

AI 캐릭터 애니메이션 제작실

그림 하나를 나누고, 움직이고, 자연스럽게 고치는 법



Coreline-AI 지음

한 그림에서 시작합니다

이 책은 작은 로봇 그림 한 장을 움직이며 ‘어디를 바꾸고 어디를 그대로 둘까’를 배우는 안내서입니다. 몸통은 제자리에 두고 팔이 인사하며, 눈이 잠깐 감겼다 열립니다. 복잡한 도구부터 배우기보다 눈에 보이는 차이를 설명하는 데서 시작합니다.

실습은 새로 작성한 SVG·HTML·CSS·JavaScript 파일로 되어 있습니다. SVG는 크기를 바꿔도 또렷하게 표시할 수 있는 벡터 그림 형식입니다. [MDN SVG](#), 확인 2026-10-18. 계정·유료 서비스·새 패키지 없이 기존 브라우저와 로컬 서버로 확인하도록 설계했습니다.

정지된 책 한 컷만으로 움직임의 속도와 이어짐을 판단할 수는 없습니다. 이 책에서는 실제 Chrome 화면과 설명 도해를 구분해 신고, 재생·일시정지·초기화의 느낌은 동봉한 실습 HTML에서 직접 확인하게 합니다. 현재 검수 기기의 움직임 줄이기 선호가 켜져 독자용 화면은 정지했으며, 움직임은 별도 승인된 QA 전용 화면에서 검수했습니다.

01

보기

02

나누기

03

움직이기

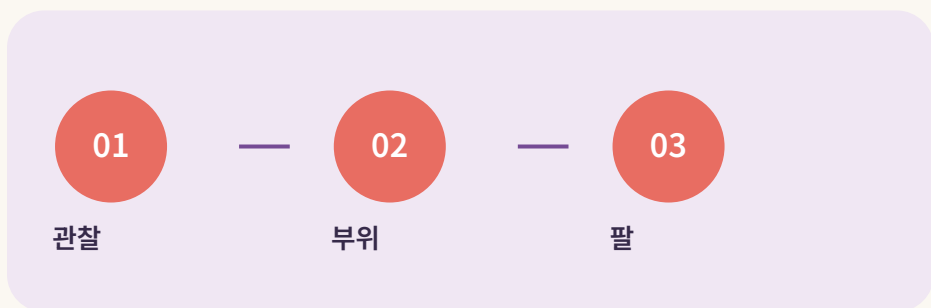
목차 · 먼저 보고 나누기

01 멈춘 그림과 움직이는 그림을 봅니다 · 5쪽. 첫 그림을 보고 몸통·팔·눈 중 무엇이 달라져야 하는지 관찰 문장을 만듭니다.

02 움직일 부분만 나눕니다 · 9쪽. 함께 움직일 선을 묶고 팔이 몸에 붙은 자리를 찾습니다. ‘어깨를 중심으로’라는 말을 화면과 코드에서 같은 뜻으로 쓰게 됩니다.

03 팔 하나부터 움직여 봅니다 · 15쪽. 재생·일시정지·초기화로 출발·중간·끝 자세를 비교합니다. 화면을 그리는 시간과 실제 움직임을 혼동하지 않습니다.

독자가 먼저 결정할 것은 화려한 효과가 아니라 원하는 동작입니다. 각 장의 작업물은 AI에게 다음 수정을 요청할 때 다시 쓸 수 있는 관찰 메모·부위 카드·움직임 카드입니다.



목차 · 고치고 내 것으로 만들기

04 반복할 때 튀는 움직임 고칩니다 · 23쪽. 일부러 잘못 둔 회전 중심을 찾고, 같은 그림을 유지한 채 바르게 돌립니다.

05 멈추고 천천히 볼 수 있게 합니다 · 31쪽. 중간 자세를 보관하고 다시 잇습니다. 움직임을 줄이려는 사람에게는 알아볼 수 있는 정지 그림을 남깁니다.

06 색과 속도를 바꿔 내 캐릭터로 · 39쪽. 정상판을 덮어쓰지 않고 색 한 가지와 팔 주기 한 가지를 차례로 바꿉니다.

끝에는 실습 파일을 다시 여는 법, 공식 자료와 이 책의 검수 범위를 정리했습니다. 실제 브라우저에서 확인한 일과 합성 시험, 미수행 기기 시험은 각각 다른 이름으로 표시합니다. 독자도 ‘내 화면에서 본 결과’를 따로 적어야 합니다.

04

반복

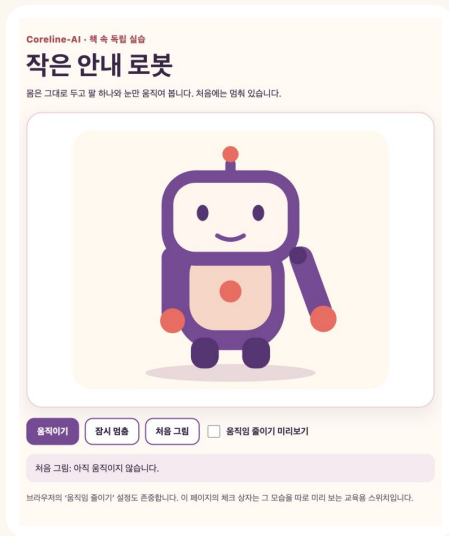
05

멈춤

06

변형

01 멈춘 그림과 움직이는 그림을 봅니다



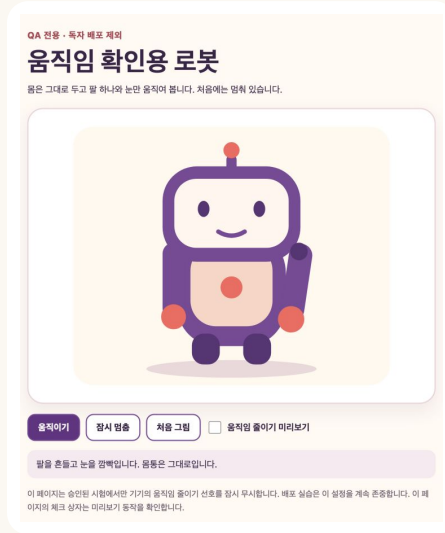
실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

아래 로봇을 잠시 그대로 보세요. 얼굴과 몸통은 중앙에 있고, 오른팔은 몸 옆에 달려 있습니다. 움직이기 버튼을 누르기 전에는 그림이 가만히 있어야 합니다. 시작부터 저절로 흔들리면 독자는 비교할 기준을 잃습니다.

로봇이 인사한다고 할 때도 모든 부분을 움직일 필요는 없습니다. 팔만 살짝 오가면 인사로 읽히고, 눈이 한 번 감겼다 열리면 표정이 살아납니다. 반대로 몸통까지 함께 흔들리면 ‘팔을 흔든다’는 관찰이 어려워집니다.

그림을 보고 먼저 세 문장을 적어 보세요. ‘몸은 제자리에 있다. 한쪽 팔이 어깨 근처에서 돈다. 눈은 잠깐 작아졌다 돌아온다.’ 이 문장이 AI에게 수정 요청을 할 때도 기준이 됩니다.

움직이는 부분을 말로 표시합니다



실제 Chrome QA 전용 화면 · 독자용 ZIP에는 이 시험 페이지가 없습니다.

‘귀엽게 움직여 줘’는 결과를 상상하게 하지만, 어느 부분이 어떻게 움직여야 하는지는 정하지 못합니다. 로봇 그림에서 손으로 가리킬 수 있는 부위부터 말해 봅니다. 이번에는 몸통을 고정하고, 오른팔과 눈만 바꿉니다.

실제 Chrome의 QA 전용 화면에서는 팔 각도가 -20도 부근에서 +20도 부근까지 바뀌는 장면을 관찰했습니다. 이는 저자가 정한 교육용 범위이며 모든 로봇 캐릭터에 좋은 값이라는 뜻은 아닙니다. 책의 정지 사진 두 장만 보고 부드럽게 이어지는지 단정할 수 없기에, 동작은 HTML에서도 확인해야 합니다.

이 장면에서 독자가 정하는 것은 ‘움직일 부위’와 ‘움직이지 않을 부위’입니다. 코드를 먼저 바꾸지 않아도 그 결정은 할 수 있습니다.

멈춤은 고장과 다릅니다

재생과 멈춤은 서로 다른 약속입니다

재생

팔과 눈이 바뀝니다



잠시 멈춤

지금 자세를 보관합니다



다시 재생

그 자리에서 이어집니다

처음 그림은 별도 버튼: 출발 자세로 돌아갑니다

설명 도해 · 실제 실행 화면이 아닙니다.

