아래의 빈칸과 서로 차이점을 서술하시오.

```
function add(x, y) { // x,y 을 뭐라하는가 _________
  return x + y;
}
add(2,5); // 들어가는 값에 대한 단어 : _____
```

위의 함수를 정의할때 x,y를
매개변수라고 한다 그리고 아래에있는
2,5를 보고는 인수라고 한다

본인이 생각하기에 이상적인 개발자는 어떤 형태인가?
해당 질문에 대해서 진지하게 한번쯤은 고민할 만 합니다. 보통 프로그래머로 취업을 한다면 인터뷰에서 무조건 물어보는 내용이기도 합니다.

이상적인 개발자란 유저들이
좋아할만한 게임을 만드는것이고
그게임을 유저들이 오랫동안
좋아할수있도록 유지보수 할 수 있는
그런 개발자라고 생각함(게임쪽에서)

선언문에서는 함수 이름을 생략할 수 없다. 만약 함수 이름을 생략하면 나오는 에러는 어떤건지 확인해보세요.

함수이름을 안정하면 함수를 어떻게
호출한다는 것인가

{} 는 블록문일까 객체 리터럴일까? 본인의 생각을 쓰고 그 이유에 대해서 서술하시오.

? 둘다 가능한것이 아닌가 if for
같은부분을 본다면 블록문일것이고
함수같은것을 호출할때는 객체
리터럴인것이 아닌가?

하단의 에러는 왜 날까?

```
var add1 = (function() {
  var a = 10;
  return function (x, y){
    return x + y + a;
  };
}());

console.log(add1(1,2)); // 13

var add2 = (function() {
  var a = 10;
  return new Function('x', 'y', 'return x + y + a;')
}());

console.log(add2(1,2)); // ReferenceError: a is not defined
```

'return x + y + a;'를 ' '이렇게
가져왔는데 어떻게 x,y,a를
구분하겠는가 문자열로 인식할텐데

아래 함수를 실행해보고 결과 값을 적으시오.

```
> function add(x, y){
  console.log(x,y);
  return x+y;
  }
  add(2, 5);
  console.log(x, y);

  2 5
  ▶ Uncaught ReferenceError: x is not defined
      at <anonymous>:6:13
```

젤밑에있는
console.log(x,y)부분을보면
x,y가 정의되지않았는데
출력하라고 하니 에러가
나는것일듯

Add에 필요한 매개변수는
2개다 하지만
console.log안에있는
함수도 매개변수를
2개로받고 반환값을
하나로 반환하기에
매개변수가 2개가
될수없기에 NaN이 뜬다

```
function add(x, y) {
return x + y;
}
console.log(add(2));

NaN

undefined
```

```
> function add(x, y) {
  console.log(arguments);
  return x + y;
  }
  console.log(add(2,5,10));

                                                    VM208:2
  ▼ Arguments(3) [2, 5, 10, callee: ƒ, Symbol(Symbol.iterator): ƒ]
      0: 2
      1: 5
      2: 10
    ▶ callee: ƒ add(x, y)
      length: 3
    ▶ Symbol(Symbol.iterator): ƒ values()
    ▶ [[Prototype]]: Object
  7                                                 VM208:5
```

앞의 2개의 매개변수만을 받아서 2,5를
더해서 7이 출력이 되긴 하나보다
출력이 된다는게 좀 놀랍다 매개변수가
여러 개니 당연히 에러가 날것이고
갑자기 배열을 출력한다라고
되어있으니 이상하긴하다

Call by Reference과 Call by Value라는게 있는지도 모르고 사용을 하고있었다 Call by Reference는 주소값을 참조시키는것으로 함수에게 인수값을 줄때 매개변수가 주소값을 참조하는것이다 Call by Value는 매개변수부분의 값이 그대로 복사되어 넘겨준다

