

## 1. 개요

<b>프로그램 개발 의도</b>	컴퓨터 비전을 활용한 재미를 위한 간단한 앱을 만들어 봄으로써, 앱 제작, 컴퓨터 비전, 인공지능 학습 및 모델 생성, 배포의 전반적인 과정을 학습하고, 개발된 모델을 배포함으로써 학습자의 역량 향상이 곧 관련 생태계 조성에 직접적인 영향을 줄 수 있음을 배운다.
<b>프로그램 목표</b>	12간지 사진을 모으고, 정선한 다음 컴퓨터 비전을 이용해 학습을 하고 모델을 만들어낸다. 앱을 제작해 인물 사진을 찍으면 그 사람이 12간지 중에서 어떤 동물과 닮았는지 보여준다. 만들어진 앱을 배포해 다른 사람들이 활용할 수 있도록 공유한다.
<b>인공지능 연계 요소</b>	머신러닝(CNN), 컴퓨터 비전, 빅 데이터
<b>관련 교과</b>	공학, 기술, 정보
<b>성취 기준</b>	<p>[기가0212-2] 기술적 문제를 문제해결 과정을 활용하여 창의 공학 설계 프로젝트를 수행할 수 있다.</p> <p>[12공학02-01] 정보통신 공학의 원리를 이해하고 동향을 파악하여, 그 원리가 응용된 사례를 탐색하고 체험한다.</p> <p>[11지능02-01] 다양한 데이터를 획득하고 데이터의 특징이 나타나도록 시각화한다.</p> <p>[11지능02-02] 데이터 분석 프로세스를 분석하고 문제 해결에 어떤 데이터가 필요한지 분석한다.</p> <p>[11지능03-03] 분류 문제와 회귀 문제를 해결하는 지도 학습 알고리즘을 설계하고 프로그램을 작성한다.</p> <p>[11지능04-01] 인문사회, 자연과학, 예술체육 분야의 문제를 분석하고 융합적으로 해결하기 위한 인공지능 모델을 협력적으로 설계하고 프로그래밍 언어를 이용하여 구현하고 배포한다.</p>

## 2. 구성

단계	내용
<b>공감하기 (Empathize)</b>	<ul style="list-style-type: none"> <li>· 사람들은 자신의 누구나 자신의 외모에 관심을 갖는 시기를 거친다.</li> <li>· 외모에 대해서 너무 자부심을 가질 필요도, 거부감을 가질 필요도 없음을 이해할 필요가 있다.</li> <li>· 자신의 외모를 12간지와 비교해주는 간단한 앱을 제작해보는 경험을 통해 사람의 외모는 차이를 보여줄 뿐, 차별의 대상이 아님을 이해한다.</li> </ul>
<b>문제 정의하기 (Define)</b>	<ul style="list-style-type: none"> <li>· 나의 외모가 12간지 중에서 어떤 동물과 닮았는지 확인해 볼 수 있는 앱을 만들어본다.</li> <li>· 스마트폰에서 구동되는 앱을 누구나 만들 수 있음을 안다.</li> </ul>
<b>아이디어내기 (Ideate)</b>	<ul style="list-style-type: none"> <li>· 12간지와 비교하기 위해서 데이터를 모으려면 어떻게 해야 하는가?</li> <li>· 모아진 데이터를 잘 선별하려면 어떻게 해야 하는가?</li> <li>· 앱을 만들 때 12간지 닮은꼴을 사용자에게 어떻게 표현하면 좋을까?</li> </ul>
<b>시제품 만들기 (Prototype)</b>	<ul style="list-style-type: none"> <li>· Personal Image Classifier를 이용해 12간지를 학습시킨다.</li> <li>· 학습시킨 모델 파일을 다운로드 받는다.</li> <li>· 앱 인벤터에 모델파일을 업로드 하고 앱을 제작한다.</li> </ul>
<b>평가하기 (Test)</b>	<ul style="list-style-type: none"> <li>· 서로 자신의 만든 작품을 설명하고, 앱을 테스트 해본다.</li> <li>· 앱이 잘 동작하면, 해당 앱을 공유한다.</li> </ul>

### 3. 수업 지도안

관련 교과	공학, 기술, 정보	대 상	고 1학년	수업 차시	1~4/4차시
수업 주제	인공지능 12간지 닭은꼴 찾기				
학습 목표	지도 학습을 통한 머신러닝을 활용해 이미지 분류를 이해하고, 이를 활용한 앱을 제작한다.				

