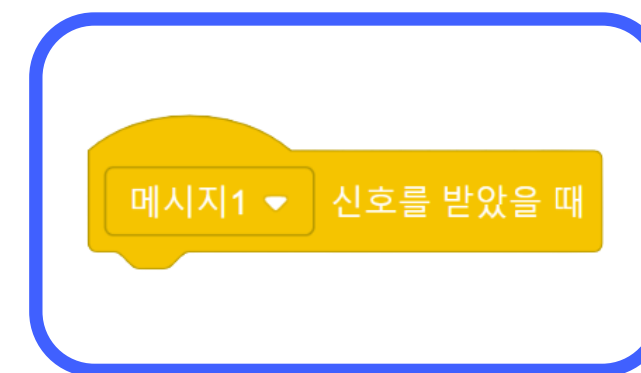
병렬 처리는 두 가지 이상의 일을 동시에 수행(처리)할 수 있게 하는 기능입니다. 이벤트 팔레트에서 아래 블록으로 두 가지 이상의 프로세스를 동시에 실행할 수 있습니다. 아래 설명을 보고 다음장에 있는 예시를 무작정! 따라해보세요.



이 블록을 두 개 이상 실행할 수 있으며, 프로그램이 시작할 때 이 블록구러미들 안에 있는 블록들이 동시에 차례로 실행됩니다.



실행중인 블록에 이 블록을 넣으면 이벤트가 발생하여 '메시지1'이라는 메시지를 전달합니다.



왼쪽 블록에서 보낸 '메시지1'의 이벤트 메시지를 받는 블록입니다. 메시지를 받은 후 아래 블록들을 차례로 실행합니다.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

