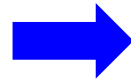


Quiz #1 예제1 소스에서 이벤트리스너와 이벤트핸들러는 각각  
어느 부분일까요?

Quiz #2 예제1에서 transition°이 없다면 어떤 결과가 나올까요?

Quiz #3 예제1의 아래 한 줄을 템플릿 리터럴 말고 일반 큰따옴표 문자열로 표현하세요.

```
i.style.transform = `rotate(${d}deg)`;
```



Quiz #4 예제1에서 자바스크립트 변수 d를 rotate함수 밖 말고  
안에 선언했다면 어떤 결과가 나올까요?

Quiz #5   예제1에서 자바스크립트 변수 d의 값을 출력하도록  
소스를 수정하세요.

- (1) 웹화면에 출력
- (2) 콘솔 출력

Quiz #6 예제1에서 아래처럼 img태그의 onclick부분을 삭제해도 동일한 결과가 나오도록 자바스크립트 1줄을 추가하세요.

```
...  
  
<script>  
  const i = document.getElementById("rotatImage");  
  let d = 0;  
  function rotate() {  
    d += 360; /* 회전각도*/  
    i.style.transform = `rotate(${d}deg)`;  
  }  
</script>  
...
```

Quiz #7   예제1에서  $d += 360;$  대신에  $d -= 360;$  을 하면 어떤  
결과가 나올까요?

Quiz #8 예제2에서 아래 2라인을 삭제하고 자바스크립트 밖에서 함수를 호출하도록 소스를 변경하세요.

```
...  
  
<script>  
  let i = document.getElementById("rotateImage");  
  i.onmouseover=small;  
  i.onmouseout=back;  
  
  function small() {    i.style.transform = "scale(0.2, 0.2)";  }  
  function back() {    i.style.transform = "scale(1, 1)";  }  
</script>  
...
```



Quiz #9 예제2를 복사해서 예제2변형.html을 만든 후 코드를 추가하세요.

➔ 그림에 마우스를 손가락으로 누른 채 있으면 그림이 두 배로 커지고, 마우스에서 손을 떼면 크기가 원래대로 돌아가도록 수정하세요. (기존에 있던 이벤트도 삭제하지 말고 그대로 유지)

Quiz #10 예제3의 style에서 linear가 있고 없과의 차이는?

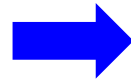
```
#followMouse { transition: transform 1s linear; }
```

Quiz #11 예제3에서 아래 1라인을 삭제하고 자바스크립트 밖에서 함수를 호출하도록 소스를 변경하세요.

```
...  
<div id="sky">  </div>  
<script>  
  const i = document.getElementById("followMouse");  
  const d = document.getElementById("sky");  
d.onmousemove = updateImage;  
  function updateImage(event) {  
    const X = event.clientX;  
    const Y = event.clientY;  
    i.style.transform = `translate(${X}px, ${Y}px)`;  
  }  
</script>  
...
```

Quiz #12 예제3의 아래 한 줄을 템플릿 리터럴 말고 일반 큰따옴표 문자열로 표현하세요.

```
i.style.transform = `translate(${X}px, ${Y}px)`;
```



Quiz #13 예제3에서 마우스 위치가 그림의 시작지점(좌측상단)을 맞춥니다. 그림의 중간부분에 마우스 위치를 맞추기 위해 아래 코드의 일부를 수정하세요.

```
const X = event.clientX;  
const Y = event.clientY;
```

```
i.style.transform = `translate(${X}px, ${Y}px)`;
```

Quiz #14 예제3을 복사해서 예제3변형.html을 만든 후 코드를  
추가하세요. ➡

그림을 더블클릭할 때마다 그림이 photo3 및 photo6으로 바뀜  
(기존에 있던 이벤트도 그대로 유지)

[힌트1] `console.log(i.src);` 해보세요.

[힌트2] `let str="i am a student";` 일 때  
`str.includes('stu')`은 true입니다.

[힌트3] `str = str.replace(student, teacher);` 하면  
`str`은 "i am a teacher"가 됩니다.

Quiz #15 앞 문제에서 만든 예제3변형.html을 조금 더 수정하세요. ➡

- (1) 마우스를 움직이면 따라오는 애니메이션 개념 말고, 마우스 움직일 때마다 동시에 움직이도록 코드를 수정하세요.
- (2) 클릭하면 90도씩 회전하도록 코드를 추가하세요.
- (3) 더블클릭하면 클릭이벤트도 동시 발생합니다. (즉, 그림 바뀌면서 회전도 함) 더블클릭했을 때는 더블클릭이벤트만 발생하고 클릭이벤트는 발생하지 않도록 코드를 추가하세요. (즉, 그림만 바뀌고 회전은 하지 않도록)

## Quiz #16 예제3에서 매개 변수 event를 다른 이름으로 바꾸면 어떻게 될까요?

...

```
<div id="sky">  </div>
```

```
<script>
```

```
  const i = document.getElementById("followMouse");
```

```
  const d = document.getElementById("sky");
```

```
  d.onmousemove = updateImage;
```

```
  function updateImage(e) {
```

```
    const X = e.clientX;
```

```
    const Y = e.clientY;
```

```
    i.style.transform = `translate(${X}px, ${Y}px)`;
```

```
  }
```

```
</script>
```

...



