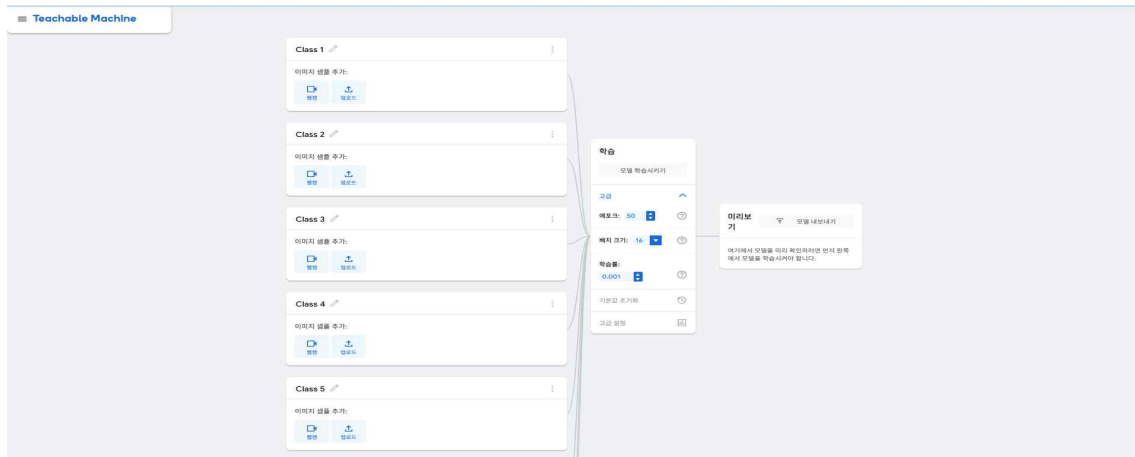
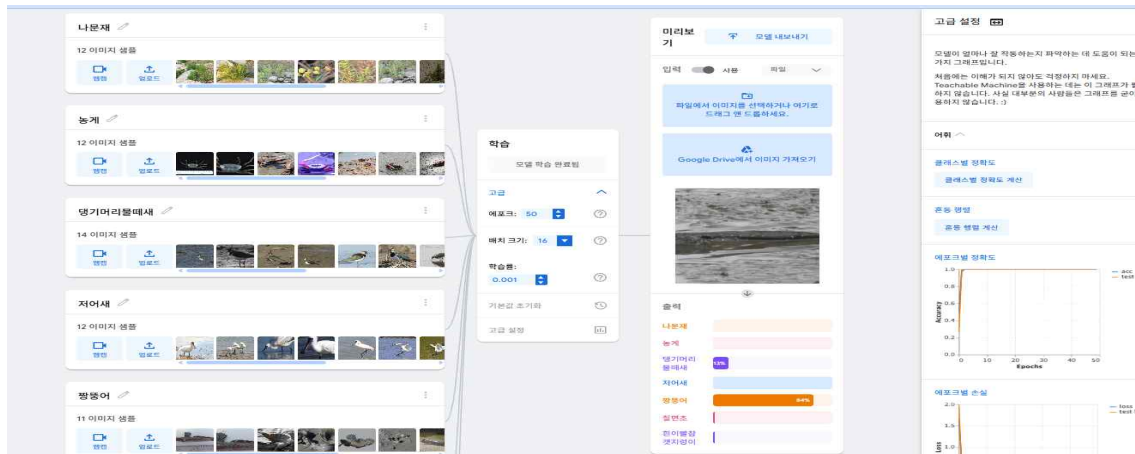


프로그램 제작 과정 (갯벌 생물)