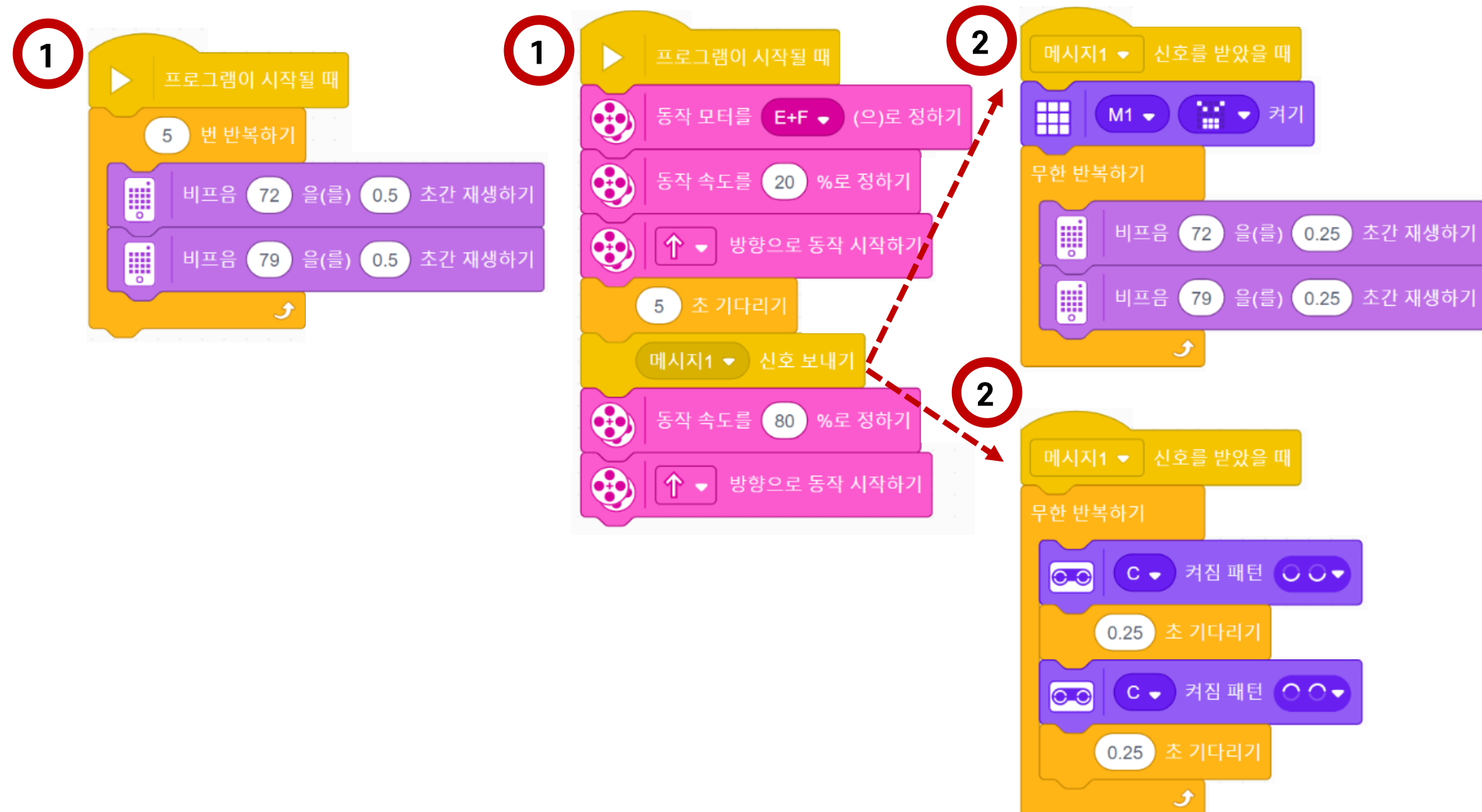
05. 검토하기

06. 더 나아가기

개념 다지기

병렬 처리 | Multi Processing

우선 이해가 안 되더라도 어떤 순서로 실행되는지 확인하기 위해 무작정 따라해보세요!



04

스트리트 로봇 파이터 | 라이트
사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

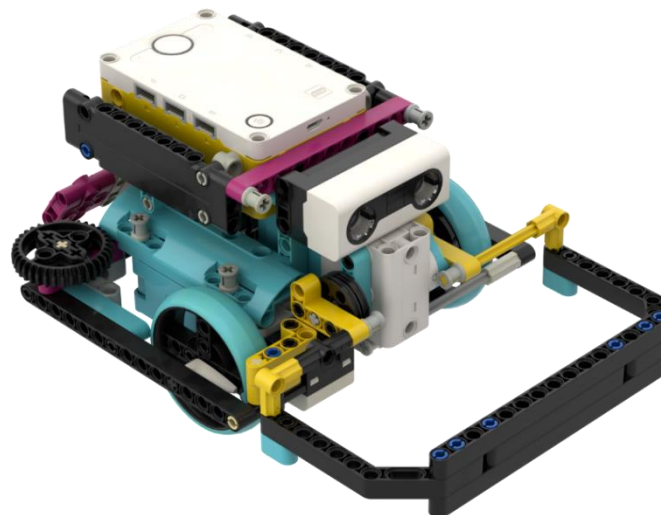
05. 검토하기

06. 더 나아가기

프로그래밍

동작 구현 순서

- 필요한 파레트** 이벤트 파레트, 음악 파레트, 사운드 파레트, 라이트 파레트, 제어 파레트, 모터 파레트, 동작 파레트
- STEP 01** 템포를 설정하고 비트를 만든다.
- STEP 02** **STEP 01**에서 만든 비트에 맞춰 음악을 만든다.
- STEP 03** 음악에 맞춰 회전을 한다.
- STEP 04** 음악에 맞춰 라이트를 켜다.



Drop the BEAT!

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

05. 검토하기

06. 더 나아가기

프로그래밍

STEP 01 템포 설정하고 리듬 만들기 60bpm



Bpm을 정하는 블록입니다. 60bpm이니 1초당 1박자를 진행하겠죠?

“노는 게 제일 좋아” 노래를 생각하며 쿵쾅쿵쾅 비트를 만들고 이를 4번 반복했습니다. 여러가지 타악기를 선택할 수 있습니다.