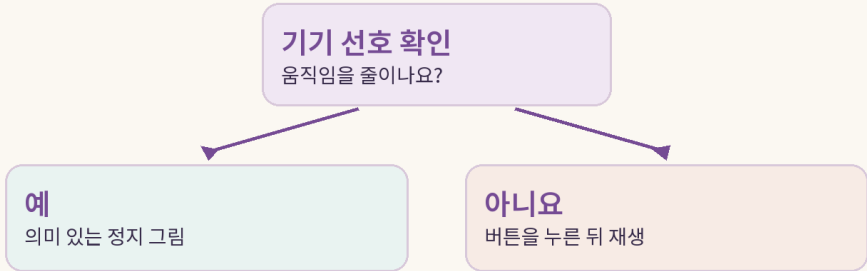
재생 중 ‘잠시 멈춤’을 누르면 움직임이 그 순간의 자세에서 멈춰야 합니다. 다시 ‘움직이기’를 누르면 처음 자세로 갑자기 돌아가기보다 그 자리에서 이어지는 편이 자연스럽습니다. ‘처음 그림’은 다른 버튼입니다. 이 버튼은 언제 눌러도 팔과 눈을 출발 자세로 돌립니다.

실제 QA 전용 화면에서 일시정지 뒤 팔 각도와 눈 모양이 다음 관찰 때까지 같았고, 다시 재생한 뒤에는 팔 각도가 변했습니다. 이 결과는 해당 Chrome 환경의 관찰입니다. 모든 화면 갱신이 부드럽다는 성능 보장은 아닙니다.

독자가 남길 관찰 메모에는 행동과 결과를 붙여 쓰세요. ‘재생 → 팔 변화 / 잠시 멈춤 → 현재 자세 유지 / 재생 → 이어서 변화 / 처음 그림 → 출발 자세’처럼 적으면, 오류가 어느 단계에서 났는지 찾기 쉽습니다.

움직임을 보지 못해도 뜻이 남게

움직임 줄이기 선호를 먼저 살핍니다



설명 도해 · 실제 실행 화면이 아닙니다.

어떤 사람은 기기에서 움직임을 줄이는 설정을 켜 둡니다. 브라우저는 이 선호를 감지할 수 있습니다. [MDN prefers-reduced-motion](#), 확인 2026-10-18. 장식적인 흔들림을 억지로 재생하지 않도록 이 실습은 시작 자세만으로도 로봇의 몸·팔·눈을 알아볼 수 있게 만들었습니다.

검수한 Chrome은 실제로 움직임 줄이기 선호가 켜져 있었습니다. 독자용 페이지에서 재생을 눌러도 로봇이 정지 상태를 유지하는 것을 확인했습니다. 움직임을 강제로 보기 위해 기기 설정을 바꾸지 않았습니다. 별도 QA 화면에서만 시험용으로 팔과 눈의 움직임을 확인했습니다.

이번 장의 작업물은 ‘움직임 관찰 메모’입니다. 빈칸에 ‘기준 그림: ____ / 움직임 부분: ____ / 고정할 부분: ____ / 재생 뒤 기대: ____ / 멈춤 뒤 기대: ____’를 적어 보세요.

02 움직임 부분만 나눕니다



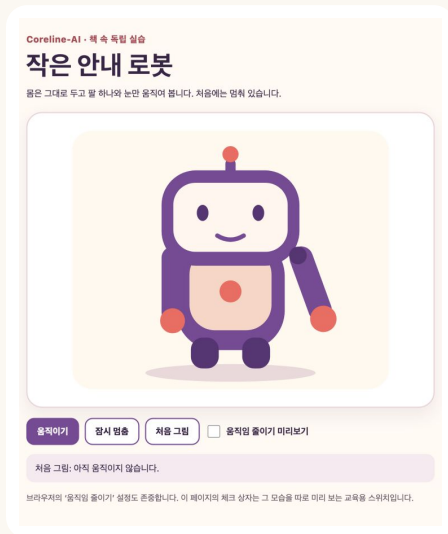
설명 도해 · 실제 실행 화면이 아닙니다.

한 장의 그림처럼 보여도 로봇은 몸통·팔·눈이 함께 놓인 결과입니다. 부분을 나눈다는 것은 그림을 잘라 파일 세 개로 저장하라는 뜻만은 아닙니다. ‘이 선들은 함께 움직이고, 이 선들은 그대로 둔다’는 경계를 정하는 일입니다.

이번 예제의 몸통은 가운데에 남습니다. 팔은 어깨 부근을 중심으로 돌고, 두 눈은 높이만 잠깐 줄었다 돌아옵니다. 이렇게 역할을 정해 두면 AI가 팔을 고치다가 얼굴 위치까지 옮긴 경우를 발견할 수 있습니다.

도해의 숫자는 부위를 찾기 위한 안내입니다. 실제 실행 화면이 아니라 설명 그림이며, 로봇 코드에서 어떤 묶음을 확인해야 하는지 연결해 줍니다.

팔이 붙는 자리를 먼저 찾습니다



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

팔 그림을 옆으로 조금 옮기는 것과 어깨에서 돌리는 것은 다릅니다. 손을 들어 인사하는 느낌을 내려면 팔이 몸에 붙은 곳을 중심으로 돌아야 합니다. 중심을 손끝에 두면 팔 전체가 엉뚱한 원을 그리며 돌아갑니다.

이 실습은 팔의 회전 중심을 그림 안의 (332, 194)에 둡니다. 좌표는 그림 안에서 위치를 부르는 방법입니다. 숫자 자체를 외울 필요는 없습니다. 먼저 실제 화면에서 팔과 몸이 만나는 곳을 찾고, 그곳의 좌표를 코드와 대조하세요.

어깨 중심이 틀어졌을 때 시에게는 ‘팔을 예쁘게’보다 ‘팔이 어깨에서 떨어져 보인다. 몸과 만나는 곳을 중심으로 회전하게 해 달라’고 말하는 편이 정확합니다.

같이 움직일 선을 한 묶음으로



설명 도해 · 실제 실행 화면이 아닙니다.

팔은 바깥 선과 손끝 장식 등 여러 조각으로 보일 수 있습니다. 그중 하나만 회전시키면 다른 조각이 제자리에 남아 팔이 찢어진 것처럼 보입니다. 함께 움직여야 하는 조각은 하나의 묶음으로 다룹니다.

SVG의 ``<g>``는 여러 그림 요소를 묶고, 그 묶음에 변형을 적용하면 안의 요소도 함께 영향을 받습니다. [MDN SVG g](#), 확인 2026-10-18. 책 속 실습에서는 팔 묶음만 회전시켜 몸통과 얼굴은 그대로 둡니다.

독자가 코드를 모두 읽지 않아도 확인할 수 있는 방법이 있습니다. 팔이 올라간 화면과 내려간 화면을 나란히 보고, 손끝 색과 선이 팔과 함께 움직였는지 찾으세요. 한 조각만 남았다면 묶음의 경계가 잘못된 것입니다.

몸통을 고정하면 비교가 쉬워집니다



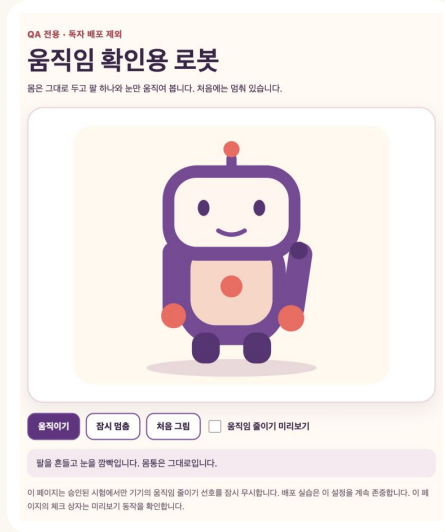
설명 도해 · 실제 실행 화면이 아닙니다.

움직임을 만들 때 모든 부분이 조금씩 흔들리면 살아 있는 느낌이 날 수 있습니다. 하지만 처음 배우는 실습에서는 무엇이 원인인지 구분하기 어려워집니다. 이번에는 몸통을 일부러 고정합니다. 눈과 팔에만 변화가 있으면 잘못 움직인 부위를 쉽게 찾을 수 있습니다.

두 장면을 비교할 때 배경·몸통·얼굴 윤곽이 같은 자리에 있어야 합니다. 실제 QA 검사에서는 몸통에 별도 회전 변형이 없고 팔 각도만 달라졌습니다. 이것은 코드와 브라우저 화면을 함께 본 제한된 검수 결과입니다.

AI가 결과물을 수정했다면 색이나 크기를 함께 바꾸지 않도록 요청하세요. 여러 조건이 한꺼번에 달라지면 팔 회전이 좋아졌는지 알기 어렵습니다. 원본을 남겨 두는 이유도 같은 조건으로 다시 비교하기 위해서입니다.

눈은 작아졌다가 돌아옵니다



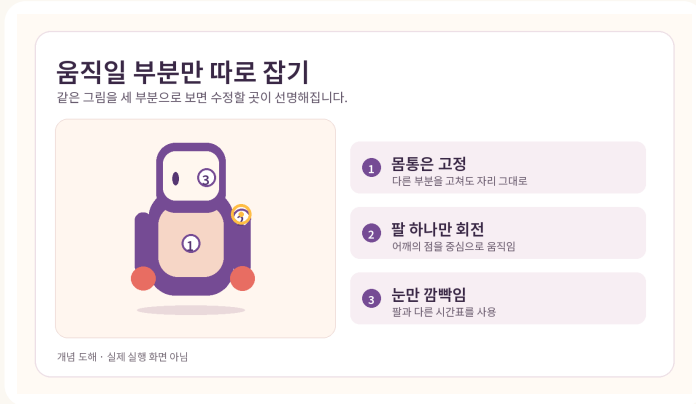
실제 Chrome QA 전용 화면 · 독자용 ZIP에는 이 시험 페이지가 없습니다.