단계	교수 학습 활동	준비물 (★) 및 지도상 유의점 (☆)
도입	<b>공감하기</b> 이미지 분류를 위한 머신러닝을 이해하고 어느 곳에 사용할 수 있는지 생각해보기(20분) <ul style="list-style-type: none"> <li>■ 이미지 분류 관련 영상 자료 찾아보기</li> <li>■ 이미지 분류 머신러닝을 활용할 수 있는 곳을 상상해보고 발표하기</li> </ul>	☆본 활동은 PC로 인터넷 활용이 가능한 교실에서 진행함
전개	<b>문제정의하기</b> 이미지 분류 인공지능을 이용한 앱 솔루션을 직접 제작하려면 어떻게 해야할까?(30분) <ul style="list-style-type: none"> <li>■ 이미지 분류는 무엇일까?</li> <li>■ 이미지 데이터는 어떻게 모을 수 있을까?</li> <li>■ 이미지 데이터를 학습시키려면 어떻게 해야할까?</li> <li>■ 스마트폰 앱을 만들려면 어떻게 해야할까?</li> </ul>	★ 학생활동으로 구성함
	<b>아이디어내기</b> 사람들에게 즐거움을 줄 수 있는 앱을 직접 만들어보자(30분) <ul style="list-style-type: none"> <li>■ (예시) 12간지 닭은꼴 찾기를 수행할 수 있는 앱을 만들어보자.</li> <li>■ 앱을 만들려면 어떻게 해야할까?</li> <li>■ python, crawling, machine learning, app inventor의 개념을 알아보자.</li> </ul>	★ 인공지능을 활용한 이미지 분류 기술을 이용해 사람들에게 즐거움을 줄 수 있는 것이 무엇이 있을지 고민해보는 시간이 필요함
	<b>시제품만들기</b> 구상한 아이디어를 구체화하여 시제품을 만들기(100분) <ul style="list-style-type: none"> <li>■ python 설치하기</li> <li>■ crawling으로 데이터 모으기</li> <li>■ machine learning으로 학습하기</li> <li>■ app inventor로 앱 제작하기</li> </ul>	★ python을 설치할 때 path 지정에 유의함 ★ PersonallImageClassifier <a href="https://classifier.appinventor.mit.edu">https://classifier.appinventor.mit.edu</a> 사이트를 이용해 데이터를 학습하고 모델을 만듦 ★ 앱인벤터에 학습시킨 모델을 넣어 앱을 제작함
정리	<b>평가하기</b> 평가 및 공유하기(20분) <ul style="list-style-type: none"> <li>■ github를 이용해 자신이 만든 앱과 관련 자료를 배포한다.</li> <li>■ 각자 만든 앱을 서로 테스트 해보고 리뷰한다.</li> </ul>	★ 서로의 앱을 테스트 해보고, 리뷰함

#### 4. 과정중심평가 계획

평가영역 및 요소	평가 기준	우수	보통	미흡
문제의 이해	이미지 분류, 데이터 획득, 학습, 솔루션 구현의 과정을 이해하고 있는가?			
아이디어 내기	이미지 분류를 이용한 적절한 분야를 찾았는가?			
제작 활동	이미지 분류를 이용한 앱을 제작하여 자신의 아이디어를 구현하고 문제를 해결하였는가?			

## 과정중심 평가지

### ☒ 관찰 평가 양식 예시

항목(예시)	평가요소	우수	보통	미흡
문제의 이해	이미지 분류, 데이터 획득, 학습, 솔루션 구현의 과정을 이해하고 있는가?			
아이디어 내기	이미지 분류를 이용한 적절한 분야를 찾았는가?			
제작 활동	이미지 분류를 이용한 앱을 제작하여 자신의 아이디어를 구현하고 문제를 해결하였는가?			