연주가 끝난 후 크래시 심벌로 끝나는 효과를 내도록 작성했습니다. 소리를 반복해서 내기 위해 제어의 반복 블록을 사용했습니다.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

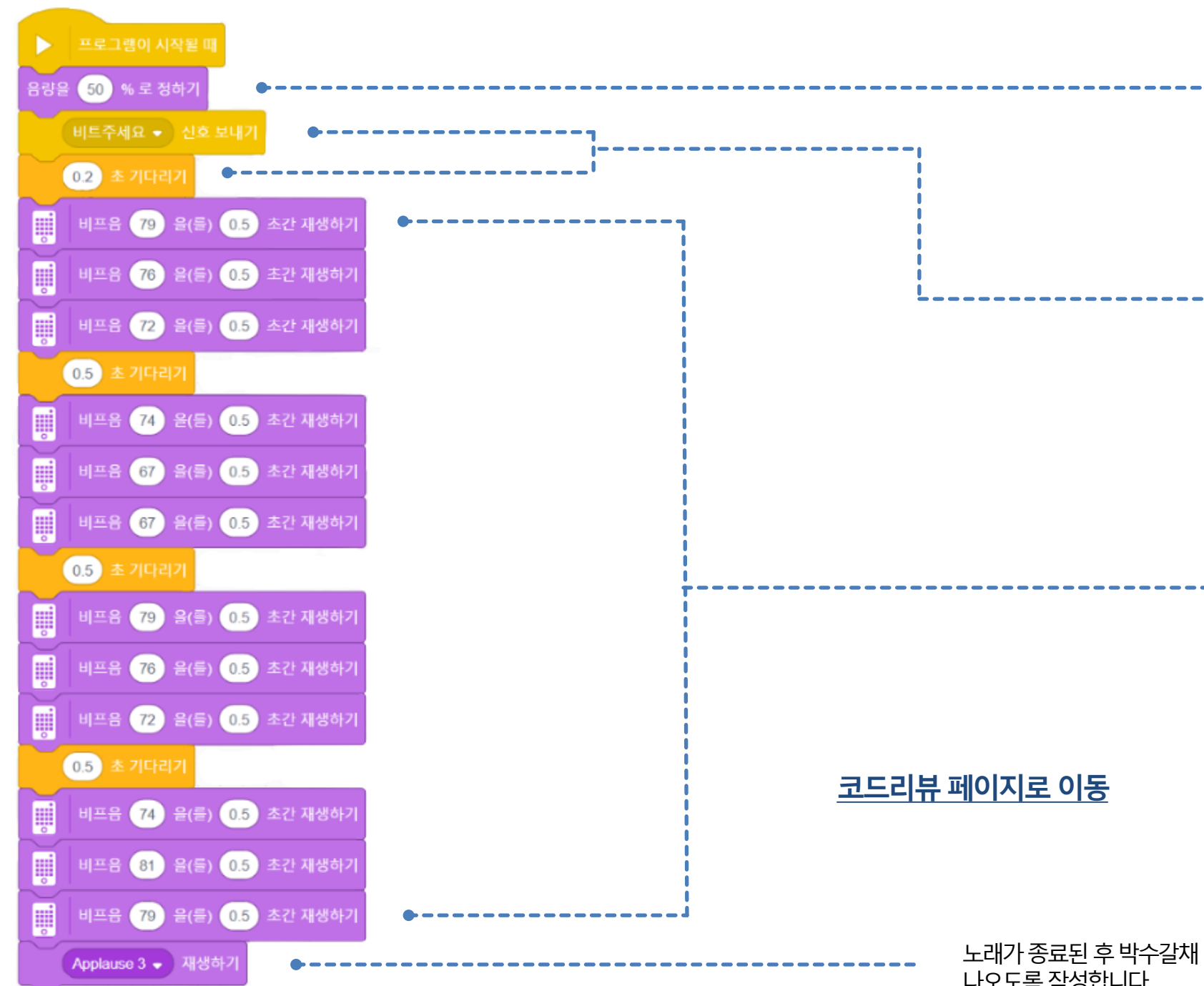
04. 프로그래밍

05. 검토하기

06. 더 나아가기

프로그래밍

STEP 02 비트에 맞춰 음악 만들기



소리를 50%로 줄였습니다.

