

콜백에서 `this`에 대한 타입 제공하기

이펙티브 타입스크립트

@Leemainsw

Javascript의 **this** 키워드

this의 값은 호출되는 방식에 따라 결정됨

this 키워드 사용법 - (1)

객체의 현재 인스턴스를 참조하는 클래스

```
class C {  
  vals = [1, 2, 3];  
  logSquares() {  
    for (const val of this.vals) {  
      console.log(val * val);  
    }  
  }  
}
```

```
const c = new C();  
c.logSquares();
```

```
const c = new C();  
const method = c.logSquares;  
method();
```

Uncaught TypeError: undefined의 'vals' 속성을 읽을 수 없습니다.

1

4

9

this 키워드 사용법

c.logSqaures() 함수의 두 가지 작업

- C.prototype.logSquares() 호출
- this의 값을 c로 바인딩

```
class C {  
  vals = [1,2,3];  
  logSquares() {  
    for (const val of this.vals) {  
      console.log(val * val);  
    }  
  }  
}
```

```
const c = new C();  
c.logSquares();
```

```
console.log(C.prototype.);
```

```
// const method = c.logS  
// method();
```

logSquares
vals

```
class C {  
  vals = [1, 2, 3];  
  logSquares() {  
    for (const val of this.vals) {  
      console.log(val * val);  
    }  
  }  
}
```

```
const c = new C();  
c.logSquares();
```

this 키워드 사용법

Call 사용해서 오류 수정하기

- call을 사용하면 명시적으로 this를 바인딩하여 수정할 수 있습니다.

```
const c = new C();  
const method = c.logSquares;  
method.call(c); // 제곱을 출력합니다.
```

this 키워드 사용법 - (2)

콜백 함수 내의 this 키워드

```
class ResetButton {  
  constructor() {  
    this.onClick = this.onClick.bind(this);  
  }  
  render() {  
    return makeButton({text: 'Reset', onClick: this.onClick});  
  }  
  onClick() {  
    alert(`Reset ${this}`);  
  }  
}
```


this 키워드 사용법 - (2)

콜백 함수 내의 this 키워드

```
class ResetButton {  
  render() {  
    return makeButton({text: 'Reset', onClick: this.onClick});  
  }  
  onClick = () => {  
    alert(`Reset ${this}`); // "this"가 항상 인스턴스를 참조합니다.  
  }  
}
```

결론

this 바인딩이 동작하는 원리를 이해해야 하고 콜백함수에서 this를 사용할 때에는 타입 정보를 명시해야 한다는 점