눈 깜빡임은 눈을 화면에서 없애는 것과 다릅니다. 눈 모양의 세로 길이를 잠깐 줄였다가 원래 길이로 돌리면, 얼굴의 위치와 표정은 유지하면서 깜빡이는 느낌을 만들 수 있습니다.

이번 예제에서 눈의 세로 반지름은 평소 11, 가장 좁을 때 약 2입니다. 이는 교육용 설계값입니다. 실제 QA 화면에서는 여러 순간을 읽어 약 2에서 11 사이로 바뀌는 것을 관찰했지만, 그 숫자만으로 모든 사람에게 자연스러운 표정이라고 판단하지는 않습니다.

만약 눈이 사라진 채 돌아오지 않는다면 ‘깜빡임이 안 된다’ 대신 ‘눈이 줄어든 뒤 원래 높이로 돌아오지 않는다’고 적으세요. 돌아오는 단계가 빠졌는지 확인할 수 있습니다.

부위 카드 한 장을 만듭니다



설명 도해 · 실제 실행 화면이 아닙니다.

이번 장의 작업물은 ‘부위 카드’입니다. 바뀌는 곳과 그대로 있어야 할 곳을 한 장에 놓으면, AI에게 다음 수정을 요청할 때도 범위가 좁아집니다. 완성 예시는 이렇습니다. ‘몸통: 고정 / 오른팔: 어깨에서 -20도부터 +20도까지 / 눈: 세로 길이만 줄었다 회복 / 색: 이번 수정에서는 유지.’

여기서 각도와 눈 길이는 이 교육 예제에 맞춘 값입니다. 새 캐릭터를 만든다면 모양과 크기를 보고 다시 정해야 합니다. 숫자를 그대로 복사하는 것보다 ‘어느 부분을 건드리지 않을까’를 먼저 확인하세요.

내 부위 카드: ‘고정할 부분: ____ / 움직일 묶음: ____ / 회전하거나 바뀌는 기준: ____ / 되돌아올 모습: ____ / 이번에 바꾸지 않을 것: ____’. 다음 장에서는 이 카드의 팔 한 줄을 실제 움직임으로 이어 봅니다.

03 팔 하나부터 움직여 봅니다



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

화면에서 ‘움직이기’를 누르는 순간부터 변화를 관찰합니다. 먼저 팔만 보고, 그다음 눈을 봅니다. 한꺼번에 두 변화를 찾으려 하면 팔의 어깨 연결이 잘못된 것을 놓칠 수 있습니다.

실제 실습 ZIP을 풀고 최상위 폴더에서 ``python3 -m http.server 8769 --bind 127.0.0.1``을 실행한 뒤, 기존 브라우저에서 ``http://127.0.0.1:8769/examples/guide-robot-v1/``을 엽니다. 이미 8769를 다른 프로그램이 사용한다면 그 프로그램을 종료하지 말고 명령과 주소의 포트를 함께 바꾸세요.

기기의 움직임 줄이기 선호가 켜져 있으면 재생 버튼을 눌러도 정지 그림을 유지합니다. 이는 고장이 아닙니다. 이 책의 실제 동작 사진은 저자가 별도 QA 화면으로 검수한 캡처이며, 독자용 ZIP에는 QA 전용 화면이 없습니다.

처음·중간·끝을 따로 봅니다

팔의 한 번: 출발 → 반대쪽 → 다시 출발



교육용 설계값 · 왕복 2.4초

설명 도해 · 각도와 2.4초는 이 책의 교육용 설계값입니다.

팔을 한 방향으로 끝없이 올리기만 하면 곧 몸에서 이상하게 꺾여 보입니다. 이번 예제는 출발 각도에서 반대쪽까지 갔다가 다시 출발 각도로 돌아오는 왕복을 한 번으로 봅니다.

저자가 정한 한 번의 길이는 2.4초, 출발은 -20° , 반대쪽은 $+20^\circ$ 입니다. 이 숫자는 속도와 움직임 범위를 설명하기 위한 설계값이며 실제 브라우저의 화면 재생률을 뜻하지 않습니다. 짧은 정지 사진 세 장은 그 순간의 자세만 보여 줍니다.

검수에서는 순수 함수의 시작·1/4·중간·3/4·끝 경계를 계산했고, 실제 QA 화면에서도 -20° 부터 $+20^\circ$ 까지 변하는 모습을 여러 번 읽었습니다. 다만 모든 프레임이 끊김 없이 보였다는 증거는 아닙니다.

화면을 그리는 시간은 일정하지 않습니다

팔의 한 번: 출발 → 반대쪽 → 다시 출발



교육용 설계값 · 왕복 2.4초

설명 도해 · 각도와 2.4초는 이 책의 교육용 설계값입니다.

브라우저는 다음 화면을 그릴 기회가 왔을 때 함수를 불러 줄 수 있습니다.

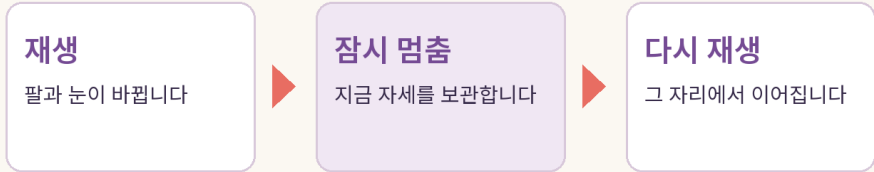
`requestAnimationFrame`은 한 번 요청할 때 다음 호출 하나를 예약하므로, 계속 움직이려면 다시 요청해야 합니다. 호출 횟수를 초당 60번이라고 가정하면 환경에 따라 빠르거나 느려 보일 수 있습니다. [MDN requestAnimationFrame](#), 확인 2026-10-18.

이 실습은 ‘지금 몇 번째 그림인가’보다 ‘출발한 뒤 실제로 얼마나 지났는가’를 기준으로 팔 자세를 정합니다. 그래서 잠깐 화면을 그리지 못했다고 모든 중간 장면을 뒤늦게 빠르게 재생할 필요가 없습니다.

초보자라면 함수 이름을 외우기보다 이 질문을 기억하세요. ‘같은 2.4초 왕복이 화면 갱신 횟수에 따라 달라지지 않는가?’ 값을 고칠 때도 사진 한 장보다 실제 재생을 함께 살펴야 합니다.

재생 버튼을 두 번 눌러도

재생과 멈춤은 서로 다른 약속입니다



처음 그림은 별도 버튼: 출발 자세로 돌아갑니다

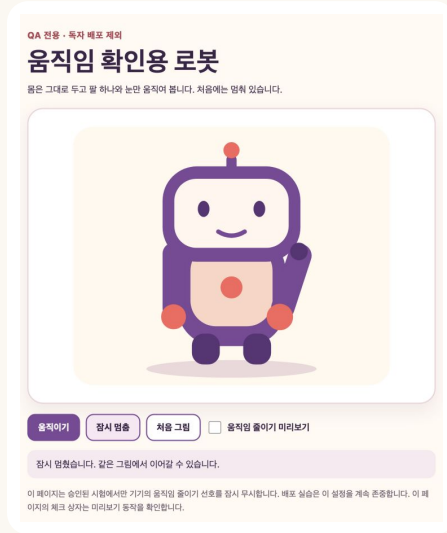
설명 도해 · 실제 실행 화면이 아닙니다.

첫 재생 뒤 버튼을 다시 눌렀을 때 두 개의 움직임이 겹치면 팔이 갑자기 빠르게 움직이거나 멈춤 버튼이 예상대로 작동하지 않을 수 있습니다. 이 실습은 이미 움직이는 동안 다시 재생을 눌러도 새로운 반복을 더 만들지 않습니다.

이 규칙을 확인하는 작은 합성 시험과 실제 QA 화면의 재생·멈춤 관찰을 분리해 기록했습니다. 합성 시험은 코드의 상태 전환을 확인하고, 화면 관찰은 실제 팔과 눈의 모습이 달라지는지 확인합니다. 어느 한쪽만으로 모든 환경의 결과를 보증하지는 않습니다.

AI에게 고쳐 달라고 할 때는 ‘버튼을 연달아 눌러도 재생이 두 번 겹치지 않게 해 달라’고 설명하세요. 겉으로 빠르게 보이는 증상과 예상한 규칙을 함께 주면 원인을 찾기 쉽습니다.

일시정지는 같은 자세를 붙잡습니다



실제 Chrome QA 전용 화면 · 독자용 ZIP에는 이 시험 페이지가 없습니다.

팔이 오른쪽으로 올라간 순간 멈춤을 눌렀다면, 멈춘 동안에도 그 자세가 남아 있어야 합니다. 팔을 출발 자세로 돌리는 것은 ‘처음 그림’의 일입니다. 두 버튼의 역할이 섞이면 이어 보기와 처음부터 보기가 구별되지 않습니다.