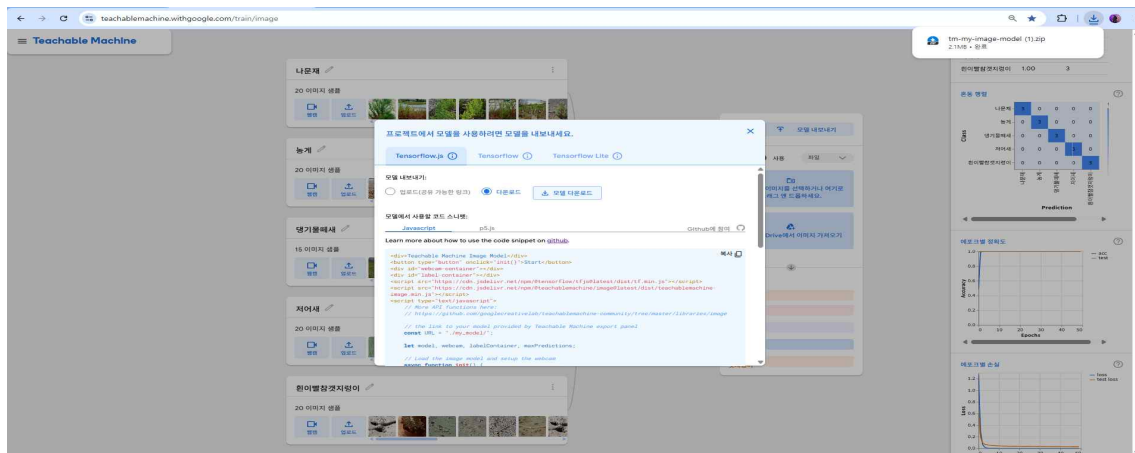
1단계: 머신러닝 기반 AI프로그램(Teachable Machine)을 활용한 생물 데이터 정리·학습



1-1 학습 횟수, 배치 크기, 학습률 미세 조정 후 정확도 검증



1-2 AI 제작 파일 다운로드



2단계: Gemini 3.1 Pro를 통한 기본 설계 및 UI/UX 개선

2-1 프로그램 PowerShell 코드 작성 과정 이미지 1

그 webcam은 필요 없는 기능이야. 그냥 이미지만 첨부할 수 있으면 되는거야. 그리고 배경 이미지는 너무 어두운 색 말고 하늘색 혹은 남태평양 바다 색으로 만들어줘. 그리고 빨리 인식이 되는 단순 하면서 정확한 프로그램이면 좋겠어. 그런 powershell 코드를 작성해줘.

웹캠 기능을 제거하고 '이미지 업로드 전용'으로 변경하면 코드가 훨씬 더 가벼워집니다!

웹캠 권한을 우회하기 위해 필요했던 복잡한 로컬 서버(`run_server.bat`)도 이제 전혀 필요가 없습니다. 파일 하나로 모든 것이 해결되며, 요청하신 대로 남태평양 바다를 연상시키는 맑은 하늘색 그라데이션과 반투명 유리 질감(Glassmorphism) 디자인을 적용했습니다.

갯벌이나 해양 생물 사진을 올렸을 때 즉각적이고 정확하게 판독할 수 있도록, 불필요한 코드를 모두 덜어낸 초경량 버전입니다. 아래 한 줄 코드를 파워셸에 붙여넣고 실행해 보세요!

```
PowerShell
$D=[Environment]::GetFolderPath('Desktop');$A=Join-Path $D 'Offline_Image_AI';if(!$?) {Test-Path $A} else {New-Item -Path $A -ItemType Directory}
```

💡 개선된 점:

- 초경량 최적화: 웹캠과 서버 코드가 사라져 명령어 길이가 절반 이하로 확 줄었습니다. 빠르고 어려움이 즉시 빌드됩니다.
- 디자인 테마: 남태평양의 에메랄드빛 바다색 톤다운 그라데이션(`linear-gradient`)과 둥글고 깨끗한 모서리를 사용하여 해양 생물 교육 자료와 잘 어울리는 화사한 느낌을 줍니다.
- 사용성: 1단계에서 모델 3개를 등록하면 바로 2단계 창이 선행해지며 활성화됩니다. 이후 갖고 계신 생물 사진을 올리면 화면에 이미지가 뜨고 퍼센트 게이지가 물결처럼 부드럽게 차오릅니다.

2-2 프로그램 PowerShell 코드 작성 과정 이미지 2



방금 만든 프로그램처럼 갯벌생물 AI 분석 프로그램도 만들어줘.

교육용이나 현장 안내용으로 바로 활용하기에 손색이 없도록, 갯벌의 청량한 느낌을 담은 민트 & 티얼(Teal) 색상 테마로 디자인을 전면 수정했습니다.

