

인터랙션스튜디오: 웹인터랙션프로그래밍



웹인터랙션프로그래밍

ododog12@naver.com

010 9215 5512

10월 3일 개천절!!!!

css animation

숫자 or infinite

animation : 이름 작동시간 속도속성 딜레이시간 반복 실행방향

ease
linear
ease-in-out
cubic-bezier(n,n,n,n)

reverse
alternate
alternate-reverse

```
.box {  
  position: absolute;  
  top: 100px;  
  left: 100px;  
  width: 100px;  
  height: 100px;  
  background: red;  
  animation: move 10s ease-in-out forwards;  
}  
  
@keyframes move {  
  0% {top: 100px; left: 100px;}  
  25% {top: 200px; left: 200px;}  
  50% {top: 300px; left: 500px;}  
  75% {top: 200px; left: 300px;}  
  100% {top: 500px; left: 400px;}  
}
```

```
animation:move 10s cubic-bezier(0.25, 0.75, 0.58, 1) forwards;
```

```
animation:move 10s cubic-bezier(0.25, 0.75, 0.58, 1) infinite;
```

```
animation:move 10s cubic-bezier(0.25, 0.75, 0.58, 1) alternate-reverse;
```

```
<div class="box"></div>  
<button class="stop">멈춰</button>  
<button class="restart">다시</button>
```

```
const box = document.querySelector('.box')  
const stop = document.querySelector('.stop')  
const restart = document.querySelector('.restart')  
  
stop.addEventListener('click', () => {  
  box.style.animationPlayState = 'paused'  
})  
  
restart.addEventListener('click', () => {  
  box.style.animationPlayState = 'running'  
})
```


실습

animation

```
<div class="loading">  
  <div class="img_cha"></div>  
  <div class="img_shadow"></div>  
  <div class="bgWrap">  
    <div class="red"></div>  
    <div class="spring"></div>  
    <div class="green"></div>  
    <div class="yellow"></div>  
  </div>  
</div>
```

```
.loading {  
  width:500px;  
  height:500px;  
  position:absolute;  
  top:50%;  
  left:50%;  
  transform: translate(-50%, -50%);  
  background:skyblue;  
}
```

```
.loading .img_cha {  
  position: absolute;  
  bottom: 35px;  
  left: 170px;  
  width: 160px;  
  height: 160px;  
  background: url('img/img_loading.svg') no-repeat;  
  z-index: 2;  
  animation: jump .5s cubic-bezier(.17,.67,.32,.99) infinite alternate;  
}  
  
@keyframes jump {  
  0% {bottom: 35px;}  
  100% {bottom: 140px;}  
}
```

```
.loading .img_shadow {  
  position: absolute;  
  bottom: 30px;  
  left: 200px;  
  width: 100px;  
  height: 20px;  
  border-radius: 50%;  
  background: rgba(0, 0, 0, .5);  
  transform: scale(1);  
  animation: shadow .5s cubic-bezier(.17,.67,.32,.99) infinite alternate;  
}  
  
@keyframes shadow {  
  0% {transform: scale(1); background: rgba(0, 0, 0, .5);}  
  100% {transform: scale(.6); background: rgba(0, 0, 0, .2);}  
}
```