실제 QA 화면에서 멈춘 뒤 같은 팔 각도와 눈 모양이 유지됐고, 다시 재생하자 그 자세에서 변하기 시작했습니다. 자동으로 재생됐다고 하지 않습니다. 독자가 명시적으로 재생을 눌러야 합니다.

내가 확인할 순서는 간단합니다. 움직이는 중간에 멈춘다. 팔과 눈을 잠시 본다. 그대로인지 확인한다. 재생을 누르고 이어지는지 본다. 마지막으로 처음 그림을 눌러 출발 자세와 비교한다. 이 순서를 적어 두면 다른 캐릭터에도 적용할 수 있습니다.

작은 화면에서 버튼이 남아야 합니다



실제 Chrome 390×844 창 · 실제 휴대폰 화면은 아닙니다.

캐릭터 그림이 좁은 화면에 맞아도 버튼이 옆으로 밀려 사라지면 실습은 끝까지 할 수 없습니다. 그림과 조작을 따로 확인해야 하는 이유입니다.

기존 Chrome의 화면 폭을 390과 320으로 줄여 보았을 때 가로 넘침은 없었고, 세 버튼과 움직임 줄이기 미리보기 안내를 볼 수 있었습니다. 버튼 높이는 확인한 화면에서 48픽셀이었습니다. 이는 데스크톱 Chrome의 크기 조정 시험이지 실제 휴대폰에서 손가락으로 눌러 본 시험이 아닙니다.

내 기기에서 확인할 때는 화면을 가로로 밀어야 버튼이 보이는지, 글자를 키웠을 때 설명이 버튼을 가리는지 살펴보세요. 문제를 발견하면 기기·창 너비·어느 버튼이 가려졌는지 함께 적어 두는 편이 좋습니다.

결과가 다르면 한 조건씩 되돌립니다

팔을 돌리는 기준점 하나가 차이를 만듭니다

정상판 · 어깨



팔이 붙어 보임

오류판 · 먼 지점



팔이 몸을 떠나 보임

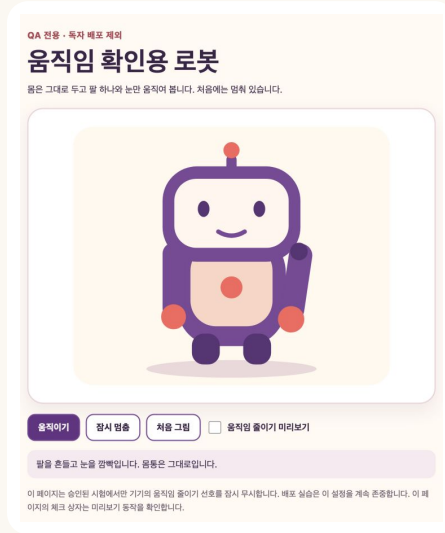
설명 도해 · 실제 실행 화면이 아닙니다.

팔이 몸에서 떨어져 보이거나, 눈이 닫힌 채 멈추거나, 일시정지 뒤 처음 자세로 돌아간다면 한 번에 전부 고치지 않습니다. 우선 기대한 동작을 하나 정하고, 그 동작에 관계된 값만 고칩니다.

예를 들어 팔이 떨어졌다면 회전 중심을 먼저 봅니다. 눈이 돌아오지 않는다면 눈의 끝 자세를 봅니다. 멈춤 뒤 자세가 바뀌었다면 시간이 어떻게 보관되는지 봅니다. 색이나 크기를 함께 바꾸면 고친 원인을 다시 확인하기 어렵습니다.

독자용 오류 기록 문장은 ‘누른 것: ____ / 기대한 자세: ____ / 실제 자세: ____ / 그대로인 부분: ____ / 마지막으로 바꾼 것: ____’입니다. AI에게 이 기록과 함께 원본 파일을 보존하라고 요청하세요. 결과물을 바로 덮어쓰기보다 차이를 보고 선택하는 사람이 최종 결정을 말합니다.

움직임 확인 카드



실제 Chrome QA 전용 화면 · 독자용 ZIP에는 이 시험 페이지가 없습니다.

이번 장의 작업물은 실습을 한 번 더 재현할 수 있는 ‘움직임 확인 카드’입니다. 완성 예시는 다음과 같습니다. ‘처음: 팔 -20도 · 눈 열림 / 재생: 팔이 어깨를 중심으로 왕복하고 눈이 깜빡임 / 멈춤: 현재 자세 유지 / 다시 재생: 명시적 버튼 뒤 이어짐 / 처음 그림: 출발 자세.’

이 예시는 새 안내 로봇의 규칙이며 실제 사람의 만족도나 보편적인 애니메이션 기준이 아닙니다. 현재 검수 기기에서는 움직임 줄이기가 켜져 독자용 페이지가 정지 상태였고, 실제 운동 관찰은 승인된 별도 QA 화면에서 했습니다. 자신의 기기에서는 다르게 보일 수 있습니다.

내 카드: ‘실습 파일/버전: ____ / 기기·브라우저: ____ / 움직임 줄이기 상태: ____ / 출발 자세: ____ / 팔 변화: ____ / 눈 변화: ____ / 멈춤·재개: ____ / 아직 확인 못 한 것: ____.’ 확인 못 한 칸을 억지로 통과로 적지 마세요.

04 반복할 때 튀는 움직임을 고칩니다



실제 Chrome의 의도적 오류판 · 정상 실습이 아닙니다.

같은 로봇 그림인데 팔만 몸에서 조금 떨어져 보입니다. 비교용 `before.html`에는 일부러 잘못된 회전 중심을 넣었습니다. 완성 실습이 아니라 오류를 찾기 위한 교육용 페이지입니다.

정상판은 어깨 근처인 (332, 194)를 중심으로 돌지만, 오류판은 (402, 194)를 중심으로 돕니다. SVG 본체는 두 페이지가 같고 회전 기준만 다릅니다. 화면에서 팔이 몸을 떠나는 모습은 실제 Chrome으로 확인했습니다. 다만 이 오류는 저자가 일부러 넣은 것이지, 참고 프로젝트에서 똑같이 발생한 실제 사고를 재현한 것이 아닙니다.

‘팔이 이상하다’보다 ‘팔이 어깨에서 분리돼 보인다. 몸과 만나는 곳을 기준으로 돌려 달라’고 요청하세요. 문제의 위치와 기대한 모습을 함께 알려 주는 것이 수정의 출발점입니다.

같은 그림, 바뀐 기준점



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

정상판으로 돌아오면 팔이 몸통 옆에서 이어집니다. 고치는 일은 새 그림을 덧씌우는 것이 아닙니다. 같은 그림의 회전 기준을 실제 어깨 위치로 돌리는 것입니다. 원본과 오류판을 나란히 보면서 팔 외의 몸·눈·색이 바뀌지 않았는지도 확인합니다.

실습 코어의 합성 시험은 일부러 넣은 잘못된 기준을 감지하고, 정상 기준을 통과시킵니다. 브라우저에서는 오류판과 정상판의 화면 차이를 보았습니다. 함수 시험과 눈으로 본 결과는 서로 보완하지만 동일한 검사가 아닙니다.

수정 확인 문장은 ‘같은 그림을 사용했다 / 팔 기준점만 바뀌었다 / 몸통은 그대로다 / 팔이 어깨에서 이어져 보인다’입니다. 다음 장에서는 이렇게 고친 움직임도 멈추고 천천히 볼 수 있어야 하는 이유를 다룹니다.

눈으로 본 오류를 한 문장으로

팔을 돌리는 기준점 하나가 차이를 만듭니다

정상판 · 어깨



팔이 붙어 보임

오류판 · 먼 지점



팔이 몸을 떠나 보임

설명 도해 · 실제 실행 화면이 아닙니다.

팔이 몸을 떠나 보이는 화면을 발견하면 먼저 멈춥니다. 숫자를 아무거나 바꾸기보다, 어떤 부분이 여전히 제자리에 있고 무엇만 어긋났는지를 말로 적습니다. 이 예제에서는 몸통과 얼굴은 그대로인데 오른팔만 어깨에서 떨어져 보입니다. 따라서 그림 전체의 위치보다 팔의 회전 기준을 먼저 살펴야 합니다.

의도적 오류판에는 회전 중심 (402, 194)가 적혀 있고 정상판에는 (332, 194)가 적혀 있습니다. 두 숫자의 차이는 이 교육 그림 안에서 가로 방향 70입니다. 이것은 저자가 만든 시험 조건이지, 원본 AIToolsLab 배달 로봇에서 측정된 오류가 아닙니다.

AI에게는 ‘새 그림을 그려 가리지 말고, 팔 묶음과 몸통은 보존한 채 회전 기준만 실제 어깨로 옮겨 달라’고 요청합니다. 수정 범위를 하나로 좁히면 고친 뒤의 비교도 쉬워집니다.

