

AI
첫걸음

인공지능 부산을 알려줘

우리 지역 문화로 배우는
자유학기제 AI 교재

부산의 상징으로
배우는
엔트리 AI

기획위원

김광수 부산광역시교육청 교육국장

이재한 부산광역시교육청 미래인재교육과장

은윤주 부산광역시교육청 지능정보교육팀 장학관

손제득 부산광역시교육청 지능정보교육팀 장학사

집필위원

김경민 | Kim KyoungMin

컴퓨팅 사고력과 문제해결력을 기르기 위한 학습에 대해 관심을 가지고 연구하고 있다. 현재 부산대학교 소프트웨어교육센터에서 근무하며 비전공자를 위한 인공지능과 데이터분석 교육에 활발히 활동하고 있다. pnumin@pusan.ac.kr

김지연 | Kim JiYeon

논리적 사고력과 창의적 문제 해결 기법을 기를 수 있는 코딩 교육에 대해 관심이 많아 다양한 소프트웨어 교육에 대해 연구하고 있으며, 웹디자인 및 프론트-엔드 개발에도 관심이 많다. 현재 신라대학교 컴퓨터 소프트웨어공학부에 재직 중이다. kimjy@sillain.ac.kr

부산시교육선도지원단

전국 최초 초·중·고·대학 학교급별 SWAI교육 교육과정 체계화, 산학관 연계 협력 강화, 교육자원 확보 방안 등을 위한 2018년부터 구성된 부산소속 교원 연구회이며, 활동을 위한 가입 문의는 helpsys@korea.kr로 연락주시기 바랍니다.

swedu-2201

부산의 상징으로 배우는

엔트리 시

인쇄일 2021년 12월 27일

발행일 2022년 1월 3일

저자 김경민, 김지연

발행인 부산광역시교육청

47119 부산광역시 부산진구 화지로 12

문의: 부산광역시교육청 미래인재교육과

전화 051) 860-0373 / 팩스 051) 860-0359

이메일 helpsys@korea.kr / 홈페이지 www.pen.go.kr

디자인 도서출판 어가

이메일 uhga@uhga.co.kr / 홈페이지 www.uhga.co.kr

등록 2016년 7월 4일 · 제2016-000133호

본 (교재/자료)는 엔트리의 교육 자료를 CC-BY 2.0 라이선스에 따라 재구성하여 사용하였습니다.

copyright©2022 by Busan Metropolitan City office of Education, Writer, PNU Software Education Center. All right reserved.

본 저작물의 출판권은 부산광역시교육청, 저자, 부산대학교 소프트웨어교육센터가 공동 소유합니다.

저작권법에 의해 보호를 받는 저작물이므로 무단전제와 복제를 금합니다.

ISBN 979-11-91109-50-4 (94000)

979-11-91109-49-8 (세트)

비매품



AI
첫걸음

인공지능 부산을 알려줘

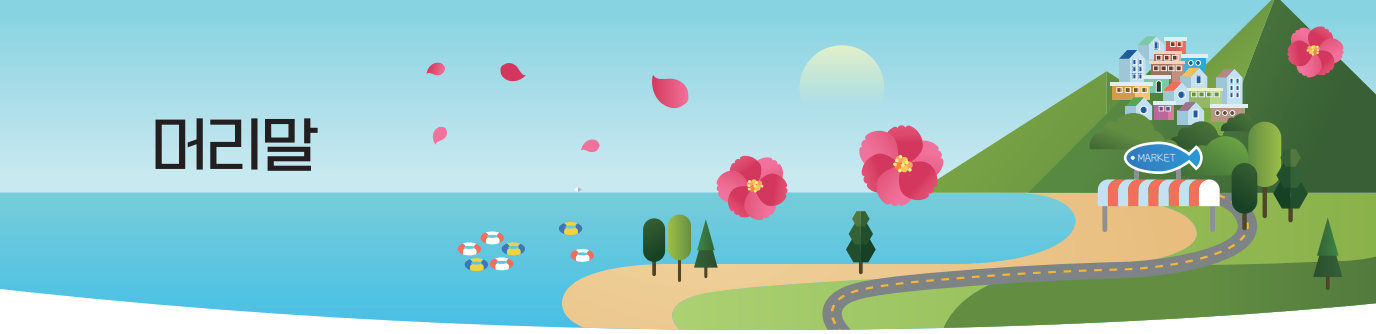
우리 지역 문화로 배우는
자유학기제 AI 교재

부산의 상징으로
배우는
엔트리 AI



본 교재는 2021년 부산광역시교육청, 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학지원사업의 지원에 의해 발간되었음(2016-0-00019)

This textbook was supported by Busan Metropolitan City office of Education, the MIST(Ministry of Science and ICT), Korea, under the National Program for Excellence in SW supervised by the IITP(Institute of Information & communications Technology Planing & Evaluation) (2016-0-00019)



디지털 대전환 시대를 맞이하여 학교 교육을 통한 소프트웨어와 인공지능 교육에 대한 중요성이 높아지고 있습니다. 소프트웨어와 인공지능 교육은 단순한 프로그래밍 작성 교육이 아니라 컴퓨팅 사고력을 기반으로 문제해결 능력을 기르는 교육입니다. 컴퓨팅 사고력은 실생활에서 발생하는 복잡한 문제를 단순한 문제로 분해하여 추상화한 후 컴퓨터를 활용하여 효율적으로 해결해 보는 과정으로 창의적 문제 해결 역량을 기를 수 있습니다. 이러한 역량은 모든 학생들이 공교육을 통해 기를 수 있어야 합니다.

본 교재는 부산광역시 교육청과 부산 지역 SW중심대학들이 자유학기제를 통해 인공지능 교육을 지원하기 위해 개발한 교재입니다. 부산 지역의 상징, 축제 및 사회 현상에 관련된 부산 지역의 문제들을 프로젝트 활동 중심으로 해결해 볼 수 있도록 구성하였습니다. 학생들은 블록 코딩으로 인공지능 기술을 학습하면서 지역의 문화와 사회문제를 해결해 볼 수 있습니다.

이 책에서는 엔트리를 활용하여 인공지능 개념을 학습하고, 우리 지역의 상징을 통해 지역 문화를 이해할 수 있는 프로젝트 활동을 수행합니다. 각 단원은 초보자들이 쉽게 프로그래밍과 인공지능을 학습할 수 있도록 주제별로 구성하였습니다.

본 교재를 활용하여 자유학기제를 통해 학생들이 자기주도적으로 소프트웨어와 인공지능 교육을 체험해 봄으로써 컴퓨팅 사고력을 향상시킬 수 있을 것으로 기대합니다.

차례



머리말 5

1장

부산의 상징을 알려줘

9

- 1 인공지능(AI)이란 무엇일까요? 11
- 2 해결 문제: 부산의 상징을 알려줘 12
- 3 엔트리로 만들어 봅시다. 12

2장

부산의 상징을 보여줘

23

- 1 음성 인식과 음성 합성이란 무엇일까요? 25
- 2 해결 문제: 부산의 상징을 보여줘 26
- 3 엔트리로 만들어 봅시다 27

3장

부산의 랜드마크를 알아보자

37

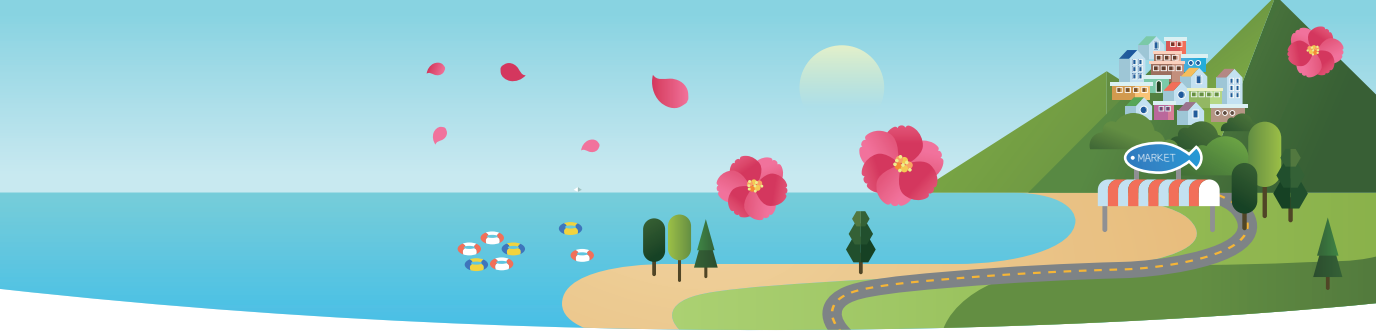
- 1 컴퓨터 비전이란 무엇일까요? 39
- 2 해결 문제: 부산의 랜드마크를 알아보자 40
- 3 엔트리로 만들어 봅시다. 41

4장

부산의 랜드마크를 잡아보자

49

- 1 얼굴 감지와 얼굴 인식은 어떤 차이가 있을까요? 51
- 2 해결 문제: 부산의 랜드마크를 잡아보자 52
- 3 엔트리로 만들어 봅시다. 52



5장

부산의 날씨는 어때

61

- 1 전문가 시스템? 63
- 2 해결 문제: 부산의 날씨는 어때 64
- 3 엔트리로 만들어 봅시다. 64

6장

부산의 여행지를 찾아보자

73

- 1 의사결정트리 75
- 2 해결 문제: 부산의 여행지를 찾아보자 76
- 3 엔트리로 만들어 봅시다. 76

7장

여기가 부산의 어디야

87

- 1 머신러닝 89
- 2 해결 문제: 여기가 부산의 어디야 90
- 3 엔트리로 만들어 봅시다. 90

8장

부산의 여행지를 추천해줘

103

- 1 데이터의 편향성 105
- 2 해결 문제: 부산의 여행지를 추천해줘 106
- 3 엔트리로 만들어 봅시다 106

부산의 상징으로 배우는 **엔트리 시**

1장 부산의 상징을 알려줘





부산의 상징으로 배우는
엔트리 AI



학습 목표

- 인공지능이 무엇인지 알아봅시다.
- 부산의 상징을 알려주는 인공지능 스피커를 만들어 봅시다.

1 인공지능(AI)이란 무엇일까요?

사람의 지능을 흉내 내는 소프트웨어, 프로그램, 인공적인 장치를 ‘인공지능’이라고 합니다. 인공지능은 프로그래밍된 대로 결과를 만드는 것이 아니고 사람이 생각하고 판단해서 결과가 나오는 것처럼 스스로 생각하고 판단할 수 있어야 합니다.

인공지능이라는 용어가 만들어진 것은 1956년 미국 다트머스 학회에서 처음 만들어져서 지금까지 사용되고 있습니다. 전통적인 인공지능은 사람이 가지고 있는 지식을 어떤 규칙의 형태로 만들어 놓고 문제가 발생하면 그 지식을 사용하여 해결했습니다. 이러한 인공지능 시스템을 전문가 시스템이라고 합니다. 사람의 지식을 규칙 형태로 만드는 것은 매우 어려운 일입니다.

최근의 인공지능 시스템은 데이터를 학습하여 유의미한 규칙을 찾아내고 센서를 통해 입력되는 데이터를 처리하여 사람이 사물을 분별하고 판단하는 것처럼 인식할 수 있습니다.

에어컨을 예로 들어 보겠습니다. 에어컨의 기능 중에 에어컨 꺼짐을 예약하는 기능이 있습니다. 예약 기능을 설정하면 사용자가 정해진 시간에 에어컨이 꺼지도록 프로그램이 작성되어 있습니다.

최신 에어컨은 인공지능이 탑재되어 있습니다. 인공지능이 탑재된 에어컨은 사용자가 에어컨을 작동하는 패턴을 학습하여 스스로 희망 온도를 설정하고, 설정된 온도가 되면 자동으로 작동되거나 사용자의 음성을 인식해서 작동시킬 수 있습니다.

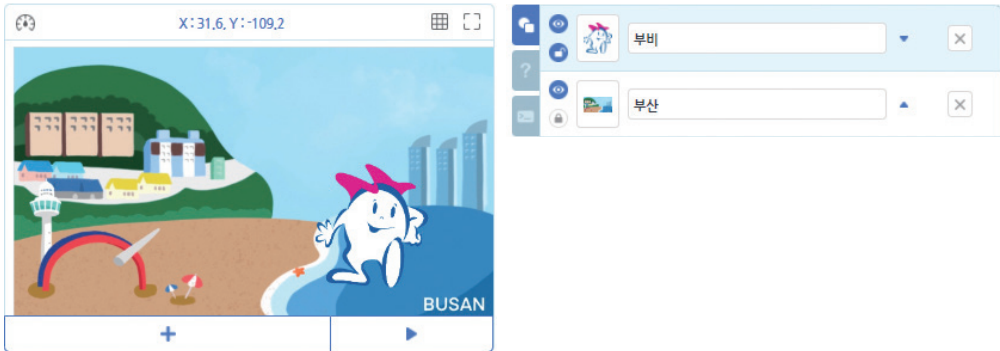
사람의 음성을 인식하여 명령을 수행하고 결과를 문장으로 읽어주는 스피커도 인공지능 기술이 탑재된 제품입니다.

이제 엔트리를 활용해서 부산의 상징을 알려주는 인공지능 스피커를 만들어 보도록 하겠습니다.



2

해결 문제: 부산의 상징을 알려줘



부산의 상징을 알아봅시다.¹ 부산의 시화는 동백꽃, 시조는 갈매기, 시어는 고등어입니다. 부산의 마스코트²는 부비입니다.

음성 인식을 통해 부비를 호출해서 부산의 상징인 동백꽃, 갈매기, 고등어가 순차적으로 소개될 수 있도록 작성해 보세요. 각 상징이 화면에 보일 때마다 컴퓨터가 읽어줄 수 있도록 작성해 보세요.

3

엔트리로 만들어 봅시다.

1) 엔트리 접속하기

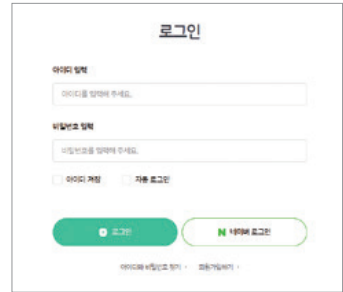
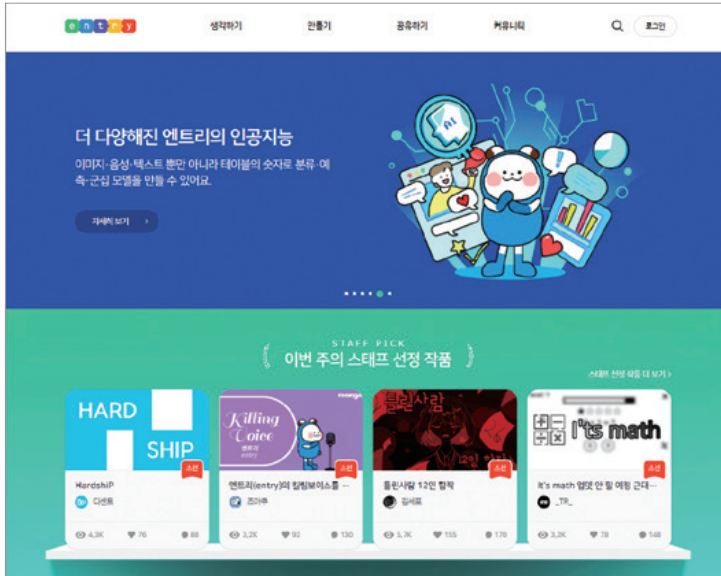
엔트리(playentry.org)는 네이버 커넥트재단에서 비영리로 운영하는 온라인 소프트웨어 교육 플랫폼으로 별도의 프로그램 설치 없이 웹브라우저에서 드래그 앤 드롭(Drag-and-drop) 방식의 블록 코딩을 지원합니다.

엔트리는 회원가입을 하지 않아도 누구나 무료로 이용할 수 있지만, 회원가입을 하면 내가 만든 작품을 저장할 수 있고, 친구들의 작품에 댓글을 달거나 '좋아요'를 눌러 줄 수 있으며, '학습하기'에서 내 학습 과정을 저장할 수 있습니다.

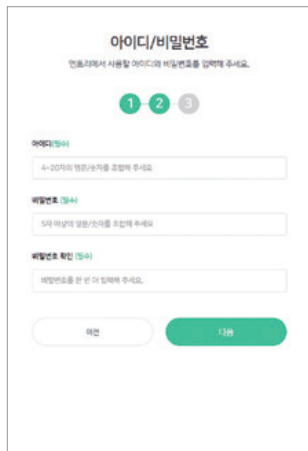
¹ <https://www.busan.go.kr/bhareasymbol>

² <https://www.busan.go.kr/bhorgsymbol01>

구글 크롬 브라우저를 실행하고 주소창에서 <https://playentry.org>를 입력합니다. 우측 상단의 [로그인] 버튼을 누릅니다. 로그인 화면의 아래쪽에 [회원가입하기]를 누릅니다.

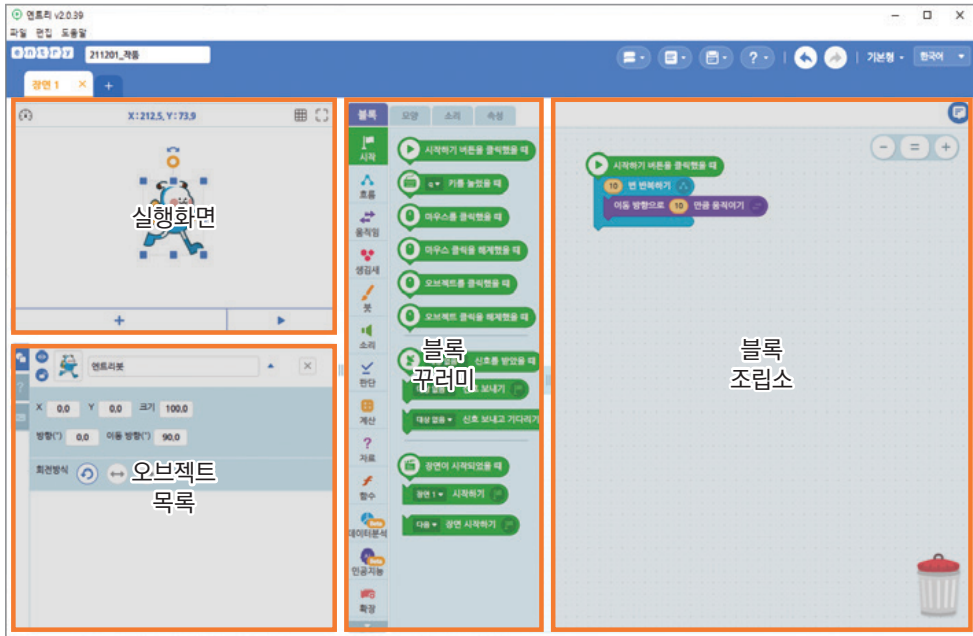


다음 단계별로 자료를 입력합니다.