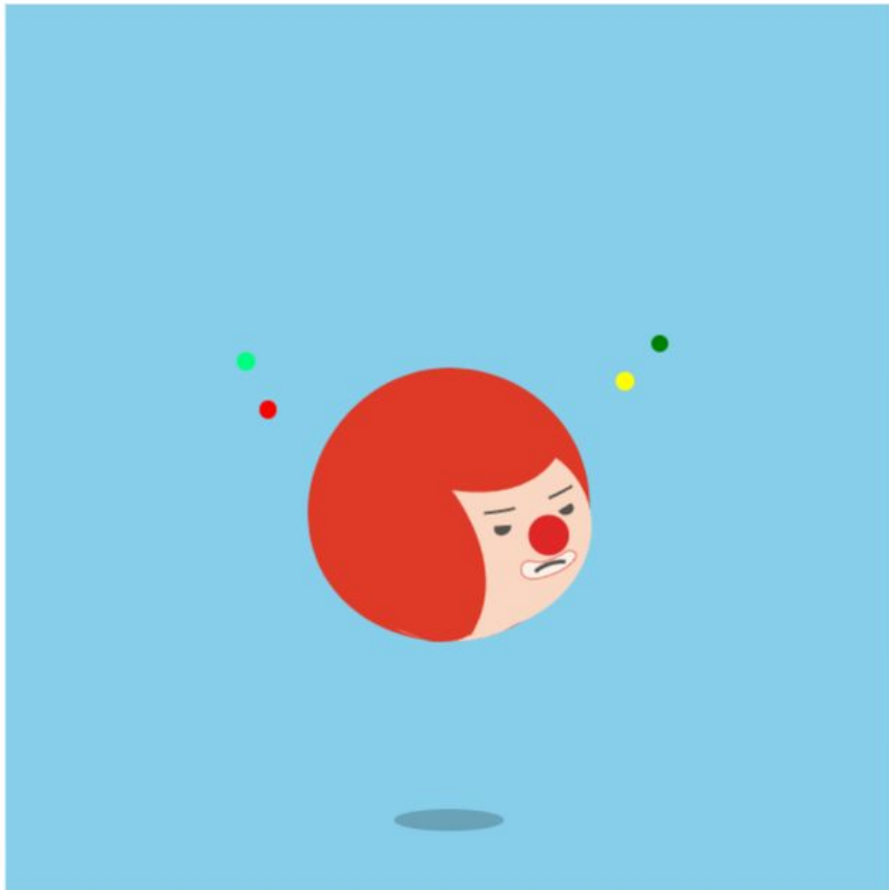
```
.loading .bgWrap > div {  
  width:10px;  
  height:10px;  
  border-radius:50%;  
  position:absolute;  
}  
  
.loading .bgWrap div.red {  
  bottom:132px;  
  left:30.33%;  
  background:red;  
  animation:flow1 1s cubic-bezier(.17,.67,.32,.99) infinite;  
}  
  
@keyframes flow1 {  
  0% {opacity:0;}  
  5% {opacity:1;bottom:132px;left:30.33%;}  
  80% {opacity:1;}  
  100% {opacity:0;bottom:320px;left:28%;}  
}
```

```
.loading .bgWrap div.spring {  
  bottom:172px;  
  left:33%;  
  background:springgreen;  
  animation:flow2 1s cubic-bezier(.17,.67,.32,.99) infinite;  
}  
  
@keyframes flow2 {  
  0% {opacity:0;}  
  5% {opacity:1;bottom:172px;left:33%;}  
  80% {opacity:1;}  
  100% {opacity:0;bottom:342px;left:23.2%;}  
}
```

```
.loading .bgWrap div.green {  
  bottom:182px;  
  right:30.33%;  
  background:green;  
  animation:flow3 1s cubic-bezier(.17,.67,.32,.99) infinite;  
}  
  
@keyframes flow3 {  
  0% {opacity:0;}  
  5% {opacity:1;bottom:182px;right:30.33%;}  
  80% {opacity:1;}  
  100% {opacity:0;bottom:352px;right:23.2%;}  
}
```



```
.loading .bgWrap div.yellow {  
  bottom:159px;  
  right:36.2%;  
  background:yellow;  
  animation:flow4 1s cubic-bezier(.17,.67,.32,.99) infinite;  
}  
  
@keyframes flow4 {  
  0% {opacity:0;}  
  5% {opacity:1;bottom:159px;right:36.2%;}  
  80% {opacity:1;}  
  100% {opacity:0;bottom:332px;right:26.4%;}  
}
```



```
<div class="toggle">  
  <label>  
    <input type="radio" name="mode" value="light" checked>  
    <span class="light">Light</span>  
  </label>  
  <label>  
    <input type="radio" name="mode" value="dark">  
    <span class="dark">Dark</span>  
  </label>  
</div>
```

