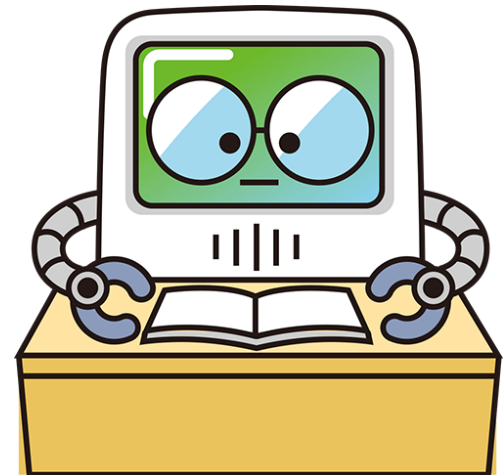


누구나 쉽게 배우는

인공지능 스타트



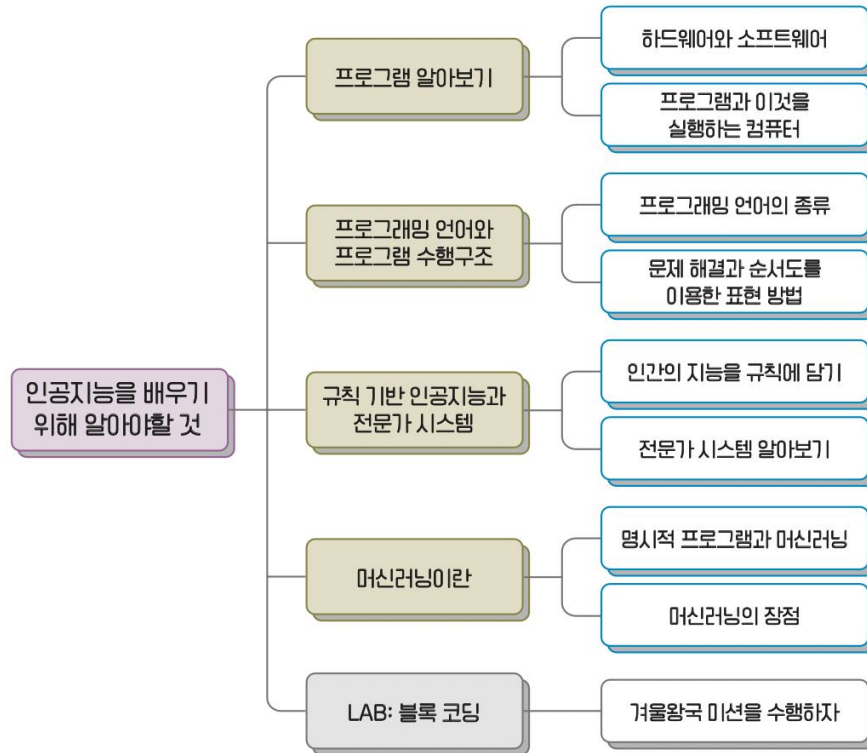
05

인공지능을 배우기 위해 알아야 할 것

학습목차

- 01 프로그램에 대해 알아보자
- 02 다양한 프로그래밍 언어와 프로그램의 수행 구조
- 03 규칙 기반 인공지능과 전문가 시스템
- 04 명시적 프로그램의 한계와 이를 극복한 머신러닝
- LAB** 블록 코딩을 익혀보자

요약
연습문제
심화 토의 문제



학습목표

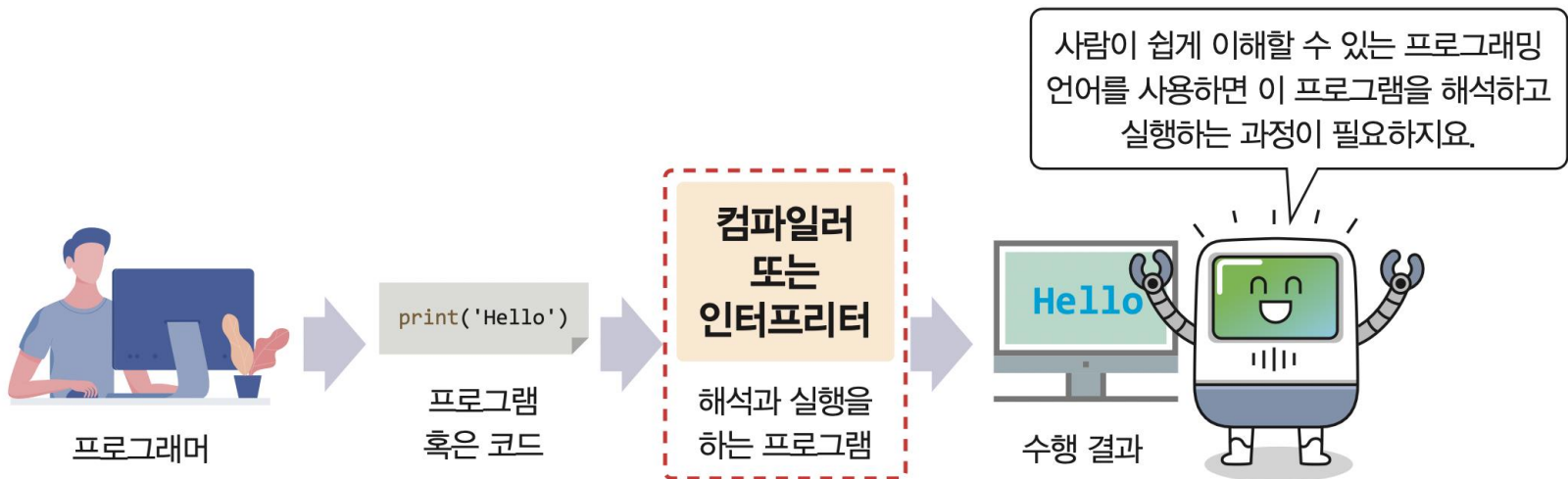
- 인공지능을 구현하는 도구인 프로그램과 프로그래밍에 대해 알아보자.
- 프로그래밍의 종류와 기본적인 제어 구조를 통해 프로그램의 작동 원리를 이해하자.
- 소프트웨어의 문제해결 표현 방법을 살펴보자.
- 인간의 지능과 유사한 일을 할 수 있도록 하는 규칙 기반 시스템에 대해 알아보자.
- 머신러닝을 통해 어떻게 명시적 규칙을 생성할 수 있을지 알아보자.