2) 엔트리 화면 살펴보기

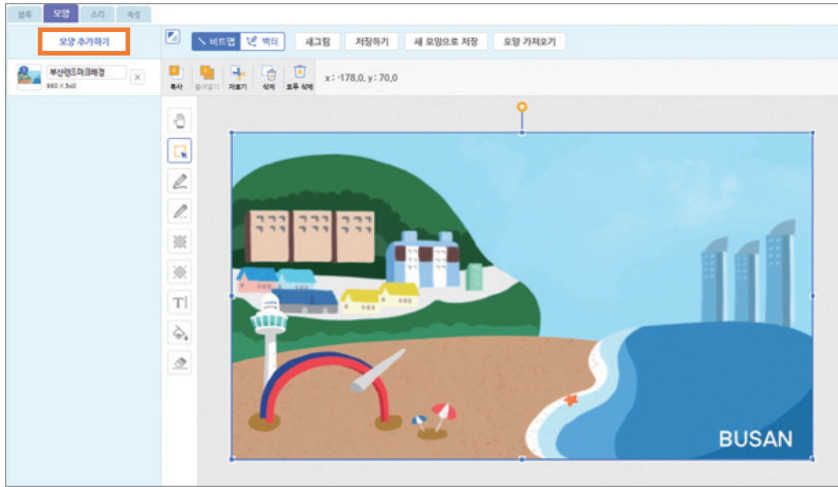
로그인 후 상단의 [만들기] 메뉴를 선택합니다. 엔트리 [만들기]에서는 게임, 애니메이션, 응용프로그램 등 다양한 작품을 만들 수 있습니다. 엔트리 [만들기] 화면은 아래와 같이 구성되어 있습니다.



3) 화면 구성하기

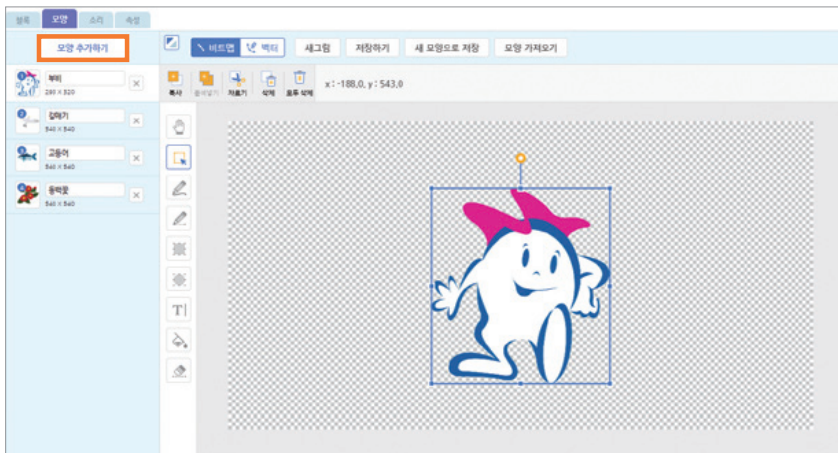
(1) 배경 추가하기

오브젝트를 추가하여 [모양 추가하기] 버튼을 눌러 [배경] 항목을 선택하고 [파일 올리기]를 이용하여 배경을 선택합니다. 파일 올리기를 통해 배경을 추가할 때는 엔트리에서 제공하는 배경을 하나 선택한 후 파일 올리기를 하면 배경에 맞게 작성됩니다.



(2) 부비 오브젝트

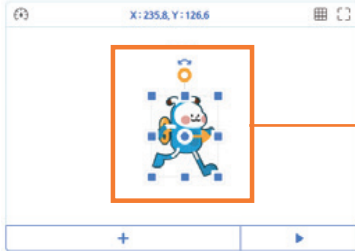
오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 [파일 올리기]를 이용하여 부비, 동백꽃, 갈매기, 고등어 모양을 선택합니다. 모양을 추가한 후에 첫 번째 모양을 선택하면 화면에 첫 번째 모양이 보입니다.





오브젝트 알아보기

오브젝트란, 실행 화면에서 보이는 다양한 캐릭터, 배경, 그림, 글상자 등을 말합니다. 프로그램에서 사용되는 오브젝트의 목록을 표시하며, 오브젝트의 다양한 값들(속성)을 설정합니다.

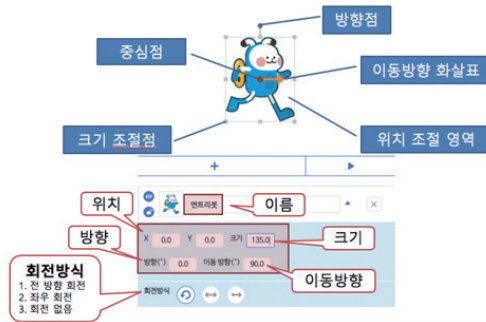


실행화면에서의 오브젝트



오브젝트 목록 - 다양한 값들

오브젝트는 이름, 위치, 크기, 방향, 이동 방향, 회전 방식의 정보를 가지고 있습니다. 이러한 정보들은 오브젝트의 핸들러를 이용하여 바꾸거나, 연필 모양의 수정 버튼을 눌러 직접 입력할 수 있습니다.



4) 인공지능 블록 불러오기

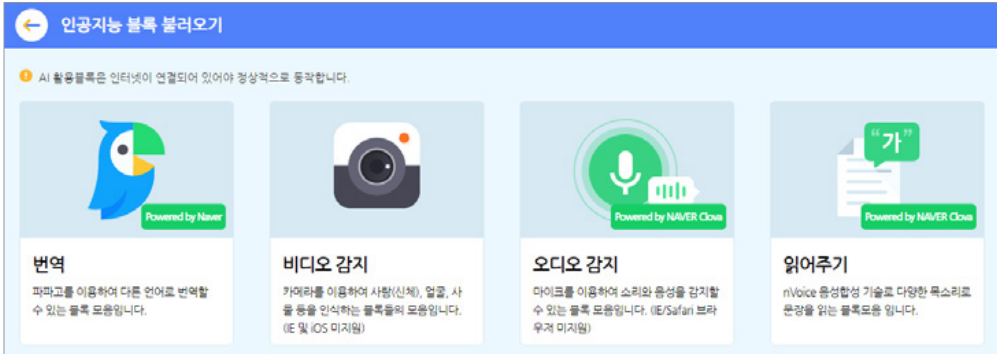
블록 꾸러미에서 블록 탭의 인공지능을 선택하고 [인공지능 불러오기] 버튼을 눌러 '오디오 감지'와 '읽어주기'를 선택합니다.





엔트리 인공지능 블록 알아보기

엔트리의 인공지능 기능을 이용하면, 실생활에 적용되는 인공지능 기술을 체험하고 활용할 수 있습니다.



■ 번역

네이버 파파고를 이용하여 입력된 글자의 언어를 감지하고, 11개 국가의 언어로 번역할 수 있습니다.

■ 비디오 감지

컴퓨터에 장착된 카메라를 이용해 비디오 화면을 시작하면, 사람(신체), 얼굴, 사물을 인식할 수 있습니다. 인식한 사람, 얼굴, 사물의 x좌표와 y좌표를 이용하여 다양한 블록과 조합해 작품을 제작할 수 있습니다.

■ 오디오 감지

컴퓨터에 장착된 마이크를 이용해서 음성을 인식할 수 있습니다. 인식한 음성을 다양한 블록과 조합해 작품에서 활용할 수 있습니다.

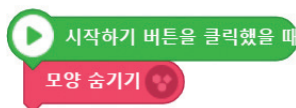
■ 읽어주기(TTS)

음성 합성 기술을 통해 입력한 글자를 사람, 캐릭터 등 다양한 목소리로 읽어주는 기능입니다.

5) 프로그램 설계하기

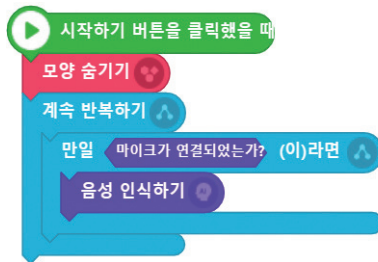
① '시작하기'를 누르면 모양을 숨깁니다.

'부비' 오브젝트를 클릭하여 '시작' 블록 꾸러미와 '생김새' 블록 꾸러미에서 블록을 선택하여 프로그램이 시작되면 '부비' 오브젝트가 보이지 않도록 합니다.



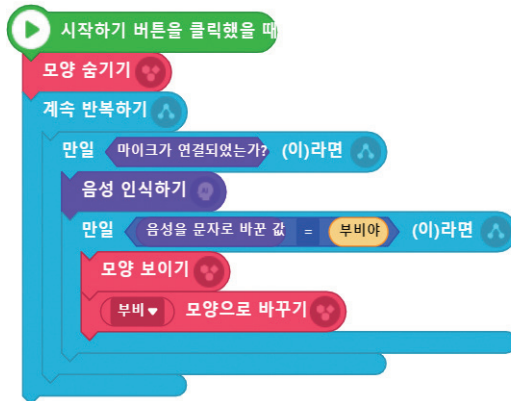
② 마이크를 인식하고 음성을 인식합니다.

프로그램이 실행되는 동안 계속해서 음성이 인식되도록 '흐름' 블록 꾸러미의 '계속 반복하기' 블록을 사용합니다. 마이크가 연결되었는지 확인하기 위해 '흐름' 블록 꾸러미의 '만일 ~라면' 블록으로 확인합니다. '인공지능' 블록 꾸러미의 오디오 감지에는 '마이크가 연결되었는가'를 확인하는 블록이 있습니다. 마이크가 연결되면 '음성 인식하기' 블록을 이용하여 음성을 인식합니다.



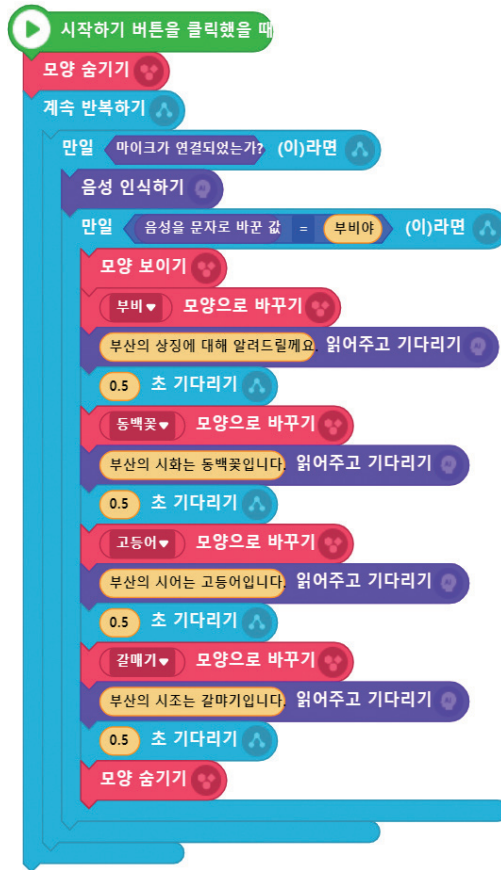
③ 음성을 인식하면 '부비' 오브젝트가 보이도록 합니다.

프로그램이 시작되면 '부비' 오브젝트가 보이지 않도록 작성했었습니다. 이제 음성을 인식합니다. 인식된 음성이 '부비야'인지 확인합니다. 인식된 음성이 '부비야'이면 '부비' 오브젝트가 보이도록 작성합니다. 이때, '부비' 오브젝트는 여러 개의 모양을 가지고 있으므로 '부비' 모양으로 바꾸어 줍니다. 모양과 관련된 블록은 '생김새' 블록 꾸러미에 있습니다.



④ 부산의 상징에 대해 알려줍니다.

음성 인식이 되면 순차적으로 '부비' 오브젝트가 부비, 동백꽃, 고등어, 갈매기로 변경합니다. 모양이 변경될 때마다 부산의 상징에 대해 음성으로 알려줍니다.

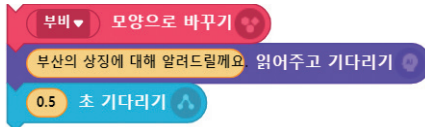


알고리즘에 대해 알아보기

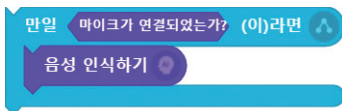
프로그램을 작성하는 것은 컴퓨터에게 일을 시키는 것입니다. 일을 시키기 위해 명령어를 순서대로 나열합니다. 이렇게 일을 시키는 명령 순서를 알고리즘이라고 합니다.

알고리즘은 순차 구조, 선택 구조, 반복 구조가 있습니다.

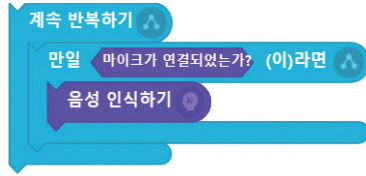
- 순차 구조는 명령을 위에서 아래로 순서대로 실행하도록 명령을 내리는 것입니다.



- 선택 구조는 주어진 조건에 따라 명령을 선택적으로 실행하도록 명령을 내리는 것입니다.

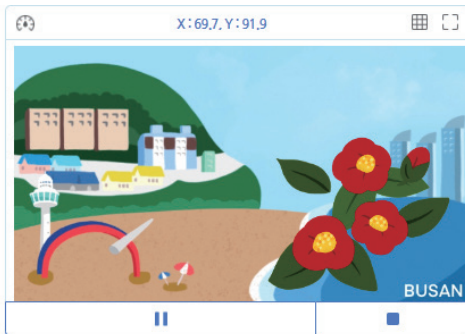
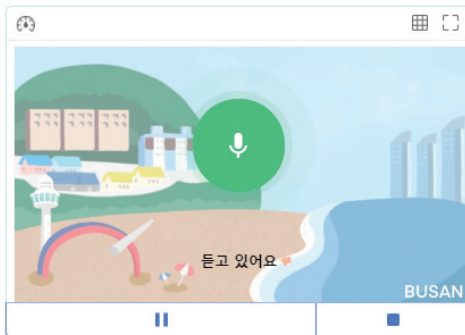


- 반복 구조는 특정 횟수나 조건을 만족할 때까지 반복하도록 명령을 내리는 것입니다.



우리가 만들어 본 프로그램에서 순차 구조, 선택 구조, 반복 구조를 찾아보세요.

6) 결과 확인하기 눌러 로그인합니다.





자신만의 해결 문제를 만들어 보세요.



해결 문제 정의하기

Blank writing area with horizontal dashed lines for defining the problem.



해결 문제 작성하기

Blank writing area with horizontal dashed lines for writing the solution.

2장

부산의 상징을 보여줘





부산의 상징으로 배우는
엔트리 AI



학습 목표

- 음성 인식과 자연어 처리에 대해 알아봅시다.
- 부산의 상징을 말하면 상징을 보여주는 프로그램을 만들어 봅시다.

1 음성 인식과 음성 합성이란 무엇일까요?

앞서 엔트리를 활용해서 부산의 상징을 알려주는 인공지능 스피커를 만들어 보았습니다. 인공지능 스피커는 음성 인식을 통해서 스피커에 내장된 음성비서가 사람의 대화를 이해하고 기능을 수행합니다.

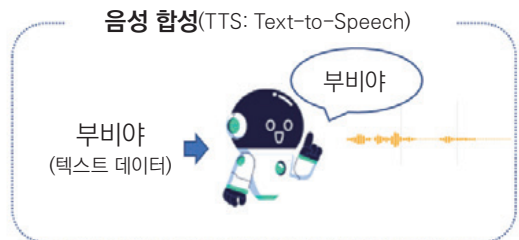
이처럼 컴퓨터가 사람과 소통할 수 있게 하는 기술을 자연어 처리(NLP: Natural Language Processing)라고 합니다. 자연어는 우리가 일상생활 속에서 말하고 듣고 읽고 쓰기를 하는 언어로 자연어 처리는 인공지능의 한 분야입니다.

자연어 처리를 이용하면 인식된 내용이 긍정인지 부정인지 분류할 수 있는 감성 분석을 할 수 있고, 구글 번역이나 파파고와 같은 번역기를 만들 수 있고, 인공지능 스피커나 챗봇 같은 대화 시스템 및 자동 응답 시스템을 만들 수 있습니다.

컴퓨터가 어떻게 사람의 말을 듣고 이해해서 답을 할 수 있을까요?

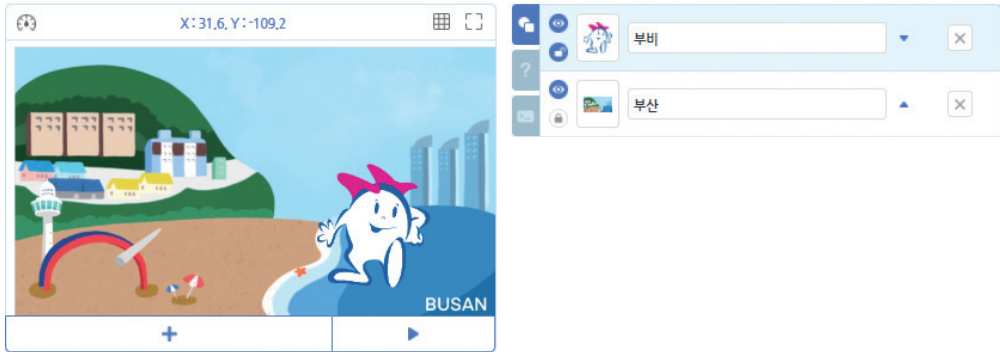
음성 인식 기술은 컴퓨터가 마이크와 같은 소리 센서를 통해서 소리 신호를 추출하고 잡음을 제거한 후, 이 신호로부터 특징을 추출합니다. 이렇게 추출된 특징으로 소리의 파동을 디지털 신호로 기록해 놓은 음성 모델 데이터베이스와 인공지능 기술을 사용하여 비교하여 글자 데이터로 변환합니다.

음성 합성 기술은 컴퓨터 글자를 음성으로 변환하는 기술입니다. 음성 합성 기술은 인공지능 기술을 통해 사람의 목소리를 학습하여 단순하게 기계음 방식으로 변환하던 방식에서 인공지능 기술을 이용해서 사람처럼 자연스럽게 말할 수 있도록 발전하고 있습니다.



2

해결 문제: 부산의 상징을 보여줘



앞 장에서는 부산의 상징을 순서대로 알려주는 프로그램을 작성했습니다. 이번 장에서는 음성 인식을 통해 인식된 글자에 따라 부산의 상징인 동백꽃, 갈매기, 고등어를 보여주도록 작성해 보세요. 또한, 동백꽃, 갈매기, 고등어가 아닌 경우는 '무슨 말인지 모르겠어요'라고 말하도록 해 보세요.

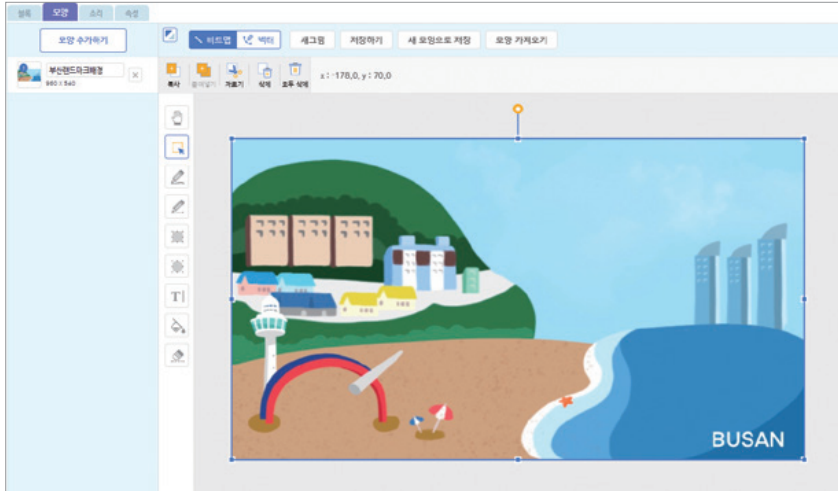
여러분이 '동백꽃'이라고 말하면 컴퓨터가 '동백꽃'으로 변환해 주는 기술이 음성 인식 기술이고 동백꽃에 대한 설명을 음성으로 읽어주는 기술이 음성 합성 기술입니다. 엔트리는 인공지능 블록에서 '오디오 감지'와 '읽어주기' 블록을 사용하여 구현할 수 있습니다.