특히 갯벌 환경의 특성을 살려, 학습된 5종의 생물(흰머리참갯지렁이, 저어새, 뱀가시물떼새, 나문재, 농게)이 50% 이상의 높은 확률로 인식될 경우, 해당 생물의 서식지 특징과 멸종위기 등급 등을 알려주는 '맞춤형 생태 해설 박스'가 자동으로 나타나도록 특화 기능을 추가했습니다. 방문객들이 사진을 판독함과 동시에 생태 정보까지 얻어갈 수 있습니다.

이전과 동일하게 파란색 파워셸(PowerShell) 창을 열고, 아래 코드를 복사하여 마우스 우클릭으로 붙여넣은 뒤 엔터(Enter)를 누르시면 바탕화면에 `Offline_TidalFlat_AI` 폴더가 생성되며 즉시 실행됩니다.

```
PowerShell
$D=[Environment]::GetFolderPath('Desktop');$A=Join-Path $D 'Offline_TidalFlat_AI';if(!$?) {Test-Path $A} else {New-Item -Path $A -ItemType Directory}
```

💡 이번 버전의 핵심 포인트

1. 생태 해설 자동 팝업: 사진을 분석했을 때 '저어새'가 1위로 나오면, 하단에 "천연기념물이자 멸종위기 야생생물 1급인 국제적 보호조류입니다..."라는 설명 상자가 함께 등장하여 뛰어난 시각적 안내를 제공합니다.
2. 에코 테마 적용: 짙은 파란색 대신, 갯벌의 자연 환경을 연상시키는 맑은 민트색(`Mint`)과 청록색(`Teal`) 톤으로 전체 UI를 부드럽게 감쌌습니다.
3. 직관적인 안내 문구: 단순한 '인공지능 모델'을 넘어 '갯벌 생태탐안 연결 완료' 등 친근한 용어로 가공되어, 이를 접하는 사람들에게 한층 더 전문적인 인상을 줍니다.

3단계: Colab 환경 내 프로토타입 검증

3-1 프로그램 검증 과정 이미지

The screenshot shows a Google Colab environment. On the left, a Jupyter notebook contains Python code for a web application. The code includes functions for starting the web application, setting up the model, and making predictions. The right side of the screen shows the Gemini chat interface with a list of instructions for using the application.

```
async function startWebcam() {
  document.getElementById("placeholder-text").innerText = "AI 구동 엔진 켜는 중...";
  try {
    // 인터넷 주소가 아닌 사용자가 발급 영토한 로컬 파일 경로를 그대로 코드 (OBS 100% 우회)
    model = await tmlage.loadFromFiles(ssoFile, binFile, metaFile);
    maxPredictions = model.getTotalClasses();

    const flip = true;
    webcam = new tmlage.Webcam(200, 200, flip);
    await webcam.setup();
    await webcam.play();
    window.requestAnimationFrame(loop);

    const container = document.getElementById("webcam-container");
    container.innerHTML = "";
    container.appendChild(webcam.canvas);

    labelContainer = document.getElementById("label-container");
    labelContainer.innerHTML = "";
    for (let i = 0; i < maxPredictions; i++) {
      labelContainer.innerHTML += `<div class="label-bar"><div id="bar-${i}" class="label-fill"></div></div>`;
    }
  } catch (e) {
    alert("카메라 혹은 모델 로딩에 실패했습니다. 카메라 권한을 확인하세요.");
    console.error(e);
  }
}

async function loop() {
  webcam.update();
  await predict();
  window.requestAnimationFrame(loop);
}

async function predict() {
  const prediction = await model.predict(webcam.canvas);
  for (let i = 0; i < maxPredictions; i++) {
    const classNode = document.getElementById(`bar-${i}`);
    const probability = (prediction[i].probability * 100).toFixed(0);
    classNode.style.width = probability + "%";
  }
}
```

Gemini

코드가 실행되지 않는 데에는 몇 가지 주요 원인이 있을 수 있습니다. 다음 사항들을 확인해 보세요:

1. **설치**: 우선 Python 코드가 담긴 웹 페이지의 실행 버튼(마우스 아이콘)을 눌러야 합니다.
2. **카메라 권한**: 브라우저 상단 주소창 근처에 카메라 허용 팝업이 뜨면 허용을 눌러야 합니다. Colab은 보안상 카메라 접근 권한을 제한합니다.
3. **파일 3개 동시 선택**: `model.json`, `weights.bin`, `metadata.json` 파일을 하나씩 올리는 게 아니라, 파일 선택 창에서 **Ctrl 키를 누른 채 3개를 모두 클릭**하거나 드래그하여 한꺼번에 업로드 해야 합니다.
4. **인터넷 연결**: 최초 실행 시에는 라이브러리를 불러오기 위해 인터넷이 필요합니다.

더 확실하게 동작하도록 오류 메시지를 강화하고 이를 개선한 코드를 아래에 다시 준비했습니다. 이 셀을 실행한 뒤 파일을 넣어보세요.

코드가 실행은 되었지만, 화면에 아무것도 나타나지 않거나 버튼이 작동하지 않는 것 같습니다. 이 문제는 주로 브라우저 보안 정책이나 파일 선택 방식 때문입니다.

해결을 위한 체크리스트입니다:

1. **출력창 확인**: 코드 셀 바로 아래에 출력창의 '시' 모음 실행기 상자가 나타났는지 확인해 주세요.
2. **파일 3개 동시 업로드**: 가장 중요한 부분입니다. 파일 선택 창에서 `model.json`, `weights.bin`, `metadata.json` 세 파일을 **Ctrl(또는 Command) 키를 누른 상태로 모두 선택**해서 한꺼번에 업로드를 눌러야 버튼이 파란색으로 활성화됩니다.
3. **카메라 권한**: 카메라 허가 버튼을 누른 후, 브라우저 주소창 왼쪽의 주소창 아이콘이나 팝업을 통해 카메라 허용을 반드시 선택해야 합니다.
4. **루키 및 사이트 권한**: 브라우저 설정에서 `googleusercontent.com`의 루키나 자바스크립트 실행이 차단되어 있는지 확인해 보세요.

만약 여전히 반응이 없다면, 브라우저의 개발자 도구(F12) 콘솔 창에 어떤 빨간색 에러 메시지가 뜨는지도 확인하는 데 어떤 도움이 필요하신가요?

+ Gemini 3 Flash >>

Gemini: 실수할 수 있으므로, 대입을 다시 한번 확인하고 신중하게 코드를 사용하시기 바랍니다. [자세히 알아보기](#)

3-2 프로그램 검증 과정 이미지

The screenshot shows a Google Colab environment. On the left, a Jupyter notebook contains Python code for a web application. The code includes functions for starting the web application, setting up the model, and making predictions. The right side of the screen shows the Gemini chat interface with a list of instructions for using the application.

```
await webcam.setup();
await webcam.play();
window.requestAnimationFrame(loop);
document.getElementById("webcam-container").appendChild(webcam.canvas);
}

async function loop() {
  await webcam.update();
  await predict();
  window.requestAnimationFrame(loop);
}

async function predict() {
  const prediction = await model.predict(webcam.canvas);
  let res = "";
  for (let i = 0; i < maxPredictions; i++) {
    res += prediction[i].className + " " + (prediction[i].probability * 100).toFixed(0) + "%<br>";
  }
  document.getElementById("label-container").innerHTML = res;
}

</script>
</body>
</html>
...
HTML(html_content);
```

완전 로컬 라이브러리 모드
CDN 연결 없이 내부 스크립트로 작동합니다.

