

6. 셔틀버스 운행

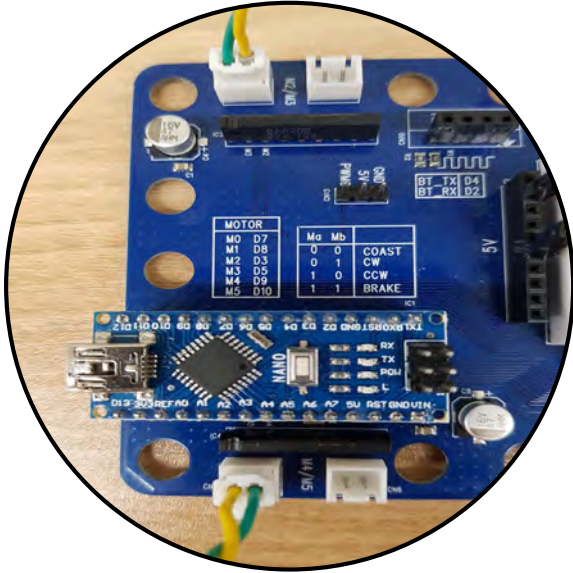


S:mplock
심플록

학습 내용

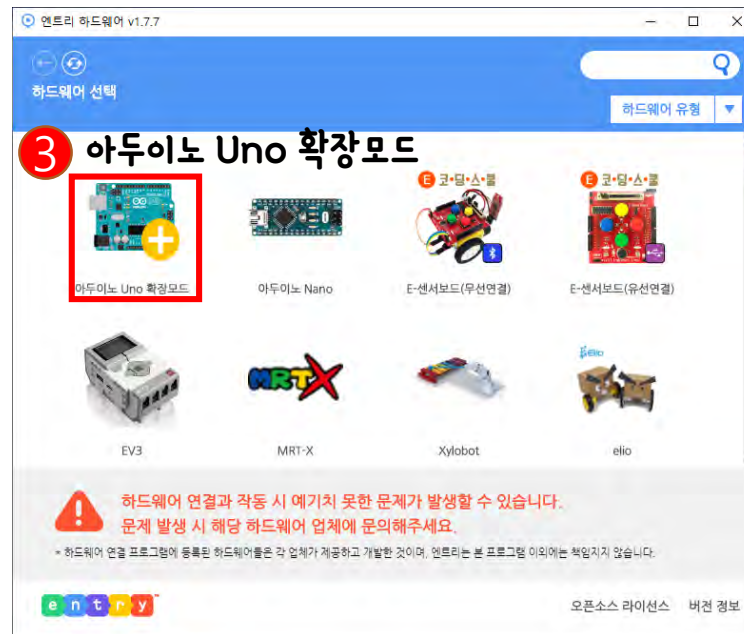
- 심플렉(키트)을 이용하여 버스 만들기
- 정해진 길을 보고 '전진', '후진', '좌회전', '우회전', '정지'함수를 이용하여 문제 해결하기

자신만의 개성 있는 셔틀버스를 만들고 심플보드와 연결해주세요.
DC 모터는 M2/M3, M4/M5 모터커넥터에 연결해주세요.



연결하기

먼저 엔트리와 심플보드를 연결시켜 줍니다.

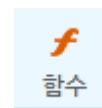


함수 만들기



속성 탭의 함수를 클릭 해줍니다.

다음으로 함수 추가하기를 클릭합니다.



함수 블록 탭에서 함수 만들기를 클릭해도 됩니다.

지난 시간에 배웠던 것들을 떠올려서
전진, 후진, 좌회전, 우회전, 정지 함수를 만들어 보세요.



