

JavaScript (ES6) For Framework

7. 이 변수, 어디까지 쓸 수 있니?

“

scope?

“

범위!

“

유효한 범위!

동일한 이름의 변수

```
function ComputerClass(name, professor, classno){  
  this.name = name;  
  this.professor = professor;  
  this.classno = classno;  
  this.printInfo = function(){  
    console.log(this.name + ' 강의' + this.classno + ' 분반입니다. 교수는 ' + this.professor + ' 입니다.');  };  
}
```

범위의 종류

전체
영역

전역 scope

지역 scope

일부
지역

범위의 종류

전역 **scope**에서 선언된 변수 : 전역 변수

지역 **scope**에서 선언된 변수 : 지역 변수

지역 scope

블록 scope

전역 **scope**에서 선언된 변수 : 전역 변수

지역 **scope**에서 선언된 변수 : 지역 변수

함수 scope

블록 스코프

블록?

블록 스코프

블록?

중괄호로 표시된 범위

블록 스코프

```
{
```

```
// 이 안에서만 유효한 범위
```

```
}
```

블록 스코프

```
{
```

```
// 이 안에서만 유효한 범위
```

```
}
```

블록 스코프

```
var name = "minchul";  
console.log(name); // minchul  
  
{  
  const name = "youngsu";  
  console.log(name); // youngsu  
}  
  
console.log(name); // minchul
```

지역변수
name이
유효한 범위

직접 쳐가며 결과를 확인해보세요.

const는 지금 당장 신경 쓸 필요는 없고,

“블록 스코프로 변수를 선언할 수 있게 하는 방법”

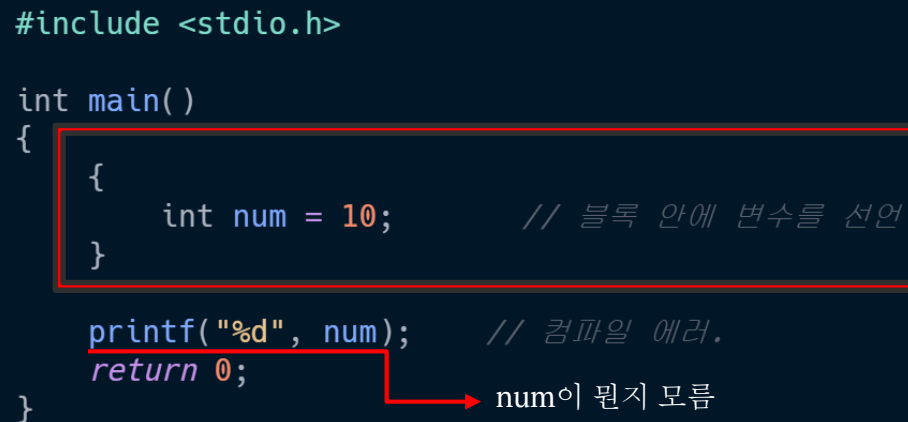
정도로 생각하셔도 좋습니다 ☺

Block scope in C

```
#include <stdio.h>

int main()
{
    {
        int num = 10;    // 블록 안에 변수를 선언
    }

    printf("%d", num);    // 컴파일 에러.
    return 0;
}
```



block

num이 뭔지 모름

Block scope in JavaScript

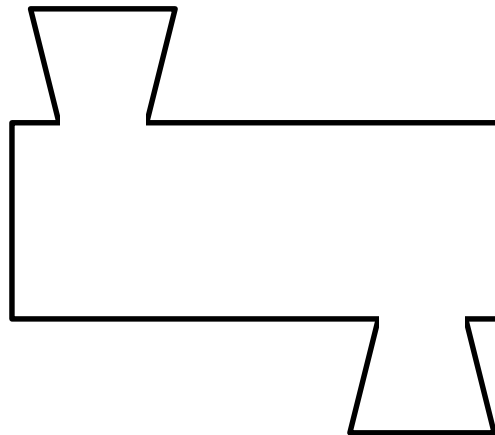
ES6 부터 쓸 수 있게 되었다..!!