“지능은 무엇을 해야 할지 모를 때 사용하는 것이다.”

_스위스의 철학자이자 발달심리학자, 장 피아제

01 프로그램에 대해 알아보자

- 프로그래머가 'Hello'를 화면에 출력하는 과정
 - 특정 소프트웨어가 이를 해석하여 컴퓨터에서 실행한다.
- 이렇게 작성된 **코드** code 를 프로그램이라고 하는데 프로그램을 실행하기 위해서는 **컴파일러** compiler 나 **인터프리터** interpreter 와 같은 해석과 실행을 담당하는 별도의 프로그램이 필요하다.



01 프로그램에 대해 알아보자

■ 좀 더 복잡한 프로그램은 똑똑한 일을 할 것이다

- 놀이공원에 키가 160 cm 이상인 고객들만 이용할 수 있는 놀이기구가 있다고 가정하고 사용자로부터 키를 입력받아 이 놀이기구를 사용할 수 있는가를 알려주는 기능을 수행하는 프로그램으로 만들어보자.

```
height = input('키를 입력하세요(단위: cm): ')
if int(height) >= 160:
    print('놀이기구 이용이 가능합니다. 즐거운 시간 보내세요!')
else:
    print('죄송하지만 놀이기구 이용이 불가능합니다.')
```

02 다양한 프로그래밍 언어와 프로그램의 수행 구조

■ 프로그래밍 언어 programming language

- 컴퓨터에게 일을 시키기 위하여 사용되는 명령의 모임
- 일정한 문법체계를 가진 명령어

- 주요 프로그래밍 언어

프로그래밍 언어	특징
Python(파이썬)	문법이 단순하며 강력한 기능을 가진 오픈 소스 프로그래밍 언어
Java(자바)	강력한 객체지향 방식의 프로그래밍 언어
C	빠른 수행 속도를 자랑하는 오랜 역사를 가진 프로그래밍 언어
JavaScript	웹브라우저에서 주로 사용되는 스크립트 방식 프로그래밍 언어

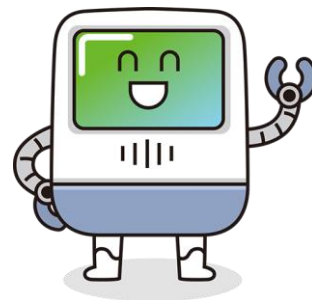
02 다양한 프로그래밍 언어와 프로그램의 수행 구조

■ 프로그램의 제어 구조

- **순차** *sequence* 구조: 여러 명령이 순차적으로 실행되는 구조로 가장 기본적인 구조이다.
- **선택** *selection* 구조: 여러 개의 명령문 중에서 조건에 따라 명령문을 선택하여 실행되는 구조이다.
- **반복** *iteration* 구조: 조건에 따라 동일한 명령이 반복되면서 실행되는 구조이다.

■ 인공지능의 기호주의

- 명시적인 프로그램으로 방대한 규칙을 입력
- 인간의 사고를 완벽하게 표현할 수 있다면 인공지능을 구현할 수 있지 않을까?



03 규칙 기반 인공지능과 전문가 시스템

■ 규칙 기반 모델 rule based model

- 다양한 조건에 대해서 개별적인 처리 기능을 가지도록 많은 규칙을 컴퓨터에게 넣어준다면, 이 컴퓨터는 궁극적으로 인간의 지능과 유사한 일을 할 수 있을 것이다.

■ 규칙 기반 시스템 rule based system

- 인공지능 구현 기법
- 규칙 집합과 규칙 엔진이라고 하는 두 가지의 요소가 필요
 - 규칙 기반 모델을 구현하기 위해

규칙 집합	여러 상황에 대한 경우의 모음
규칙 엔진	규칙에 대한 조건을 평가, 행동으로 실행하는 것을 담당.