Quiz #17    아래 소스에서 함수 2개는 그대로 두고 그림을 클릭했을 때 두 함수가 모두 호출되도록 소스를 수정하세요. ➡ 객체 변수.addEventListener 호출

```
<!DOCTYPE html><html><head>
  <meta charset="UTF-8">  <title>복수 이벤트</title>
  <style>
    #sky { width:800px; height: 400px; background:cyan; }
    #aircraft { transition: transform 1s linear; position:absolute; }
  </style>
</head><body>
  <div id="sky">    </div>
  <script>
    const i = document.getElementById("aircraft");
    i.style.left = "300px";    i.style.top = "150px";    let kak = 0;
    i.onclick = showImagePopup;
    i.onclick = rotateImage;

    function showImagePopup(){ alert('비행기를 클릭하셨습니다. '); }
    function rotateImage() {
      kak -= 90;    i.style.transform = `rotate(${kak}deg)`;
    }
  </script>
</body></html>
```

## Quiz #18 예제4 소스를 수정하세요

- (1) 버블링 말고 캡처링 전파 방식으로 바꿔보세요. (세 번째 인자 true)
- (2) 부모 자식 모두 버블링과 캡처링 둘 다 구현 후 실행시켜보세요.  
팝업창 순서는?
- (3) function 이라는 키워드를 사용하지 않고 함수를 표현한다면?

Quiz #19 예제4에서 어느 순간에 이벤트가 더 이상 전파되지 않고 멈추도록 하려면? → 이벤트 변수.`stopPropagation()`; 호출

Quiz #20 예제3에서는 event객체에 있는 clientY를 사용하였고, 예제5에서는 event객체에 있는 deltaY를 사용하였습니다. 두 Y값의 차이는 무엇일까요?

Quiz #21   예제5에서 addEventListener 안에서 사용자정의함수를 호출하도록 소스를 구현했는데 오류가 있습니다. 오류의 원인은 무엇일까요?

```
<script>
  const i = document.getElementById("largerImage");

  document.addEventListener('wheel', changelImage(e));

  function changelImage(e) {
    if (e.deltaY < 0) {
      i.width *= 1.1;
      i.height *= 1.1;
    } else {
      i.width /= 1.1;
      i.height /= 1.1;
    }
  }
</script>
```

Quiz #22 photo6.png가 한 바퀴 도는 q22.html을 완성하세요. (한 바퀴 도는 시간은 1초. addEventListener 안에서 화살표 함수 사용)

(1) 페이지가 load할 때 한 바퀴

(2) 그림을 클릭할 때에도 한 바퀴

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8"> <title>q22</title>
  <style>
    body { text-align: center; }
    /* 여기 1줄 */
  </style>
</head>
<body>
  
  <script>
    /* 여기 코딩 */
  </script>
</body>
</html>
```

Quiz #23 예제6에서 focus 이벤트 대신 click 이벤트로 하면 어떤 차이가 있을까요?

Quiz #24 예제7에서 아래와 같이 CSS를 별도 파일에도 넣고 style 태그에도 넣었습니다. 굳이 두 가지 방법을 다 사용하는 이유는 무엇일까요?



Quiz #25 예제7에서 `<label for="messageBox">입력하세요:</label>` 처럼 라벨 태그가 있는 경우와, label 시작태그 끝태그를 제거하고 그냥 `입력하세요:` 라는 글자만 있는 경우의 차이는?

Quiz #26 예제7에서 e.preventDefault(); 가 있는 경우와 없는 경우의 차이는?

Quiz #27 0장 확인문제 29번에서 div 에 텍스트 표현하기 위해서 innerHTML  
을 사용했습니다.

그런데 예제7에서는 innerText를 사용했습니다.  
innerHTML과 innerText의 차이는 무엇일까요?

## Quiz #28 예제7에 코드를 추가하세요.

- (1) 아무 것도 입력하지 않고 submit하면 result부분에 **You typed: nothing** 이라고 출력되도록 코드를 추가하세요.
- (2) 공백만 입력하면 **You typed: spaces only** 라고 출력되도록 코드를 추가하세요.

Quiz #29 예제7의 script 부분을 아래와 같이 수정했습니다. 빈 부분을 채우세요.

```
<script>  
  const f = document.getElementById("myForm");  
  const dr = document.getElementById("result");  
  
  f.addEventListener("submit", submitMessage);  
  f.addEventListener("reset", resetMessage);
```

```
/* 여기 코딩 */
```

```
</script>
```

```
</body>  
</html>
```

Quiz #30 예제8에서 select의 첫 option인 ==선택== 은 첫 화면에만 나오고,  
다음부터는 select 박스에 나오지 않도록 소스를 수정하세요.

Quiz #31 예제8에서 과일을 선택하면 해당 과일 사진이 나오도록 코드를 추가하세요.

(과일 사진은 원본 비율을 맞추되 화면에 나오는 크기는 가로 세로 200px이 넘지 않도록 설정하세요.)

Quiz #32 예제9에서 위·아래 화살표(ArrowUp, ArrowDown)를 눌렀을 때 y값이 -50 ~ 50 범위에서 왔다갔다 할 수 있도록 코드를 추가하세요.