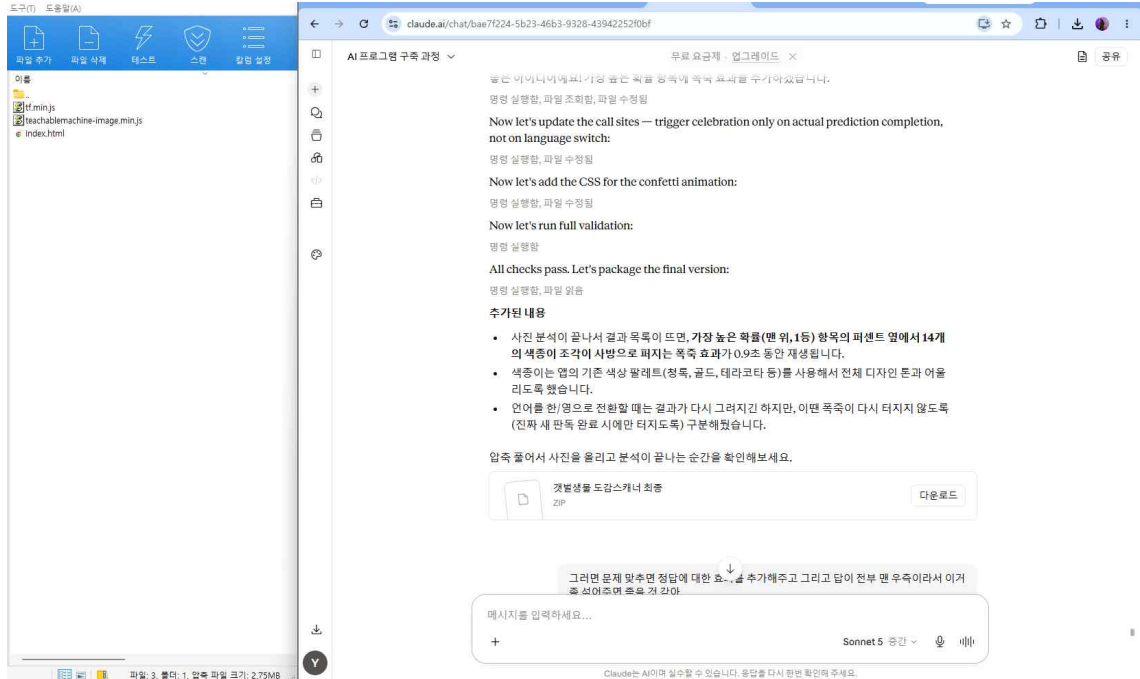
파일 선택 | 선택된 파일 없음

카메라 시작

[illegible]

6단계: 바탕화면 파일 생성 확인

6-1 (파일 생성 시 index, tf.min, teachablemachine-image.min 생성 확인 후 index 실행)



7단계: Claude AI를 활용하여 프로그램 수정

7-1 프로그램 UI 및 내용 수정



7-2 프로그램 제작 간 학습 활용 이미지 및 정보 출처 표시

갯벌생물 도감 스캐너
시료 관찰하고, 도감으로 배워요

KO

EN

스캐너

퀴즈

도감

갯벌 생태계 지표종

장동어
Boleophthalmus pectinirostris

몸길이 10~15cm 정도의 작은 물고기예요. 눈이 머리 꼭대기에 볼록 튀어나와 있어서 갯벌 위에서도 주변을 잘 살필 수 있어요. 몸은 푸른빛을 띠고 옆구리에는 하얀 작은 점들이 흩어져 있어요.

참고 사항 - 썰물 때는 갯벌 위를 기어다니며 먹이를 찾고, 밀물 때는 자기가 파놓은 굴속에 숨어요. 공기로도 숨을 쉴 수 있어서 물 밖에서도 살 수 있는 신기한 물고기예요.

서식 위치
 ※ 실제 서식 범위는 더 넓을 수 있어요

정보 출처 - MBRIS(해양생물자원통합정보시스템), 해양환경정보포털 - MBRIS에서 자세히 보기 >

AI 학습에 사용된 이미지 출처는 다음과 같습니다

나문재

철면초

농계

원이빨참갯지렁이

저어새

등기물매새

뽕종어

• iNaturalist - CC0 1.0 Universal (CC0 1.0) >

프로그램 내용 출처 : MBRIS(해양생물자원통합정보시스템)

※ 종 아이콘은 특징을 쉽게 알아볼 수 있도록 그런 캐릭터 일러스트이며, 서식지 지도는 학습용 참고 지도입니다

7-3 교육 자료(이미지) 입력 방식 단축

model.json

weights.bin

model.zip

metadata.json

JSON

BIN

ZIP

JSON

프로그램 구동을 위한 필요 파일 3종을 프로그램을 구동할 때마다 첨부하는게 아니라, 그냥 처음부터 첨부되어서 첨부하는 과정을 간략화 할 수 있어?