03 규칙 기반 인공지능과 전문가 시스템

■ 전문적인 규칙 집합을 바탕으로 만든 전문가 시스템

- 규칙 기반 모델이 인간의 지능과 유사한 일을 하기 위해서는 방대한 규모의 지식이 요구

■ 이 모델에서 사용하는 지식

- 어떤 주제나 분야에 대해 이론적으로 그리고 실제로 이해하는 것
- 지식을 소유한 사람을 **전문가** expert 라고 부름

■ 전문가 시스템

- 문제 영역의 전문가로부터 획득된 지식을 이용하여 추론하며
- 동시에 전문적으로 고도의 현실적 문제를 전문가와 동등한 레벨에서 해결하는 지적 인공지능 시스템



03 규칙 기반 인공지능과 전문가 시스템

■ 전문가 시스템이라고 불리기 위한 네 가지 조건

1. 문제 영역의 전문 지식을 이용하여 추론할 것
2. 전문가로부터 획득된 지식일 것
3. 전문적으로 고도의 현실적 문제를 대상으로 할 것
4. 능력이 전문가와 동등할 것

■ 1960년대 후반 미국의 스탠퍼드 대학에서 개발된 **DENDRAL**

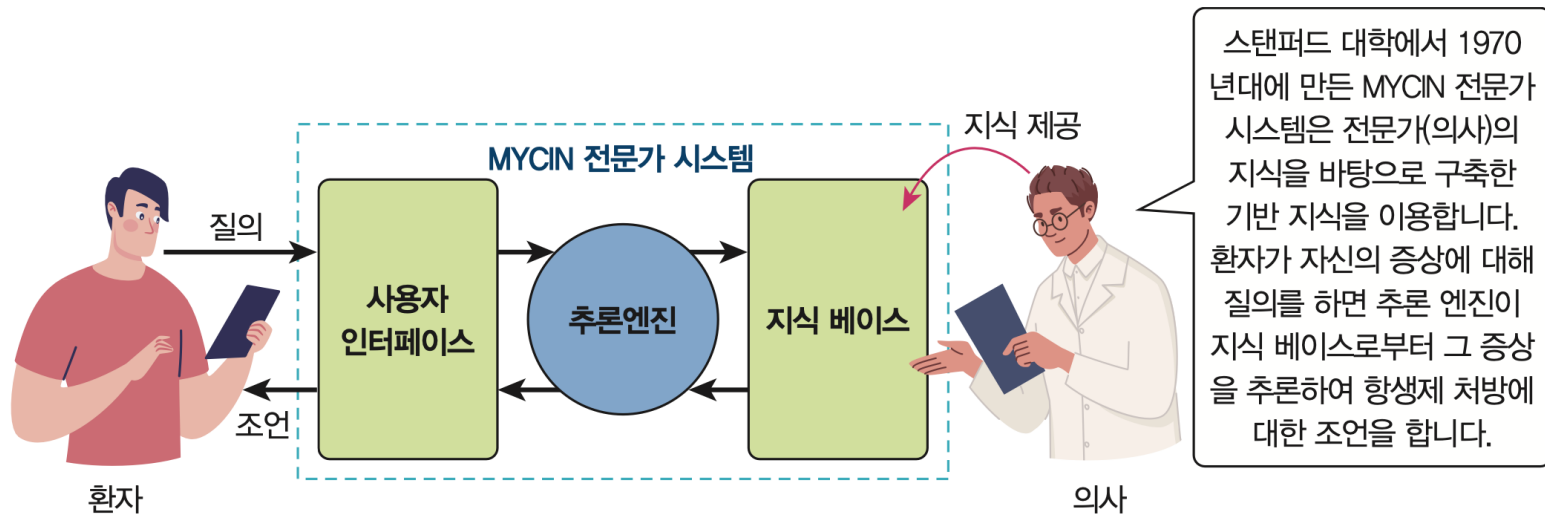
■ 최초의 전문가 시스템

- 어떤 유기 화합물의 구성요소와 그 화합물의 샘플로부터 얻어진 질량 스펙트로그램을 이용하여 그 화합물의 **분자구조를 결정**하는, 화학 전문가의 역할을 담당하는 전문가 시스템