3 엔트리로 만들어 봅시다

1) 화면 구성하기

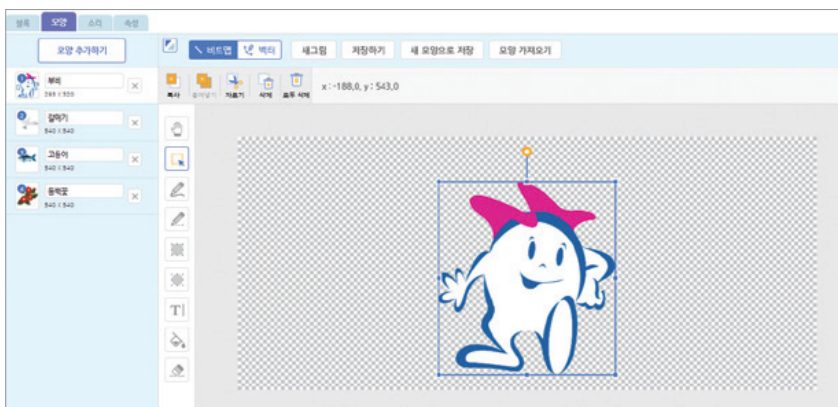
(1) 배경 추가하기



오브젝트를 추가하여 [모양 추가하기] 버튼을 눌러 [배경] 항목을 선택하고 파일 올리기를 이용하여 배경을 선택합니다.

(2) 부비 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 부비, 동백꽃, 갈매기, 고등어 모양을 선택합니다. 모양을 추가한 후에 첫 번째 모양을 선택하면 화면에 첫 번째 모양이 보입니다.



2) 인공지능 블록 불러오기

블록 꾸러미에서 블록 탭의 [인공지능]을 선택하고 [인공지능 불러오기] 버튼을 눌러 [오디오 감지]와 [읽어주기]를 선택합니다.



3) 프로그램 설계하기

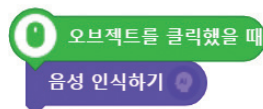
① '시작하기'을 누르면 모양을 숨기고 음성 인식이 가능해지면 '부비' 오브젝트가 보이게 합니다.

'부비' 오브젝트를 클릭하여 '시작' 블록 꾸러미와 '생김새' 블록 꾸러미에서 블록을 선택하여 프로그램이 시작되면 '부비' 오브젝트가 보이지 않도록 합니다. 마이크가 연결되었는지 확인하기 위해 '인공지능' 블록 꾸러미의 오디오 감지에는 '마이크가 연결되었는가' 블록을 이용하여 '생김새' 블록 꾸러미에서 '모양으로 바꾸기' 블록을 이용하여 '부비' 오브젝트의 모양을 '부비'로 바꾸어 줍니다.



② '부비' 오브젝트를 클릭하면 음성 인식을 합니다.

'부비' 오브젝트가 클릭이 되면 음성이 인식되도록 '음성 인식하기' 블록을 이용하여 음성을 인식합니다.



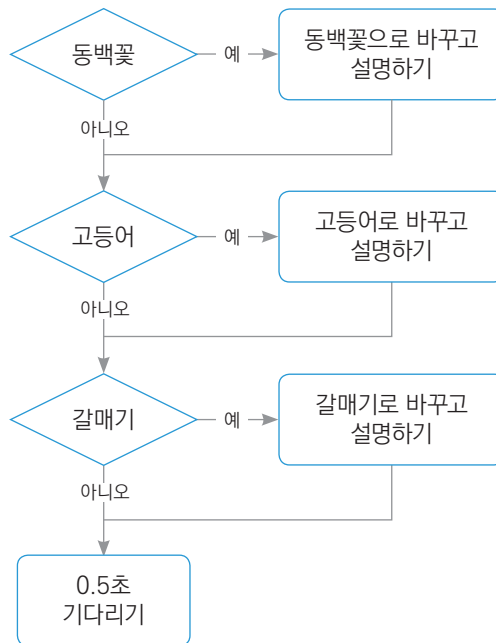
③ 음성을 인식하면 인식된 글자가 무엇인지 확인합니다.

인식된 글자가 무엇인지 확인하기 위해 '흐름' 블록 꾸러미에서 '만일~이라면' 블록을 이용하여 인식된 글자가 무엇인지 확인합니다.

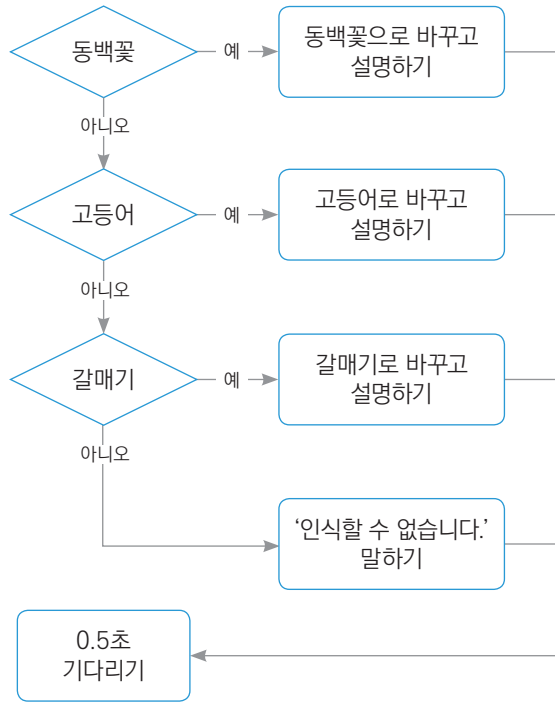


그런데 이렇게 확인을 하면 동백꽃, 고등어, 갈매기는 처리할 수 있습니다. 그러나 그 이외의 글자가 인식되었을 경우에 처리할 수 없게 됩니다.

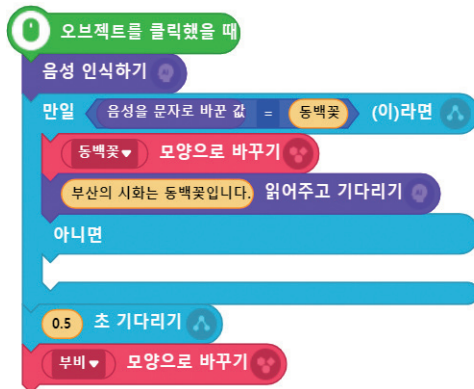
다음 그림을 살펴봅시다.



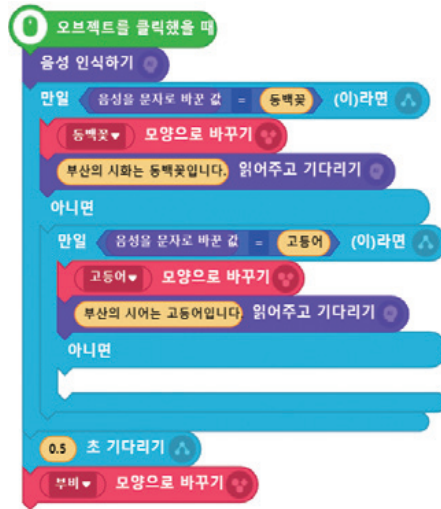
'동백꽃'인지 판단해서 맞은 경우나 아닌 경우 모두 다시 '고등어'인지 판단하게 됩니다. 또, '고등어'인지 판단해서 맞은 경우나 아닌 경우 모두 다시 '갈매기'를 판단하게 됩니다. 다음과 같이 한다면 '동백꽃'이면 '고등어'인지 다시 비교하지 않게 됩니다.



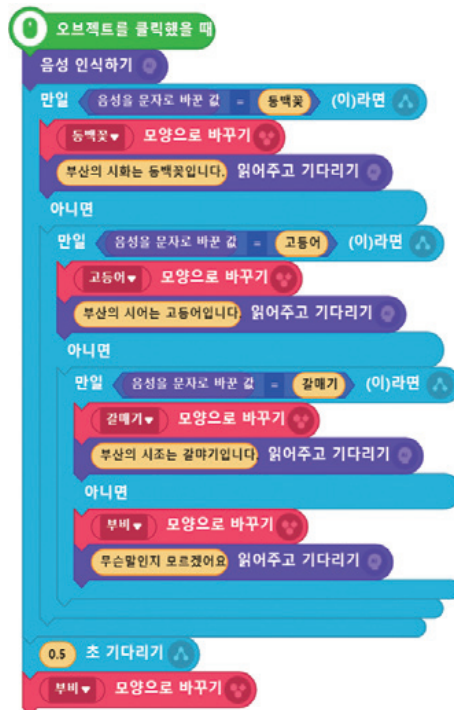
'흐름' 블록 꾸러미에서 '만약 ~이라면 아니라면' 블록을 사용하여 수정합니다. 즉, 인식된 글자가 동백꽃이면 '동백꽃' 모양으로 바꾸고 동백꽃 설명을 읽어주고 0.5초 기다린 후에 '부비' 모양으로 바꿉니다.



동백꽃이 아닌 경우는 ‘고등어’인지 판단하게 됩니다. ‘고등어’인지 확인되면 ‘고등어’ 모양으로 바꾸고 고등어 설명을 읽어주고 0.5초 기다린 후에 ‘부비’ 모양으로 바꿉니다.



이제 고등어가 아닌 경우 ‘갈매기’인지 확인하고 ‘갈매기’ 모양으로 바꾸고 갈매기 설명을 읽어 줍니다. ‘갈매기’가 아니면 ‘무슨 말인지 모르겠어요’라고 읽어주고 0.5초 기다린 후에 ‘부비’ 모양으로 바꿉니다.



④ 설명하는 동안은 클릭이 되지 않도록 한다.

‘부비’ 오브젝트를 클릭하면 설명이 진행되는 동안에도 클릭이 됩니다. 그럼 설명을 다 들을 수가 없습니다.

그럼 어떻게 처리해야 할까요? ‘부비’ 오브젝트를 클릭했을 때, ‘부비’ 오브젝트의 모양이 ‘부비’인 경우만 음성을 인식하고 처리하도록 작성하면 됩니다. 각 상징의 설명을 추가하여 작성해 봅시다.

```

1 오브젝트를 클릭했을 때
만일 부비 의 모양 이름 = 부비 (아)라면
  음성 인식하기
  만일 음성을 문자로 바꾼 걸 = 동백꽃 (아)라면
    동백꽃 모양으로 바꾸기
    진녹색의 잎과 진홍색의 꽃의 조화는 푸른 바다와, 사랑이 많은 시민의 정신을 그려내고, 싱싱하고 빛이 나는 진녹색 활엽은 시민의 젊음과 의욕을 읽어주고 기다리기
  아니면
    만일 음성을 문자로 바꾼 걸 = 고등어 (아)라면
      고등어 모양으로 바꾸기
      태평양을 누비는 강한 힘으로 목표를 향해 끊임없이 도약하는 해양수산도시 부산을 상징해 읽어주고 기다리기
    아니면
      만일 음성을 문자로 바꾼 걸 = 갈매기 (아)라면
        갈매기 모양으로 바꾸기
        새하얀 날개와 몸은 백의민족을 상징한다. 끈기 있게 먼 뱃길을 따라 하늘을 나는 갈매기의 강인함은 부산 시민의 정신을 잘 나타내기 때문에 부산의 새로 상 읽어주고 기다리기
      아니면
        부비 모양으로 바꾸기
        우순달린지 모르겠어요 읽어주고 기다리기
  0.5 초 기다리기
  부비 모양으로 바꾸기
  
```



부산의 상징에 대한 설명

■ 동백꽃

진녹색 잎과 진홍색 꽃의 조화는 푸른 바다와 사랑이 많은 시민의 정신을 그려내고, 싱싱하고 빛이 나는 진녹색 활엽은 시민의 젊음과 의욕을 나타낸다.

■ 갈매기

새하얀 날개와 몸은 백의민족을 상징한다. 끈기 있게 먼 뱃길을 따라 하늘을 나는 갈매기의 강인함은 부산 시민의 정신을 잘 나타내기 때문에 부산의 새로 선정되었다.

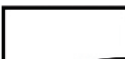
■ 고등어

태평양을 누비는 강한 힘으로 목표를 향해 끊임없이 도약하는 해양수산도시 부산을 상징한다.

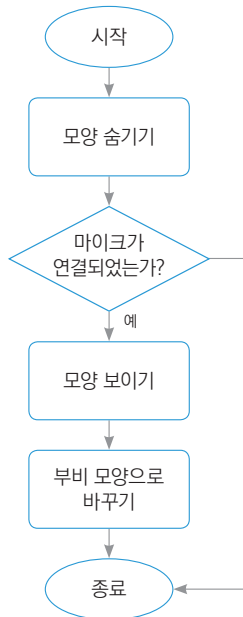


순서도 대해 알아보기

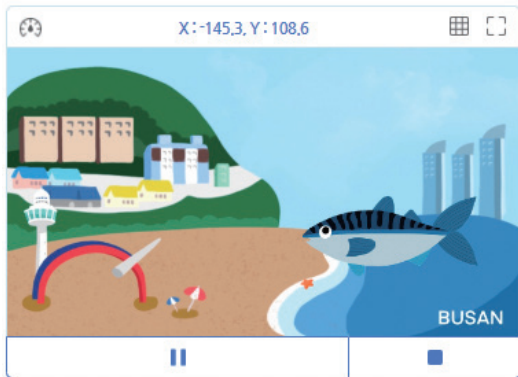
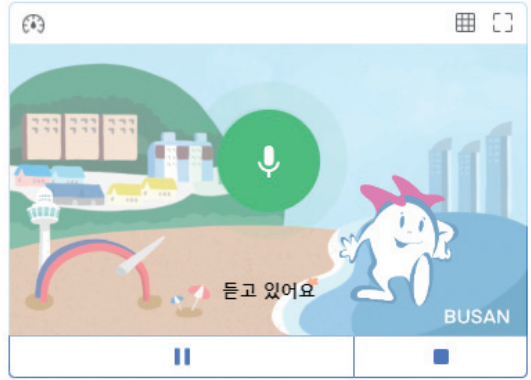
알고리즘은 컴퓨터에게 지시하는 명령 순서라고 했습니다. 순서도는 미리 정의된 기호로 이러한 알고리즘을 그림으로 표현한 것입니다. 순서도 기호에는 다음과 같은 것이 있습니다.

기호	의미
	순서도의 시작과 끝을 나타내는 기호
	값을 계산하거나 대입 등을 나타내는 기호
	처리해야 할 작업을 나타내는 기호
	조건이 참이면 '예', 아니면 '아니오'로 판하는 기호
	데이터 입력에 사용되는 기호
	문서로 인쇄할 것을 나타내는 기호

순서도와 엔트리 코드를 비교해 보세요.



4) 결과 확인하기





자신만의 해결 문제를 만들어 보세요.



해결 문제 정의하기

Blank writing area with horizontal dashed lines for defining the problem.



해결 문제 작성하기

Blank writing area with horizontal dashed lines for writing the solution problem.



3장

부산의 랜드마크를 알아보자





부산의 상징으로 배우는
엔트리 AI



학습 목표

- 컴퓨터 비전에 대해 알아보시다.
- 부산의 랜드마크에 대해 알아보는 프로그램을 만들어 봅시다.

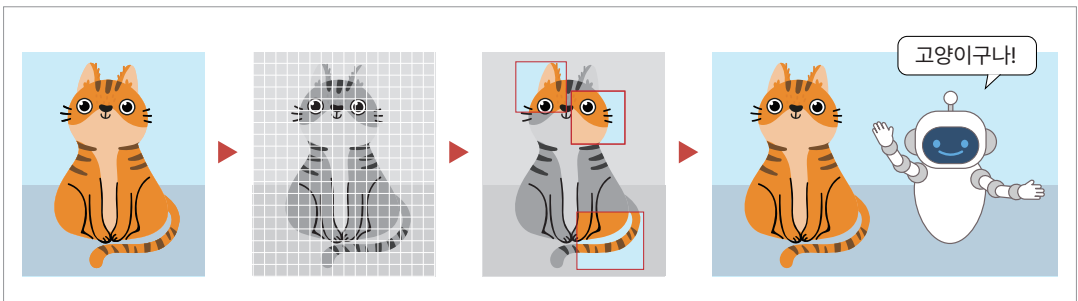
1 컴퓨터 비전이란 무엇일까요?

컴퓨터 비전(CV: Computer Vision)은 컴퓨터가 사람의 눈처럼 사물을 인식하여 구분할 수 있는 기술로 인공지능 기술이 접목되면서 다양한 분야에서 사용되고 있습니다.

컴퓨터는 이미지 데이터를 받아서 컴퓨터가 처리할 수 있는 형태로 바꿉니다. 이러한 과정을 이미지 처리(Image Processing)라고 합니다. 이미지는 주로 픽셀 형태로 표현되고 저장됩니다.

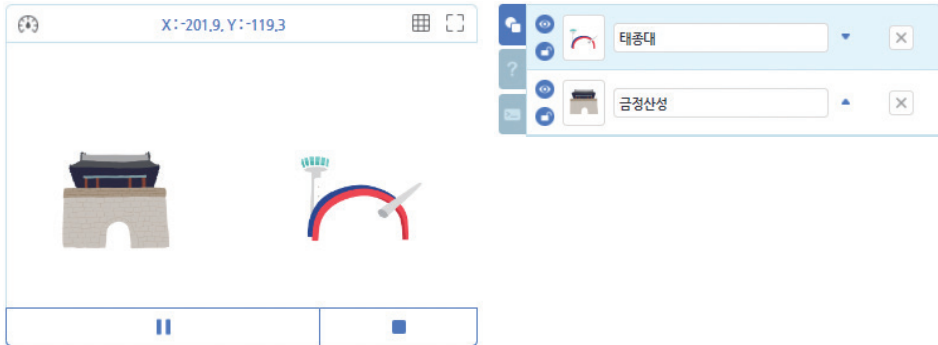
컴퓨터는 숫자로 데이터를 표현하고 저장합니다. 컴퓨터에 저장되어 있는 이미지도 숫자 형태로 표현됩니다. 이미지는 픽셀(pixel)이라는 작은 사각형들로 이루어져 있습니다. 픽셀은 화소라고도 하며 이미지에 픽셀이 많을수록 고화질의 이미지가 됩니다. 컴퓨터는 이 픽셀에 있는 숫자로 이미지를 인식합니다. 픽셀에 0과 1로 저장된 이미지는 흑백 이미지를 표현하고 픽셀에 0~255까지의 숫자를 저장하여 숫자가 커질수록 밝아지도록 명암을 표현하여 회색조 이미지를 표현합니다. 컬러 이미지는 빨강(R), 녹색(G), 파랑(B)의 조합으로 표현합니다.

이렇게 숫자로 표현된 이미지로부터 특징을 찾아내는 과정을 특징 추출(feature extraction)이라고 합니다. 예를 들면, 고양이를 판단하기 위해서 ‘귀가 뽕족하다’, ‘꼬리가 있다’, ‘수염이 있다’ 등의 특징을 찾아 이것들을 조합해서 고양이라고 컴퓨터가 인식하게 됩니다.