파일을 재검증한 후 안에 직접 통합하기로 결정했다

동일한 모델 파일이네요. 이 모델을 index.html 안에 직접 내장해서, 실행할 때마다 3개 파일을 첨부하지 않아도 자동으로 불러오도록 만들겠습니다.

명령 실행함

Safe to embed directly. Now let's insert the embedded model data and rewrite the loading logic:

JSON 직렬화를 통해 안전한 문자열 변환 방식을 검토했다

Now let's implement the `autoLoadModel()` function and rework the UI to remove manual file selection:

명령 실행함, 파일 조직함

Now let's replace this with an auto-loading version using the embedded data:

레이아웃을 간소화하고 파일 업로드 단계를 단일화했습니다

Now let's add proper styling for the new standalone status banner:

명령 실행함, 파일 조직함, 파일 수정함

Now let's wire up `autoLoadModel()` to run on page init:

명령 실행함, 파일 수정함

No leftover references. Now let's run full validation:

JSON 데이터 검증 후 파일 첨부 과정 간소화했다

이제 모델 3개 파일(model.json, metadata.json, weights.bin)이 index.html 안에 내장되었으니, 배포 폴더에서는 더 이상 필요 없습니다. 정리해서 패키징할게요.

명령 실행함, 파일 원복

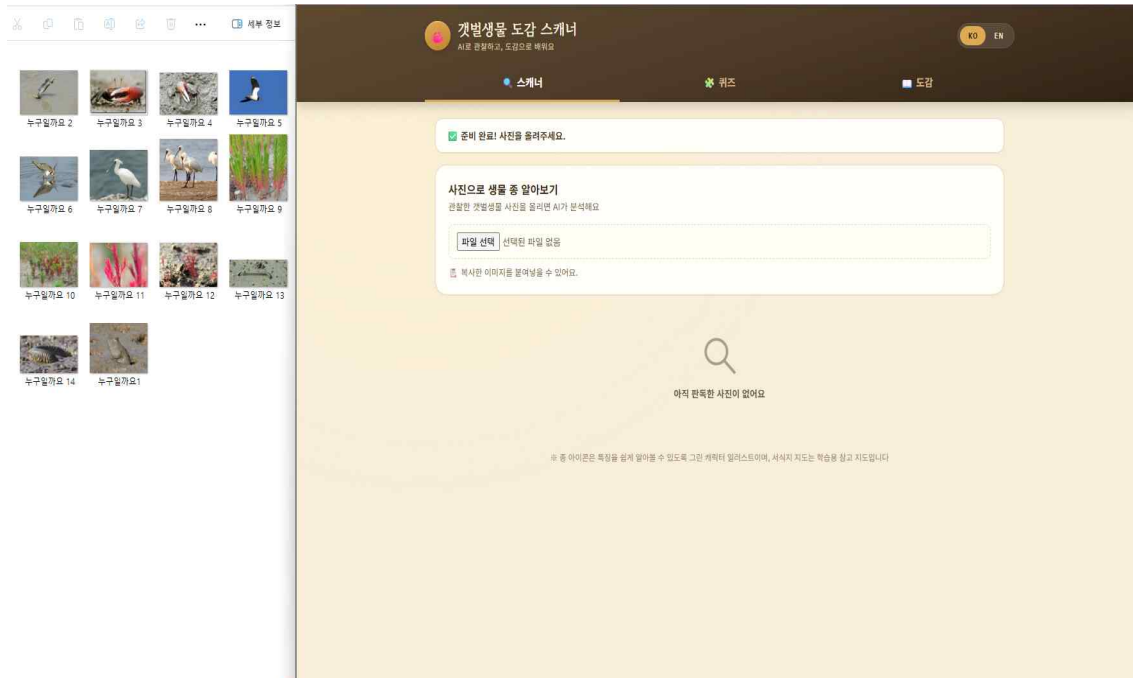
적용 완료