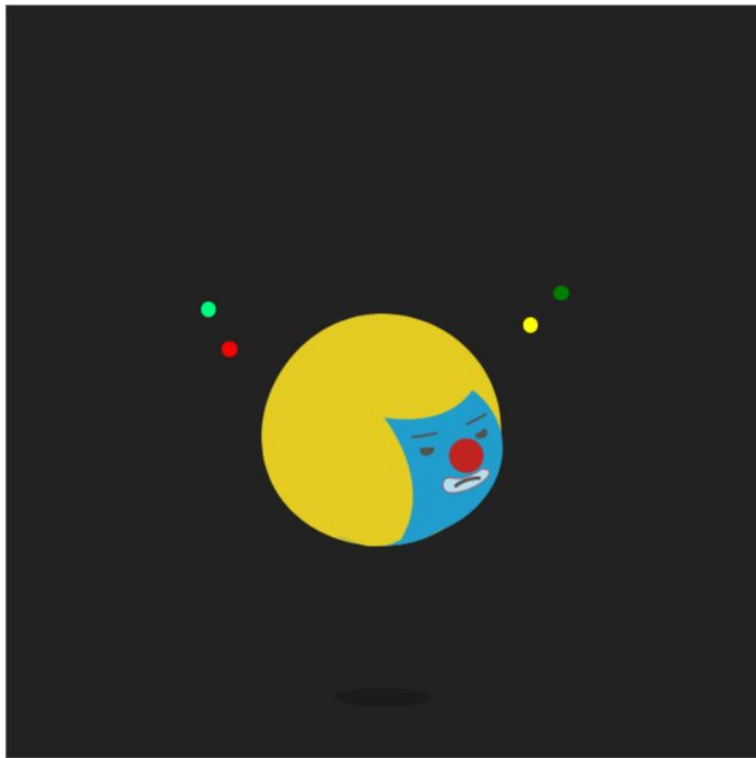
```
.toggle {  
  position: absolute;  
  top: 20px;  
  left: 50%;  
  transform: translateX(-50%);  
}  
.toggle input[type="radio"] {  
  display: none;  
}  
.toggle label {  
  cursor: pointer;  
}  
.toggle span {  
  display: inline-block;  
  padding: 10px 20px;  
  border-radius: 20px;  
  background: #ddd;  
  color: #555;  
  font-size: 16px;  
  margin: 0 5px;  
  transition: background-color 0.3s ease, color 0.3s ease;  
}  
.toggle input[type="radio"]:checked + span {  
  background: #333;  
  color: white;  
}
```

```
const radio = document.querySelectorAll('input[type="radio"]')
const loading = document.querySelector('.loading')
const imgChar = document.querySelector('.img_cha')

radio.forEach(radio => {
  radio.addEventListener('click', () => {
    if(radio.value === 'light') {
      loading.style.background = 'skyblue'
      imgChar.style.background = 'url("img/img_loading.svg") no-repeat'
    } else if(radio.value === 'dark') {
      loading.style.background = '#222'
      imgChar.style.background = 'url("img/img_loading_b.svg") no-repeat'
    }
  })
})
```

Light

Dark



실습

animation

```
<ul>
```

```
<li>1970</li>
```

```
<li>1975</li>
```

```
<li>1980</li>
```

```
<li>1985</li>
```

```
<li>1990</li>
```

```
<li>1995</li>
```

```
<li>2000</li>
```

```
<li>2005</li>
```

```
<li>2010</li>
```

```
<li>2015</li>
```

```
<li>2020</li>
```

```
<li>2025</li>
```

```
</ul>
```



```
* {  
  padding:0;  
  margin:0;  
  box-sizing: border-box;  
}  
  
body {  
  background:oldlace;  
}  
  
ul {  
  position:absolute;  
  top:100px;  
  left:50%;  
  list-style: none;  
  border-left:5px solid #000;  
}  
  
li {  
  opacity: 0;  
  position:relative;  
  padding:70px 0;  
  font-weight:bold;  
  
  transition: all 1.5s ease-out;  
}
```

```
li:nth-child(odd) {  
  left:50px;  
}  
  
li:nth-child(even) {  
  left:-100px;  
}  
  
li.show {  
  animation:show .5s ease-in-out forwards;  
}  
  
@keyframes show {  
  0% {transform: translateY(50px);opacity: 0;}  
  100% {transform: translateY(0);opacity: 1;}  
}
```

