스코프 클로저 정리

*//스코프란 ? 특정 위치에서 접근 할 수 있는 함수, 변수 등의 군락을 일컷는다.*

*//전역 스코프란 ? 현재 있는 자바스크립트가 실행하고 있는 위치(함수,블럭) 바깥에 있는*

*//스코프를 일컫는다. 전역변수와 비슷한 내용이며 변수, 함수 등을 포함하기에 좀 더 넓은 의미 .*

let global = "i'm global";

*//지역 스코프란 ? 블럭스코프, 함수스코프가 있으며 함수와 블럭 내부에서만 접근 가능한*

*//변수, 함수 등의 군락을 일 컫는다. 지역변수와 비슷한 내용이며 , 변수, 함수 등을 포함 하기에 좀 더 넓은 의미*

{

let local = "i'm local";

*//local변수는 현재 이 블락에서만 유효하며 블락 외부에서는 접근 불가능*

}

function localFunc(){

let local = "i'm local";

*//local변수는 현재 이 함수에서만 유효하며 블락 외부에서는 접근 불가능*

}

*//로컬 변수는 지역변수이자, 지역 스코프를 가졌기에 아래 콘솔 로그는 유효하지 않다.*

*// console.log(local);*

*//블락은 어느언어에서나 스코프를 구분 짓기 위해 사용하는 것이다. 함수, 반복문, 제어문 등*

if(10 < 20){

let answer = true;

}else{

let answer = false;

}

*// answer는 블락 내부에서 선언되었기에 지역 스코프를 가진다. 아래 콘솔로그는 유효하지 않다.*

*// console.log(answer);*

*//var는 global 스코프를 가진다. 그렇기에 블락 스코프는 적용되지 않으며, 두번 선언시*

*//제일 나중에 선언된 var 변수가 기존 var 변수의 내용을 덮어쓴다.*

var globalScope = "hello";

{

var globalScope = "hi";

}

console.log(globalScope) *// hi가 출력된다.*

*//함수스코프는 가능하다.*

function change(){

var globalScope = "third";

}

change()

console.log(globalScope)

*//클로저란 함수를 특정스코프에서 정의하여 , 함수를 특정 스코프로 접근 할 수 있게 하는 것*

let globalFunc;

{

*//globalFunc는 변수 password에 접근 불가능*

let password = "1234";

*//globalFunc에 함수를 할당 함으로 password에 접근 하도록 클로저를 형성*

globalFunc = function(*setPasswrod*){

console.log(password);

password = setPasswrod;

console.log(password);

}

}

globalFunc("4242")

globalFunc("1234")

*//스코프 중첩이란 ? 함수 스코프나 블럭 스코프가 여러겹 중첩되어 스코프를 공유하는 상태를 말한다.*

{

const b = 20

{

const a = 20

{

console.log(a);

console.log(b);

}

}

}

*//변수 숨김이란 ? 스코프가 중첩되어있는 경우 최상위 스코프에 있는 변수,함수 를 중간 층에서*

*//또 선언되어 최상위 스코프에 있는 변수,함수를 못쓰는 경우*

{

const a = 30

{

const a = 20

{

*//이 스코프에서는 a가 30이라고 되어있는 변수에 접근 불가능*

console.log(a);

}

}

}

*//동적 스코프 . 함수를 호출 하는 위치에 따라 스코프가 변하는 것*

function lookAround(){

*//현재 위치에서는 a변수가 없다.*

console.log(a);

console.log("look around")

}

{

var a = 20;

lookAround();

}

{

var a = 30;

lookAround();

}