### ☒ 포트폴리오(github, 웹사이트 등) 평가 양식 예시

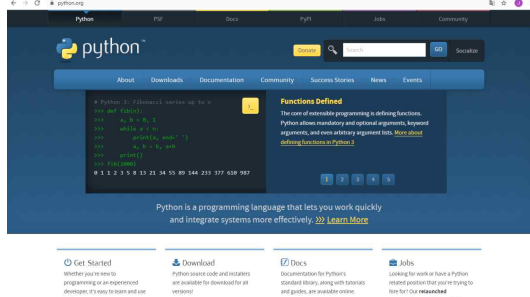
항목(예시)	평가요소	우수	보통	미흡
작품 소개	자신의 작품을 이해하고, 쉽고 간결하게 설명하고 있는가?			
설계	최종 솔루션까지 체계적으로 설계되어 있는가?			
제작과정	작품의 제작과정이 상세하여 이해하기 쉬운가?			
결과	최종 결과를 잘 설명하고 있는가?			

### ☒ 완제품 평가 양식 예시

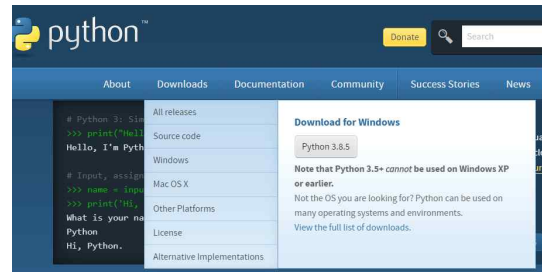
항목(예시)	평가요소	우수	보통	미흡
기능성	작품이 원하는 의도대로 작동하고 있는가?			
심미성	자신이 구현하고자 하는 솔루션에 심미적 요소가 적용되어 있는가?			

## 5. 교수 · 학습 자료

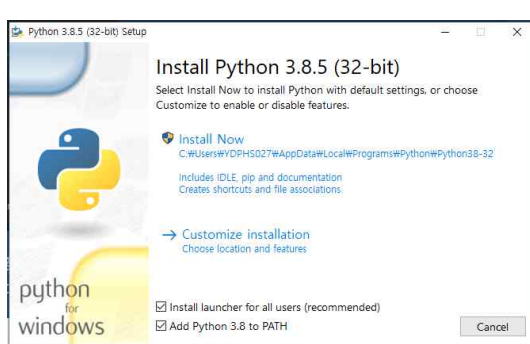
## python 설치와 crawling (<https://github.com/mtinet/chineseZodiac> 을 참고하세요.)