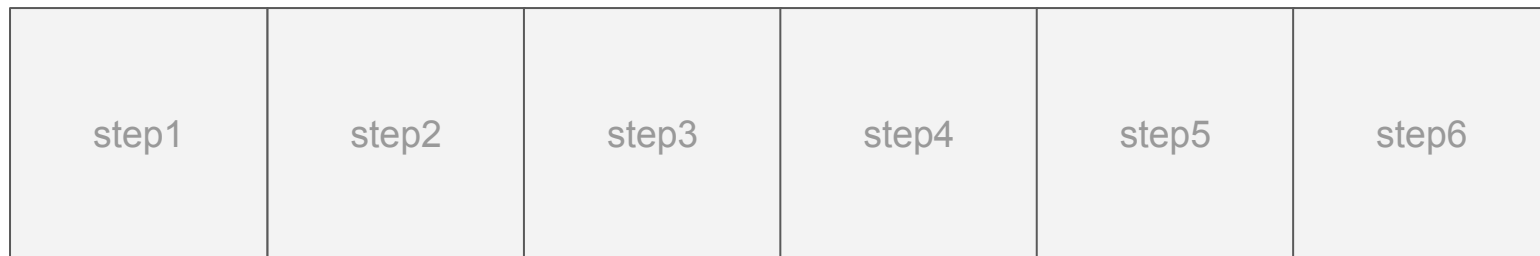
```
document.addEventListener("DOMContentLoaded", function() {  
  const times = document.querySelectorAll("li");  
  const observer = new IntersectionObserver((sections) => {  
    sections.forEach(section => {  
      if (section.isIntersecting) {  
        section.target.classList.add("show");  
      }  
      else {  
        section.target.classList.remove("show");  
      }  
    });  
  }, {  
    threshold: 0.1  });  
  
  times.forEach(time => {  
    observer.observe(time);  
  });  
});
```

step animation

animation

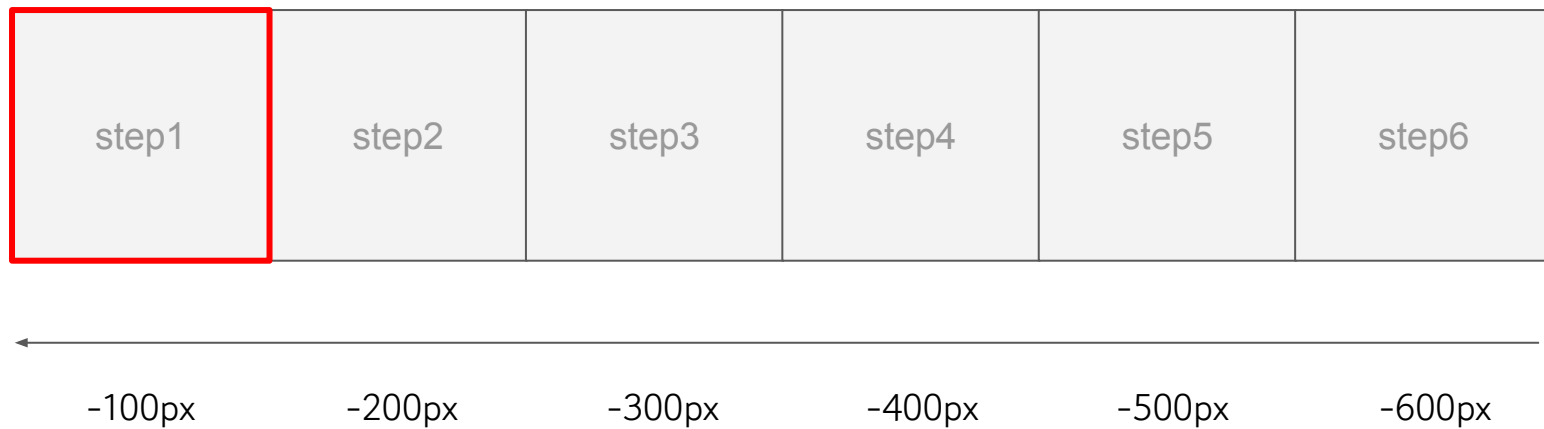
step

100px



600px

1. 컷 동일한 크기를 유지
2. 컷 사이에 여백 없이 밀착



step animation



```
<div class="box"></div>
```

```
body {  
  overflow: hidden;  
}  
  
.box {  
  position: absolute;  
  top: 100px;  
  left: 100px;  
  width: 415px;  
  height: 370px;  
  background: url('img/run.png') no-repeat;  
  background-size: cover;  
  animation: gif 1s infinite steps(10), run 10s linear infinite;  
}  
  
@keyframes gif {  
  100% { background-position: -4150px 0; }  
}  
  
@keyframes run {  
  0% { left: 0; }  
  100% { left: 100%; }  
}
```