2

해결 문제: 부산의 랜드마크를 알아보자



랜드마크는 어떤 지역에서 두드러지는 지형물이나 사물을 말합니다. 부산의 랜드마크는 어떤 것이 있을까요? 부산에는 금정산성, 태종대, 용두산공원, 광안대교, 해운대, 감천 문화마을, 흰여울 마을 등 아주 많이 존재합니다.

이번 프로그램에서는 부산의 랜드마크인 금정산성과 태종대에 대해 알아보기 위한 프로그램을 작성합니다. 얼굴을 인식해서 랜드마크를 선택하고 선택된 랜드마크에 대해 설명을 해 줍니다.

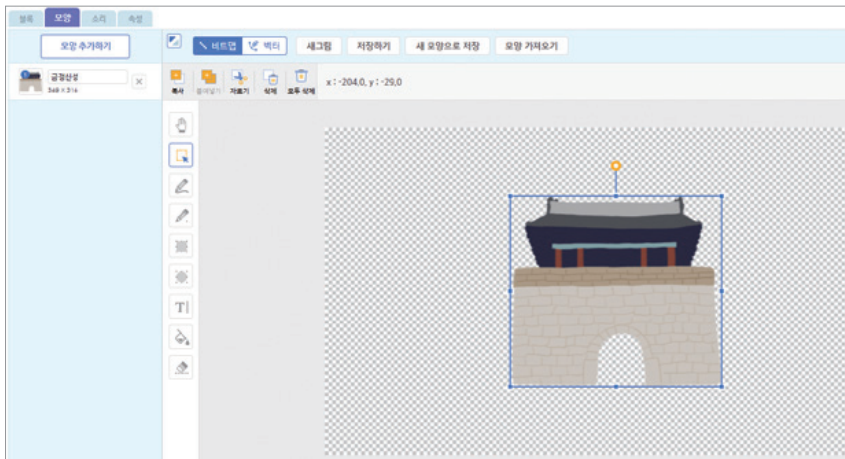


3 엔트리로 만들어 봅시다.

1) 화면 구성하기

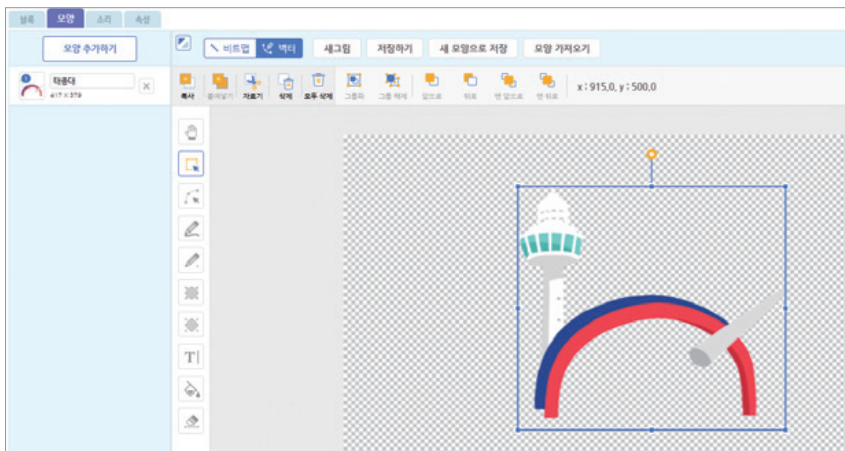
(1) 금정산성 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 금정산성 이미지를 추가합니다.



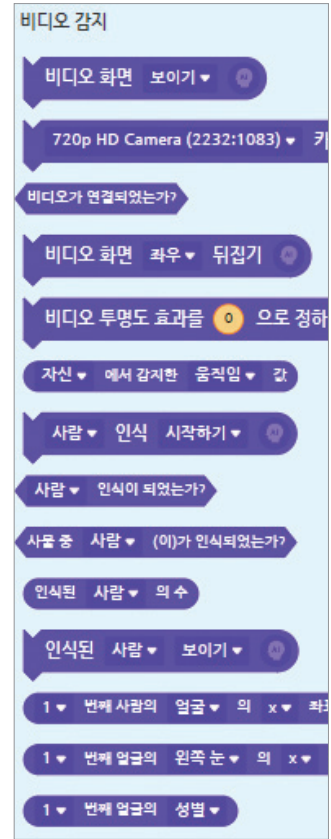
(2) 태종대 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 태종대 이미지를 추가합니다.



2) 인공지능 블록 불러오기

블록 꾸러미에서 블록 탭의 인공지능을 선택하고 인공지능 불러오기 버튼을 눌러 오디오 감지와 비디오 감지를 선택합니다. 엔트리의 비디오 감지 블록에는 사람을 인식하는 블록이 이미 만들어져 있습니다.



엔트리가 인식하는 사물

엔트리가 인식하는 사물의 종류는 다음과 같습니다.

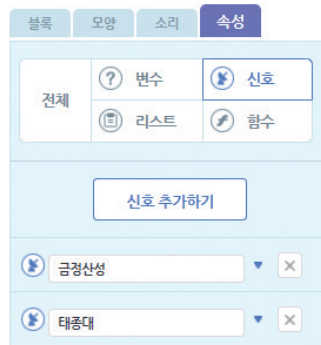
사람	자전거	자동차	오토바이	샌드위치	오렌지	브로콜리	당근
비행기	버스	기차	트럭	핫도그	피자	도넛	케이크
보트	신호등	소화전	정지 표지판	의자	소파	화분	침대
주차 미터기	벤치	새	고양이	식탁	변기	텔레비전	노트북
개	말	양	소	마우스	리모컨	키보드	핸드폰
코끼리	곰	얼룩말	기린	전자레인지	오븐	토스터	싱크대
배낭	우산	핸드백	텍타이	냉장고	책	시계	꽃병
여행 가방	원반	스키	스노우보드	가위	테디베어	헤어드라이어	칫솔
공	연	야구 배트	야구 글러브				
스케이트보드	서프보드	테니스라켓	병				
와인잔	컵	포크	ナイ프				
숟가락	그릇	바나나	사과				

3) 신호 만들기

코드를 실행시키기 위해서는 반드시 [시작] 카테고리의 블록으로 시작되어야 합니다. [시작] 카테고리의 블록들은 해당 코드가 언제 시작되어야 할지를 결정해 줍니다.

한 작품에 여러 개의 오브젝트가 있을 때, 한 오브젝트의 코드를 실행하는 도중에 다른 오브젝트의 코드의 실행이 시작되도록 해 주려면 어떻게 해야 할까요? 이렇게 오브젝트 간 상호 작용이 필요할 때 '신호'를 사용합니다. 신호는 원하는 만큼, 원하는 이름으로 만들 수 있습니다.

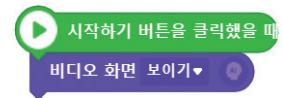
'금정산성'과 '태종대'라는 신호를 만들어 각 오브젝트에서 신호를 받으면 해당 오브젝트의 크기를 크게 해 주고, 설명해 주고, 크기를 줄여 주도록 작성합니다. 블록 꾸러미에서 속성 탭을 선택하고 [신호 추가하기] 버튼을 눌러 2개의 신호를 작성합니다.



4) 프로그램 설계하기

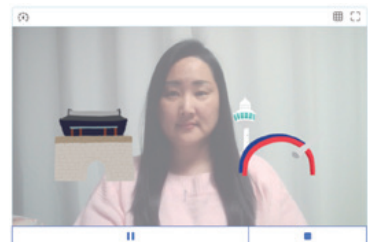
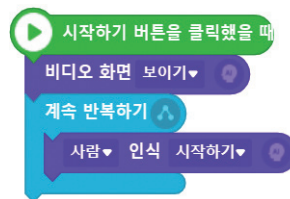
① 카메라를 통해 사물을 인식하도록 합니다.

'금정산성' 오브젝트에서 '시작하기 버튼을 클릭했을 때'를 누르면 카메라를 통해 인식할 수 있도록 합니다.



② 사람을 인식하도록 합니다.

프로그램을 시작하면 카메라가 켜지고 계속해서 사람을 인식하도록 작성합니다. 프로그램이 수행되는 동안 계속해서 사람을 인식하도록 '제어' 블록 꾸러미에서 '계속 반복하기' 블록을 사용합니다.

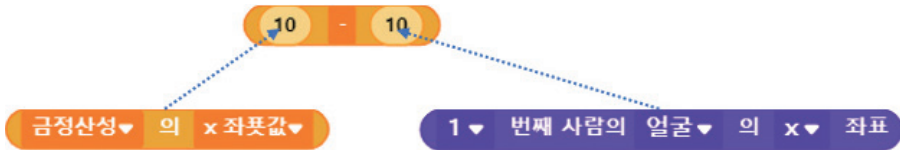


③ 오브젝트와 사람 얼굴과 x좌표의 차를 계산합니다.

사람을 인식해서 사람의 얼굴로 '금정산성' 오브젝트와 '태종대' 오브젝트를 선택합니다. 어떻게 선택되었는지를 확인할 수 있을까요?

카메라로 인식된 사람의 얼굴 x좌표와 오브젝트의 x좌표의 거리차를 계산해서 50보다 작으면 오브젝트가 선택되었다고 해 보겠습니다.

'계산' 블록 꾸러미에서 '빼기' 블록과 '~의 x좌표값' 블록을 가져옵니다. '인공지능' 블록 꾸러미에서 '~번째 사람의 얼굴의 x좌표' 블록을 가져와서 각각 빼기 블록에 넣어 줍니다.



오브젝트와 얼굴의 x좌표의 차의 절댓값을 계산하기 위해 '계산' 블록 꾸러미에서 '~의 제곱' 블록을 가져와서 제곱을 절댓값으로 바꾸어 주고 위에서 작성한 빼기 블록을 넣어 줍니다.



④ 오브젝트가 선택되었는지 확인합니다.

오브젝트와 얼굴의 x좌표의 차이가 50보다 작으면 해당 오브젝트가 선택되었다고 확인하고 해당 오브젝트 신호를 보내고 신호를 받은 오브젝트가 실행이 끝날 때까지 기다립니다.





엔트리 '신호 보내기'와 '신호 보내고 기다리기'

■ 신호 보내기

선택한 신호를 보냅니다.

■ 신호 보내고 기다리기

선택한 신호를 보내고 해당 신호를 받은 블록들이 실행이 끝날 때까지 기다립니다.

⑤ 신호처리를 합니다.

오브젝트가 신호를 받으면 해당 오브젝트의 크기를 50만큼 크게 바꾸고 설명을 읽어주고 2초 후에 다시 크기를 -50만큼 바꾸어 원래 크기로 만들어 줍니다.

크기를 바꾸기 위해서는 '생김새' 블록 꾸러미에서 '크기를 ~만큼 바꾸기' 블록을 선택합니다. 아래의 내용으로 설명을 변경해 보세요.



랜드마크에 대한 설명

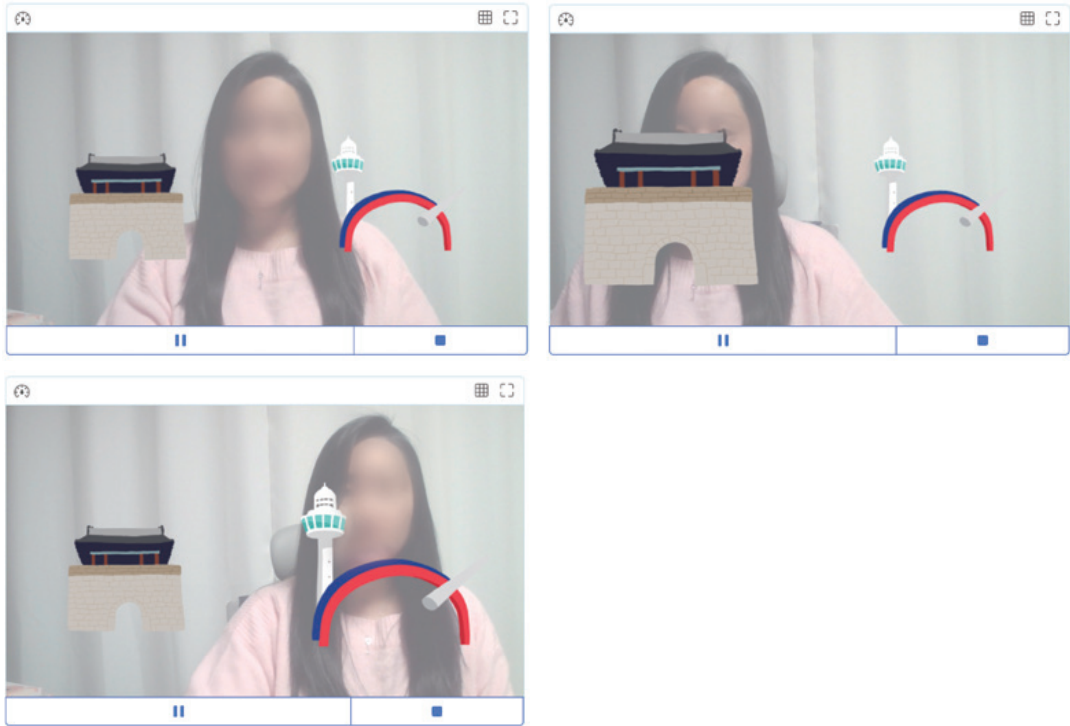
■ 금정산성

사적 제215호 금정산성은 숙종 29년(1703)에 건설된 길이 18,845m, 성벽 높이 1.5~3m로 대한민국 최대 규모인 산성입니다.

■ 태종대

국가 지정 명승지 제17호인 태종대는 신라 태종 무열왕이 들러 활쏘기를 하고 연회를 개최했던 곳으로 울창한 숲과 기암괴석이 남해 바다와 어우러진 비경으로 유명합니다.

5) 결과 확인하기





자신만의 해결 문제를 만들어 보세요.



해결 문제 정의하기

A large rectangular area with horizontal dashed lines for writing, intended for defining the problem.



해결 문제 작성하기

A large rectangular area with horizontal dashed lines for writing, intended for writing the solution to the problem.

4장

부산의 랜드마크를 잡아보자





부산의 상징으로 배우는
엔트리 시



학습 목표

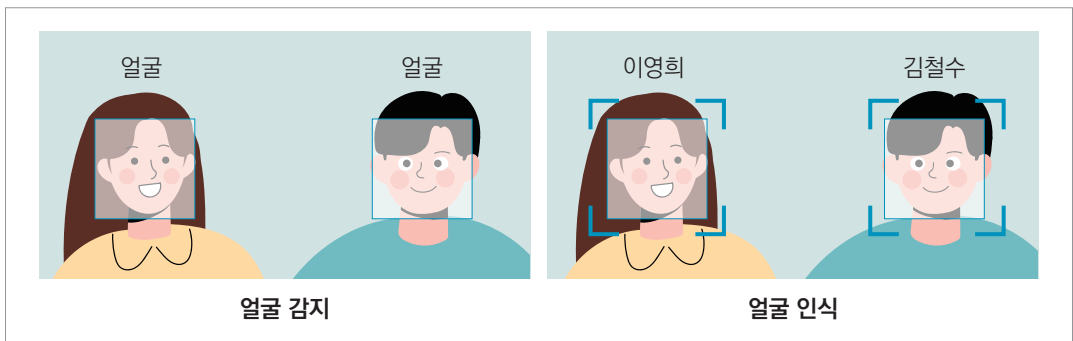
- 컴퓨터 비전 기술을 이용한 얼굴 감지와 얼굴 인식에 대해 알아봅시다.
- 얼굴을 따라다니는 부비가 랜드마크를 잡아보는 프로그램을 만들어 봅시다.

1 얼굴 감지와 얼굴 인식은 어떤 차이가 있을까요?

컴퓨터 비전(CV: Computer Vision)은 컴퓨터가 사람의 눈처럼 사물을 인식할 수 있다고 했습니다. 이러한 컴퓨터 비전 기술을 통해 인식된 이미지에서 사람의 얼굴을 찾아내는 것이 얼굴 감지 기술입니다. 얼굴 감지 기술을 통해 이미지 속 얼굴의 윤곽, 눈·코·입 위치, 표정 값을 제공해 줍니다.

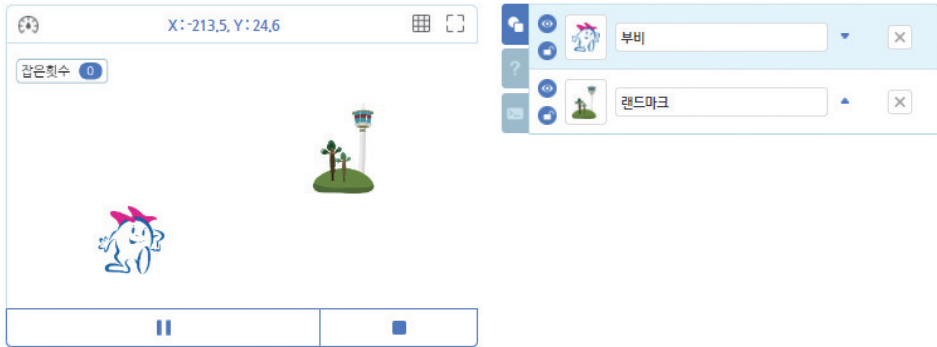
얼굴을 감지하고 감지된 얼굴이 누구인지 알아내는 것을 얼굴 인식이라고 합니다. 얼굴 감지를 통해 눈, 눈썹, 코, 입, 턱 등의 얼굴 특징이 변하는 부위를 분석해 특징이 되는 데이터를 추출합니다. 검출된 얼굴 정보에서 특징점을 파악하여 개개인의 얼굴 정보와 일치하는 작업이 얼굴 인식 기술의 핵심입니다. 인공지능 알고리즘과 GPU를 활용한 연산 속도의 증가는 기존 인공지능 연산 속도를 획기적으로 증가시켜 인공지능을 활용한 얼굴 인식 기술이 더욱 빠르게 발전하고 있습니다.

얼굴 인식 기술을 활용하여 출입 통제, 범죄 수사, 금융 결재, CCTV 영상 분석 영역까지 다양하게 확대되고 있습니다.



2

해결 문제: 부산의 랜드마크를 잡아보자



사람을 인식해서 '부비' 오브젝트가 얼굴을 따라다니도록 작성해 봅니다. '랜드마크' 오브젝트는 1초에 한 번씩 임의의 위치에 나타나면서 모양도 변경해 봅니다. '부비' 오브젝트와 '랜드마크' 오브젝트가 충돌하게 되면 잡은 횟수를 증가시키도록 작성해 봅니다. 이때, 충돌된 '랜드마크'가 무엇인지 읽어줍니다.

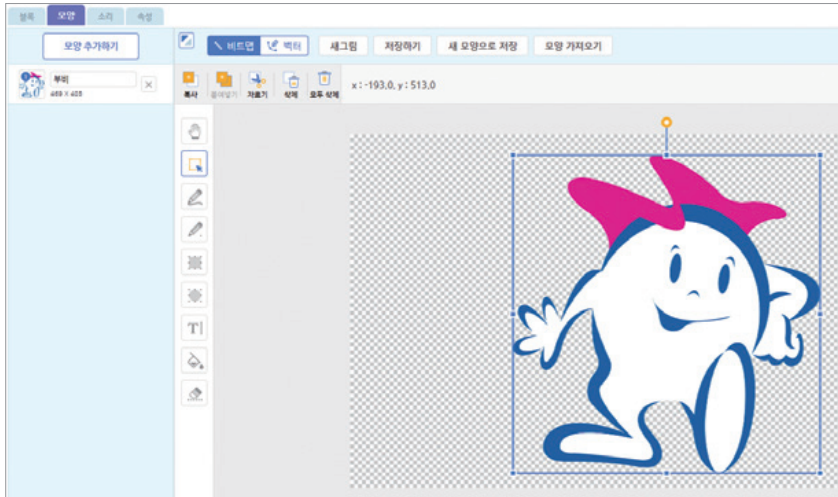
3

엔트리로 만들어 봅시다.