왕복의 양끝이 이어져야 합니다

팔의 한 번: 출발 → 반대쪽 → 다시 출발



교육용 설계값 · 왕복 2.4초

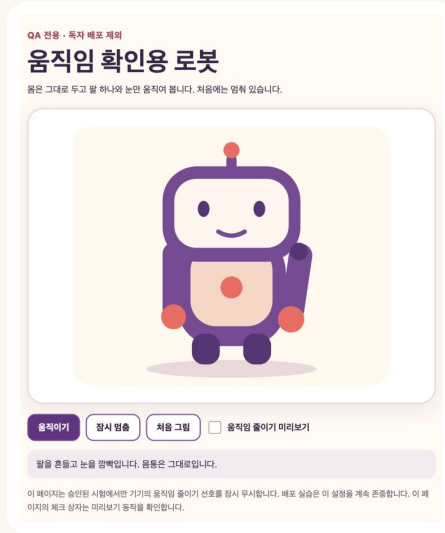
설명 도해 · 각도와 2.4초는 이 책의 교육용 설계값입니다.

팔이 어깨에 붙어 있어도 반복의 끝에서 출발 자세로 갑자기 튀면 인사가 끊겨 보입니다. 이번 왕복은 -20° 에서 출발해 $+20^\circ$ 에 닿은 뒤 다시 -20° 로 돌아옵니다. 끝 자세와 다음 반복의 첫 자세가 같으므로, 경계에서 큰 위치 차이가 생기지 않도록 설계했습니다.

그렇다고 숫자만 같으면 언제나 자연스러운 것은 아닙니다. 방향을 바꾸는 순간의 속도도 눈에 보이는 느낌을 바꿉니다. 이 책은 복잡한 곡선 계산보다 출발·가운데·끝 화면을 먼저 비교합니다. 사람이 보고 ‘멈칫한다’고 느낀다면 어느 순간인지 기록해야 다음 수정이 가능합니다.

순수 함수 시험은 0, $1/4$, 중간, $3/4$, 끝 시점의 각도를 확인했습니다. 실제 QA 화면에서는 여러 순간의 팔 각도를 읽었습니다. 두 시험을 합쳐도 모든 프레임이 완벽하다는 증거는 아닙니다.

눈이 닫힌 채 남는다면



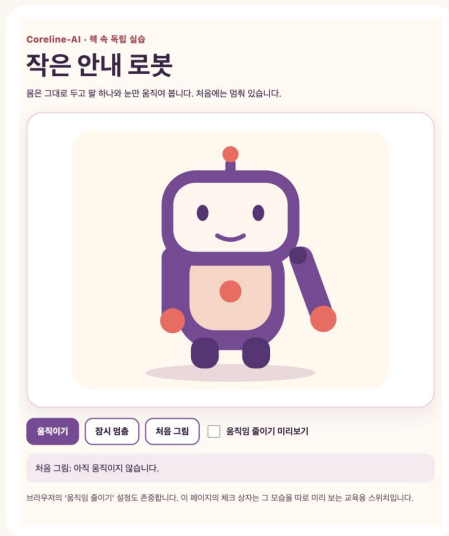
실제 Chrome QA 전용 화면 · 독자용 ZIP에는 이 시험 페이지가 없습니다.

팔만 살피다 보면 눈이 감긴 뒤 돌아오지 않는 문제를 놓칠 수 있습니다. 눈 깜빡임은 짧게 좁아졌다가 다시 열리는 왕복입니다. 이 예제에서는 열린 눈의 세로 반지름이 11이고, 가장 좁을 때 약 2가 되도록 정했습니다.

눈이 계속 좁다면 ‘표정이 이상하다’고만 하지 말고 ‘줄어든 눈의 높이가 다음 반복의 출발값으로 돌아오지 않는다’고 설명하세요. 팔의 회전 기준을 고친 작업과 눈의 왕복을 동시에 고치지 않는 것도 중요합니다. 서로 다른 문제를 섞으면 무엇이 나아졌는지 비교하기 어렵습니다.

QA 전용 Chrome 표본에서는 눈의 세로 반지름이 약 2부터 11까지 달라진 장면을 확인했습니다. 이는 현재 기기의 움직임 줄이기 선호를 바꾸지 않고 별도 시험 화면에서 본 결과이며, 독자용 실습 ZIP의 일반 재생을 같은 환경에서 입증한 것은 아닙니다.

고쳤다면 같은 조건으로 다시 봅니다



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

수정 전과 뒤를 비교할 때 그림 크기·배경·색까지 바뀌었다면 팔 기준점이 좋아졌는지 알 수 없습니다. 이 예제의 `before.html`과 정상 `index.html`은 그림 본체가 같고 기준점만 다르게 준비했습니다. 그래서 ‘팔이 떨어졌다/붙었다’는 한 질문에 집중할 수 있습니다.

실제 Chrome에서 두 페이지를 열어 보았을 때 오류판은 팔이 분리돼 보였고 정상판은 몸 옆에서 이어졌습니다. 합성 시험은 기준점 값의 옳고 그름을 따로 확인했습니다. 화면 관찰은 보이는 결과를, 함수 시험은 정해 둔 값을 확인합니다. 두 결과가 맞아도 실제 모든 움직임을 검수했다는 뜻은 아닙니다.

독자도 비교할 때 창 크기와 같은 출발 자세를 맞추세요. 이전 파일을 덮어썼다면 비교할 기준이 사라집니다. 고치기 전에 복사본을 남기는 작은 습관이 수정의 이유를 설명할 수 있게 합니다.

고장 기록을 수정 요청으로 바꿉니다

팔을 돌리는 기준점 하나가 차이를 만듭니다

정상판 · 어깨



팔이 붙어 보임

오류판 · 먼 지점



팔이 몸을 떠나 보임

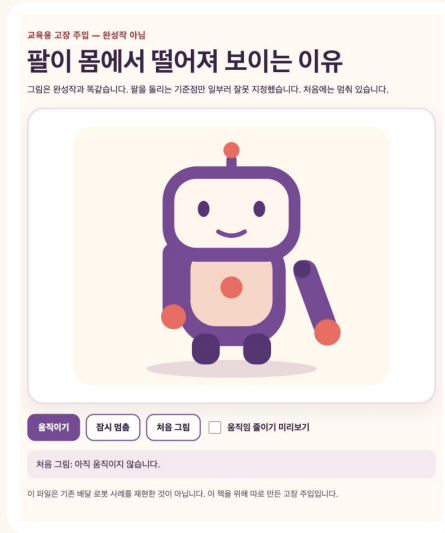
설명 도해 · 실제 실행 화면이 아닙니다.

AI에게 고장 화면만 보여 주면 눈에 띄는 색이나 크기를 바꿔 문제를 가릴 수도 있습니다. 원하는 수정과 보존할 조건을 함께 적어야 합니다. 예를 들어 ‘팔이 어깨에서 떨어져 보입니다. 오른팔 묶음의 회전 중심을 몸과 만나는 곳에 맞추고, 몸통·눈·색·버튼은 바꾸지 마세요. 수정 전 파일을 남겨 주세요’라고 말합니다.

이 문장은 실제 제품에 입력해 성공률을 측정한 기록이 아니라, 이번 실습에 맞춰 만든 교육용 요청 예시입니다. AI가 코드를 제안해도 적용 여부는 사람이 결정합니다. 낯선 외부 라이브러리나 서비스 연결을 새로 요구하면 이번 로컬 실습 범위 밖이므로 먼저 이유를 확인합니다.

수정 뒤에는 오류판과 정상판을 같은 창에서 번갈아 봅니다. 팔이 붙었다는 인상뿐 아니라 다른 부위가 그대로인지 확인해야 ‘원한 한 가지를 고쳤다’고 말할 수 있습니다.

이번 장의 작업물: 오류·복구 기록



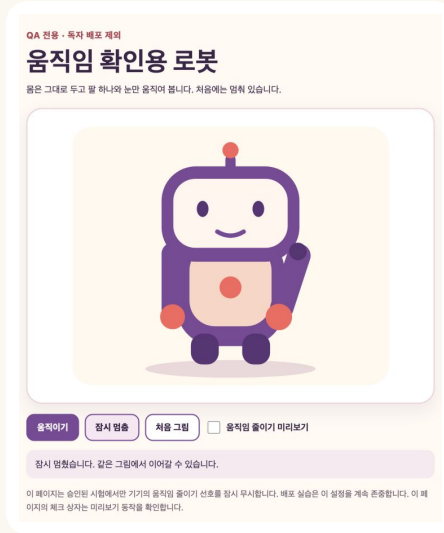
실제 Chrome의 의도적 오류판 · 정상 실습이 아닙니다.

완성 예시는 다음과 같습니다. ‘증상: 팔이 어깨에서 떠 보임 / 기대: 팔이 몸과 만나는 곳을 중심으로 회전 / 원인 후보: 회전 기준 x좌표 / 수정: 오류판 (402, 194)에서 정상판 (332, 194)로 / 다시 본 것: 몸통·눈·색은 그대로, 팔 연결이 회복.’ 이 숫자와 결론은 새 교육 예제의 의도적 오류에만 해당합니다.

내 기록에는 ‘실습 파일: ____ / 오류 장면: ____ / 그대로인 부위: ____ / 의심한 값: ____ / 바꾼 값: ____ / 같은 조건의 다시 보기: ____ / 아직 불확실한 점: ____’을 적습니다. 확인하지 못한 원인을 추측으로 단정하지 않는 것이 중요합니다.