메시지 브로드캐스트 블록입니다. '비트주세요'라고 메시지를 보내면 이 메시지를 받는 블록이 이벤트 신호를 받아 아래 블록들을 실행합니다. 메시지 받는 블록은 STEP 01에서 작성한 비트에 작성합니다. 다음 페이지를 참고하세요. 0.2초 대기블록을 넣은 이유는 STEP 01에서 작성한 비트 블록꾸러미와 동시에 실행하는데 걸리는 delay 때문입니다.

노는게 제일 좋은 노래입니다. 음을 찾아 구성하고, 60bpm에 맞추어 각 비트음마다 적절한 시간을 입력합니다. STEP 01에서 만든 bpm과 같아야 한다는 점, 잊지 마세요.

[코드리뷰 페이지로 이동](#)

노래가 종료된 후 박수갈채 사운드가 나오도록 작성합니다.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

05. 검토하기

06. 더 나아가기

프로그래밍

STEP 02 비트에 맞춰 음악 만들기 - 이벤트 변경



'비트주세요'라는 이벤트 메시지를 보냈으니 비트를 진행하는 코드꾸러미에서 이 메시지를 받는 코드를 작성하면 됩니다. 이 블록을 찾아 이전에 작성한 '프로그램이 실행될 때' 블록을 바꿔주세요.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

05. 검토하기

06. 더 나아가기

프로그래밍

STEP 03 음악과 비트에 맞춰 회전하기



템포를 60bpm으로 설정한 거 기억나죠? 비트를 05박으로 4번 쪼갠기(?) 때문에 비트 한 구간에 맞추어 2초로 작성했습니다. 왼쪽과 오른쪽 모터의 속도를 보면 이제 어떤 방향으로 진행하는지 알겠죠?

음악에 맞추어 후진한 후 왼쪽과 오른쪽으로 Point Turn을 합니다.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