1) 화면 구성하기

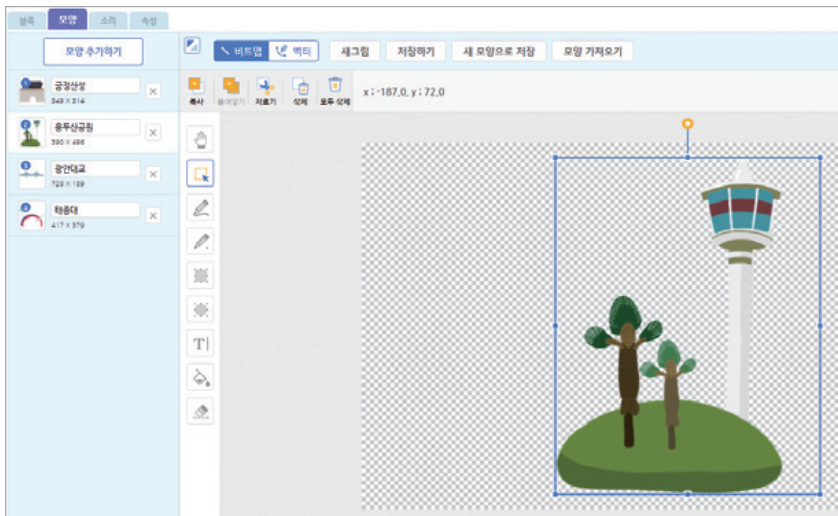
(1) 부비 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 부비 이미지를 추가합니다.



(2) 랜드마크 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 금정산성, 용두산공원, 광안대교, 태종대 이미지를 추가합니다.



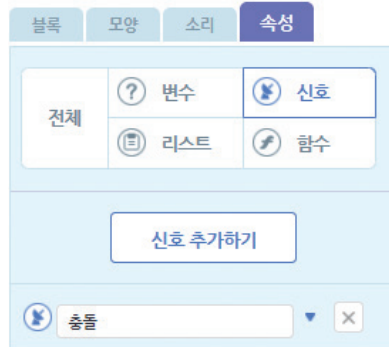
2) 인공지능 블록 불러오기

블록 꾸러미에서 블록 탭의 인공지능을 선택하고 인공지능 불러오기 버튼을 눌러 [[비디오 감지]와 [읽어주기]를 선택합니다.



3) 신호 만들기

'부비'와 '랜드마크' 오브젝트가 충돌하면 '부비' 오브젝트가 '랜드마크' 오브젝트로 신호를 보내주기 위해 '충돌' 신호를 작성합니다.

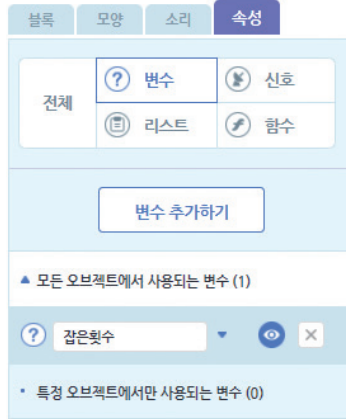


4) 변수 만들기

프로그램에서는 어떠한 정보를 기억하기 위해 '변수'라는 공간이 필요합니다. 변수는 상자와 같으며, 그 안에는 숫자나 문자를 넣을 수 있습니다. 상자에 무엇이 들어있는지 알기 쉽게 이름표를 붙여 주는데 그것이 변수의 이름입니다. 변수 안의 값은 프로그램 실행 중에 블록 명령을 통해 변

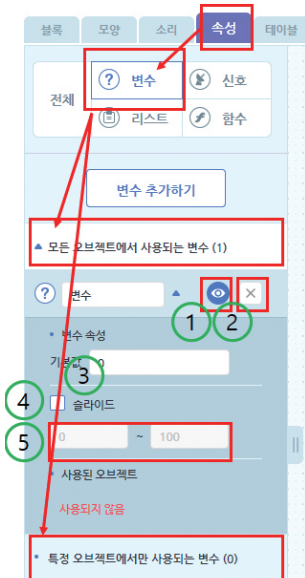
경될 수 있습니다.

'부비'와 '랜드마크' 오브젝트가 충돌이 일어나면 충돌 횟수를 기억하기 위해 '잡은횟수' 변수를 작성합니다. 블록 꾸러미의 속성 탭에서 변수를 선택하고 [변수 추가하기] 버튼을 누르고 모든 오브젝트에서 사용하는 변수 '잡은횟수'를 작성합니다.



엔트리 변수 만들기

변수는 어떤 정보를 저장하고, 지우기도 하고, 다른 값을 가져오기도 할 수 있는 저장소입니다. 엔트리에서 변수를 만든 후 값과 속성을 변경할 수 있습니다.

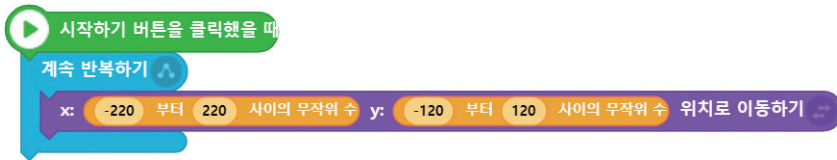


- ① 눈 아이콘: 변수를 화면에서 보여주거나 보여주지 않도록 합니다.
- ② 변수 삭제: 변수를 삭제합니다.
- ③ 기본값: 시작할 때의 초깃값을 정합니다. 기본으로 0이 됩니다.
- ④ 슬라이드: 플레이할 때 변수를 조절할 수 있는 슬라이드를 제공합니다.
- ⑤ 슬라이드 최솟값/최댓값: 슬라이드의 최솟값/최댓값을 입력합니다.

5) 프로그램 설계하기

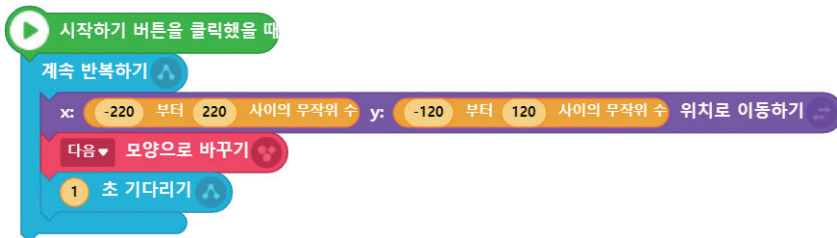
① ‘랜드마크’ 오브젝트를 임의의 위치로 이동합니다.

‘시작하기 버튼을 클릭했을 때’ 계속 반복해서 임의의 위치로 이동합니다. 위치 이동은 움직임 블록 꾸러미에서 ‘x~ y:~ 위치로 이동하기’ 블록을 사용합니다. 이때, 위치는 ‘계산’ 블록 꾸러미 ‘~부터~사이의 무작위 수’ 블록을 이용하여 x의 범위와 y의 범위를 지정합니다.

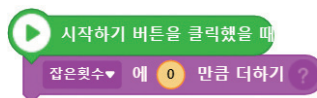


② ‘랜드마크’ 오브젝트가 임의의 위치로 변경할 때마다 모양을 바꿉니다.

‘랜드마크’ 오브젝트의 모양이 금정산성, 용두산공원, 광안대교, 태종대 순으로 추가되어 있으므로 순서대로 변경되도록 작성합니다. 1초마다 변경하기 위해 ‘흐름’ 블록 꾸러미에서 ‘~초 기다리기’ 블록을 사용합니다.



③ ‘부비’ 오브젝트가 시작할 때 ‘잡은횟수’ 변수를 0으로 초기화합니다.



④ 사람을 인식합니다.

사람을 인식하기 위해 ‘인공지능’ 블록 꾸러미에서 ‘비디오 화면을 보이기’ 블록을 사용하여 카메라를 켜고 ‘사람 인식 시작하기’ 블록을 사용하여 사람을 인식합니다.



- ⑤ ‘부비’ 오브젝트가 얼굴을 따라 움직이도록 합니다.
 얼굴 인식 후 좌표를 ‘부비’ 오브젝트의 좌표로 지정합니다.



- ⑥ 잡은 횟수 변수를 변경합니다.

‘부비’ 오브젝트가 ‘랜드마크’ 오브젝트에 닿으면 ‘잡은횟수’ 변수의 값을 1씩 증가시키고 ‘랜드마크’ 오브젝트에게 ‘충돌’ 신호를 보내고 기다립니다.



⑦ 랜드마크가 무엇인지 알려줍니다.

'랜드마크' 오브젝트가 '충돌' 신호를 받으면 어떤 랜드마크인지 랜드마크의 이름을 읽어주고 기다립니다.



6) 결과 확인하기





자신만의 해결 문제를 만들어 보세요.



해결 문제 정의하기

Blank writing area with horizontal dashed lines for defining the problem.



해결 문제 작성하기

Blank writing area with horizontal dashed lines for writing the solution.

5장

부산의 날씨는 어때





부산의 상징으로 배우는
엔트리 AI

**학습
목표**

- 전문가 시스템에 대해 알아보시다.
- 날씨를 알려주는 챗봇 프로그램을 만들어 봅시다.

1 전문가 시스템?

전문가 시스템은 사실과 규칙으로 지식을 표현하는 시스템으로 인공지능 기술의 한 분야입니다. 사람이 가진 전문적인 지식을 컴퓨터에 기억시켜 일반인들이 이 전문지식을 사용할 수 있도록 하는 시스템입니다.

우리가 무언가를 관찰하여 얻은 정보와 이미 알고 있는 지식을 비교하면 새로운 사실을 논리적으로 추론할 수 있습니다. 예를 들어, 교복을 입은 사람을 보았습니다. 우리는 이미 교복은 학생이 입는다는 것을 알고 있습니다. 그러니 교복을 입은 사람을 보면 학생이라고 추론할 수 있습니다. 이것이 논리적 추론입니다.

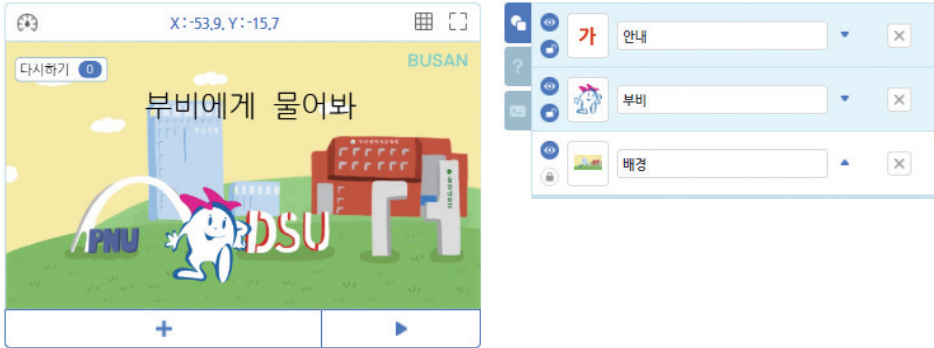
전문가 시스템은 이러한 논리적인 추론을 바탕으로 만들어집니다. 사람이 가지고 있는 지식을 어떤 규칙의 형태로 만들어 컴퓨터에 저장해 놓고 문제가 주어졌을 때 그 지식을 사용하여 해결합니다. 전문가 시스템을 이용하면 전문가가 부족한 상황에서도 전문가처럼 문제를 해결할 수 있습니다.

논리적 추론이 적용된 전문가 시스템으로 인공 대화 시스템인 엘리자가 있습니다. 엘리자는 MIT의 조셉 와이젠바움 교수가 1966년에 만든 인공 대화 시스템입니다. 엘리자에 의해 인류는 처음으로 컴퓨터와 대화하게 되었습니다. 당시 엘리자는 단순히 '생각과 답변'을 하는 것이 아니었는데요. 예를 들면, 사람이 엘리자에게 "배가 아픕니다."라고 말하면, 엘리자는 "왜 배가 아파?"라고 대응했습니다. 그러나 엘리자의 치명적인 단점은 사용자가 일일이 사람들의 예상 질문과 답변을 프로그래밍해 주어야 했고, 예상치 못한 질문이나 말을 건넨다면 답변할 수 없었습니다.

이처럼 사람의 지식을 모두 규칙 형태로 표현하여 저장하는 것이 매우 어렵고 전문가마다 가지고 있는 규칙이 다른 경우에는 하나의 공통된 규칙을 정하기 어렵습니다.

2

해결 문제: 부산의 날씨는 어때



부산의 날씨를 알려주는 챗봇을 만들어 봅시다. 챗봇을 만들기 위해서 규칙을 정리해 봅시다. 날씨 정보는 실시간으로 제공되는 공공데이터를 활용합니다.

질문	규칙	답변
기온과 미세먼지 무엇을 알려드릴까요?	기온	기온은 ○도입니다.
	미세먼지	미세먼지는 ○입니다.
	그 외	이해를 못 했어요
다시 하고 싶으면 다시라고 해 주세요.	다시	기온과 미세먼지 무엇을 알려드릴까요?
	다시가 없으면	날씨 정보를 종료합니다.

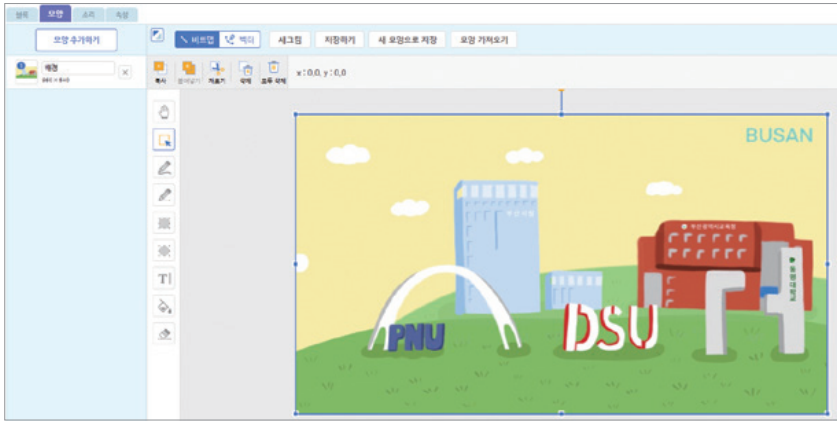
3

엔트리로 만들어 봅시다.

1) 화면 구성하기

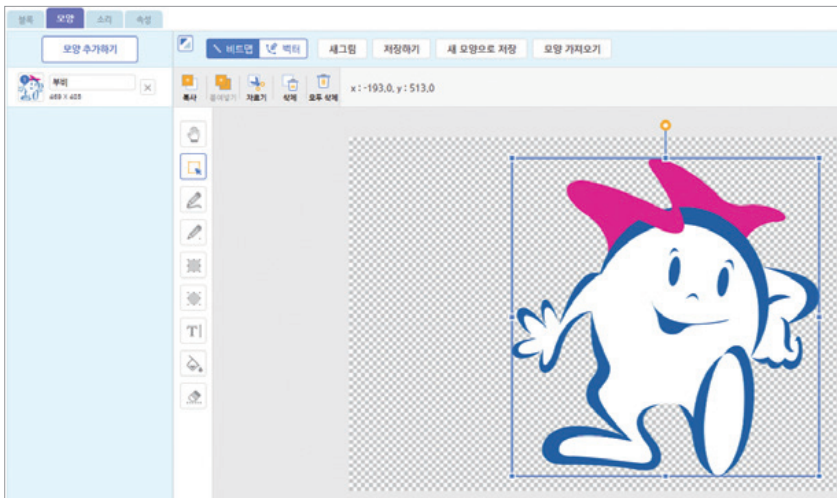
(1) 배경 추가하기

오브젝트를 추가하여 [모양 추가하기] 버튼을 눌러 [배경] 항목을 선택하고 파일 올리기를 이용하여 배경을 선택합니다. 파일 올리기를 통해 배경을 추가할 때는 엔트리에서 제공하는 배경을 하나 선택한 후 파일 올리기를 하면 배경에 맞게 작성됩니다.



(2) 부비 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 과일 올리기를 이용하여 부비 이미지를 추가합니다.



(3) 안내 오브젝트

오브젝트 추가하기(+)를 선택하고 오브젝트 선택에서 '글상자'를 선택하여 글자를 작성하고 적용하기 버튼을 누릅니다.



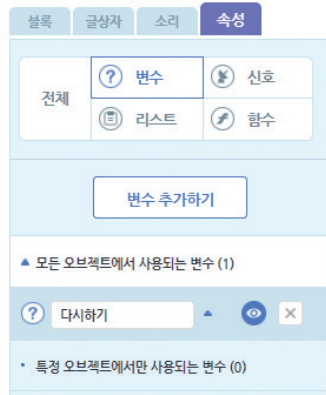
2) 인공지능 블록 불러오기

블록 꾸러미에서 블록 탭의 인공지능을 선택하고 인공지능 불러오기 버튼을 눌러 오디오 감지와 읽어주기를 선택합니다.



3) 변수 만들기

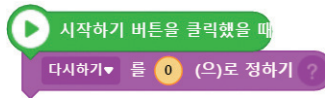
반복문을 제어하기 위해 '다시하기' 변수를 추가합니다.



4) 프로그램 설계하기

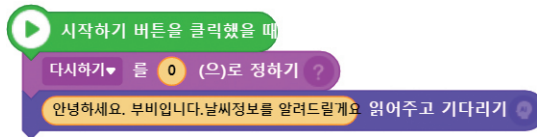
① 변수 초기화합니다.

프로그램이 시작되면 반복을 제어하기 위해 만들어 놓은 '다시하기' 변수를 0으로 초기화합니다.



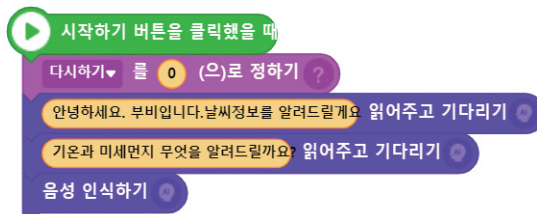
② 인사말을 합니다.

'안녕하세요. 부비입니다. 날씨 정보를 알려드릴게요'라고 인사말을 작성합니다.



③ 질문하고 답변을 기다립니다.

'기온과 미세먼지 무엇을 알려드릴까요?'라고 질문을 하고 답변을 기다립니다.



④ ‘기온과 미세먼지 무엇을 알려드릴까요?’ 질문에 답변을 합니다.

정해진 규칙에 따라 답변을 합니다. 답변 내용에 기온이 있으면 기온을 알려주고, 미세먼지가 있으면 미세먼지를 알려주고 정해진 규칙이 없으면 ‘이해를 못 했어요’라고 합니다.