함수 정의하기 **전진**

디지털 3 ▾ 변 핀 끄기 ▾

디지털 5 ▾ 변 핀 켜기 ▾

디지털 9 ▾ 변 핀 끄기 ▾

디지털 10 ▾ 변 핀 켜기 ▾

함수 정의하기 **좌회전**

디지털 3 ▾ 변 핀 켜기 ▾

디지털 5 ▾ 변 핀 끄기 ▾

디지털 9 ▾ 변 핀 끄기 ▾

디지털 10 ▾ 변 핀 켜기 ▾

함수 정의하기 **정지**

디지털 3 ▾ 변 핀 끄기 ▾

디지털 5 ▾ 변 핀 끄기 ▾

디지털 9 ▾ 변 핀 끄기 ▾

디지털 10 ▾ 변 핀 끄기 ▾

함수 정의하기 **우회전**

디지털 3 ▾ 변 핀 끄기 ▾

디지털 5 ▾ 변 핀 켜기 ▾

디지털 9 ▾ 변 핀 켜기 ▾

디지털 10 ▾ 변 핀 끄기 ▾

함수 정의하기 **후진**

디지털 3 ▾ 변 핀 켜기 ▾

디지털 5 ▾ 변 핀 끄기 ▾

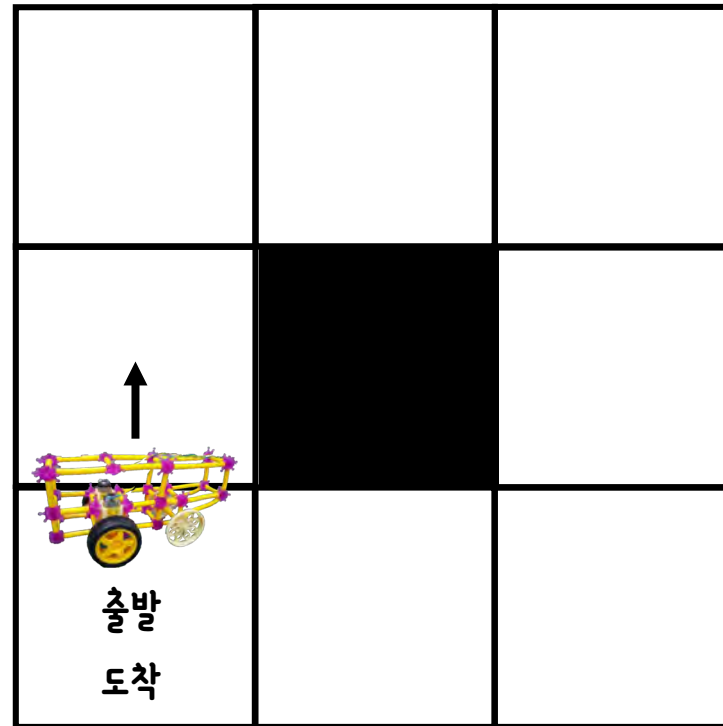
디지털 9 ▾ 변 핀 켜기 ▾

디지털 10 ▾ 변 핀 끄기 ▾

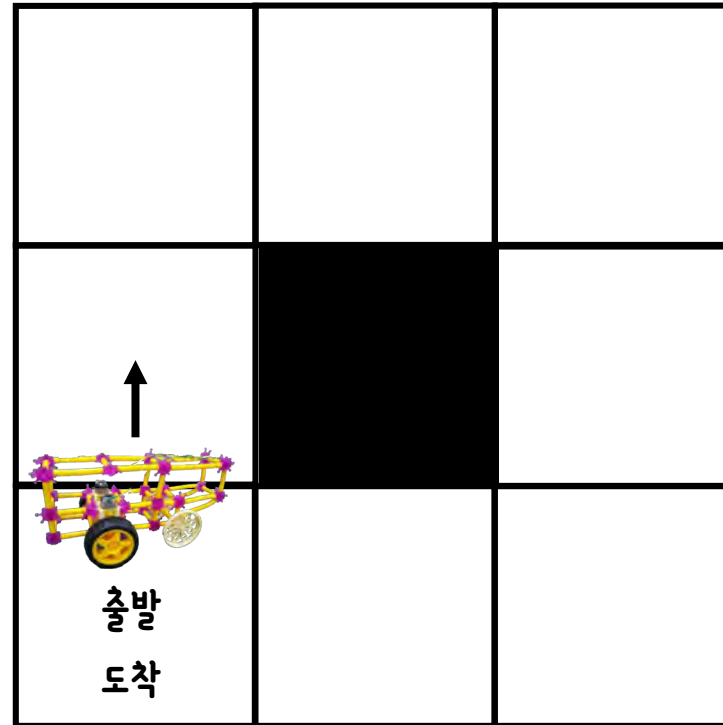
문제1

한 칸당 1초가 소요됩니다.