- model.json, metadata.json, weights.bin 3개 파일을 index.html 코드 안에 직접 내장했습니다. 페이지를 열면 자동으로 "AI 모델 불러오는 중..." → "준비 완료! 사진을 올려주세요"로 바뀌며, 별도로 파일을 선택할 필요가 없습니다.
- 내장된 데이터가 원본 파일과 바이트 단위로 완전히 동일함지 검증했습니다 (model.json/metadata.json 문자열 일치, weights.bin 디코딩 결과 일치).
- 이제 배포 폴더는 index.html, tf.min.js, teachablemachine-image.min.js 3개 파일만 있으면 됩니다. index.html 하나만 열면 바로 사진 업로드(또는 Ctrl+V 붙여넣기)로 종 관측이 가능합니다.

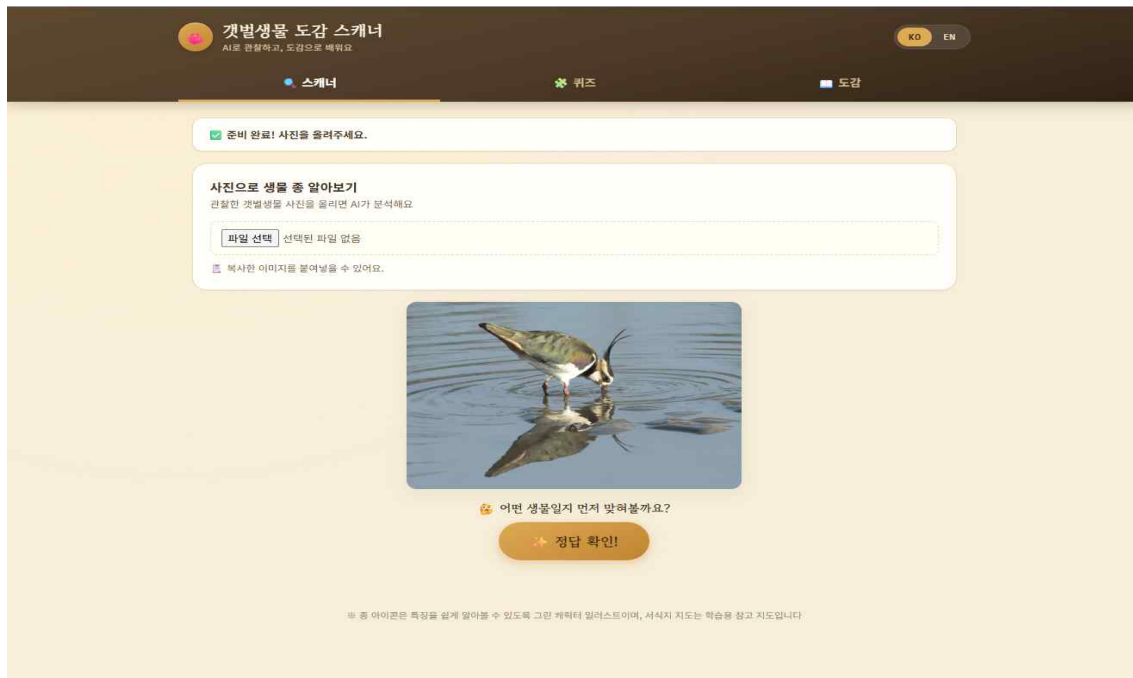
다만 파일 크기가 커진 점은 참고해주세요 (index. ↓, 31이 약 3MB) — 모델 가중치(weights.bin, 약 2.1MB)를 텍스트로 인코딩해 뒀다 보니 그렇게 됐습니다. 실행 속도나 기능에는 영향 없습니다.

8단계 프로그램 실행 시 화면

8-1 프로그램 실행 시 (프로그램 학습에 사용되지 않은 이미지 준비)



8-2 이미지 첨부 시 화면 (준비된 사진 중 아이들이 알아보고 싶은 생물종 선택권 제공 후 첨부)



8-3 정답 확인 (아이들의 의견 수렴 후 정답 확인 클릭)



댕기물떼새 99.4%

저어새 0.4%

평둥어 0.1%

농게 0.0%

칠면초 0.0%

나문재 0.0%

흰야빨갯갯지렁이 0.0%

댕기물떼새 Northern Lapwing [혼한 겨울 철새](#)

몸길이가 30cm쯤 되는 흔한 겨울 철새예요. 머리와 목, 등이 반짝이는 초록빛 도는 검은색이고, 뒷머리의 깃털이 위로 솟아 있어서 멀리서 보면 댕기를 맨 것처럼 보여요.