질문	규칙	답변
기온과 미세먼지 무엇을 알려드릴까요?	기온	기온은 ○도입니다.
	미세먼지	미세먼지는 ○입니다.
	그 외	이해를 못 했어요

음성을 인식했을 때 규칙에 해당하는 ‘기온’이나 ‘미세먼지’ 글자가 인식된 문자에 포함되어 있는지 확인하기 위해서 ‘계산’ 블록 꾸러미에서 ‘~에서 ~의 시작 위치’ 블록을 사용합니다. 예를 들어, ‘음성을 문자로 바꾼 값에서 기온의 시작 위치’ 블록을 사용하면 인식된 문자에서 기온이란 글자의 시작 위치의 값을 알려줍니다. 만약 0이라면 해당 글자가 없는 것입니다. 따라서, 다음 코드는 음성으로 인식된 글자에서 기온이란 글자가 있는 경우가 됩니다.

```
음성을 문자로 바꾼 값에서 기온의 시작 위치 != 0
```

또한, 대답을 해주기 위해 글자를 연결해서 말하기 위해서는 ‘계산’ 블록 꾸러미에서 ‘~와 ~을 합치기’ 블록을 사용합니다. 두 개의 합치기 블록을 이용하여 대답을 작성합니다. 이때, 기온에 대한 정보는 ‘확장’ 블록 꾸러미에서 날씨 확장 블록을 불러와서 사용합니다.

```
기온은 과(와) 현재 부산 전체의 기온(°C) 과(와) 도 입니다. 를 합치기 를 합치기
```

규칙을 적용하여 코드를 완성하면 다음과 같습니다.

```
시작하기 버튼을 클릭했을 때
다시하기 블록 0 (으)로 정하기
안녕하세요. 부비입니다. 날씨정보를 알려드릴게요. 읽어주고 기다리기
기온과 미세먼지 무엇을 알려드릴까요? 읽어주고 기다리기
음성 인식하기
만일
음성을 문자로 바꾼 값에서 기온의 시작 위치 != 0 (아)라면
기온은 과(와) 현재 부산 전체의 기온(°C) 과(와) 도 입니다. 를 합치기 를 합치기 읽어주고 기다리기
아니면
만일
음성을 문자로 바꾼 값에서 미세먼지의 시작 위치 != 0 (아)라면
미세먼지는 과(와) 현재 부산 전체의 미세먼지농도(ug) 과(와) 입니다. 를 합치기 를 합치기 읽어주고 기다리기
아니면
이해를 못 했어요. 읽어주고 기다리기
```



엔트리 확장 블록 알아보기

엔트리의 확장 블록은 실시간으로 공공데이터를 확인할 수 있습니다.



■ 날씨

웨더아이가 제공하는 기온, 강수량, 미세먼지 농도 등 날씨와 관련된 정보를 확인할 수 있는 블록을 제공합니다.

■ 생활안전 국민행동요령

국민안전처에서 제공하는 생활 속 안전을 위해 지켜야 하는 행동 요령에 대한 정보를 확인할 수 있는 블록을 제공합니다.

■ 자연재난 국민행동요령

국민안전처에서 제공하는 자연재난 발생 시 지켜야 하는 행동 요령에 대한 정보를 확인할 수 있는 블록을 제공합니다.

■ 행사

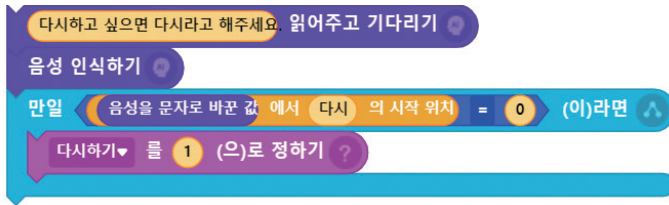
한국관광공사에서 제공하는 국내 지역별 다양한 행사와 축제와 관련된 정보를 확인할 수 있는 블록을 제공합니다.

⑤ ‘다시 하고 싶으면 다시라고 해주세요.’ 질문에 답변을 합니다.

정해진 규칙에 따라 답변을 합니다. 답변 내용에 ‘다시’가 있으면 ‘기온과 미세먼지 무엇을 알려드릴까요?’ 다시 묻고 그렇지 않으면 ‘날씨 정보를 종료합니다.’라고 대답합니다.

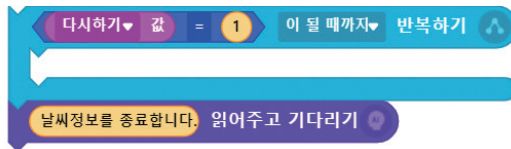
질문	규칙	답변
다시 하고 싶으면 다시라고 해주세요.	다시	기온과 미세먼지 무엇을 알려드릴까요?
	다시가 없으면	날씨 정보를 종료합니다.

이때, 다시 시작한다는 것은 앞선 과정을 반복하게 된다는 것입니다. 만약, 다시 하지 않는다면 ‘다시하기’ 변수에 1을 설정하여 반복을 종료하도록 합니다.



⑥ 다시하기 변수를 이용하여 반복을 처리합니다.

인식된 글자에 ‘다시’라는 글자가 있으면 반복문이 수행되어 질문을 다시 할 수 있도록 합니다. 이를 위해, ‘다시하기’ 변수를 사용합니다. 즉, 인식된 글자에 ‘다시’라는 글자가 없으면 ‘다시하기’ 변수의 값이 1이 됨으로 1이 될 때까지 반복할 수 있도록 하면 됩니다. 이를 위해, ‘흐름’ 블록 꾸러미에서 ‘참이 될 때까지 반복하기’ 블록을 사용합니다.



반복문을 적용하여 완성된 코드는 다음과 같습니다.



5) 결과 확인하기





자신만의 해결 문제를 만들어 보세요.



해결 문제 정의하기

Blank writing area with horizontal dashed lines for defining the problem.



해결 문제 작성하기

Blank writing area with horizontal dashed lines for creating the problem.

6장

부산의 여행지를 찾아보자





부산의 상징으로 배우는
엔트리 AI



학습 목표

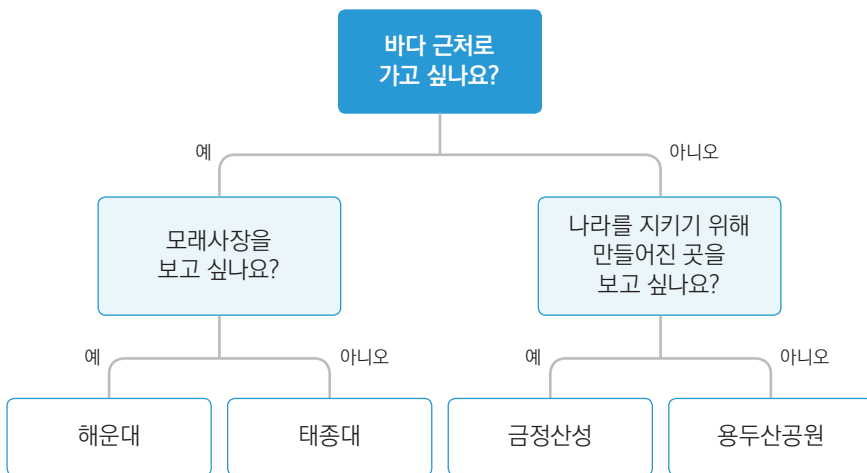
- 의사결정트리에 대해 알아봅시다.
- 의사결정트리를 이용해 여행지를 찾아주는 프로그램을 만들어 봅시다.

1 의사결정트리

앞서 논리적 추론을 바탕으로 한 전문가 시스템에 대해 살펴보았습니다. 이번 장에서는 논리적 추론을 통해 우리가 알고 있는 지식을 규칙으로 만들고 이러한 규칙을 바탕으로 순서도 형태로 이진 트리를 만들어 봅시다. 이 이진 트리를 의사결정트리(decision tree)라고 하고 예측이나 분류를 할 때 사용되는 모델입니다.

의사결정트리는 트리의 가장 위 지점(루트 노드)을 기준으로 가지(링크 혹은 분기)라는 화살표를 통해 여러 개의 방향으로 나눕니다. 예측이나 분류는 최종적으로 가지가 끝나는 지점(리프 노드)에서 결정됩니다.

예를 들어, 부산의 여행지를 추천하기 위한 의사결정트리를 만들어 보겠습니다. 추천 여행지는 해운대, 태종대, 금정산성, 용두산공원이 있습니다. 우리가 알고 있는 지식을 활용하여 규칙을 만들어 봅시다.



2

해결 문제: 부산의 여행지를 찾아보자



앞에서 만들어 본 의사결정트리를 이용하여 부산의 여행지를 찾아봅시다. 여행지는 리스트를 이용하여 저장한 후 사용합니다.

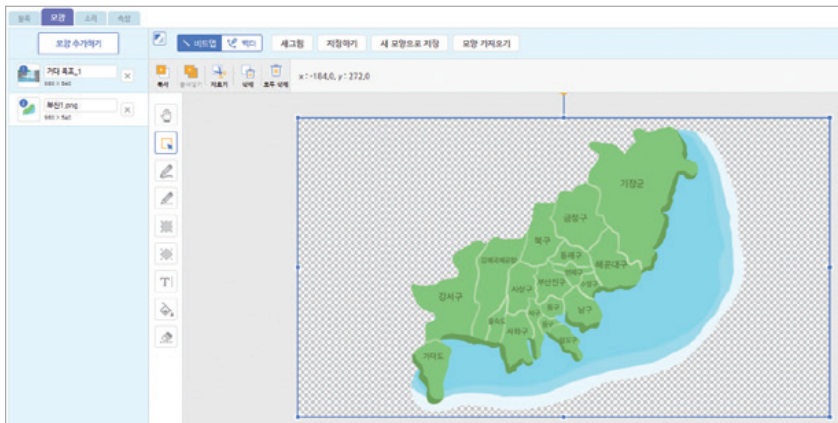
3

엔트리로 만들어 봅시다.

1) 화면 구성하기

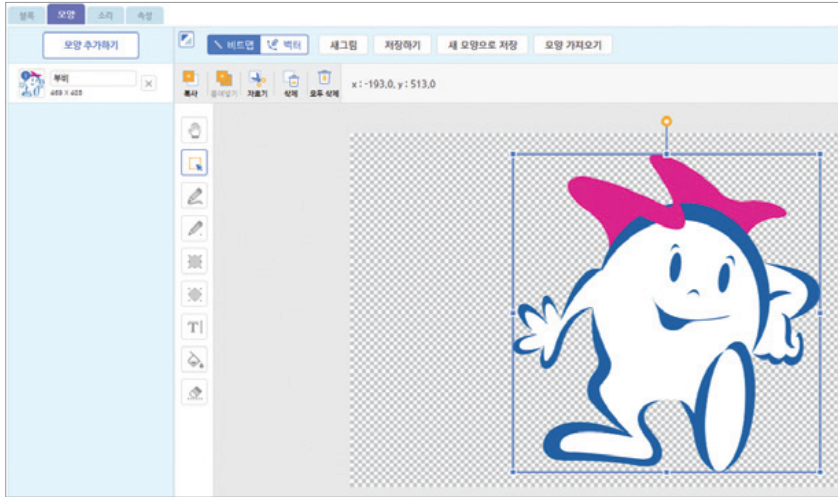
(1) 배경 추가하기

오브젝트를 추가하여 [모양 추가하기] 버튼을 눌러 [배경] 항목을 선택하고 파일 올리기를 이용하여 배경을 선택합니다.



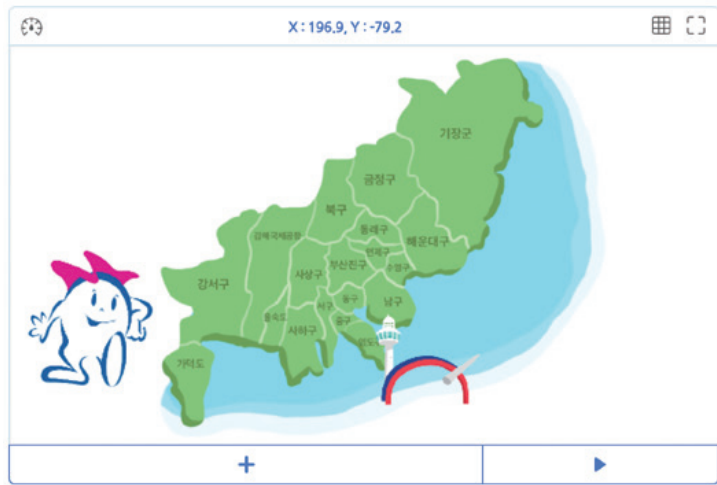
(2) 부비 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 부비 이미지를 추가합니다.



(3) 태종대 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 태종대 이미지를 추가하고 영도 지역에 배치합니다.



(4) 해운대 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 해운대 이미지를 추가하고 해운대 지역에 배치합니다.



(5) 용두산공원 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 용두산공원 이미지를 추가하고 중구 지역에 배치합니다.



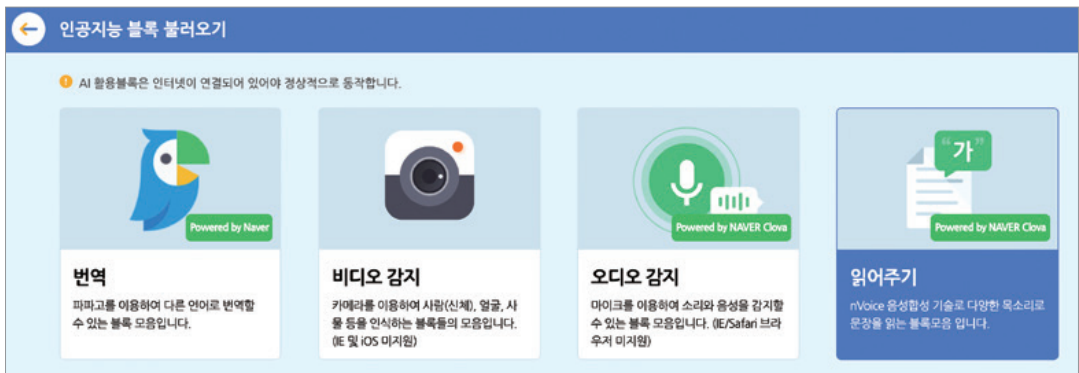
(6) 금정산성 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 금정산성 이미지를 추가하고 금정구 지역에 배치합니다.



2) 인공지능 블록 불러오기

블록 꾸러미에서 블록 탭의 인공지능을 선택하고 인공지능 불러오기 버튼을 눌러 읽어주기를 선택합니다.



3) 리스트 만들기

해운대, 태종대, 금정산성, 용두산공원 여행지를 리스트에 저장합니다. 블록 꾸러미에서 '속성' 탭을 선택하고 '리스트'를 선택하고 [리스트 추가하기] 버튼을 눌러 '관광지' 리스트를 만듭니다.



엔트리 리스트 알아보기

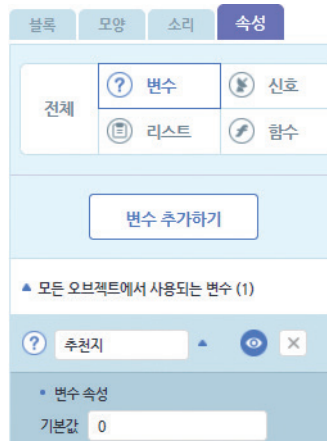
같은 종류의 데이터들이 여러 개일 때, 자료마다 번호를 붙여 '리스트'로 묶어 관리하면 편리합니다. 리스트는 번호를 붙여 관리하는 목록과 같으며, 그 안에는 원하는 만큼의 숫자나 문자를 넣을 수 있습니다. 데이터를 목록으로 만들어 저장하는 공간을 '리스트'라고 하고, 리스트 안에 들어 있는 데이터를 '리스트 항목'이라고 합니다. 또한, 리스트 항목의 위치를 알려주는 번호를 '리스트 번호'라고 합니다. 블록 꾸러미에서 '속성' 탭을 선택하고 '리스트'를 선택하고 [리스트 추가하기] 버튼을 눌러 만들 수도 있고 '자료' 블록 꾸러미에서 [리스트 만들기] 버튼을 눌러 만들 수도 있습니다. 리스트가 생성되면 '자료' 블록 꾸러미에서 리스트와 관련된 블록을 확인할 수 있습니다.



블록	설명
리스트▼ 의 1 번째 항목	선택한 리스트에서 입력한 순서에 있는 항목 값입니다.
10 항목을 리스트▼ 에 추가하기 ?	입력한 값을 선택한 리스트의 마지막 항목에 추가합니다.
1 번째 항목을 리스트▼ 에서 삭제하기 ?	선택한 리스트의 입력한 순서에 있는 항목을 삭제합니다.
10 을(를) 리스트▼ 의 1 번째에 넣기 ?	입력한 값을 선택한 리스트의 입력한 순서의 항목으로 추가합니다. (입력한 항목 뒤에 있는 항목들은 순서가 하나씩 밀려납니다.)
리스트▼ 1 번째 항목을 10 (으)로 바꾸기 ?	선택한 리스트에서 입력한 순서에 있는 항목의 값을 입력한 값으로 바꿉니다.
리스트▼ 항목 수	선택한 리스트가 보유한 항목의 개수입니다.
리스트▼ 에 10 이 포함되어 있는가?	선택한 리스트가 입력한 값을 가진 항목을 포함하는 경우 '참'으로 판단합니다.
리스트 리스트▼ 보이기 ?	선택한 리스트의 '리스트 창'을 실행 화면에 보이게 합니다.
리스트 리스트▼ 숨기기 ?	선택한 리스트의 '리스트 창'을 실행 화면에서 숨깁니다.

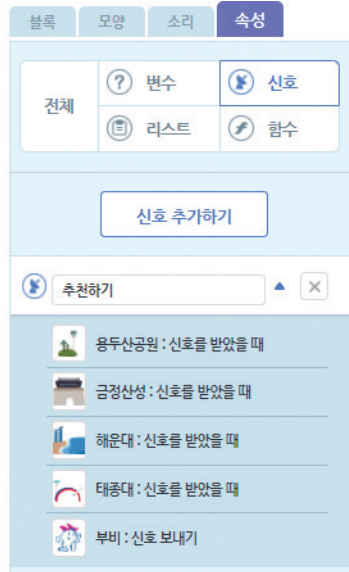
4) 변수 만들기

'관광지' 리스트에서 여행지를 찾기 위한 변수 '추천지'를 작성합니다.



5) 신호 만들기

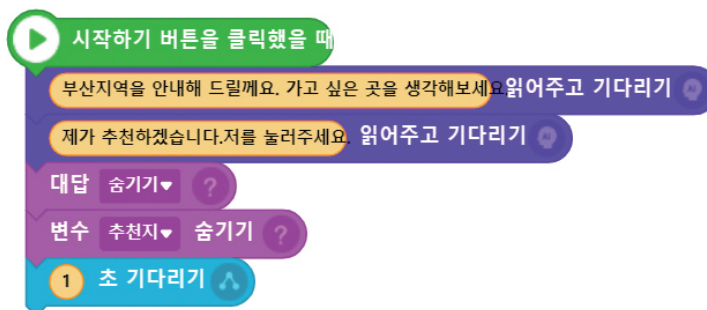
여행지가 결정되면 관광지 오브젝트들에게 신호를 보내어 해당하는 오브젝트를 표시할 수 있도록 '추천하기' 신호를 작성합니다. '부비' 오브젝트가 신호를 보내면 관광지 오브젝트들이 신호를 받습니다.