■ 1970년대 초에 개발된 MYCIN

- **감염성 질병을 진단**하고, 이 진단을 바탕으로 항생제를 **처방**하며 그 추론을 자세히 설명할 수 있는 대화형 프로그램

03 규칙 기반 인공지능과 전문가 시스템



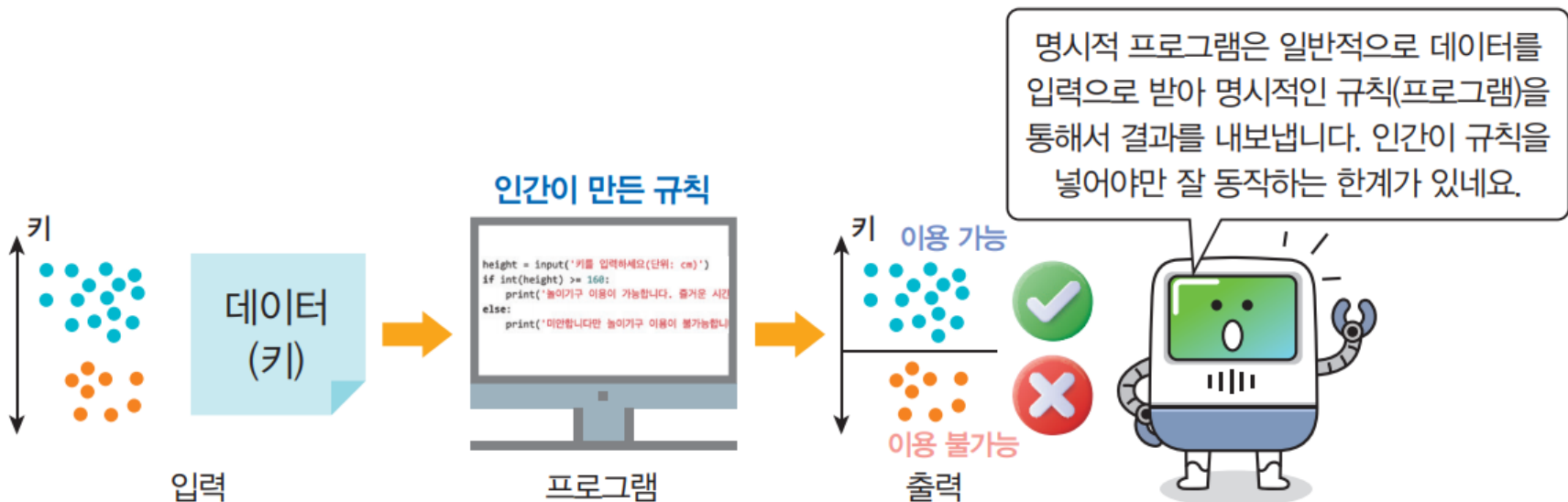
- 사용자 인터페이스: 사용자와 상호작용하는 방식을 제공
- 지식 베이스: 지식 저장공간으로서 사실과 규칙으로 구성
- 추론 엔진: 저장된 지식이 어떻게 활용될 수 있는가를 지원

04 명시적 프로그램의 한계와 이를 극복한 머신러닝

■ 명시적 프로그램 예

- 키를 cm 단위의 수치값으로 입력받아서 명시적인 규칙(프로그램)을 통해 결과를 내 보냄
- 이 때문에 반드시 인간이 규칙을 넣어야만 잘 동작하는 한계가 있다.

■ 이러한 명시적인 규칙이 매우 복잡할 경우 이를 일일이 프로그래밍하는 것이 어려운 경우가 많음



명시적 프로그래밍의 동작 방식

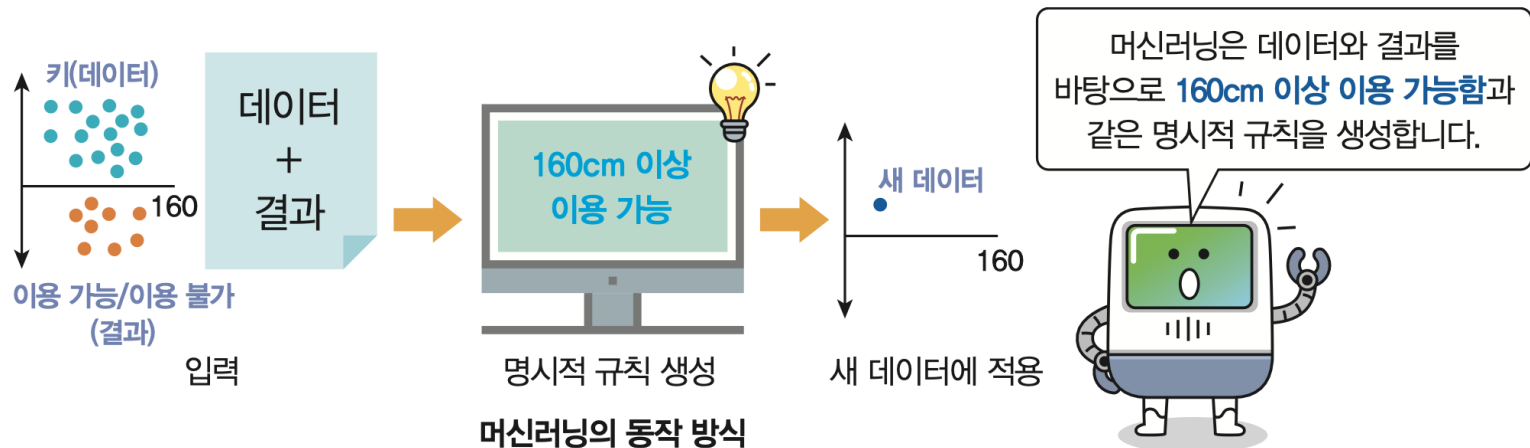
04 명시적 프로그램의 한계와 이를 극복한 머신러닝

■ 머신러닝의 예

- 여러 사람의 키를 입력으로 받고 각각의 값(데이터)과 함께 이용 가능 혹은 이용 불가능이라는 결과를 넘겨준 다음
- "키가 160cm 이상은 놀이기구 이용 가능"이라는 규칙을 컴퓨터가 생성할 수 있다면
- 이 프로그램은 명시적 프로그램에 비하여 그 적용 범위가 넓어질 수 있을 것