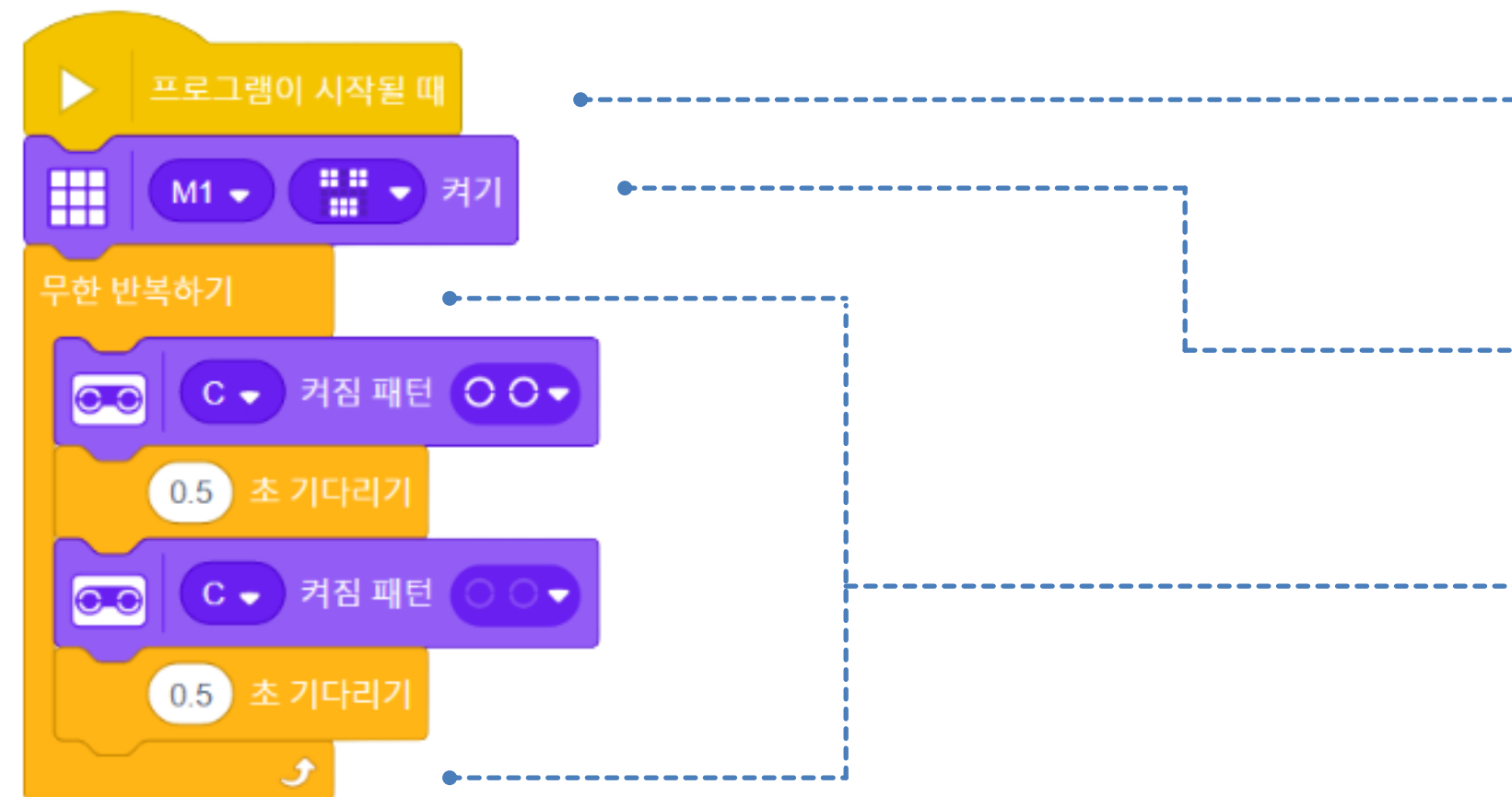
04. 프로그래밍

05. 검토하기

06. 더 나아가기

프로그래밍

STEP 04 음악과 비트에 맞춰 라이트 켜기



이 코드를 작성하면 '프로그램이 시작될 때' 블록은 총 2개가 되겠네요. 즉 병렬 처리 프로세스가 2개가 있다는 겁니다. 두 개의 프로그램이 동시에 실행이 된다는 얘기입니다.

매트릭스 라이트를 무한히 켵니다.

거리 센서에 달린 LED를 60bpm 비트에 맞추어 제어합니다. 한 번만 실행되는게 아니라 여러 번 실행하기 위해 무한 반복하기 블록으로 살포시 감싸줍니다.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

05. 검토하기

06. 더 나아가기

검토하기

코드 리뷰

완성된 코드를 살펴보고 혹시 놓친 부분이 있는지 확인해보세요. 살펴보고 싶은 목록을 클릭하면 해당 페이지로 이동합니다.

