

애니메이션 알아야 할것

항목	권장	이유
변경 속성	transform, opacity 우선	width, top등의 화면을 다시 계산하거나 그리지 않도록 줄이기
will-change 속성	필요한 곳에만 일시 사용	브라우저에게 미리 작동을 알려 움직임을 부드럽게 해주는 속성이지만, 너무 많이 쓰면 메모리 낭비가 심함
긴 애니메이션	과도한 반복 지양	피로도/전력 소모

필수 속성 요약 표

속성	값(예)	설명	짧은 예시
animation-name	fadeIn등	사용할 keyframes 이름	animation-name: fadeIn;
animation-duration	1s, 500ms	1회 재생 시간	animation-duration: 600ms;
animation-timing-function	ease, linear, ease-in, ease-out, ease-in-out, steps(n), cubic-bezier(...)	속도 곡선	animation-timing-function: ease-out;
animation-delay	0s, 1.2s	시작 지연 시간	animation-delay: .3s;
animation-iteration-count	1, 3, infinite	반복 횟수	animation-iteration-count: infinite;
animation-direction	normal, reverse, alternate(왔다갔다:홀수역방향) alternate-reverse	진행 방향	animation-direction: alternate;
animation-fill-mode	none, forwards, backwards, both none (기본) → 시작 전/끝난 후 모두 원래 상태 forwards → 애니메이션이 끝난 후 마지막 상태 유지 backwards → 애니메이션 시작 전 처음 상태 적용 both → 시작 전에는 처음 상태, 끝난 후에는 마지막 상태 둘 다 유지		animation-fill-mode: both;
animation-play-state	running, paused	재생/일시정지	animation-play-state: paused;
animation (축약)	name duration timing delay count direction fill play-state	한 줄에 작성	animation: fadeIn 600ms ease-out .2s 1 normal both;
축약 순서: name → duration → timing-function → delay → iteration-count → direction → fill-mode → play-stat			

@keyframes 기본 패턴

@keyframes fadeIn { from { opacity: 0; } to { opacity: 1; } }	keyframes floatY { 0% { transform: translateY(0); } 50% { transform: translateY(-8px); }
--	--