정해진 길에 맞게 블록을 코딩해봅시다.



출발 → 전진 → 전진 → 우회전 → 전진 → 전진 → 우회전 → 전진 → 전진 → 우회전 → 전진 → 전진 → 우회전 → 도착



만들어보기

만들어본 함수를 이용해봅시다.



전진을 1초 동안 유지하는 코드입니다.

그렇다면 한 칸에 1초이므로 전진을 두 칸 하면 2초 기다리기를 하면 되겠죠?



‘전진 → 전진 → 우회전’ 코드입니다.

‘전진 → 전진 → 우회전’은 4번 반복되므로 반복문을 이용하면 코드가 간결 해집니다.



길을 다 찾으면 정지를 시켜줍니다.

만들어보기



‘전진 -> 전진 -> 우회전’이 4번 반복 하므로 반복 블록을 추가해줍니다.



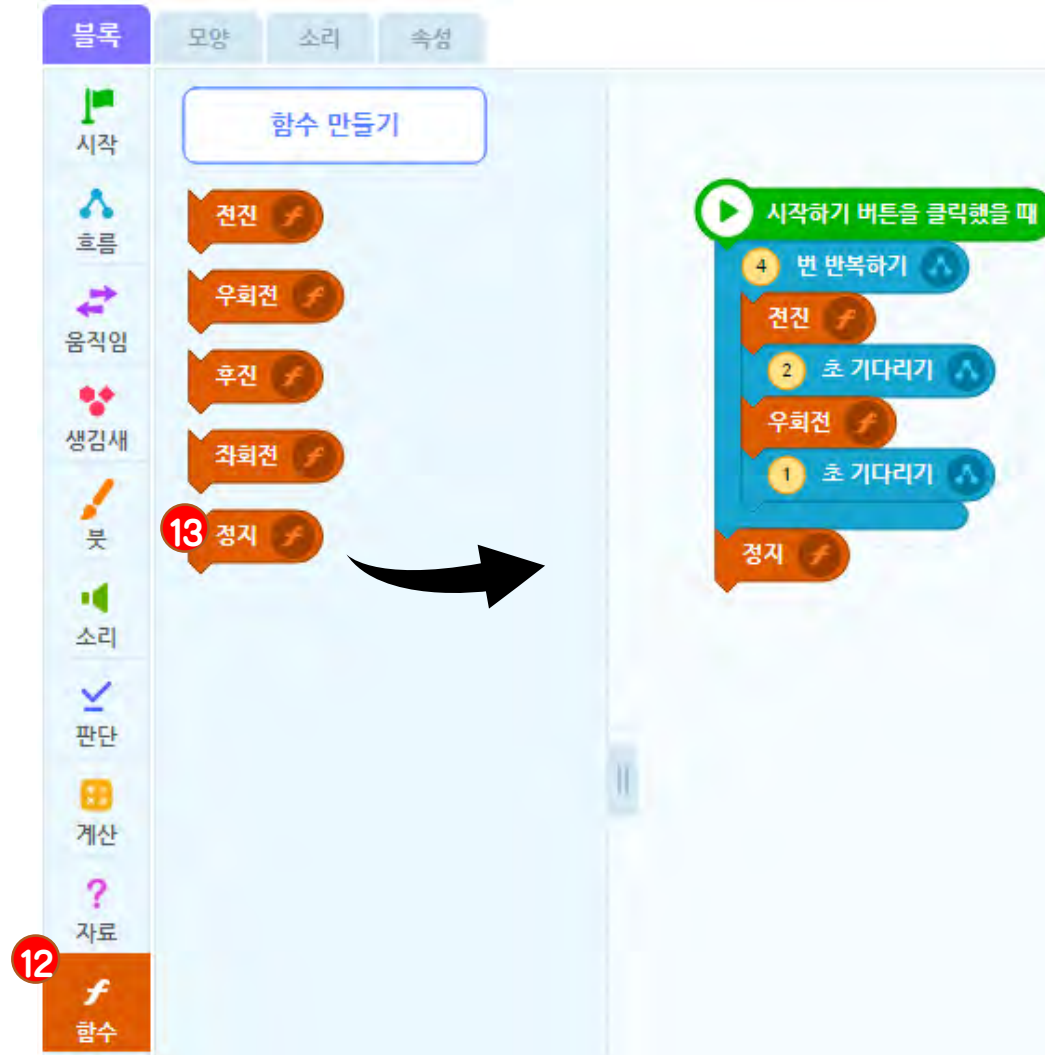
‘전진 -> 전진 -> 우회전’에서 필요한 전진과 우회전 함수를 추가합니다.



