

예제만들기 000, (1만 시간의 법칙)

- 로딩화면, URL복사/ ClipboardJS

!Tips & Links

- ※ clipboard.js <https://clipboardjs.com/> // <https://github.com/zenorocha/clipboard.js/>
- ※ ClipboardEvent.clipboardData <https://developer.mozilla.org/ko/docs/Web/API/ClipboardEvent/clipboardData>
- ※ Document.execCommand() <https://developer.mozilla.org/ko/docs/Web/API/Document/execCommand>
- ※ placeholder 또는 임시이름 / foo, bar(푸바), tmp, dummy

!예제 내용

- ※ 섹션 정렬 flex(column)임 / position relative 50% -50%말고 flex로 align center, justify center
- ※ result 내보기 전에 로딩화면 있음 ※ URL 복사, 사용할 수 있게 정리 ※ function 하나 당 기능 하나만

※ 클립보드 복사:

①	라이브러리 / clipboard.js	이거 사용, 링크참조
②	execCommand("copy")	(Deprecated) 사용종료될 수 있음
③	ClipboardEvent.clipboardData	(Experimental) 사용에 주의

Clipboard.js

특정요소를 클릭했을 때
클립보드에 복사하는
스크립트

(b는 소문자)

data-clipboard-text="~"	el의 data(data-clipboard-text)값을 복사
data-clipboard-target="#~"	타겟의 value를 복사
data-clipboard-action="cut"	잘라내기 + 클립보드에 복사

<pre>{ target: ~ } next요소 선택</pre>	<pre>new Clipboard('.nextSiblingBtn', { target: function(trigger) { return trigger.nextElementSibling; } })</pre>
<pre>{ text: ~ } 복사할 text 선택</pre>	<pre><button class="ariaLabelBtn" aria-label="~"> new Clipboard('.ariaLabelBtn', { text: function(trigger) { return trigger.getAttribute('aria-label') } })</pre>

※ success, error

타겟.on('이벤트', 작동)
동적 이벤트 바인딩

```
let 변수 선언=new ClipboardJS(".class") // 변수에 저장 후  
변수.on('success', function(){ ~~ }) // 클릭 후 복사에 성공  
변수.on('error', function(){ ~~ }) // 클릭 후 복사에 실패
```

※ execCommand()로

코드 짜는 법
(Deprecated이라 사용x)

```
let tmp_input=document.createElement('input'); // input el 생성  
document.body.appendChild(tmp_input); // 바디에 append  
tmp_input.value=url; // value에 url 넣기  
tmp_input.select(); // 텍스트 필드 선택  
document.execCommand("copy"); // 클립보드에 복사  
document.body.removeChild(tmp_input); // el 삭제
```

※ 클린코드 습관잡기:

함수는 동사로 / 이름 축약X =>가독성
코드 짠 후 청소
정보 추가해도 사용가능
필요시 configuration_object 사용해 정리
boolean 보내지 않기(함수1=>액션1, if사용x)

```
▽ 함수("box", 2, "red", "up");  
configuration_object({  
  type: "box",  
  qty: 2,  
  color: "red",  
  direction: "up",  
});
```