■ 이렇게 생성된 명시적 규칙은 새로운 데이터에 적용 가능

- 그림과 같이 160cm 보다 큰 값이 들어올 경우 "이용 가능"을 출력



04 명시적 프로그램의 한계와 이를 극복한 머신러닝

■ 머신러닝

- 데이터에서 학습을 통해 규칙을 찾는 프로그램
- 인간이 미처 발견하지 못한 패턴이나 규칙을 찾을 수도 있음
- 전통적인 프로그램보다 더 발전된 형태의 프로그램으로 볼 수 있음

■ 현재 다양한 분야에서 큰 성공을 거두고 있는 인공지능 기술

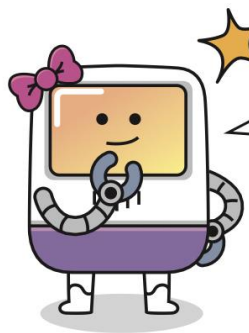
- 머신러닝(machine learning)이 핵심적인 기술이 되어 가고 있으며
- 앞으로 우리는 이에 관하여 상세하게 살펴볼 예정

명시적 프로그램의 한계

- 사람이 세부적인 규칙을 일일이 알려주어야 한다.
- 규칙이 너무 많아서 이를 표현 하기가 힘든 경우도 많다.
- 규칙이 변경되는 경우 빠르게 대응하기 힘들다.

머신러닝의 장점

- 데이터에서 학습을 통해 기계가 규칙을 찾는다.
- 인간이 발견하지 못한 패턴이나 규칙을 찾을 수도 있다.
- 새로운 데이터가 들어오면 그에 맞는 규칙을 만들어낸다.



머신러닝은 명시적 프로그램의 한계를 극복하고자 만들어진 기법입니다. 즉 데이터에서 규칙을 학습하는 능력을 가진 프로그램이지요.

LAB 05 : 블록 코딩을 익혀보자

- 실습목표 : 이번 실습에서는 블록 코딩을 통해서 간단하게나마 코딩의 원리를 익혀볼 것이다. 이 세상에는 수천 가지 이상의 프로그래밍 언어가 있으며 한 달에 하나씩을 배운다 하더라도 모두 다 익힐 수는 없을 것이다. 하지만 모든 프로그래밍 언어가 가지는 코딩의 기본적인 원리를 쉽게 익힐 수 있는 웹사이트가 있는데 바로 아래의 웹사이트이다.

<https://code.org/>

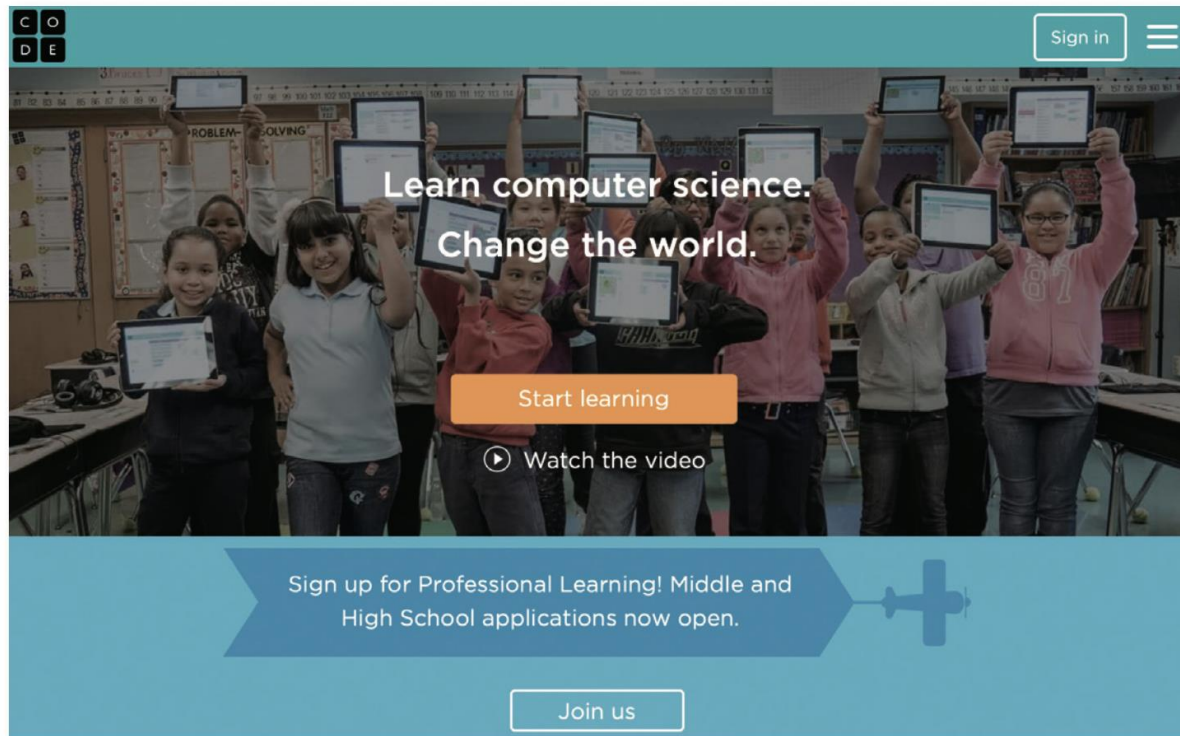
code.org의 특징은 다음과 같다.