Block scope in JavaScript

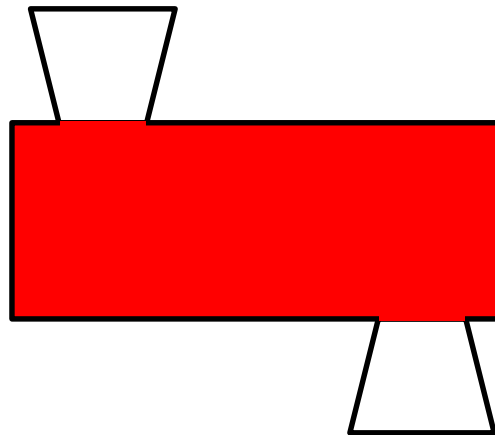
ES6 부터 쓸 수 있게 되었다..!!

(근데 기본적으로는 함수 `scope`를 따른다..)

함수 scope



함수 scope



함수 scope

```
function funcScope () {  
  var num = 10;
```

```
  // 여기서는 함수 scope 안에서 num을 console.log 했으니 잘 출력된다  
  console.log(num);  
}
```

```
// 여기서는 num이 뭔지 몰라서 Uncaught ReferenceError: num is not defined 가 뜬다  
console.log(num);
```


전역 scope

어떤 것으로도 둘러싸여 있지 않을 때
let, const가 쓰이지 않은 일반적인 블록 스코프

전역 scope

어떤 것으로도 둘러싸여 있지 않을 때

let, const가 쓰이지 않은 일반적인 block 스코프



Block scope를
쓸 수 있게 하는
ES6 문법

전역 scope

```
var num = 10;

console.log("전역에서 출력하는 num : ", num);

function numTest(){
    console.log("함수 안에서 출력하는 num : ", num);
}

numTest();
```

직접 쳐 보며 결과를 확인해보세요

전역 scope

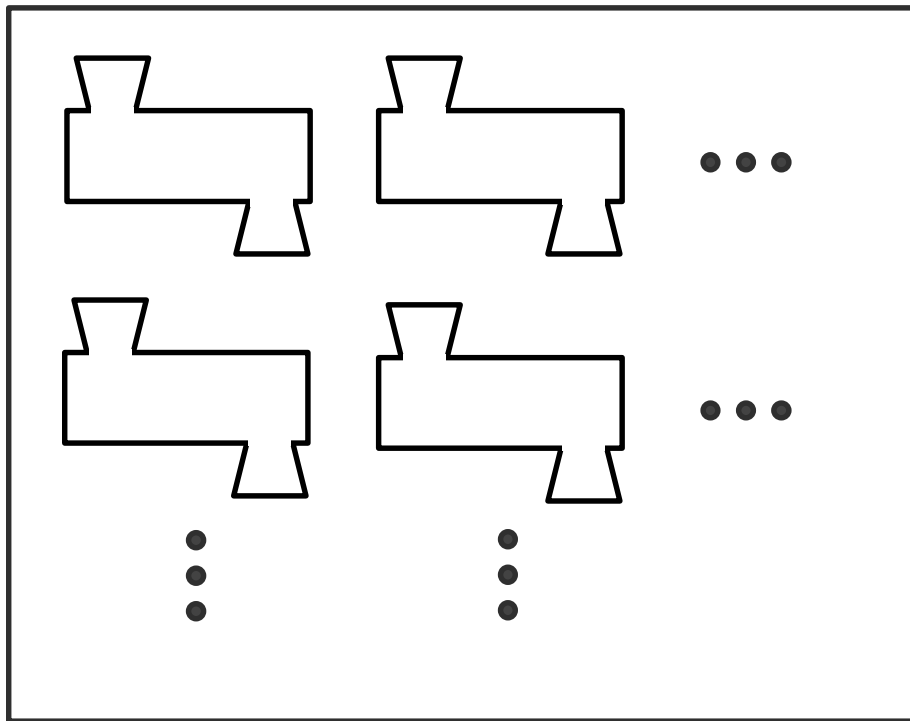
```
{  
    var num = 10; // block 안에 들어있는 전역 변수 num  
}  
  
console.log("전역에서 출력하는 num", num);  
  
function numTest(){  
    console.log("함수 안에서 출력하는 num", num);  
}  
  
numTest();
```