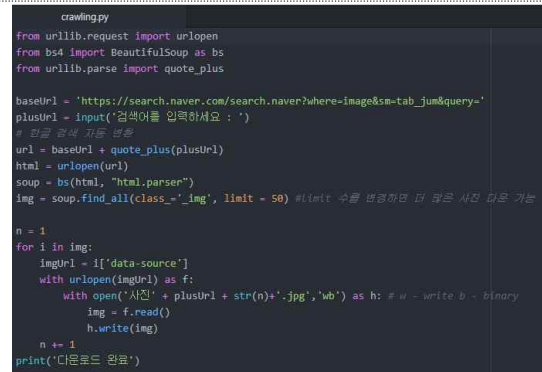
<https://www.python.org/>



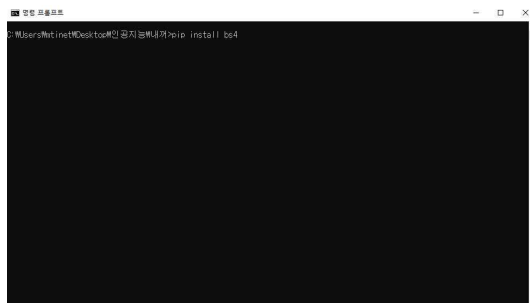
사용하는 운영체제에 맞는 버전의  
python을 다운로드



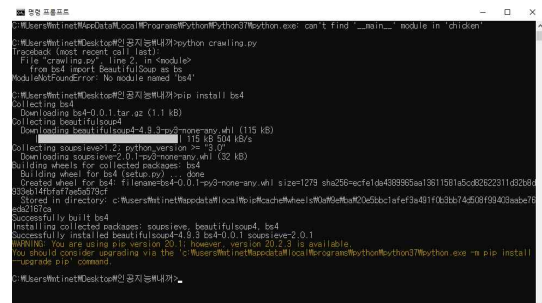
Add Python 3.8 to PATH를  
체크하고 설치



crawling을 위한 python 코드  
<https://github.com/mtinet/chineseZodiac>



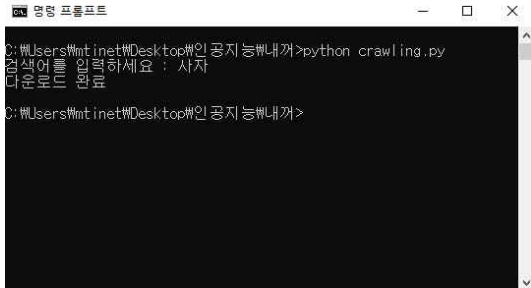
library 설치  
(cmd를 열고 pip install bs4 입력)



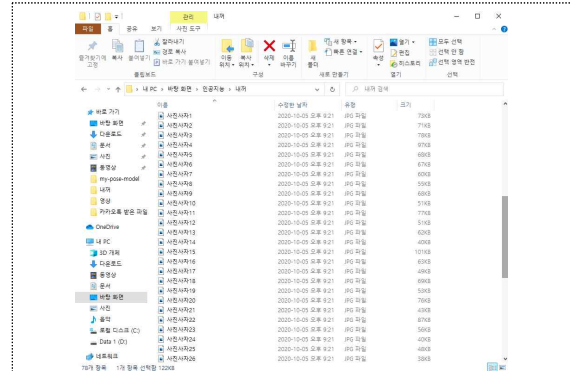
library 설치 완료

## 5. 교수 · 학습 자료

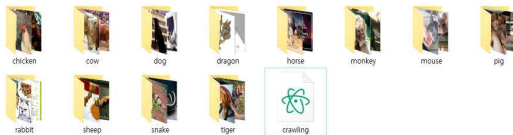
## python 설치와 crawling, 이미지 분류를 위한 머신러닝 (Personal Image Classifier)



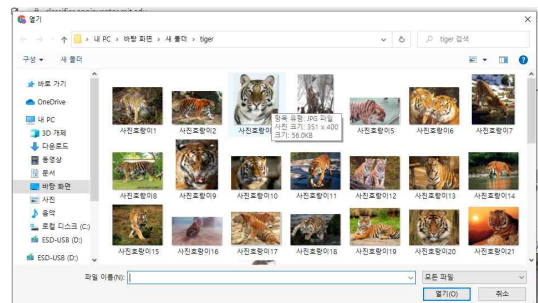
python을 이용해 crawling 코드가  
들어있는 파일을 실행-검색어 입력  
(python crawling.py)



crawling 결과로 다운로드 된 이미지  
파일



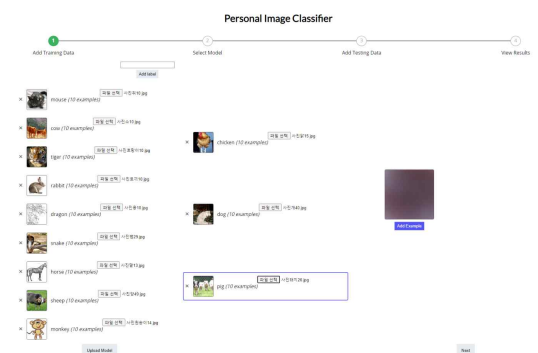
crawling이 완료되어 정리한 이미지  
데이터 폴더



폴더에 저장된 자료가 적당한지  
확인하고 자료를 정선했음



personal image classifier에  
레이블을 생성하고 이미지 데이터를  
업로드 하기



각각의 모든 레이블에 데이터를 추가

## 5. 교수 · 학습 자료

# 이미지 분류를 위한 머신러닝 (Personal Image Classifier)

## Personal Image Classifier

1

2

3

4

Add Training Data

Select Model

Add Testing Data

View Results

Choose Model: MobileNet

Create Model:

Convolution 5 5 1

7,7,256 -> 3,3,5

Flatten Remove Layer: 3,3,5 -> 45

Fully Connected 100 Remove Layer: 45 -> 100

Fully Connected 100 -> Number of Labels

Add Layer

Hyperparameters:

- Learning Rate: 0.0001

- Epochs: 20

- Training Data Fraction: 0.4

- Optimizer: Adam

Train model

Training Time: 00:00:00.000

Back

Next

모델 옵션을 선택하고 Train Model  
버튼을 눌러 학습함

**Personal Image Classifier**


1. Add Training Data

- mouse (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- cow (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- tiger (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- rabbit (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- dragon (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- snake (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- horse (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- sheep (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- monkey (2 examples)  $\frac{100}{100} = 1.0000$  jpg

2. Select Model

- ☐ chicken (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- ☐ dog (2 examples)  $\frac{100}{100} = 1.0000$  jpg
- ☐ pig (2 examples)  $\frac{100}{100} = 1.0000$  jpg

3. Add Testing Data

- ☐  [Download](#)

4. View Results

Done adding testing examples?

학습된 모델을 테스트 하기 위한  
테스트 데이터를 업로드

**1 Add Training Data**

- image (7 examples) 0.8 KB → 0.812 giga
- tree (7 examples) 0.8 KB → 0.847 giga
- tiger (7 examples) 0.8 KB → 0.819 giga
- table (7 examples) 0.8 KB → 0.8113 giga
- cat (7 examples) 0.8 KB → 0.8102 giga
- snake (7 examples) 0.8 KB → 0.8102 giga
- snake (7 examples) 0.8 KB → 0.8102 giga
- monkey (7 examples) 0.8 KB → 0.8103 giga

**2 Select Model**

0.000000 (7 examples) 0.8 KB → 0.8100 giga

**3 Add Testing Data**

dog (7 examples) 0.8 KB → 0.8101 giga

**4 View Results**

Don't add more images?

**5 Download**

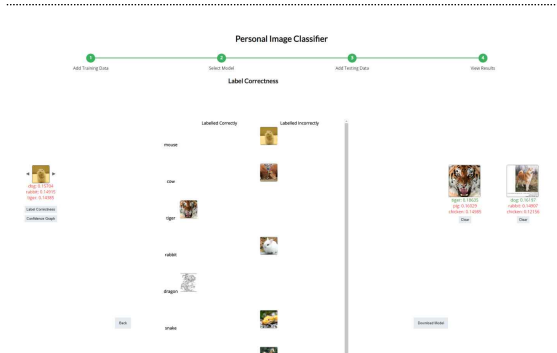
테스트 데이터를 업로드 완료

The screenshot displays the 'Personal Image Classifier' application. At the top, a horizontal timeline with four green circular markers indicates the workflow: 'Add Training Data', 'Select Models', 'Train/Testing Data', and 'View Results'. Below the timeline, the 'Label Correctness' section is divided into two columns. The 'Labelled Correctly' column contains four image thumbnails, each with a red 'Correct' label. The 'Labelled Incorrectly' column contains four image thumbnails, each with a red 'Incorrect' label. A vertical scrollbar is positioned between these two columns. At the bottom left, there is a 'Train' button, and at the bottom right, there is a 'Download Model' button. The interface is clean and modern, with a white background and a light gray border.

