<CSS animation>

* 사용법
  + @keyframes 애니메이션 이름 { 애니메이션 제작}
* animation-name : 애니메이션 이름
* **animation-duration : 애니메이션 실행 시간**
* **animation-delay: element : 로드 후 애니메이션이 시작되기까지의 시간**
* **animation-iteration-count : 애니메이션이 몇 번 반복할지 지정 (1, 1.5, infinite 등)**
* animation-direction : 애니메이션 종료 후 역방향(or 정방향)으로 시작할지 지정
  + normal(매 사이클 정방향) , reverse(매 사이클 역방향)
  + alternate(각 주기마다 방향을 뒤집음, 정방향 시작), alternate-reverse( ‘’ , 역방향 시작)
* animation-play-state : 애니메이션을 멈추거나 다시 시작
* animation-timing-function : easy(default), transition과 동일
* **animation-fill-mode : 끝난 후 상태 지정**
  + none(default), forwards(마지막 설정값 유지), backwards(첫번째 설정값 유지)
  + both(애니메이션은 앞 뒤 양쪽 모두의 규칙을 따르므로 애니메이션 속성이 양방향으로 확장)

<web layout>

* grid : 가이드라인, 가로로 12개의 박스로 나눔(세로줄 11개)
* 가터 : 박스 사이에 공간을 두기 위해 (padding)과 비슷한 공간을 두는 것

<javascript>

* join(연결문자) : 배열 객체의 데이터를 연결문자 기준으로 1개의 문자형 데이터로 반환
* reverse() : 배열 객체의 데이터 순서를 거꾸로 바꾼 후 반환
* sort() : 오름차순으로 정렬
* slice(index1, index2) : 원하는 인덱스 구간만큼 잘라서 배열 객체로 가져옴
* splice() : 지정 데이터를 삭제하고 그 구간에 새 데이터를 삽입
* concat() : 2개의 배열을 하나로 결함
* pop() : 마지막 인덱스에 저장된 데이터를 삭제
* push(new data) : 마지막 인덱스에 새 데이터를 삽입
* shift() : 첫번째 인덱스에 저장된 데이터를 삭제
* unshift(new data) : 제일 앞에다가 새 데이터를 삽입
* length : 배열의 개수