참고 사항 · 배는 하얗고 꼬리 위쪽은 붉은 밤색이에요. 눈이나 발, 강 하구의 습지에서 무리 지어 지내며 곤충이나 지렁이, 풀씨 등을 먹어요.

서식 위치
댕기물떼새도 철새라서 한곳에 계속 살지 않아요. 이동하는 동안 서해안 갯벌 여러 곳에 잠시 머물다 떠나요.

99.4% 일치율

정보 출처 · 국립생물자원관, 해양환경정보포털, MBRI(S)해양생물지형도정보시스템) · 해양환경정보포털에서 자세히 보기

8-4 퀴즈 확인 (아이들 참여 유도)

갯벌 친구들 퀴즈

정답이 아니어도 괜찮아요, 정답을 맞힐 때까지 계속 클릭하세요!

● ● ● ● ● ● ● ●

문제 4. 뒷머리 깃털이 위로 솟아 있는 모습이 우리나라 전통 머리 장식 '댕기'를 닮아서 이런 이름이 붙은 새는 누구일까요?

힌트: 댕기

칠면초

댕기물떼새

저어새

정답이에요! 배는 하얗고 꼬리 위쪽은 붉은 밤색이에요. 눈이나 발, 강 하구의 습지에서 무리 지어 지내며 곤충이나 지렁이, 풀씨 등을 먹어요.

[다음 문제 →](#)

* 중 아이들은 특징을 쉽게 알아볼 수 있도록 그린 캐릭터 일러스트이며, 서식지 지도는 학습용 참고 지도입니다

갯벌생물 도감 스캐너
 AI로 관찰하고, 도감으로 배우요

스캐너

위즈

도감

저어새
Black-faced Spoonbill

보호대상해양생물, 멸종위기 야생생물 1급

몸 전체가 하얗고, 부리는 검고 길쭉하여 끝이 손가락처럼 넓적한 여름 철새예요. 부리를 물속에 넣고 좌우로 휘저으며 물고기나 새우를 잡아 먹어서 '저어새'라는 이름이 붙었어요.

참고 사항 - 우리나라 서해안의 사람이 살지 않는 섬에서 주로 알을 낳고 새끼를 키웁요. 전 세계 저어새 대부분이 우리나라에서 번식할 만큼 소중한 새라서 멸종위기 야생생물 1급으로 보호받고 있어요.

지정 사유 - 전 세계에 남아있는 개체 수가 아주 적어서, 전 세계가 특별히 보호하고 있는 종이에요.

서식 위치
저어새는 칠레에서 한국에 계속 살지 않아요. 이동하는 동안 서해안 갯벌 여러 곳에 잠시 머물다 떠나요.

정보 출처 - MBIRIS(해양생물자원통합정보시스템), 해양환경정보포털, 국가유산청(국가유산포털), 시흥환경교육원(www.) **해양환경정보포털에서 자세히 보기** >

댕기물떼새
Northern Lapwing

흔한 겨울 철새

몸길이가 30cm쯤 되는 흔한 겨울 철새예요. 머리와 목, 등이 반짝이는 초록빛 또는 검은색이고, 뒷머리와 깃털이 위로 솟아 있어서 멀리서 보면 댕기를 맨 것처럼 보여요.

참고 사항 - 배는 하얗고 꼬리 위쪽은 붉은 밤색이에요. 눈이나 발, 갈 하구의 습지에서 무리 지어 지내며 곤충이나 지렁이, 풀씨 등을 먹어요.

서식 위치
댕기물떼새도 칠레에서 한국에 계속 살지 않아요. 이동하는 동안 서해안 갯벌 여러 곳에 잠시 머물다 떠나요.

정보 출처 - 국립생물자원관, 해양환경정보포털, MBIRIS(해양생물자원통합정보시스템) **해양환경정보포털에서 자세히 보기** >