- 비영리 교육단체인 code.org에서 운영하며 다양한 코딩 콘텐츠가 있다.
- 블록 코딩 방식을 사용하여 누구나 쉽게 코딩의 원리를 알 수 있다.
- 단계별 코딩을 할 수 있는 과정을 운영하고 있다.
- 코딩의 원리를 알려주는 동영상 콘텐츠가 제공된다.

LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

- 단계 1 : code.org 웹사이트에 접속한다.



LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

- 단계 2 : 화면 오른쪽 위의 Sign in 메뉴를 선택하여 로그인을 한다. 로그인 계정이 없을 경우 계정을 생성한다. 계정을 생성하기 위해서는 오른쪽과 같이 구글, 페이스북, 마이크로소프트 계정을 사용해서 아이디를 만들 수 있다.

The image shows the code.org login and sign-up interface. It is divided into two main sections by a dashed red line. The left section is for existing users, and the right section is for new users.

Left Section (Existing Users):

- Header:** Have an account already? Sign in
- Form:** Email or username (input field), Password (input field)
- Links:** [Forgot your password?](#)
- Buttons:** Sign in (orange), Create an account (blue)

Right Section (New Users):

- Header:** Enter your 6 letter section code
- Form:** Section Code (ABCDEF) (input field), Go (orange button)
- Buttons:** Continue with Google (dark red), Continue with Facebook (dark blue), Continue with Microsoft (dark blue)

Callouts:

- Left Callout:** code.org 계정이 있을 경우 이메일이나 사용자 이름과 비밀번호를 이용해서 로그인합니다.
- Right Callout:** code.org 계정이 없을 경우 구글, 페이스북, 마이크로 소프트 계정을 사용해서 아이디를 만듭니다.

OR

LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

- 단계 3 : 로그인을 한 후 다음과 같은 주소를 통해서 안나 엘사와 함께하는 코드 페이지로 이동하도록 하자.

<https://studio.code.org/s/frozen/lessons/1/levels/1>

만일 위의 페이지를 찾기가 어렵다면 메인 메뉴의 과정 카탈로그에서 Hour of Code 하위 학습 카테고리에는 **겨울왕국**을 찾아보도록 하자.

Hour of Code

모든 학습 코스를 해 볼 시간이 없다면, 모든 사람들을 위해 설계된 1시간짜리 학습 튜토리얼의 학생, 선생님들과 함께 참여해보세요.

메인 화면의 과정 카탈로그 아래를 찾아보면 **Hour of Code**라는 학습 카테고리가 있습니다. 여기서 찾아보세요.

아워오브코드 튜토리얼 더 보기 >

세계 180 개가 넘는 나라의 수 백만 명 이상



댄스 파티

Featuring Katy Perry, Shawn Mendes, Panic! At The Disco, Lil Nas X, Jonas Brothers, Nicki Minaj, and 34 more!



마인크래프트

창의성, 문제해결 능력, 코드를 사용해 언더워터 월드를 탐험하고 만들어 보세요!



겨울왕국

얼음의 마술과 아름다움을 탐구하는 안나, 엘사와 함께 코드를 사용하세요.



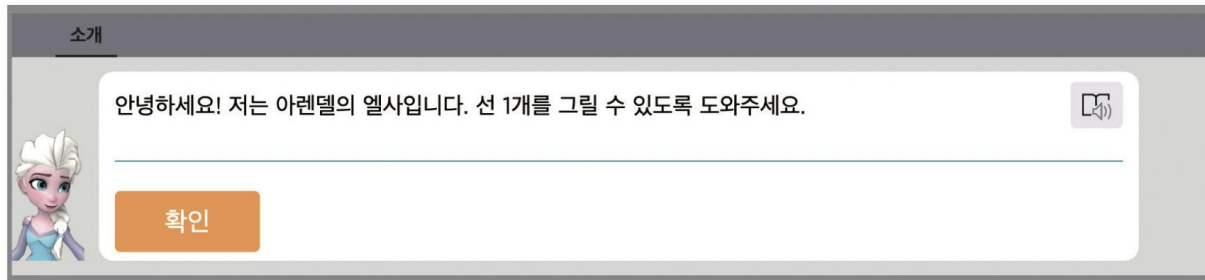
더 보기

아워오브코드 튜토리얼 더 보기

LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

- 단계 4 : 이 과정을 시작하면 사용법을 안내하는 영상과 다음과 같이 수행 작업을 알려주는 창이 나타난다. 첫 번째 과정은 선 1개를 그리는 간단한 미션이다.



LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

- 단계 5: 이 과정의 화면을 살펴보면 왼쪽에는 블록 코드의 수행 결과를 볼 수 있으며 실행 버튼으로 블록 코드를 실행시킬 수 있다. 상세한 설명은 아래의 풍선 글과 화면을 참조하라.