이제 팔이 붙어 보인다고 해서 작업이 끝난 것은 아닙니다. 움직임을 멈추고 다시 봤을 때도 같은 규칙이 유지돼야 합니다. 다음 장에서는 그 상태를 어떻게 보관하고 이어 가는지 살펴봅니다.

05 멈추고 천천히 볼 수 있게 합니다



실제 Chrome QA 전용 화면 · 독자용 ZIP에는 이 시험 페이지가 없습니다.

팔이 움직이는 장면을 따라가기 어렵다면 잠시 멈춰 자세를 보아야 합니다. 이 때 멈춤 버튼이 로봇을 출발 자세로 되돌려 버리면 중간 자세를 살필 수 없습니다. 이번 실습은 ‘잠시 멈춤’과 ‘처음 그림’을 다른 행동으로 둡니다.

실제 QA 화면에서는 팔이 오른쪽으로 올라간 때 멈춤을 누른 뒤 각도와 눈 모양이 유지됐습니다. 다시 ‘움직이기’를 누르자 그 자리에서 변화가 이어졌습니다. 현재 Chrome의 관찰이지 모든 기기의 성능 시험은 아닙니다.

버튼을 누르기 전에 내가 기대하는 결과를 한 문장으로 적어 두세요. ‘잠시 멈춤은 현재 자세 유지, 처음 그림은 출발 자세.’ 이 문장을 기준으로 화면을 보면 오류를 기능으로 착각할 가능성이 줄어듭니다.

멈춘 시간을 움직임에 더하지 않습니다

재생과 멈춤은 서로 다른 약속입니다

재생

팔과 눈이 바뀝니다



잠시 멈춤

지금 자세를 보관합니다



다시 재생

그 자리에서 이어집니다

처음 그림은 별도 버튼: 출발 자세로 돌아갑니다

설명 도해 · 실제 실행 화면이 아닙니다.

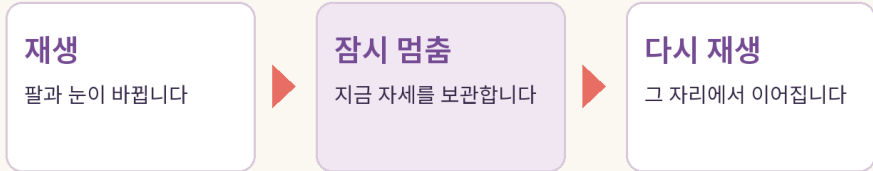
중간에서 10초 쉬었다가 다시 재생했다고 팔이 10초치 움직임을 한꺼번에 따라잡아서는 안 됩니다. 일시정지는 화면을 잠깐 가리는 일이 아니라, 그동안 흐른 시간을 애니메이션 진행량에서 빼는 결정입니다.

이번 예제는 재생 중에만 경과 시간을 더하고, 멈춘 동안에는 현재 경과 시간을 보관합니다. 다시 재생하면 그 이후의 시간부터 더합니다. 그래서 중간 자세에서 이어지는 행동을 만들 수 있습니다. 이 규칙은 저자가 정한 실습 설계이며, 특정 브라우저가 자동으로 대신해 주는 기능이라고 소개하지 않습니다.

만약 멈춘 뒤 돌아왔더니 팔이 갑자기 다른 쪽에 있다면 ‘일시정지 중에도 시간이 더해지는지’부터 확인하세요. 눈이 함께 튜다면 같은 시간 규칙을 눈에도 적용하는지 살펴봅니다.

다시 누르기와 처음부터 보기

재생과 멈춤은 서로 다른 약속입니다



처음 그림은 별도 버튼: 출발 자세로 돌아갑니다

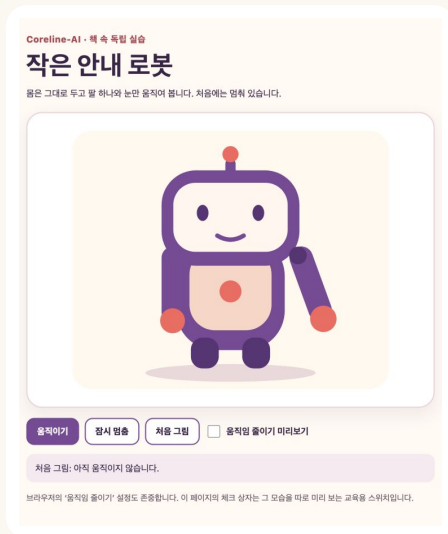
설명 도해 · 실제 실행 화면이 아닙니다.

재생 중 재생 버튼을 다시 누를 때는 새 반복을 쌓지 않습니다. 멈춤 뒤 재생은 이어 가기이고, 처음 그림은 출발 자세로 돌리기입니다. 같은 버튼을 여러 번 눌러도 이 역할이 바뀌지 않아야 독자가 자신의 행동을 믿을 수 있습니다.

합성 함수 시험에서는 이미 재생 중인 상태에 재생을 다시 요청해도 새로운 시계를 만들지 않는 규칙을 확인했습니다. 실제 QA에서는 멈춤·재생·초기화의 화면 반응을 보았습니다. 배경 탭에서 얼마나 오래 있었는지까지 같은 시험으로 입증한 것은 아닙니다.

확인할 때는 ‘재생 → 재생 다시 누름 → 잠시 멈춤 → 다시 재생 → 처음 그림’ 순서로 진행해 보세요. 중간의 팔 자세가 어떻게 남는지 설명할 수 있다면 버튼 이름과 동작이 맞는지 판단하기 쉬워집니다.

보이지 않는 탭에서 일어나는 일



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

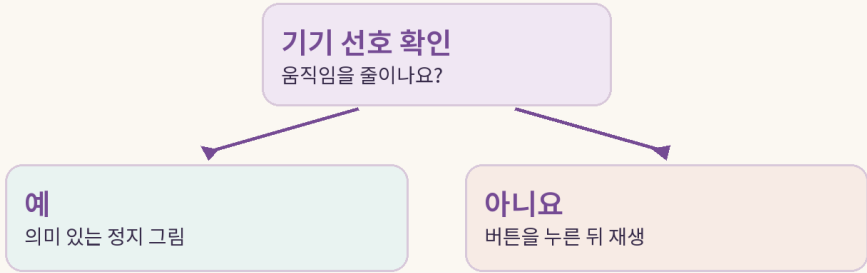
다른 탭을 보다가 돌아왔을 때 로봇이 보이지 않는 동안에도 계속 움직였는지는 독자가 알아채기 어렵습니다. 이 실습은 문서가 숨겨지면 재생을 멈추고, 돌아온 뒤에도 자동으로 이어 가지 않도록 작성했습니다. 사용자가 다시 재생을 누르는 것이 분명한 경계입니다.

브라우저의 `visibilitychange` 이벤트는 문서가 보이거나 숨겨지는 변화를 알려 줍니다. [MDN visibilitychange](#), 확인 2026-10-18. 다만 이 책의 현재 Chrome 연결 환경에서는 실제 탭 이탈을 독립적으로 입증하지 못해 그 동작을 통과로 기록하지 않았습니다. 코드와 합성 시험의 정책 확인과 실제 탭 전환 시험은 구분해야 합니다.

독자가 직접 확인했다면 어떤 브라우저에서 어느 탭으로 옮겼다가 왔는지, 돌아온 순간 로봇이 멈춰 있었는지를 기록하세요. 미확인이라면 그대로 남겨 둡니다.

움직임 줄이기 선호를 존중합니다

움직임 줄이기 선호를 먼저 살핍니다



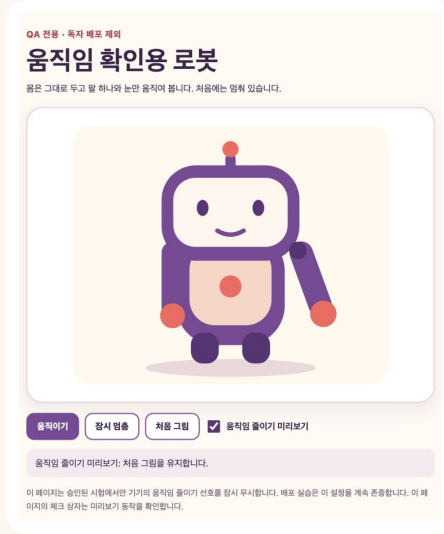
설명 도해 · 실제 실행 화면이 아닙니다.

움직임을 줄이고 싶은 사람에게 로봇이 저절로 흔들리는 화면을 강요할 이유는 없습니다. 브라우저는 기기의 움직임 줄이기 선호를 감지할 수 있습니다. [MDN prefers-reduced-motion](#), 확인 2026-10-18. 이번 실습은 그 선호가 있으면 팔과 눈의 장식적 움직임 대신 처음 그림을 유지합니다.

저자의 검수 기기에서는 실제 선호가 켜져 있었습니다. 독자용 정상판에서 재생을 눌러도 정지 자세가 유지됐고, 안내 문구도 표시됐습니다. OS 설정을 바꿔 반대 조건을 시험했다고 주장하지 않습니다. 별도 QA 전용 페이지에서만 움직임을 확인했습니다.

움직임을 볼 수 없더라도 몸통·팔·눈의 관계를 이해할 수 있어야 합니다. 정지 그림과 설명이 남아 있는지 먼저 보는 것이 접근성 점검의 출발입니다.

