

이론문제

1. 자바스크립트 이벤트에 대한 설명으로 틀린 것은?

- ① 마우스를 클릭하면 이벤트가 발생한다.
- ② 이벤트를 발생시킨 DOM 객체를 이벤트 타겟이라고 부른다.
- ③ 이벤트는 반드시 이벤트 타겟 객체에서만 처리된다.
- ④ 발생한 이벤트에 관한 여러 정보를 가진 객체를 이벤트 객체라고 부른다.

2. 다음과 같은 자바스크립트 코드가 있을 때, 버튼을 클릭하면 실행 결과는 무엇인가?

```
var button = document.getElementById("button");
button.addEventListener("click", function () { alert("ONE") });
button.addEventListener("click", function () { alert("TWO") });
button.addEventListener("click", function () { alert("THREE") });
```

- ① ONE 문자열을 가진 경고창이 출력된다.
- ② THREE 문자열을 가진 경고창이 출력된다.
- ③ ONE, TWO, THREE 문자열을 가진 3개의 경고창이 순서대로 출력된다.
- ④ THREE, TWO, ONE 문자열을 가진 3개의 경고창이 순서대로 출력된다.

3. 이벤트 리스너 중 같지 않은 것은?

- ① div1.addEventListener("onclick", function() { this.innerHTML = "hello"; });
- ② div1.onclick = function() { this.innerHTML = "hello"; }
- ③ div1.onclick = function(e) { this.innerHTML = "hello"; }
- ④ div1.onclick = f; function f() { this.innerHTML = "hello"; }

4. 다음 함수와 태그가 있을 때 물음에 답하여라.

```
function f() { // obj 객체의 배경색을 orchid로 변경하는 함수
  obj.style.backgroundColor = "orchid";
}

<div id="div1">안녕하세요. 환영합니다.</div>
```

(1) id가 "div1"인 DOM 객체를 알아내는 코드를 작성하라.

var obj = document.getElementById("div1");

(2) <div> 태그를 클릭하면 f() 함수가 실행하도록 addEventListener()를 이용하여 이벤트 리스너를 등록하라.

obj. addEventListener("click", f) ;

(3) <div> 태그를 클릭하면 f() 함수가 실행하도록 onclick 프로퍼티에 이용하여 이벤트 리스너를 등록하라.

obj. onclick = f ;

(4) f()를 익명 함수로 작성하여 onclick 리스너로 등록하라. onclick 프로퍼티를 이용하라.

obj. onclick = function () { obj.style.backgroundColor = "orchid" } ;

5. 이벤트 캡처 단계는 어디서부터 어디까지 이벤트가 흘러가는 과정인가?

- ① window 객체에서부터 타겟 객체까지
- ② 타겟 객체에서부터 window 객체까지
- ③ document 객체에서부터 타겟 객체까지
- ④ 타겟 객체에서부터 document 객체까지

6. 이벤트 버블에 대해 잘못 말한 것은?

- ① 익스플로러에서 처음으로 제안한 이벤트 흐름 모델
- ② 이벤트가 타겟 객체에서 DOM 트리를 따라 window 객체로 흘러가는 모델
- ③ 이벤트 버블이나 이벤트 캡처 단계 중 한 단계에서만 이벤트 리스너 작성 가능
- ④ 이벤트에 따라 이벤트 버블 단계는 없을 수도 있다.

7. 다음 자바스크립트 코드가 실행되면 웹 페이지에 작성된 어떤 onclick 리스너도 실행되지 못한다. 이유를 설명하라.

```
window.addEventListener("click", function(e) { e.stopPropagation(); }, true);
```

- HTML 페이지 내의 onclick 리스너가 실행되기 전에 제일 먼저 위의 window 객체의 onclick 리스너가 실행된다. 이때, e.stopPropagation() 함수를 호출하기 때문에 더 이상 click 이벤트가 전파되지 않게 된다.
8. 다음 중 버블 단계에서 실행되도록 작성된 onclick 리스너를 모두 골라라.
- ① window.onclick = function(e) { alert(e.type); }
 - ② window.addEventListener("click", function(e) { alert(e.type) }, true);
 - ③ window.addEventListener("click", function(e) { alert(e.type) }, false);
 - ④ window.onclick = f; function f(e) { alert(e.type); }

9. 다음은 HTML 페이지의 일부분이다.

```
<body>
<h3>이벤트 흐름</h3>
<hr>
<p id="p">이벤트가 어디서 시작하여 어디로 흘러가는지
<strong id="strong">이벤트 흐름</strong>을
아는 것은 매우 중요하다.
</p>
</body>
```

- (1) 앞의 코드에서 ‘이벤트 흐름’ 부분을 클릭하였을 때 경고창에 “strong+이벤트 흐름”을 출력하려고 한다. 다음 코드를 완성하라.

```
var strong = document.getElementById("strong");
strong.onclick = function (e) {
                  alert (this.tagName + "+" + e.target.innerHTML);
                  }
```

- (2) 앞의 코드에서 ‘이벤트 흐름’ 부분을 클릭하였을 때 경고창에 “p+이벤트 흐름”을 출력하려고 한다. 다음 코드를 완성하라.

```
var p = document.getElementById("p");
p.addEventListener("click", function (e) {
                  alert (this.tagName + "+" + e.target.innerHTML);
                  }, true);
```

- (3) 문제 (1)과 (2)의 코드를 모두 HTML 페이지에 작성하였을 때, “이벤트 흐름”을 클릭하면 출력 결과는 무엇인가?

- ① 문제 (1)의 경고창이 출력된 후 문제 (2)의 경고창이 출력된다.
- ✓ ② 문제 (2)의 경고창이 출력된 후 문제 (1)의 경고창이 출력된다.
- ③ 문제 (1)의 경고창만 출력된다.
- ④ 문제 (2)의 경고창만 출력된다.

10. 다음 코드를 addEventListener() 메소드를 이용하여 다시 작성하라.

```
p.onclick = function (e) { alert("hello"); }
```

```
p.addEventListener("click", function (e) {
    alert ("hello"); }, false);
```