The image shows a block coding interface with several callout boxes explaining its features:

- 블록 코드의 수행 결과를 볼 수 있음** (You can see the execution results of the block code): Points to the stage area on the left where a character is visible.
- 블록 코드를 실행시킬 수 있다. 슬라이드 막대가 토끼 쪽으로 갈수록 실행 속도가 빠르다.** (You can execute the block code. The execution speed increases as the slider moves towards the rabbit): Points to the '실행' (Run) button and the slider below it.
- 블록을 드래그하여 작업공간으로 이동할 수 있다. 블록의 왼쪽 위의 홈이 다른 블록과 맞아야 한다.** (You can move the block to the workspace by dragging it. The home on the top left of the block must match the other blocks): Points to the '앞으로' (Forward) block in the block palette.
- 작업공간으로 블록의 상태에 의해서 명령이 수행된다.** (The command is executed in the workspace according to the state of the block): Points to the '실행하면' (When clicked) block in the workspace.

The interface includes a '소개' (Intro) section with a character and a message: '안녕하세요! 저는 아렌델의 엘사입니다. 선 1개를 그릴 수 있도록 도와주세요.' (Hello! I am Elsa from Arendelle. Please help me draw a line). Below this is a '블록' (Blocks) palette with three blocks: '앞으로' (Forward) with '다음 만큼 이동: 100 pixels', '오른쪽으로' (Turn right) with '다음 만큼 돌기: 90 도', and '왼쪽으로' (Turn left) with '다음 만큼 돌기: 90 도'. The workspace shows '작업공간: 1 / 2 블록' (Workspace: 1 / 2 blocks) and buttons for '다시 시작하기' (Restart) and '</> 코드 보기' (View code).

LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

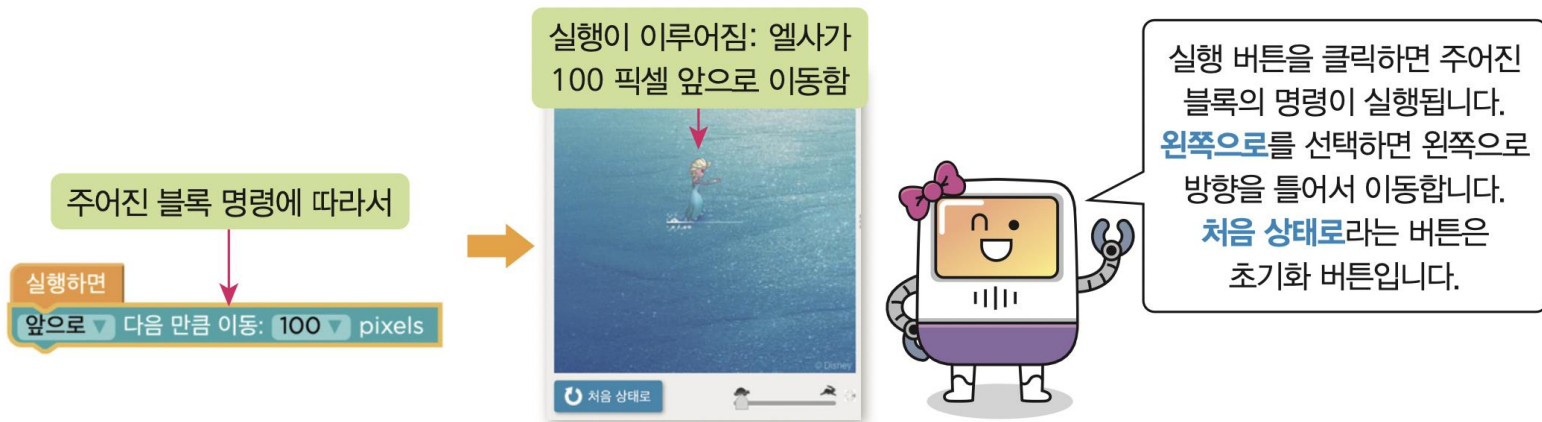
- 단계 6 : 이제 블럭(블록) 영역에 있는 블록 중에서 '앞으로 다음 만큼 이동 : 100 pixels' 블록을 드래그해서 '실행하면' 블록의 아래에 가져다 놓도록 한다.



LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

- 단계 7 : 100 pixels의 화살표를 선택하면 50/100/150/200/300 pixel의 선택 가능한 이동 범위가 나타난다. 디폴트로 표시된 100 pixels를 선택한 다음 실행을 클릭하자. 엘사가 **100 픽셀 앞으로** 이동하는 것을 볼 수 있을 것이다. 만일 앞으로 대신 왼쪽으로를 선택했다면 엘사는 **왼쪽으로** 방향을 틀어서 이동할 것이다.



- 화면 하단의 실행을 선택하면 실행 버튼은 사라지고 '처음 상태로'라는 버튼이 나타난다. 이 버튼은 초기화 버튼으로 모든 실행이 중지된다.