step animation



```
<div class="box"></div>  
<input type="range" class="fast" min="2" max="20" step="2">빠르게
```

```
const fast = document.querySelector('.fast')  
const box = document.querySelector('.box')  
  
box.style.animation = `gif 1s infinite steps(10), run ${fast.value}s  
linear infinite`;  
  
fast.addEventListener('input', function() {  
  box.style.animation = `gif 1s infinite steps(10), run  
${fast.value}s linear infinite`;  
  console.log(fast.value);  
});
```

실습

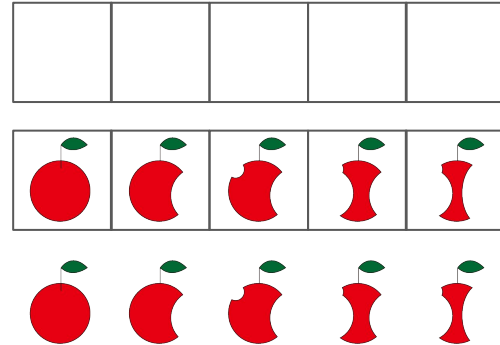
step animation

본인만의 스텝 애니메이션을 만들어 보자

아주 간단한 소스를 제작한 후 step animation을 완성한다.

소스 제작법

1. 컷을 위한 레이아웃 박스를 생성한다. (모든 컷이 같은 크기여야 한다)
2. 컷 안에 변화되는 애니메이션을 그린다.
3. 레이아웃 박스 선을 투명도를 주어 눈에 보이지 않게 처리한다.
4. png 파일로 저장한다.



* 만약 배경색을 넣고 싶다면 넣어도 무방하다. > 이경우 jpg로 출력해도 괜찮음

과제

애니메이션 + js 자유롭게

기한 : 10월 9일 밤 11:59까지

제출 : LMS에 제출

파일명 : 학번_이름

이벤트 타입

mouse / keyboard

class5

이벤트 타입 : 마우스 이벤트

마우스 이벤트

`click` : 마우스 클릭 했을때

`mouseenter` : 마우스 커서가 요소 위에 있을때

`mouseleave` : 마우스 커서가 요소 밖으로 나갈때

`mouseup` : 마우스 눌렀을때

`mousedown` : 마우스 땀을때

`dblclick` : 더블클릭

`mousemove` : 마우스 이동

class5

이벤트 타입 : 마우스 이벤트

```
<div class="box"></div>  
<h1></h1>
```

```
.box {  
  width:100px;  
  height:100px;  
  background:aqua;  
}
```

class5

이벤트 타입 : 마우스 이벤트

```
const box = document.querySelector('.box')
const h1 = document.querySelector('h1')

box.addEventListener('click', () => {
  h1.innerHTML = '클릭'
})

box.addEventListener('mouseover', () => {
  h1.innerHTML = '들어옴'
})

box.addEventListener('mouseout', () => {
  h1.innerHTML = '나감'
})

box.addEventListener('mouseup', () => {
  h1.innerHTML = '땀다'
})

box.addEventListener('mousedown', () => {
  h1.innerHTML = '누르는중'
})

box.addEventListener('dblclick', () => {
  h1.innerHTML = '더블클릭'
})
```

실습

class5

이벤트 타입 : 키보드 이벤트

키보드 이벤트

keydown : 키 눌렀을때

keyup : 키 떨어졌을때

event.대상자.key : key값 리턴

class5

이벤트 타입 : 키보드 이벤트

```
<h1>'<strong></strong>' 를 눌렀어요</h1>
```

```
<script>  
  let keyValue = document.querySelector('strong')  
  document.addEventListener('keydown', (key) => {  
    keyValue.innerText = key.key;  
  });  
</script>
```

실습