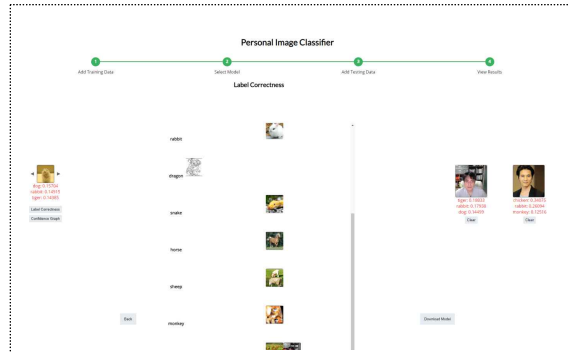
## 테스트 데이터가 잘 분류되는지 확인

## 5. 교수 · 학습 자료

# 이미지 분류를 위한 머신러닝 (Personal Image Classifier) 앱인벤터를 이용한 앱 제작 (<https://github.com/mtinet/chineseZodiac> 을 참고하세요.)



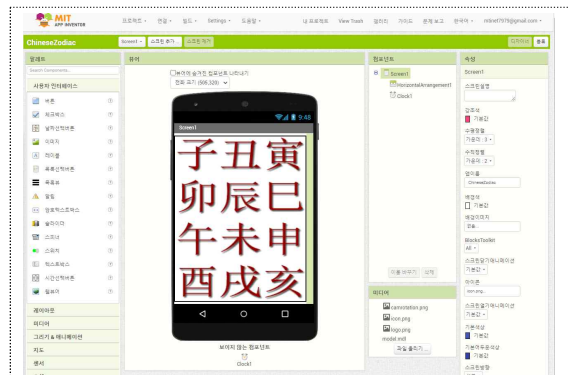
각각의 테스트 데이터 분류 비중 확인



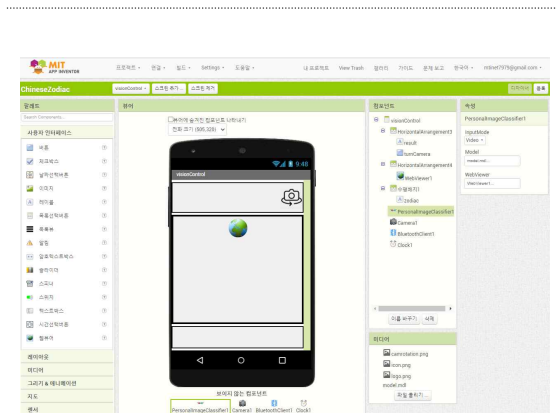
인물 사진을 이용해 모델 성능을 확인,  
Download Model 버튼을 눌러  
model.mdl 파일을 다운로드