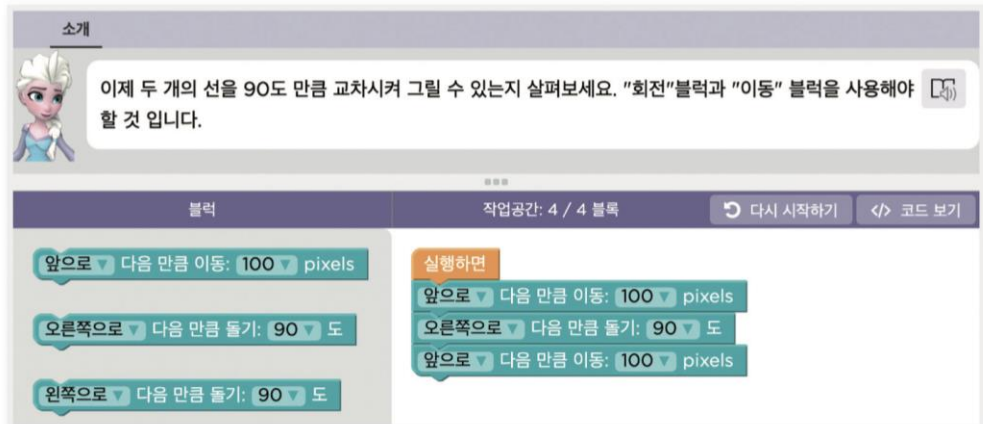
LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

- 단계 8 : 이제 다음과 같은 설명 창이 나타난다. 계속하기 버튼을 선택하면 2단계로 이동한다.



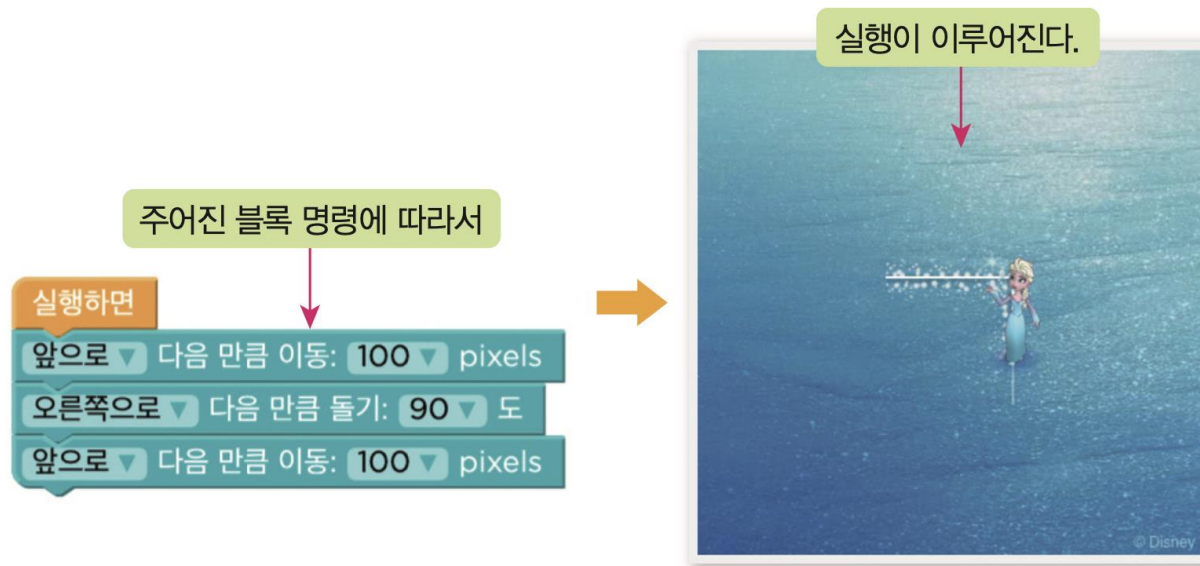
- 단계 9 : 2단계에서는 이전 단계에 비해서 할 일이 좀 늘어났다. 하지만 다음과 같이 블록을 잘 배치시킨다면 앞으로 100픽셀만큼 이동한 후 오른쪽으로 90도를 돌아서 다시 앞으로 100픽셀만큼 이동하는 결과를 얻을 수 있을 것이다.



LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

- 단계 10 : 실행버튼을선택하면다음과같이 ㄱ자모양의직선을그리는것을볼수있다.단계 5-6에서와 같이 주어진 블록 명령을 수행하는 것을 볼 수 있다.



LAB 05 : 블록 코딩을 익혀보자

■ 실습하기

- 단계 11 : 이와 같은 방법으로 각 단계를 수행하게 되면 상단에 다음과 같은 녹색 기호가 하나씩 생겨날 것이다. 20단계에서는 임의의 그림을 그려볼 수 있으며 최종적으로 모두 20개의 녹색 기호가 화면 상단에 채워지면 미션이 완료된 것이다. 그림에 나타난 것과 같이 진한 분홍색의 ㄷ자 모양 블록은 반복적으로 일을 수행하는 블록이다. 그림에서는 반복 3번으로 표시되어 있으므로 패턴이 3개 나타나 있다. 녹색의 원 만들기 버튼은 크기를 지정하여 원을 그린다. 다음으로 색을 설정하는 블록도 있으며 이들을 잘 조립하면 재미있는 무늬를 만들 수 있다.

