JavaScript Arrow Function

- description : JavaScript Arrow Function
- author : 오션
- email : shlim@repia.com
- lastupdate : 2021-04-26

The source of this article

JavaScript Arror Function
화살표 함수(Arrow functions)은 ES6에 도입되었습니다.

화살표 함수를 사용하면 더 짧은 함수 구문을 작성할 수 있습니다.

Before

```
let hello;

hello = function () {
  return "Hello World!";
}

document.getElementById("demo").innerHTML = hello();
```

With Arrow Function

```
let hello;

hello = () => {
  return "Hello World!";
}

document.getElementById("demo").innerHTML = hello();
```

화살표 함수(Arrow Function)를 사용하면 코드는 더 짧아집니다!
함수에 스테이트먼트가 하나만 있고 그 스테이트먼트가 값을 반환하는 경우, 소괄호와 `return` 키워드를 제거할 수 있습니다.

Arrow Functions Return Value by Default

```
let hello;

hello = () => "Hello World!";

document.getElementById("demo").innerHTML = hello();
```

Note: 이것은 함수가 스테이트먼트 하나 만을 가졌을 때 적용됩니다.

Arrow Function With Parameters:

```
let hello;

hello = (val) => "Hello " + val;

document.getElementById("demo").innerHTML = hello("Universe!");  // Hello
Universe!
```

실제로, 오직 하나의 매개변수(parameter)만을 가지고 있는 경우, 마찬가지로 괄호를 사용하지 않을 수 있습니다.

Arrow Function Without Parentheses

```
let hello;

hello = val => "Hello " + val;

document.getElementById("demo").innerHTML = hello("Universe!");  // Hello
Universe!
```

What About ''this''?

`this`를 다루는 것은 처리는 일반 함수와 비교하여 화살표 함수에서도 다릅니다.

요컨대, 화살표 함수를 사용하면 `this`에 대한 바인딩이 없습니다.

일반 함수에서, `this` 키워드는 함수를 호출한 오브젝트를 나타내며, 창, 문서, 버튼 등이 될 수 있습니다.

화살표 함수에서 `this` 키워드는 항상 화살표 함수를 정의한 오브젝트를 나타냅니다.

차이점을 이해하기 위해, 두 가지 예를 살펴 보겠습니다.

두 예제는 메서드를 두 번 호출합니다. 첫 전째는 페이지가 로딩될 때, 사용자가 버튼을 클릭 할 때 다시 한 번 메서드를 호출합니다.

첫 번째 예제는 일반 함수(regular function)를 사용하고, 두 번째 예제는 화살표 함수(arrow function)를 사용합니다.

window 오브젝트가 함수의 “ 소유자 ” 이기 때문에, 첫 번째 예제는 두 개의 다른 오브젝트(window와 버튼)를 반환하고,
두 번째 예제는 window 오브젝트를 두 번 반환한다는 결과를 보여줍니다.

일반 함수를 사용하면, `this`는 함수를 호출하는 오브젝트를 나타냅니다.

화살표 함수를 사용하면, `this`는 해당 함수의 소유자를 나타냅니다.

함수로 작업할 때 이러한 차이점을 기억하십시오.
때로는 일반 함수의 동작이 원하는 것이지만, 그렇지 않은 경우 화살표 함수를 사용합니다.

오션, Javascript, Arrow, Function

From:
http://rwiki.repia.com/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다.** - 2023.12

Permanent link:
http://rwiki.repia.com/doku.php?id=wiki:javascript:javascript_note:js_arrow_function&rev=1619415968

Last update: **2022/03/10 19:52**