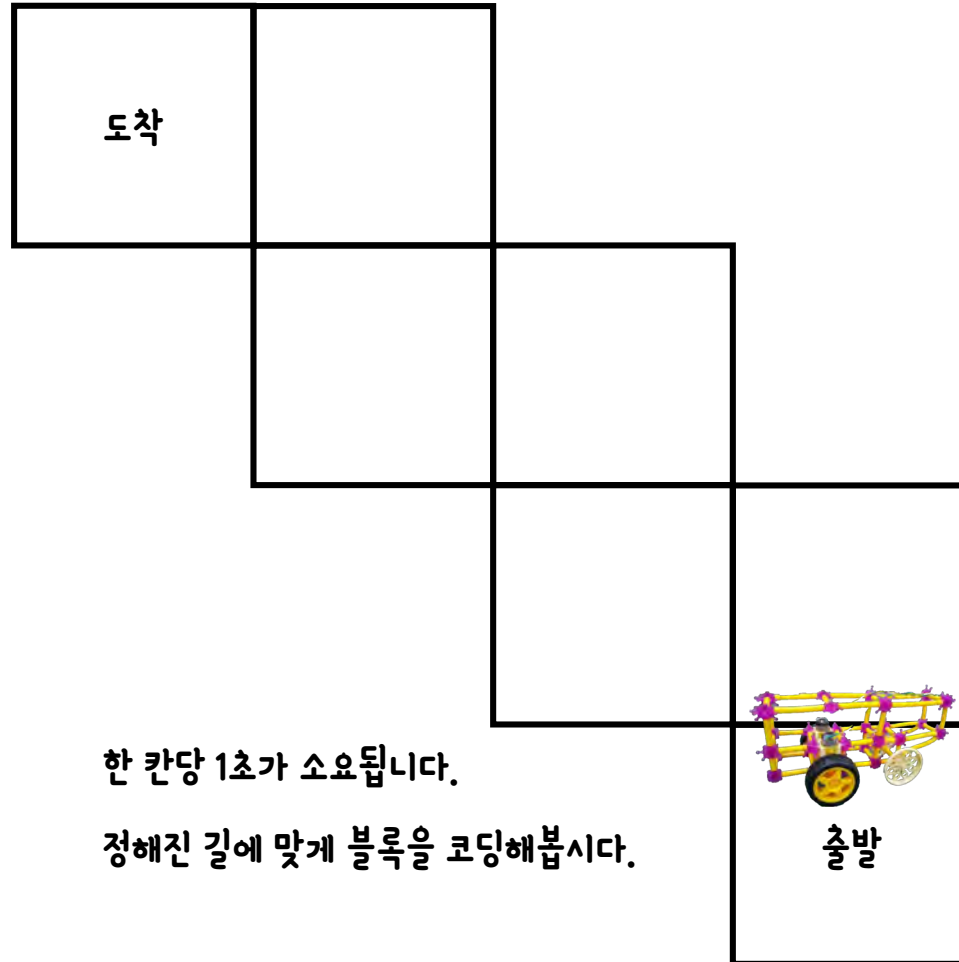
전진 2번, 우회전 1번 이므로 각각 2초와 1초 기다리기를 넣어줍니다.



길을 다 찾았다면 정지를 시켜줍니다.



