

CSS ANIMATION STUDY 16

- openPopup(target)
- closePopup(target)
- 완성

openPopup(target)

- 실제로 popup이 펼쳐진다.
- openPopup => remove close class => add open class => done

```
popup.classList.remove("close");
requestAnimationFrame(() => {
  requestAnimationFrame(() => {
    popupItem.style.width = `${width}px`;
    popup.style.left = 0;
    popup.style.top = 0;
    popup.style.width = `100%`;
    popup.style.height = `100%`;
    popup.classList.add("open");
  });
});
```

openPopup(target)

```
requestAnimationFrame(() => {  
    requestAnimationFrame(() => {  
    });  
});
```

- 1 frame을 건너 뛴 수 있는 효과가 있다.
- requestAnimationFrame가 화면에 그려지기 전에 호출하기 때문에 2번 호출 해야한다.

closePopup(target)

- popup을 클릭하면 닫힌다.

```
popup.addEventListener("click", e => {  
  closePopup(popupTarget);  
});
```

closePopup(target)

- closePopup => remove open class => done => add close class

```
function closePopup(target) {  
  const rect = target.getBoundingClientRect();  
  const left = rect.left;  
  const top = rect.top;  
  const width = rect.width;  
  const height = rect.height;  
  popup.style.left = `${left}px`;  
  popup.style.top = `${top}px`;  
  popup.style.width = `${width}px`;  
  popup.style.height = `${height}px`;  
  popupItem.style.width = popup.style.width;  
  popup.classList.remove("open");  
}
```

closePopup(target)

```
popup.addEventListener("transitionend", e => {  
  if (e.target !== popup) {  
    return;  
  }  
  if (!popup.classList.contains("open")) {  
    popup.classList.add("close");  
  }  
})
```

- open 클래스를 제거하면서 transition이 발생이 끝나면 close를 추가한다.

완성

- <https://codepen.io/daybrush/pen/XBxpVb?editors=0110>

스터디 끝

- 스터디 자료
- [https://github.com/younkue/css/tree/master/animation study](https://github.com/younkue/css/tree/master/animation%20study)