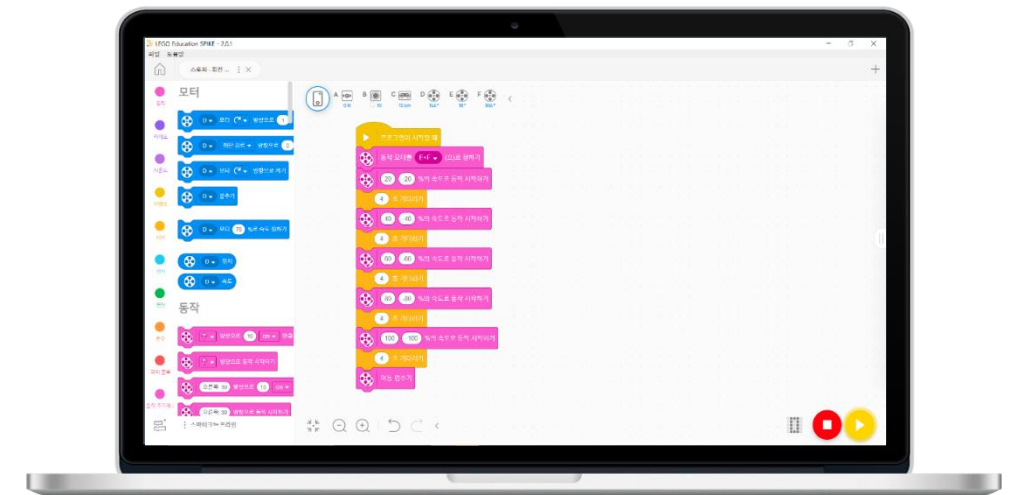
STEP 01 템포 설정하고 리듬 만들기

STEP 02 비트에 맞추어 음악 만들기

비트에 맞추어 음악 만들기 - 이벤트 변경

STEP 03 음악과 비트에 맞추어 회전하기

STEP 04 음악과 비트에 맞추어 라이트 켜기



04

스트리트 로봇 파이터 | 라이트
사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

04. 프로그래밍

05. 검토하기

06. 더 나아가기

더 나아가기

도전! 스파이크™ 프라임 코딩 마스터

다양한 방법으로 즐겁게 휴식을 즐기는 멀티 로버를 만들어보세요!

미션 1

범인 발견한 경찰차 구현하기 [예시코드보러가기](#)

10페이지에서 일단 작성해본 코드를 응용하여
범인을 발견하여 추격하는 멀티 로버를 구현해보세요.

미션 2

노래에 따라 춤추는 멀티 로버 구현하기

이젠 여러분들이 비트와 음악을 직접 만들어
그 음악에 따라 춤추는 멀티 로버를 만들어보세요.

04 스트리트 로봇 파이터 | 라이트 사운드

01. 생각하기

02. 생각 다듬기

03. 개념 다지기

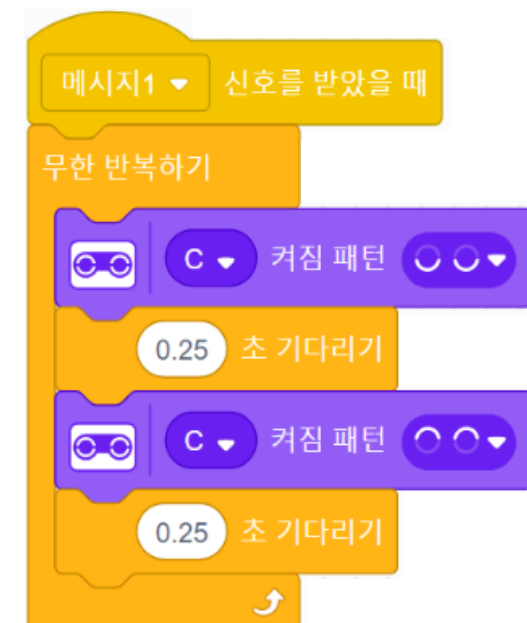
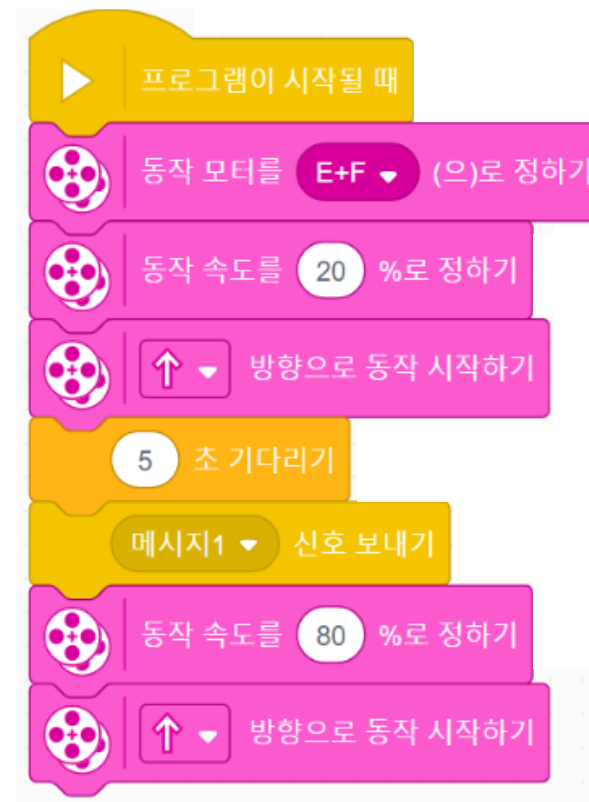
04. 프로그래밍

05. 검토하기

06. 더 나아가기

더 나아가기

미션 1 예시코드



다음 세대의 시간을 가치 있게!