문제2

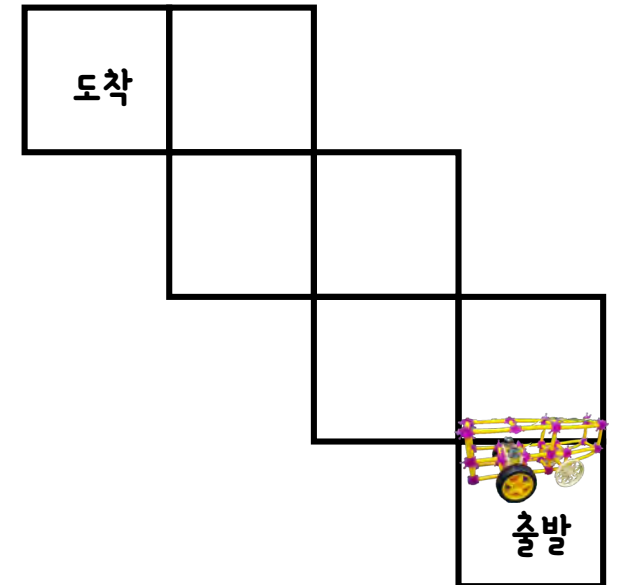


한 칸당 1초가 소요됩니다.
정해진 길에 맞게 블록을 코딩해봅시다.

출발 → 전진 → 좌회전 → 전진 → 우회전 → 전진 → 좌회전 → 전진 → 우회전 →
 → 전진 → 좌회전 → 전진 → 우회전 → 전진 → 좌회전 → 전진 → 우회전 → 도착

블록이 너무 길어 지기 때문에 반복되는 구간을 찾습니다.

‘전진 → 좌회전 → 전진 → 우회전’이 4번 반복되고 있습니다.



만들어보기



‘전진 -> 좌회전 -> 전진 -> 우회전’이 4번 반복 하므로 반복 블록을 추가해줍니다.



전진 → 좌회전 → 전진 → 우회전

The image shows a Scratch code editor interface. On the left sidebar, the '함수' (Function) block is highlighted with a red circle containing the number '6'. The main workspace is divided into '모양' (Appearance), '소리' (Sound), and '속성' (Properties) tabs. The '함수 만들기' (Make Function) dialog is open, showing a list of movement blocks: 전진 (Forward), 후진 (Backward), 좌회전 (Turn Left), 우회전 (Turn Right), and 정지 (Stop). A red circle with the number '7' is positioned over the '4번 반복하기' (Repeat 4 times) block in the script area, which contains a sequence of four movement blocks: 전진, 좌회전, 전진, and 우회전. A black arrow points from the '7' circle to the '4번 반복하기' block.

각각 1칸씩 움직이므로 1초 기다리기 블록을 넣어줍니다.

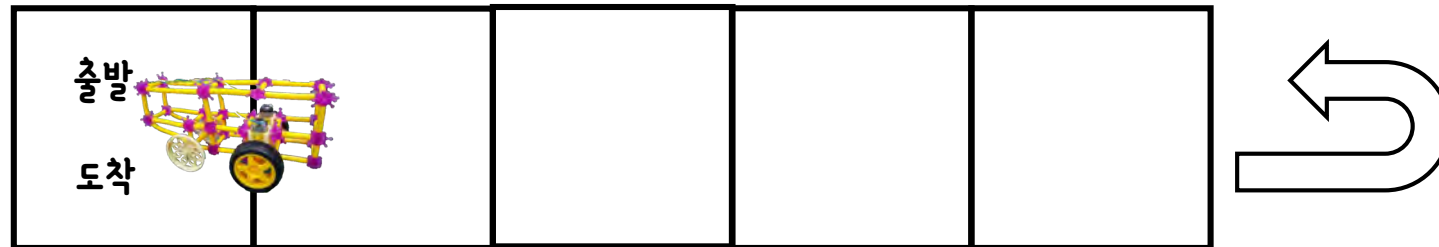


길을 다 찾았으므로 마지막에 정지를 시켜줍니다.



문제3

한 칸당 1초가 소요됩니다.
정해진 길에 맞게 블록을 코딩해봅시다.



회전하는 시간은 직접 해보면서 조절하시면 됩니다.



or



or



강의 요약

- 함수를 사용하여 문제 풀어보기
- DC모터 제어하기

다음시간에는 엔트리를 이용해서 LED 제어를 배워 보겠습니다.

본 강의 자료는 www.simplock.co.kr 에서 다운받으실 수 있습니다.