미리보기는 실제 기기 설정이 아닙니다



실제 Chrome QA 전용 화면 · 독자용 ZIP에는 이 시험 페이지가 없습니다.

실습 화면에는 ‘움직임 줄이기 미리보기’ 체크 상자가 있습니다. 이 상자를 누르면 처음 그림을 유지하는 모습을 따로 볼 수 있습니다. 하지만 화면 안의 체크 상자는 운영체제 설정을 바꾸지 않습니다. 둘의 결과가 같아 보여도 검수 근거는 다릅니다.

현재 기기의 실제 선호가 켜져 있어 독자용 페이지는 이미 정지 상태였습니다. 그래서 저자는 움직임 확인을 위해 사용자 승인을 받은 별도 QA 페이지를 만들었습니다. 그 페이지는 선호를 시험 목적으로만 무시하고 독자 ZIP에는 넣지 않았습니다.

독자가 자신의 기기에서 미리보기를 확인할 때는 체크 전·후의 안내 문구와 그림을 함께 봅니다. ‘체크가 되었다’만 확인하면 실제로 로봇이 멈추는지 놓칠 수 있습니다.

좁은 창과 키보드에서도



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

움직임이 잘 보여도 버튼을 누를 수 없으면 실습은 완성되지 않습니다. 검수에서는 기존 Chrome 창 너비를 390과 320으로 줄여 가로 넘침과 버튼 배치를 보았습니다. 세 버튼은 화면 안에 남았고 확인한 높이는 48픽셀이었습니다.

키보드 Enter로 재생·잠시 멈춤·처음 그림을 눌러 상태 문구와 자세가 바뀌는 것도 QA 페이지에서 확인했습니다. 이 결과는 데스크톱 Chrome의 좁은 창과 키보드 조작입니다. 실제 휴대폰 터치·큰 글자·화면 읽기 도구 검수는 하지 않았습니다.

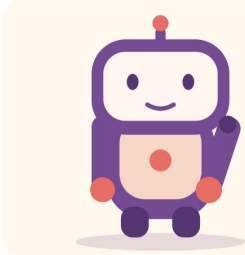
내 환경에서 버튼이 겹치거나 가려진다면 ‘모바일이 안 된다’고만 쓰지 말고 화면 폭과 어느 버튼이 보이지 않는지 적으세요. 새 기기에서 생긴 문제는 다음 수정의 입력이 됩니다.

이번 장의 작업물: 멈춤 확인표

QA 전용 · 독자 배포 제외

움직임 확인용 로봇

몸은 그대로 두고 팔 하나와 눈만 움직여 봅니다. 처음에는 망쳐 있습니다.



움직이기
잠시 멈춤
처음 그림
☐ 움직임 줄이기 미리보기

잠시 멈췄습니다. 같은 그림에서 이어갈 수 있습니다.

이 페이지는 승인된 시험에서만 기기별 움직임 줄이기 선택을 잠시 무시합니다. 배포 실습은 이 설정을 계속 존중합니다. 이 페이지의 체크 결과는 미리보기 동작을 확인합니다.

실제 Chrome QA 전용 화면 · 독자용 ZIP에는 이 시험 페이지가 없습니다.

완성 예시는 ‘재생 중 팔 변화 관찰 / 잠시 멈춤 뒤 같은 자세 유지 / 다시 재생은 직접 버튼을 누른 뒤 / 처음 그림에서 팔과 눈 출발값 / 움직임 줄이기 선택이 켜진 기기에서는 정지 그림과 안내 문구’입니다. 이 표는 현재 새 예제의 설계와 범위 안에서 작성한 것입니다.

내 기록에는 ‘브라우저와 창 너비: ____ / 기기 선호: ____ / 멈춘 순간의 자세: ____ / 다시 본 순간의 자세: ____ / 재생 버튼을 다시 눌렀는가: ____ / 처음 그림 결과: ____ / 직접 해 보지 못한 시험: ____’을 적습니다. 미수행을 빈칸이 아니라 미수행이라고 적으면 다음 사람이 검수 범위를 알 수 있습니다.

다음 장에서는 정상판을 보존한 채 색과 속도를 바꿉니다. 멈춤 규칙이 안정된 상태에서 한 가지씩 바뀌어야 내 캐릭터의 변화가 무엇 때문인지 알 수 있습니다.

06 색과 속도를 바꿔 내 캐릭터로



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

완성된 로봇을 내 취향에 맞게 바꿀 때 가장 흔한 실수는 원본부터 덮어쓰는 것입니다. 그러면 마음에 들지 않는 결과가 나와도 무엇을 바꾸었는지 비교하기 어렵습니다. 먼저 정상판 폴더를 그대로 두고 별도 변형 폴더를 만듭니다.

이 책의 작은 변형은 코럴 포인트를 청록으로 바꾸고 팔의 왕복 길이를 2.4초에서 3.2초로 늘리는 것입니다. 두 값 모두 저자가 정한 실습 예시이며 선호도 시험이나 표준 속도가 아닙니다. 색과 속도를 동시에 바꾸기보다 각각 확인한 뒤 결합합니다.

실제 Chrome에서는 변형판의 청록 포인트가 보였습니다. 현재 기기의 움직임 줄이기 선호 때문에 독자용 변형판의 재생 속도 차이는 눈으로 검수하지 못했습니다. 확인한 것과 아직 확인 못 한 것을 구분한 채 다음 단계로 갑니다.

색 하나만 바꾼 첫 비교



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

코럴을 청록으로 바꾸면 얼굴 위 표시, 몸통 가운데 점, 손끝 같은 포인트가 함께 바뀝니다. 팔의 위치나 눈 모양은 그대로여야 합니다. 색만 바꾸었다고 말하려면 어떤 부분의 색이 달라졌고 어떤 부분은 달라지지 않았는지 확인해야 합니다.

정상판의 포인트 값은 `#e86d62`, 변형판은 `#3b9da3`입니다. 이 16진 색 값은 이 예제의 설계값이며 색각 보조나 접근성 수준을 보증하지 않습니다. 화면에서 색만으로 상태를 구분하지 않도록 버튼 문구와 설명도 유지합니다.

실제 Chrome에서 변형판의 다섯 포인트가 청록으로 표시된 것을 확인했습니다. 화면 색은 기기마다 달라 보일 수 있으므로, 독자는 원본과 변형을 같은 창에서 번갈아 보며 의도치 않은 색 변화가 없는지 살핍니다.

속도는 느낌이 아니라 길이부터

팔의 한 번: 출발 → 반대쪽 → 다시 출발



교육용 설계값 · 왕복 2.4초

설명 도해 · 각도와 2.4초는 이 책의 교육용 설계값입니다.

팔이 ‘조금 느리게’ 움직였으면 좋겠다는 말은 사람마다 다르게 받아들입니다. 이번 변형에서는 한 번 왕복하는 시간을 2.4초에서 3.2초로 늘립니다. 숫자가 커지면 같은 각도 범위를 돌아오는 데 더 긴 시간이 필요하므로 더 천천히 보이도록 설계한 것입니다.

이 설명은 코드의 계산과 의도에 관한 것이지, 실제 독자용 브라우저에서 속도를 측정했다는 말이 아닙니다. 현재 시험 기기에서는 움직임 줄이기 선호가 커져 독자용 두 판이 모두 정지했습니다. 순수 함수로 왕복 시간의 경계를 확인했고, 독자의 기기에서 실제 느낌은 별도로 확인해야 합니다.

AI에게 바꿀 때는 ‘팔의 왕복 시간 한 값만 3200ms로 바꾸고, 각도 범위·눈 깜빡임·몸통 고정은 유지해 달라’고 요청하세요. 숫자 하나만 고쳐도 비교할 조건이 분명해집니다.

두 변경을 섞기 전에



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

색과 속도를 한꺼번에 바꾸면 결과가 마음에 들어도 어느 변경이 영향을 주었는지 설명하기 어렵습니다. 첫 비교에서는 색만 보고, 두 번째 비교에서는 시간 값만 확인합니다. 마지막에 둘을 합친 변형판을 봅니다.

색은 정지 화면에서도 확인할 수 있지만 속도는 정지 사진으로 판단할 수 없습니다. 이 차이를 인정하면 검수 기록도 달라집니다. ‘청록 포인트는 실제 Chrome에서 보임 / 팔 속도는 코드 · 합성 시험으로만 확인, 현재 기기의 독자용 재생에서는 미확인’이라고 적는 것이 정확합니다.

확인하지 못한 항목 때문에 이미 확인한 색 결과까지 버릴 필요는 없습니다. 반대로 색이 맞았다고 해서 속도도 통과했다고 적지 않습니다. 작은 변경을 하나씩 나누는 이유입니다.

AI에게 보존 범위를 함께 말합니다

재생과 멈춤은 서로 다른 약속입니다

재생

팔과 눈이 바뀝니다



잠시 멈춤

지금 자세를 보관합니다



다시 재생

그 자리에서 이어집니다

처음 그림은 별도 버튼: 출발 자세로 돌아갑니다

설명 도해 · 실제 실행 화면이 아닙니다.