직접 쳐 보며 결과를 확인해보세요

전역 scope의 사용을 자제하라는 이유

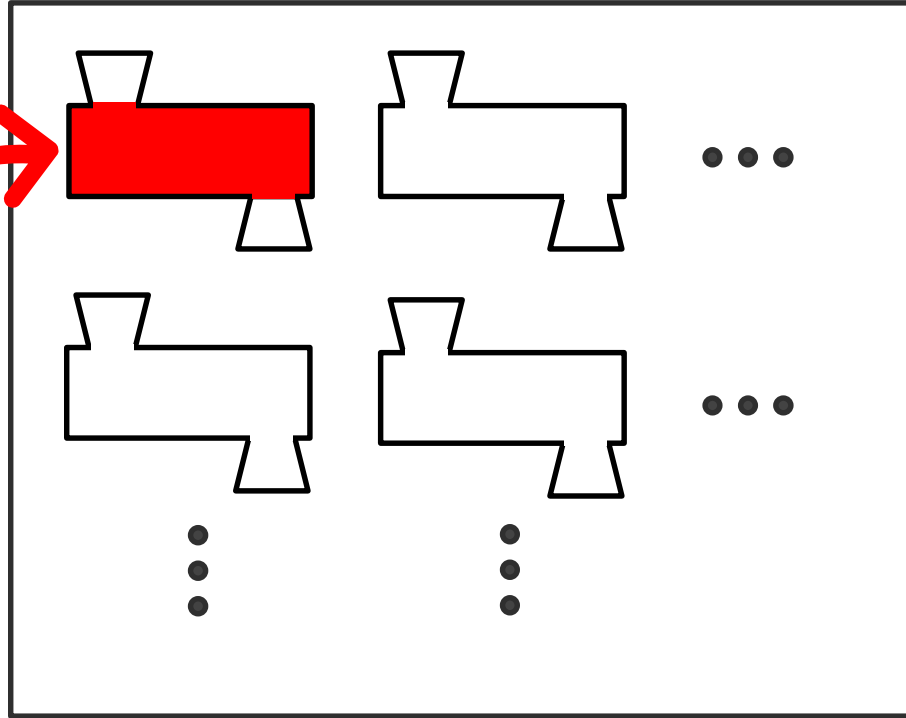
전역변수는 가급적 사용을 자제할 것

전역 scope의 사용을 자제하라는 이유



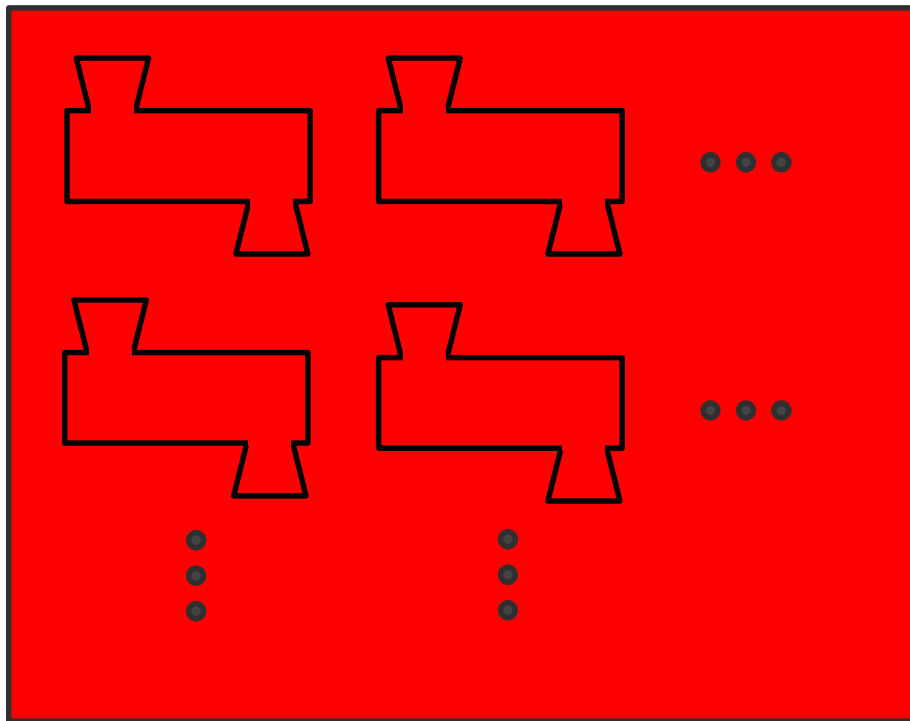
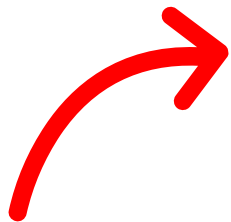
전역 scope의 사용을 자제하라는 이유