6) 프로그램 설계하기

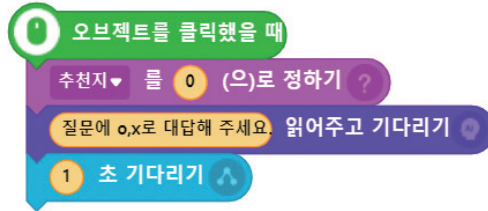
① '부비' 오브젝트가 안내를 시작합니다.

프로그램이 시작되면 '부비' 오브젝트는 관광지 추천을 위해 안내를 합니다. 이때, 질문을 받아 저장하는 '대답'과 '추천지' 변수는 보이지 않도록 숨겨 줍니다.



② ‘부비’ 오브젝트를 누를 때마다 ‘추천지’ 변수를 초기화하고 안내합니다.

‘부비’ 오브젝트를 누를 때마다 새로운 ‘추천지’를 결정할 수 있도록 초기화하고 ‘질문에 o, x로 대답해 주세요.’라고 안내합니다.



③ 의사결정트리를 구현합니다.

의사결정트리를 구현하기 위해 ‘만일 ~이라면 ~ 아니라면’ 블록을 사용하여 구현합니다. 최종적으로 가지가 끝나는 지점(리프 노드)에서 관광지 리스트 위치의 값을 ‘추천지’ 변수에 저장합니다.

의사결정트리가 작성되면 각 관광지 오브젝트로 ‘추천하기’ 신호를 보냅니다.



④ 관광지 오브젝트를 처리합니다.

의사결정트리기가 실행되면 '추천지' 변수에 추천정보가 저장됩니다. '추천하기' 신호를 받은 오브젝트는 자신의 '추천지' 변수의 값을 확인하고 자신에 해당되는 값이면 리스트에서 관광지 이름을 가져와서 읽어주고 크기를 확대합니다.

```

추천하기▼ 신호를 받았을 때
만일 추천지▼ 값 = 1 (이)라면
  관광지▼ 의 추천지▼ 값 번째 항목 읽어주기
  크기를 30 만큼 바꾸기
  1 초 기다리기
  크기를 -30 만큼 바꾸기
  
```

해운대

```

추천하기▼ 신호를 받았을 때
만일 추천지▼ 값 = 2 (이)라면
  관광지▼ 의 추천지▼ 값 번째 항목 읽어주기
  크기를 30 만큼 바꾸기
  1 초 기다리기
  크기를 -30 만큼 바꾸기
  
```

태종대

```

추천하기▼ 신호를 받았을 때
만일 추천지▼ 값 = 3 (이)라면
  관광지▼ 의 추천지▼ 값 번째 항목 읽어주기
  크기를 30 만큼 바꾸기
  1 초 기다리기
  크기를 -30 만큼 바꾸기
  
```

금정산성

```

추천하기▼ 신호를 받았을 때
만일 추천지▼ 값 = 4 (이)라면
  관광지▼ 의 추천지▼ 값 번째 항목 읽어주기
  크기를 30 만큼 바꾸기
  1 초 기다리기
  크기를 -30 만큼 바꾸기
  
```

용두산공원

7) 결과 확인하기





자신만의 해결 문제를 만들어 보세요.



해결 문제 정의하기

Blank writing area with horizontal dashed lines for defining the problem.



해결 문제 작성하기

Blank writing area with horizontal dashed lines for creating the problem.

7장

여기가 부산의 어디야





부산의 상징으로 배우는
엔트리 시

**학습
목표**

- 데이터를 학습하는 머신러닝에 대해 알아봅시다.
- 이미지 학습 모델을 만들어 봅시다.

1**머신러닝**

인공지능은 컴퓨터가 사람처럼 지능적인 행동을 할 수 있도록 하는 모든 것을 의미한다고 했습니다. 앞서 전통적인 방법으로 인공지능을 만드는 논리적 추론에 대해 살펴보았습니다. 최근의 인공지능을 만드는 방법은 컴퓨터를 사람처럼 학습시켜 스스로 규칙을 만들도록 하는 머신러닝 방식을 사용합니다. 머신러닝은 컴퓨터가 데이터로부터 학습하여 스스로 규칙을 찾아가면서 점점 지능적으로 변화하도록 하는 것입니다. 전통적인 인공지능이 사람의 지식을 흉내 내는 것이라면 머신러닝은 사람의 학습을 흉내 내는 것입니다. 머신러닝을 통해 전통적인 인공지능이 가진 한계를 극복하고 있습니다.

인공지능의 학습은 데이터를 통해서 이루어집니다. 머신러닝은 학습하는 방법에 따라 지도 학습(Supervised Learning), 비지도 학습(Unsupervised Learning), 강화 학습(Reinforcement)으로 나뉩니다.

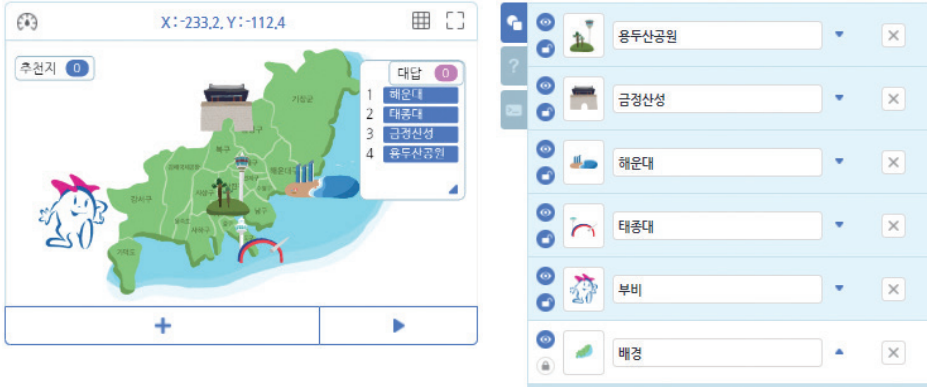
지도 학습은 다양한 데이터를 보여주고 이 데이터들의 정답을 미리 알려주고 학습하는 방법입니다. 이때, 정답을 레이블(label)이라고 합니다. 예를 들면, 여러 종류의 강아지 사진을 보여주고 이 사진은 강아지라고 학습시킵니다. 마찬가지로 여러 종류의 고양이 사진을 보여주고 이 사진은 고양이라고 학습시킵니다. 학습이 완료되면 새로운 사진을 보여줬을 때 강아지인지 고양이인지 구분해 줍니다.

비지도 학습은 정답 없이 학습을 하는 방법입니다. 예를 들면, 강아지와 고양이 사진을 비지도 학습 모델로 학습시키면 패턴이 비슷한 것끼리 분류해 줍니다.

강화 학습은 어떤 환경에서 컴퓨터가 현재의 상태를 인식하여 선택 가능한 행동들 중 보상을 최대화하는 방향으로 행동을 선택합니다. 몇 번의 시행착오를 통해 학습을 하다 보면 컴퓨터는 보상을 최대한으로 많이 받을 수 있도록 행동하게 됩니다. 예를 들어, 벽돌 깨기 게임을 통해 벽돌을 깨면 점수를 주고 바닥에 공이 닿으면 벌점을 주면 인공지능은 게임을 하면서 어떻게 하면 점수를 많이 얻을 수 있을지 학습을 하여 효율적인 게임을 하게 됩니다.

2

해결 문제: 여기가 부산의 어디야



이번 장에서는 관광지의 사진을 지도 학습의 이미지 분류 모델로 학습을 시킵니다. 각 관광지의 사진을 학습한 후 새로운 사진을 보여주고 어디인지 알아보도록 작성합니다.

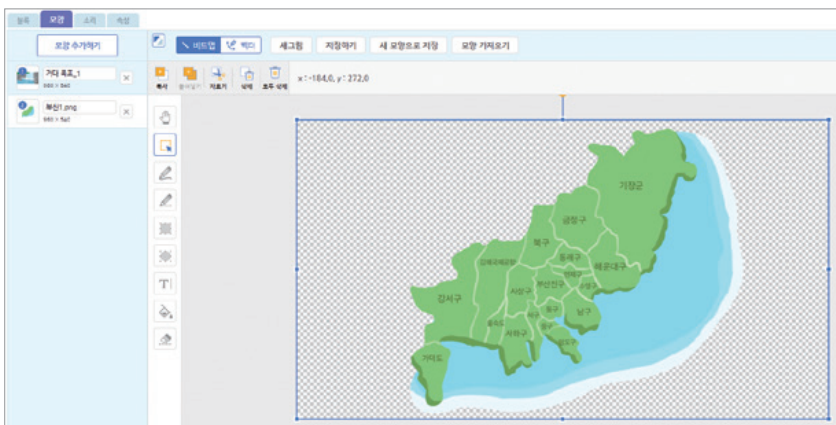
3

엔트리로 만들어 봅시다.

1) 화면 구성하기

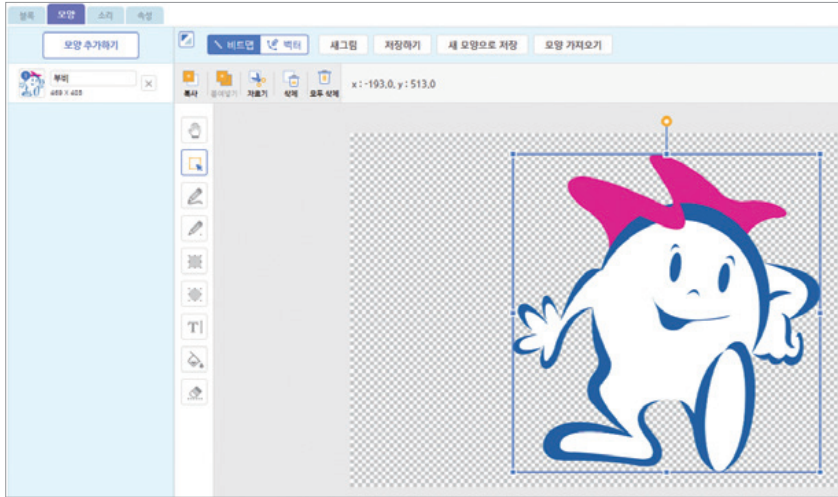
(1) 배경 추가하기

오브젝트를 추가하여 [모양 추가하기] 버튼을 눌러 [배경] 항목을 선택하고 파일 올리기를 이용하여 배경을 선택합니다.



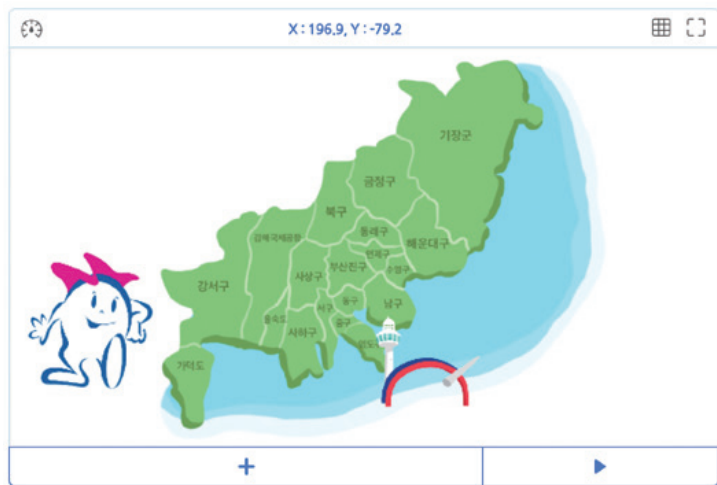
(2) 부비 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 부비 이미지를 추가합니다.



(3) 태종대 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 태종대 이미지를 추가하고 영도 지역에 배치합니다.



(4) 해운대 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 해운대 이미지를 추가하고 해운대 지역에 배치합니다.



(5) 용두산공원 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 용두산공원 이미지를 추가하고 중구 지역에 배치합니다.



(6) 금정산성 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 금정산성 이미지를 추가하고 금정구 지역에 배치합니다.




2) 인공지능 블록 불러오기

블록 꾸러미에서 블록 탭의 인공지능을 선택하고 인공지능 불러오기 버튼을 눌러 읽어주기를 선택합니다.

←
인공지능 블록 불러오기


! AI 활용블록은 인터넷이 연결되어 있어야 정상적으로 동작합니다.



Powered by Naver

번역


파피고를 이용하여 다른 언어로 번역할 수 있는 블록 모음입니다.



Powered by NAVER Clova

비디오 감지


카메라를 이용하여 사람(신체), 얼굴, 사물 등을 인식하는 블록들의 모음입니다. (E 및 iOS 미지원)



Powered by NAVER Clova

오디오 감지

마이크를 이용하여 소리와 음성을 감지할 수 있는 블록 모음입니다. (E/Safari 브라우저 우저 미지원)



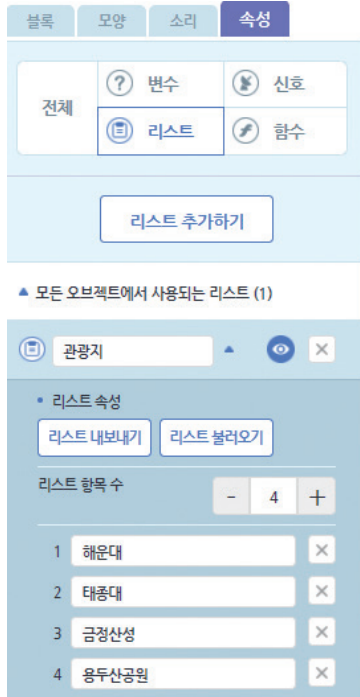
Powered by NAVER Clova

읽어주기

nVoice 음성합성 기술로 다양한 목소리로 문장을 읽는 블록모음입니다.

3) 리스트 만들기

해운대, 태종대, 금정산성, 용두산공원 여행지를 리스트에 저장합니다. 블록 꾸러미에서 '속성' 탭을 선택하고 '리스트'를 선택하고 [리스트 추가하기] 버튼을 눌러 '관광지' 리스트를 만듭니다.



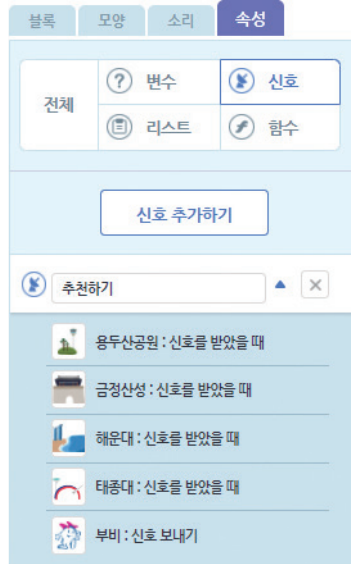
4) 변수 만들기

'관광지' 리스트에서 여행지를 찾기 위한 변수 '추천지'를 작성합니다.



5) 신호 만들기

여행지가 결정되면 관광지 오브젝트들에게 신호를 보내어 해당하는 오브젝트를 표시할 수 있도록 '추천지' 변수를 작성합니다. '부비' 오브젝트가 신호를 보내면 관광지 오브젝트들이 신호를 받습니다.



6) 함수 만들기

관광지를 확인하는 '관광지보기' 함수를 작성합니다.





함수 알아보기

함수란 어떤 일을 수행하는 명령들을 하나로 묶어서 이름을 부여하고 필요할 때마다 함수 이름을 호출하여 사용할 수 있는 프로그래밍 방법입니다. 반복적으로 사용되는 일을 함수로 한번 만들어 두면 계속해서 필요할 때마다 쓸 수 있어 블록 조립에 걸리는 시간이 짧아집니다.

예를 들면, 관광지 오브젝트 리스트에서 해당하는 항목의 이름을 가져와서 읽어주고 크기를 변경하는 코드는 모두 같습니다. 이 부분을 함수로 만들어 보겠습니다.

'함수' 블록 꾸러미에서 [함수 만들기] 버튼을 눌러 함수를 작성합니다. 함수 이름은 '관광지보기'로 작성하고 공통적으로 사용되는 블록을 함수 블록에 추가합니다.



함수로 작성한 후 각 관광지 오브젝트의 코드를 다음과 같이 수정하면 됩니다.



해운대



태종대



금정산성



용두산공원

7) 학습 모델 만들기

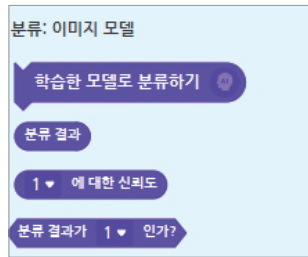
‘인공지능’ 블록 꾸러미에서 [인공지능 모델 학습하기] 버튼을 누릅니다. ‘새로 만들기’ 탭을 선택하고 ‘분류: 이미지’를 선택하고 [학습하기] 버튼을 누릅니다.



각 관광지에 대한 이미지 학습을 위한 이미지를 미리 찾아 놓도록 합니다. 학습하기 화면에서 각 관광지에 대한 사진을 올려줍니다. 이때, 관광지에 대한 레이블 값으로 ‘관광지’ 리스트 위치의 값을 사용합니다. 이 레이블을 ‘추천지’ 변수의 값으로 사용합니다. 데이터 입력이 완료되고 [모델 학습하기] 버튼을 클릭하여 학습이 완료되면 결과에서 파일을 업로드하거나 촬영하여 학습된 결과를 테스트해 볼 수 있습니다.



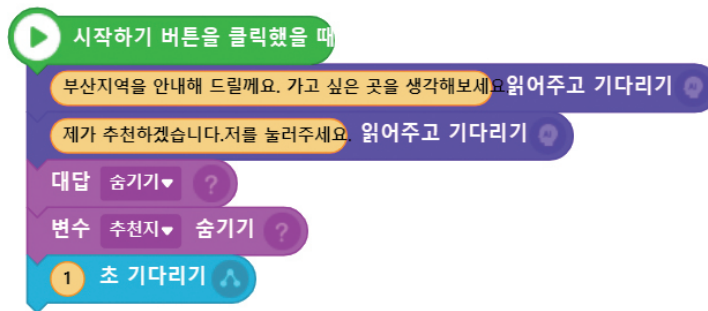
이미지 모델이 학습이 완료되면 '인공지능' 블록 꾸러미에 학습과 관련된 블록들이 생성됩니다.



8) 프로그램 설계하기

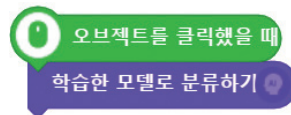
① '부비' 오브젝트가 안내를 시작합니다.

프로그램이 시작되면 '부비' 오브젝트는 관광지 추천을 위해 안내를 합니다. 이때, 질문을 받아 저장하는 '대답'과 '추천지' 변수는 보이지 않도록 숨겨 줍니다.



② '부비' 오브젝트를 누르면 학습을 기준으로 분류합니다.

'부비' 오브젝트를 누르면 앞서 학습한 이미지 모델을 통해 생성된 블록에서 '학습한 모델로 분류하기' 블록을 이용하여 분류합니다.



③ 추천지를 결정하고 추천 신호를 보냅니다.

학습한 모델로 분류가 되면 분류 결과를 반환합니다. 반환된 분류 결과를 '추천지' 변수에 저장하고 '추천하기' 신호를 보냅니다.



④ 관광지 오브젝트를 처리합니다.

의사결정트리가 실행되면 '추천지' 변수에 추천 정보가 저장됩니다. '추천하기' 신호를 받은 오브젝트는 자신의 '추천지' 변수를 확인하고 자신에 해당되는 값이면 리스트에서 관광지 이름을 가져와서 읽어주고 크기를 확대합니다.