변형을 요청할 때는 원하는 새 모습뿐 아니라 건드리면 안 되는 규칙을 적습니다. ‘정상판은 보존하고 새 폴더에 변형판을 만들어 줘. 포인트 색은 청록으로, 팔 왕복은 3.2초로 하되 몸통 위치·눈 깜빡임·재생·멈춤·처음 그림은 바꾸지 마. 바뀐 파일과 확인할 행동을 알려 줘.’

이 문장은 이번 실습을 설명하기 위한 교육용 요청 예시입니다. 특정 AI 서비스에 이 문장을 입력해 동일한 결과를 보장하거나 성공률을 측정한 기록이 아닙니다. AI가 새 패키지 설치나 외부 업로드를 제안하면 이번 로컬 실습에 왜 필요한지 먼저 확인합니다.

사람은 수정안을 받아들일지 결정하고, 정상판을 보존하며, 눈으로 색을 비교하고 가능한 환경에서 속도 차이를 관찰합니다. AI의 설명만으로 ‘검수 완료’라고 적지 않습니다.

같은 출발 화면으로 비교합니다



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

원본과 변형판을 비교하려면 둘 다 ‘처음 그림’에서 시작하세요. 한쪽은 팔을 든 순간이고 다른 쪽은 출발 자세라면 색과 속도에 집중하기 어렵습니다. 같은 창 크기에서 두 판을 차례로 열고, 눈에 보이는 포인트와 버튼 문구부터 확인합니다.

변형판은 원본의 로컬 CSS·JavaScript를 상대 경로로 참조합니다. 그래서 ZIP을 풀 때 두 실습 폴더의 상대 위치를 바꾸면 화면이 깨질 수 있습니다. 새 폴더에서 열기 시험이 필요한 이유입니다. 샘플 실습 ZIP v3는 압축 CRC·상대 경로·합성 시험을 다시 확인했고, Chrome에서 정상판과 변형판의 정지 화면을 실제로 열었습니다.

색이나 그림이 빠졌다면 코드를 다시 쓰기 전에 파일을 어디에 풀었는지, 두 폴더가 나란히 있는지 확인하세요. 파일 구조의 문제를 움직임 오류로 오해하지 않는 편이 빠릅니다.

작은 변형의 범위를 지킵니다



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

내 캐릭터를 만들고 싶다고 해서 얼굴·팔·몸통·배경을 모두 새로 그릴 필요는 없습니다. 이 책은 한 가지 색과 한 가지 시간 값을 바꾸는 작은 변형을 끝까지 확인하게 합니다. 한 번의 성공을 만든 뒤라야 다음 변경에서도 무엇이 원인인지 알 수 있습니다.

새로운 그림을 불러오거나 다른 사람의 캐릭터를 복사할 계획이라면 사용권리와 파일의 외부 참조를 따로 살펴야 합니다. 이번 실습 ZIP에는 원본 AIToolsLab 배달 로봇의 그림·코드·ZIP을 넣지 않았고, 새로 그린 안내 로봇만 사용했습니다. 이것이 다른 자산의 사용 권리를 자동으로 보증한다는 뜻은 아닙니다.

책의 숫자와 색은 교육 예제의 값입니다. 내 캐릭터에 맞지 않으면 조금씩 바꿀 수 있지만, 그때도 비교할 정상판과 바꾸지 않을 조건을 먼저 남겨 두세요.

이번 장의 작업물: 변형 비교표



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

완성 예시는 ‘원본: 코럴 #e86d62, 팔 왕복 2400ms / 변형: 청록 #3b9da3, 팔 왕복 3200ms / 화면 확인: 청록 포인트 다섯 곳 실제 Chrome에서 관찰 / 코드·합성 확인: 두 시간 값 / 미수행: 움직임 줄이기 선택을 끈 독자용 브라우저에서 실제 속도 비교’입니다. 한 표 안에서도 증거의 종류를 나누어 적었습니다.

내 비교표에는 ‘원본 폴더: ____ / 변형 폴더: ____ / 바꾼 색: ____ / 화면에서 본 색: ____ / 바꾼 왕복 시간: ____ / 실제 재생에서 본 느낌: ____ / 유지된 기능: ____ / 미확인 항목: ____’을 적습니다. 미확인 칸을 지우지 않아야 다음 검수가 가 이어서 확인할 수 있습니다.

이제 캐릭터가 완벽하다고 선언할 필요는 없습니다. 한 그림을 나누고, 한 동작을 만들고, 잘못된 기준을 고치고, 확인 가능한 작은 변형을 남겼다면 이 책의 목표에 도달했습니다.

실습 파일을 다시 여는 법



실제 Chrome의 새 실습 화면 · 정지 사진은 움직임을 대신하지 않습니다.

실습 ZIP을 새 폴더에 풀고 두 실습 폴더가 같은 `examples` 안에 있는지 봅니다. 최상위 폴더에서 `python3 -m http.server 8769 --bind 127.0.0.1`을 실행한 뒤 브라우저에서 `http://127.0.0.1:8769/examples/guide-robot-v1/`을 엽니다. 포트가 이미 쓰이면 다른 번호를 명령과 주소에 함께 적습니다.

`index.html`은 정상판, `before.html`은 의도적 오류판입니다. 변형판은 별도 폴더의 `index.html`입니다. 움직임이 보이지 않으면 기기의 움직임 줄이기 선호와 안내 문구를 확인하세요. 다른 사람의 설정을 바꾸어 재생을 강요하지 않습니다.

이 로컬 서버는 같은 컴퓨터에서 교육 파일을 보는 용도입니다. 외부 게시가 아닙니다. 종료할 때는 실행 창에서 중단합니다.

확인 기록을 읽는 방법



실제 Chrome 390×844 창 · 실제 휴대폰 화면은 아닙니다.

이 책의 화면 캡처는 새 안내 로봇을 실제 Chrome에서 연 장면입니다. 움직이는 팔·눈의 캡처는 사용자 승인을 받은 별도 QA 전용 페이지에서 만들었고, 그 페이지는 독자용 ZIP에 넣지 않았습니다. 좁은 화면은 데스크톱 Chrome의 창 크기 시험이지 실제 휴대폰 시험이 아닙니다.

합성 함수 시험 13개는 각도·눈 높이·일시정지 시간 계산·의도적 오류 감지를 확인합니다. 이것은 독자의 기기에서 그림이 예쁘게 보이는지를 대신하지 않습니다. 반대로 사진 한 장이 모든 시간 계산의 정확성을 증명하지도 않습니다.

아직 확인하지 못한 것은 실제 휴대폰의 터치·큰 글자·화면 읽기 도구, 현재 기기의 움직임 줄이기 선호를 끈 독자용 재생, 실제 탭 이탈 뒤의 동작입니다. 이 빈칸을 숨기지 않고 남겨 둡니다.

공식 자료와 책의 설계값

SVG의 크기 변경과 그림 묶음, 브라우저의 다음 화면 요청, 움직임 줄이기 선택호는 [MDN SVG](#), [SVG ``<g>``](#), [requestAnimationFrame](#), [prefers-reduced-motion](#) 문서를 2026-10-18 확인해 설명했습니다. 탭 가시성은 [visibilitychange](#)를 참고했습니다.

팔 주기 2400ms, 변형 3200ms, 눈 주기 3600ms, 각도 -20도부터 +20도, 코럴·청록 색은 저자가 이 그림에 맞춰 고른 교육용 설계값입니다. MDN의 공식 권장 숫자나 실험 결과로 받아들이지 마세요.

책 속 AI 수정 요청은 실제 AI 제품에서 실행해 성능을 검증한 대화록이 아닌 교육용 예시입니다. 사용한 도구와 기기, 파일 버전을 독자가 기록하고 결과를 직접 비교해야 합니다.

다음 한 장의 그림을 시작할 때

재생과 멈춤은 서로 다른 약속입니다

재생

팔과 눈이 바뀝니다



잠시 멈춤

지금 자세를 보관합니다



다시 재생

그 자리에서 이어집니다

처음 그림은 별도 버튼: 출발 자세로 돌아갑니다

설명 도해 · 실제 실행 화면이 아닙니다.

새 캐릭터를 만들 때는 이 책의 로봇을 그대로 복제하기보다 세 가지 질문을 가져가세요. 무엇이 움직이는가. 어디를 중심으로 움직이는가. 멈추고 다시 볼 때 어떤 모습이 남아야 하는가. 먼저 한 문장으로 답하고, 바꾸지 않을 부분을 적습니다.

AI에게는 그 작은 범위만 요청하고 원본을 보존하게 합니다. 결과를 받은 뒤에는 정지 그림, 한 번의 움직임, 멈춤, 다시 보기, 좁은 창을 차례로 봅니다. 한 번에 모든 효과를 더하지 않으면 실패했을 때 다시 시작할 수 있습니다.

이 책의 작업물은 로봇 그림 그 자체보다 관찰 메모·부위 카드·움직임 카드·오류 기록·변형 비교표입니다. 다른 도구와 그림을 쓰더라도 이 기록은 사람이 원하는 결과를 설명하고 승인하는 바탕이 됩니다. 출판과 배포의 최종 판단도 사람의 몫입니다.