지역 변수
영향 범위



전역 scope의 사용을 자제하라는 이유

전역 변수
영향 범위

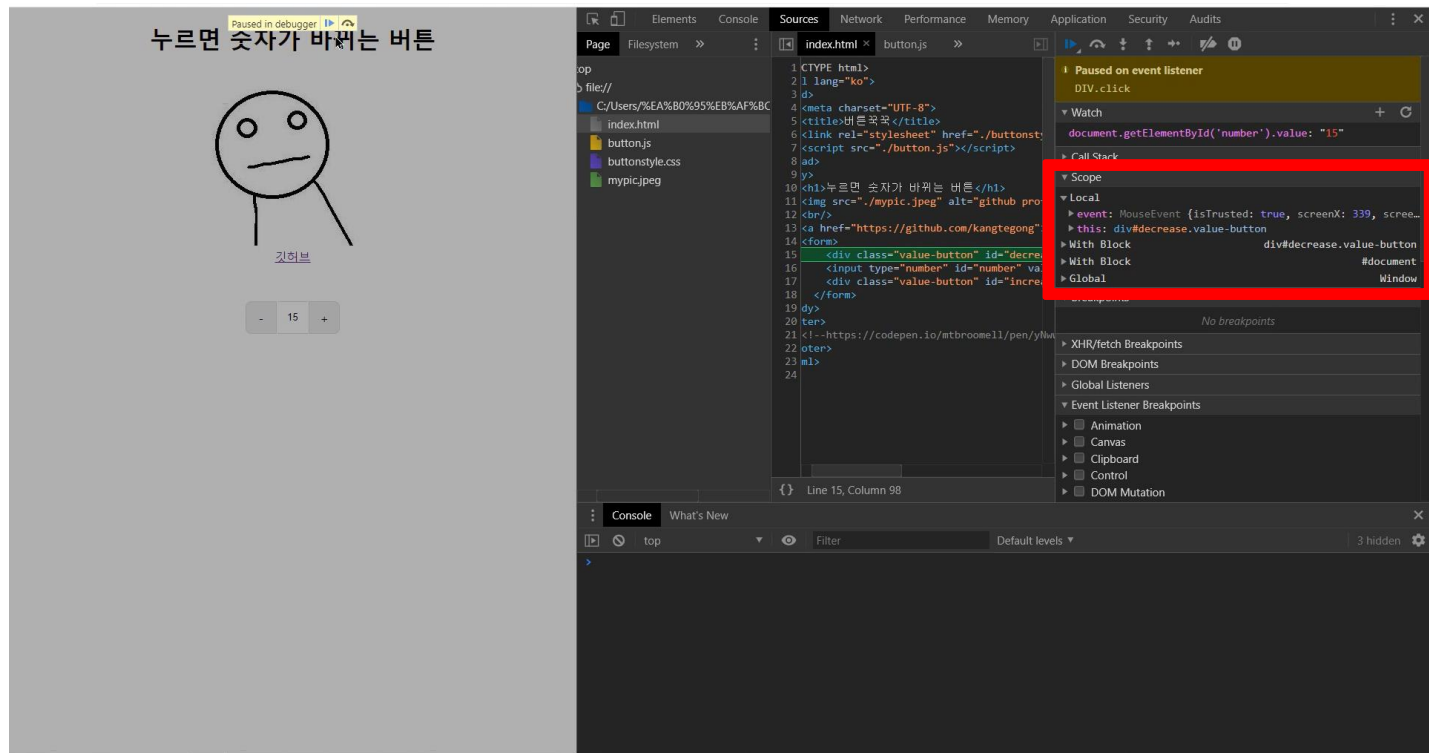


전역 scope의 사용을 자제하라는 이유

전역변수는 가급적 사용을 자제할 것
쓰더라도 값이 변경되지 않게 할 것

덧. Scope 찾아보기

덧. Scope 찾아보기 : chrome browser



다음강의

Section 8

this의 모든 것

Lecturer Github : <https://github.com/kangtegong/>

수업자료링크 : <https://github.com/kangtegong/develop-javascript>

Email : tegongkang@gmail.com