해운대



태종대



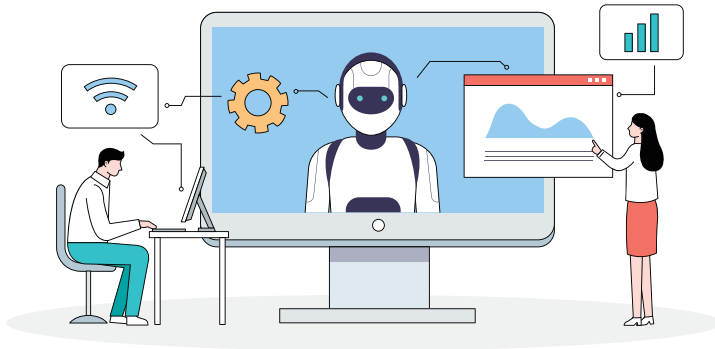
금정산성



용두산공원

9) 결과 확인하기

The screenshot displays a web application interface for data entry. The main window features a map of Busan with a list of landmarks: 1. 해운대 (Haeundae), 2. 테총대 (Tejondae), 3. 금강산성 (Gyeonggangsanseong), and 4. 용두산공원 (Yongdusan Park). A sidebar on the right is titled '데이터 입력' (Data Entry) and contains an '업로드' (Upload) button, a photo of Haeundae Lighthouse, an '적용하기' (Apply) button, and a legend for '작품 정지하기' (Pause) and '작품 일시정지' (Stop).





자신만의 해결 문제를 만들어 보세요.



해결 문제 정의하기

Blank writing area with horizontal dashed lines for defining the problem.



해결 문제 작성하기

Blank writing area with horizontal dashed lines for writing the solution to the problem.

8장

부산의 여행지를 추천해줘





부산의 상징으로 배우는
엔트리 AI



학습 목표

- 데이터의 편향성에 대해 알아봅시다.
- 텍스트 학습 모델을 만들어 봅시다.

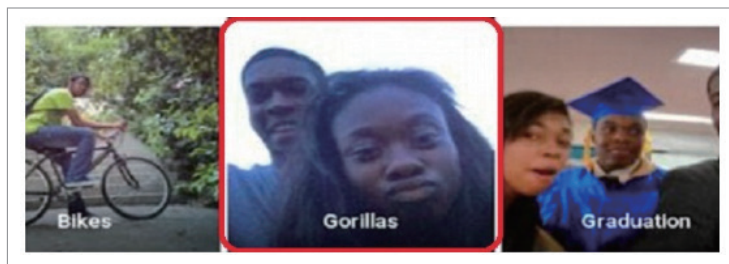
1 데이터의 편향성

앞서 머신러닝의 지도 학습은 다양한 데이터를 보여주고 이 데이터들의 정답을 미리 알려주고 학습하는 방법이라고 했습니다. 지도 학습 모델을 사용하면 무언가를 예측하거나 분류를 할 수 있습니다.

인공지능을 활용한 예측과 분류는 어떤 차이가 있을까요? 예측은 데이터를 학습하여 숫자를 예측합니다. 예를 들면, 과거의 성적 데이터를 통해 앞으로의 시험 성적을 예측하는 것입니다. 분류는 어떤 대상이 두 개 이상의 그룹 중에서 어떤 그룹에 속하는지 구분하는 것입니다. 앞서 작성해 본 이미지를 보고 관광지를 찾아주는 것도 분류의 문제입니다.

이처럼 예측과 분류의 문제는 필요한 적절한 데이터를 준비하는 것이 중요합니다. 데이터가 적절하지 못하면 정확한 예측과 분류가 어렵습니다.

예를 들면, 인공지능 기술을 활용하여 자동 태깅과 분류 기능을 서비스한 구글 포토는 2015년 6월 미국에 사는 흑인 남성이 흑인 친구와 찍은 사진을 고릴라라고 분류한 사실이 알려져 인종차별 논란으로 이어졌습니다.



출처: <http://www.hani.co.kr/arti/science/future/877637.html>

이러한 사회적, 윤리적 문제들은 학습 데이터의 부족, 편향성, 사용된 알고리즘의 왜곡과 편향으로 인해 지속적으로 발생되고 있습니다. 결국 인공지능을 만들 때 사람이 어떻게 만드는지에 따라 선택 방향이 결정됩니다. 다가올 인공지능 시대에 새로운 윤리의식이 필요한 이유입니다.

2

해결 문제: 부산의 여행지를 추천해줘



이번 장에서는 부산을 관광하고 싶은 사람이 원하는 정보를 텍스트로 입력하면 입력된 텍스트 데이터를 학습하여 분류한 후 관광지를 추천해 보도록 작성합니다.

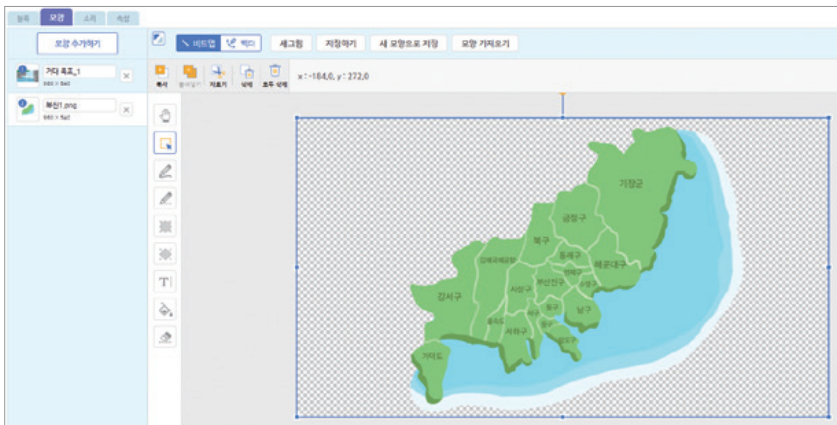
3

엔트리로 만들어 봅시다

1) 화면 구성하기

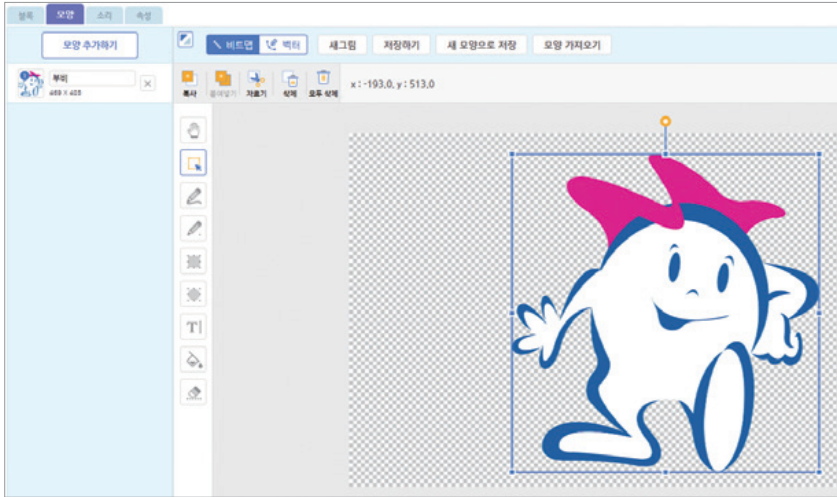
(1) 배경 추가하기

오브젝트를 추가하여 [모양 추가하기] 버튼을 눌러 [배경] 항목을 선택하고 파일 올리기를 이용하여 배경을 선택합니다.



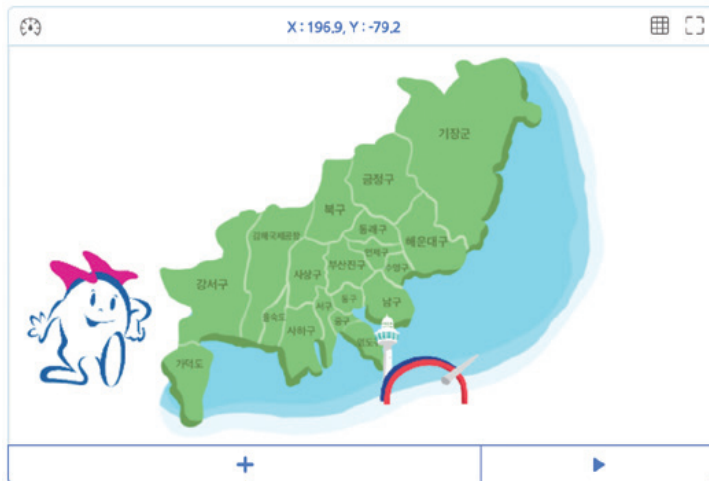
(2) 부비 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 부비 이미지를 추가합니다.



(3) 태종대 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 태종대 이미지를 추가하고 영도 지역에 배치합니다.



(4) 해운대 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 해운대 이미지를 추가하고 해운대 지역에 배치합니다.



(5) 용두산공원 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 파일 올리기를 이용하여 용두산공원 이미지를 추가하고 중구 지역에 배치합니다.



(6) 금정산성 오브젝트

오브젝트 추가하기(+)를 선택하고 블록 꾸러미의 모양 탭에서 [모양 추가하기] 버튼을 누르고 과일 올리기를 이용하여 금정산성 이미지를 추가하고 금정구 지역에 배치합니다.

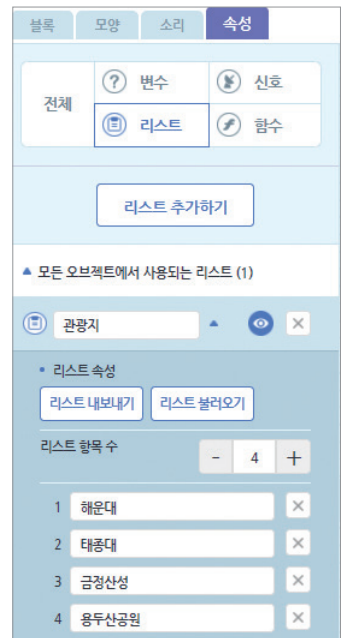


2) 인공지능 블록 불러오기

블록 꾸러미에서 블록 탭의 인공지능을 선택하고 인공지능 불러오기 버튼을 눌러 읽어주기를 선택합니다.

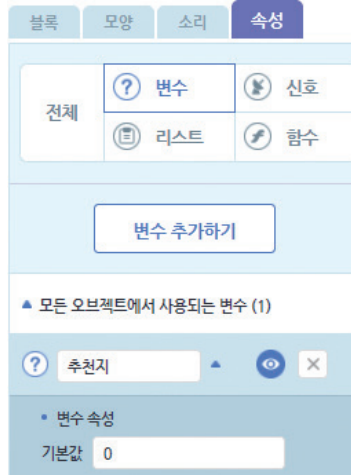
3) 리스트 만들기

해운대, 태종대, 금정산성, 용두산공원 여행지를 리스트에 저장합니다. 블록 꾸러미에서 '속성' 탭을 선택하고 '리스트'를 선택하고 [리스트 추가하기] 버튼을 눌러 '관광지' 리스트를 만듭니다.



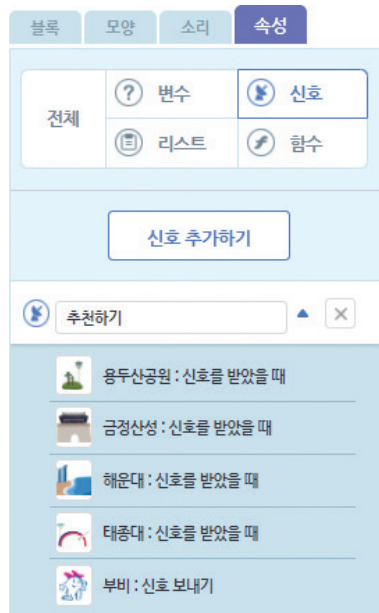
4) 변수 만들기

'관광지' 리스트에서 여행지를 찾기 위한 변수 '추천지'를 작성합니다.



5) 신호 만들기

여행지가 결정되면 관광지 오브젝트들에게 신호를 보내어 해당하는 오브젝트를 표시할 수 있도록 '추천지' 변수를 작성합니다. '부비' 오브젝트가 신호를 보내면 관광지 오브젝트들이 신호를 받습니다.



6) 함수 만들기

관광지를 확인하는 ‘관광지보기’ 함수를 작성합니다.



7) 학습 모델 만들기

‘인공지능’ 블록 꾸러미에서 [인공지능 모델 학습하기] 버튼을 누릅니다. ‘새로 만들기’ 탭을 선택하고 ‘분류: 텍스트’를 선택하고 [학습하기] 버튼을 누릅니다.



각 관광지에 대한 텍스트 자료를 작성합니다. 이때, 관광지에 대한 레이블 값으로 ‘관광지’ 리스트 위치의 값을 사용합니다. 이 레이블을 추천지 변수의 값으로 사용합니다. 데이터 입력이 완료되고 [모델 학습하기] 버튼을 클릭하여 학습이 완료되면 학습된 결과를 테스트해 볼 수 있습니다.

← 분류: 텍스트 모델 학습하기

1 모델 학습은 인터넷이 연결되어 있어야 정상적으로 동작합니다. 튜토리얼 보기

자유학년제-텍스트

데이터 입력 + 클래스 추가하기

1 5개 ×

모델이 학습할 텍스트 데이터를 아래에 작성해 주세요.
클래스 및 5개 이상의 데이터를 입력해야 합니다.
각각의 데이터는 필드로 구분합니다. (예: 맛있는. 맛있는)

모래사장을 걷고 싶어, 해수욕을 하고 싶어, 아쿠아리움을 보고 싶어, 누리마루에 가고 싶어, 바닷가를 산책하고 싶어

2 6개 ×

모델이 학습할 텍스트 데이터를 아래에 작성해 주세요.
클래스 및 5개 이상의 데이터를 입력해야 합니다.
각각의 데이터는 필드로 구분합니다. (예: 맛있는. 맛있는)

절벽을 보고 싶어, 태종 무열왕에 대해 알고 싶어, 등대를 보고 싶어, 열차를 타고 싶어, 바위를 보고 싶어, 바다를 보고 싶어

3 5개 ×

모델이 학습할 텍스트 데이터를 아래에 작성해 주세요.
클래스 및 5개 이상의 데이터를 입력해야 합니다.
각각의 데이터는 필드로 구분합니다. (예: 맛있는. 맛있는)

성곽을 보고 싶어, 임진왜란에 대해 알고 싶어, 산성마을에 가고 싶어, 막걸리를 먹고 싶어, 산악자전거를 타고 싶어

4 5개 ×

모델이 학습할 텍스트 데이터를 아래에 작성해 주세요.
클래스 및 5개 이상의 데이터를 입력해야 합니다.
각각의 데이터는 필드로 구분합니다. (예: 맛있는. 맛있는)

타워에 올라가고 싶어, 공원을 산책하고 싶어, 꽃시계를 보고 싶어, 이순신장

학습

입력한 데이터와 조건으로 모델을 학습합니다.

모델 학습하기

학습을 완료했습니다.

결과

학습한 모델의 결과를 확인해 보세요.

입력하기

입력된 텍스트 : 모래사장에 가고 ...

1	50.23%
3	23.74%
2	14.13%
4	11.87%



엔트리 텍스트 모델 학습 데이터

각 관광지를 분류하기 위한 텍스트 자료를 다음과 같이 사용하였습니다. 자신만의 자료로 변경해 보세요.

■ 해운대

모래사장을 걷고 싶어, 해수욕을 하고 싶어, 아쿠아리움을 보고 싶어, 누리마루에 가고 싶어, 바닷가를 산책하고 싶어

■ 금정산성

성곽을 보고 싶어, 임진왜란에 대해 알고 싶어, 산성마을에 가고 싶어, 산악자전거를 타고 싶어

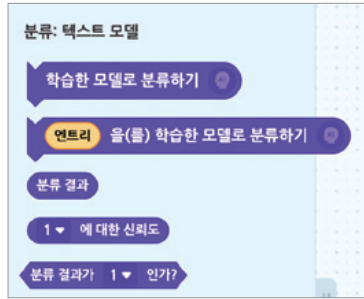
■ 태종대

절벽을 보고 싶어, 태종 무열왕에 대해 알고 싶어, 등대를 보고 싶어, 열차를 타고 싶어, 바위를 보고 싶어, 바다를 보고 싶어

■ 용두산공원

타워에 올라가고 싶어, 공원을 산책하고 싶어, 꽃시계를 보고 싶어, 이순신 장군 동상을 보고 싶어, 용탑을 보고 싶어

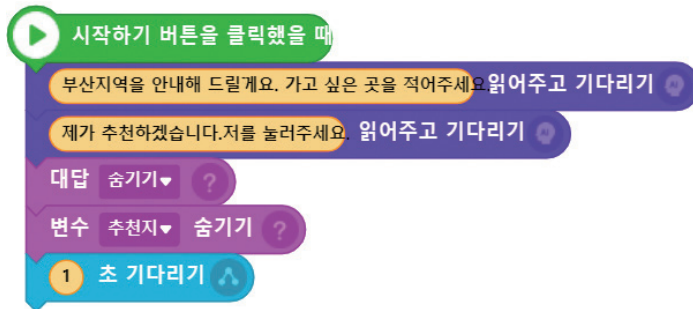
텍스트 모델의 학습이 완료되면 ‘인공지능’ 블록 꾸러미에 학습과 관련된 블록들이 생성됩니다.



8) 프로그램 설계하기

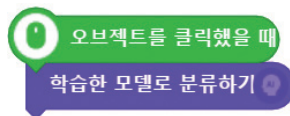
① ‘부비’ 오브젝트가 안내를 시작합니다.

프로그램이 시작되면 ‘부비’ 오브젝트는 관광지 추천을 위해 안내를 합니다. 텍스트 모델을 사용하므로 안내에 ‘부산지역을 안내해 드릴게요. 가고 싶은 곳을 적어주세요.’라고 합니다.



② ‘부비’ 오브젝트를 누르면 학습을 기준으로 분류합니다.

‘부비’ 오브젝트를 누르면 앞서 학습한 이미지 모델을 통해 생성된 블록에서 ‘학습한 모델로 분류하기’ 블록을 이용하여 분류합니다.



③ 추천지를 결정하고 추천 신호를 보냅니다.

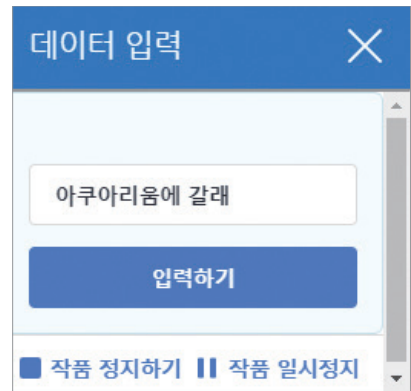
학습한 모델로 분류가 되면 분류 결과를 반환합니다. 반환된 분류 결과를 '추천지' 변수에 저장하고 '추천하기' 신호를 보냅니다.



④ 관광지 오브젝트를 처리합니다.

의사결정트리가 실행되면 '추천지' 변수에 추천정보가 저장됩니다. '추천하기' 신호를 받은 오브젝트는 자신의 '추천지' 변수의 값을 확인하고 자신에 해당하는 값이면 리스트에서 관광지 이름을 가져와서 읽어주고 크기를 확대합니다.

9) 결과 확인하기





자신만의 해결 문제를 만들어 보세요.



해결 문제 정의하기

A large rectangular area with horizontal dashed lines for writing, intended for defining the problem.



해결 문제 작성하기

A large rectangular area with horizontal dashed lines for writing, intended for writing the solution to the problem.