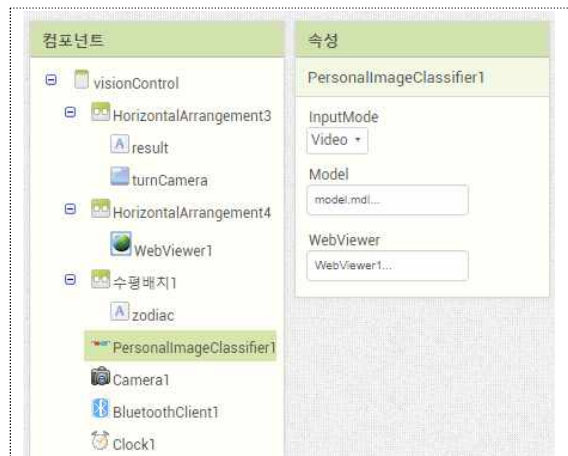
모든 레이블을 합산하면 1이 나옴



앱 표지



visionControl 장면

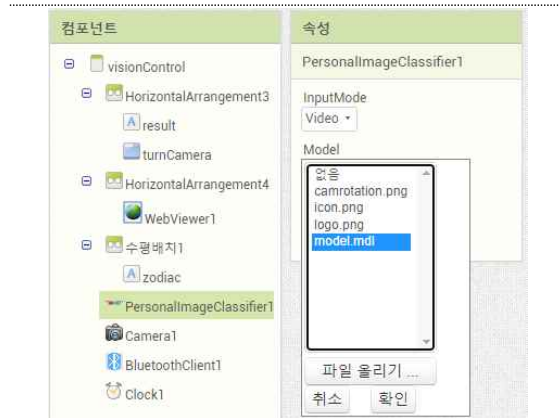


personal image classifier 모듈에  
model.mdl 파일 업로드 하기



## 5. 교수 · 학습 자료

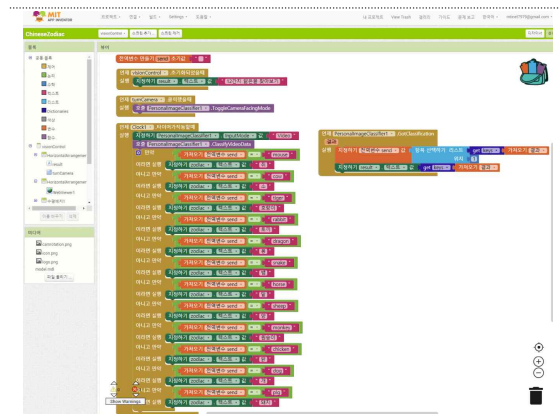
앱인벤터를 이용한 앱 제작  
(<https://github.com/mtinet/chineseZodiac> 을 참고하세요.)



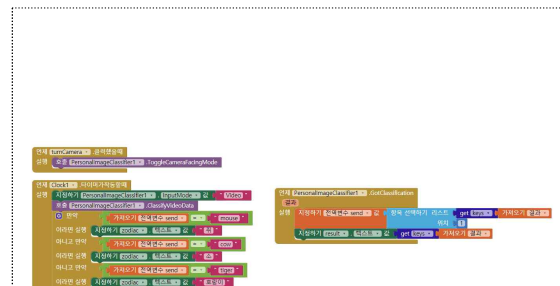
personal image classifier 모듈에  
model.mdl 파일 업로드 하기



screen1 장면이 자동전환되도록 하는  
코드 블럭



visionContol 장면의 전체 코드 블럭



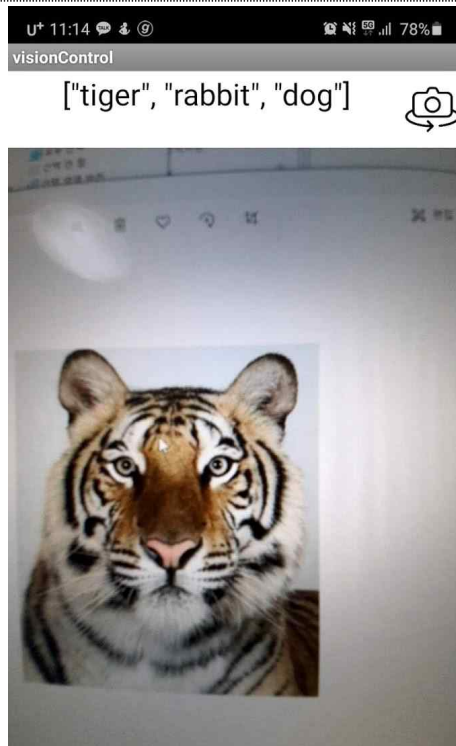
이미지 분류 모델을 이용해 실제로  
분류 결과를 추론하고 결과를 문자로  
보여주는 코드 블럭

子

앱 아이콘에 사용된 이미지

子丑寅  
卯辰巳  
午未申  
酉戌亥

앱의 시작 화면에 보여지는 이미지



호랑이

최종 테스트를 위해 학습데이터를  
테스트 데이터로 사용한 결과



토끼

최종 테스트를 위해 인물 사진을  
테스트 데이터로 사용한 결과



인물 테스트 이미지

## 6. 학생 활동지 1

\_\_\_학년 \_\_\_반 이름: \_\_\_\_\_

### <인공지능 12간지 뚝은꼴 찾기>

여러분이 제작한 앱의 이름은 무엇인가요?	
제작한 앱이 여러분의 목적대로 잘 동작하나요?	
위 질문의 대답을 하게 된 근거는 무엇인가요?	
그렇다면, 이러한 문제점을 개선하려면 어떻게 해야할까요?	