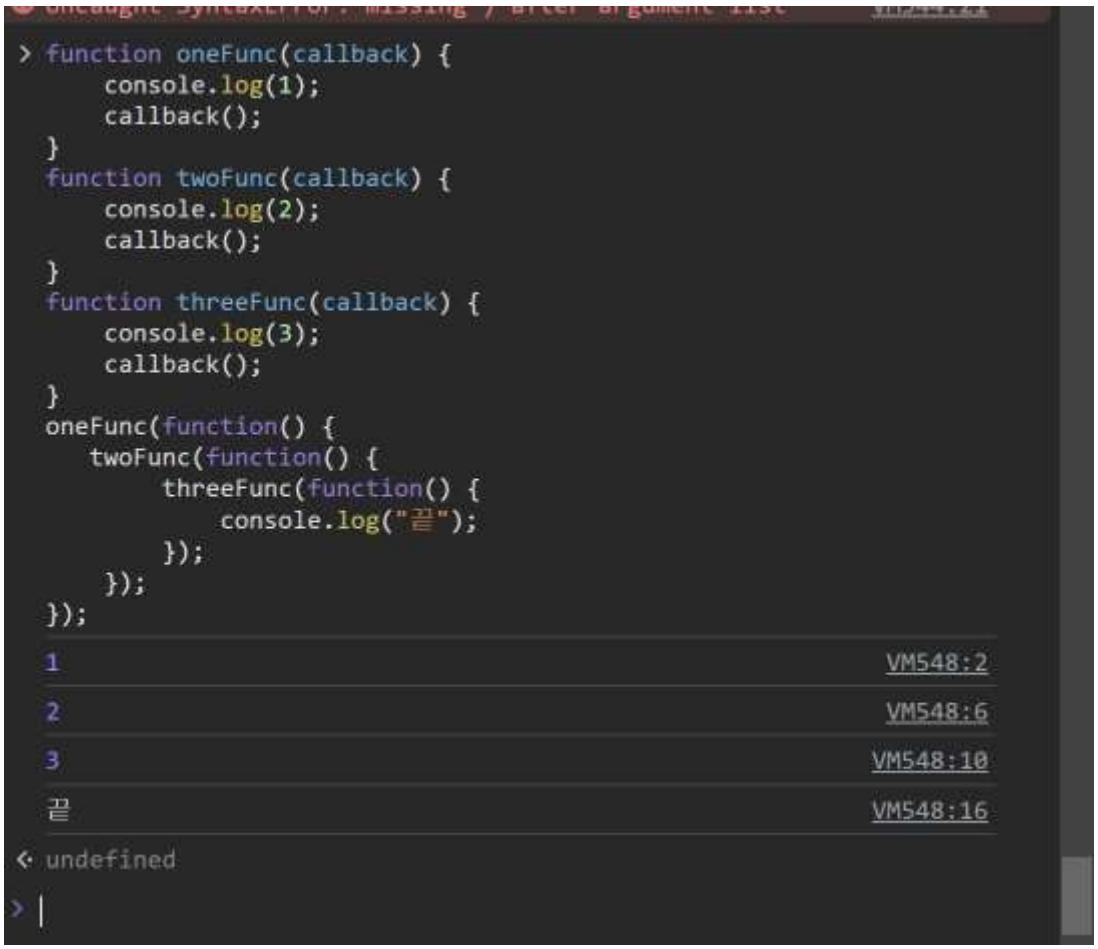
재귀함수로 팩토리얼을 구현해보시오. 그리고 해당 코드에 대한 리뷰를 해보세요.

```
> function factorial(n) {
    if (n === 0) {
      return 1;
    } else {
      return n * factorial(n - 1);
    }
  }

  const n = 5;

  console.log(factorial(n));
  120
< undefined
```

.n의 값을 -1씩해가며 곱해간다

5*factorial(4)
5*4*factorial(3)
.
.
.
5*4*3*2*1=120

callback 지옥이라는 말이 유명하다. 직접 지옥을 만들어보자.
그리고 calllback 지옥이 왜 위험한지 서술하시오.



```
function oneFunc(callback) {
    console.log(1);
    callback();
}
function twoFunc(callback) {
    console.log(2);
    callback();
}
function threeFunc(callback) {
    console.log(3);
    callback();
}
oneFunc(function() {
   twoFunc(function() {
        threeFunc(function() {
            console.log("끝");
        });
    });
});
```

```
1                                   VM548:2
2                                   VM548:6
3                                   VM548:10
끝                                  VM548:16
undefined
```

Callback지옥이라곤하지만 간단하게 만들어봄 뭘로만들지 생각하다가 너무 오래걸릴것 같아서 그냥 함수안에 함수안에 함수 느낌으로 만듬

아무런 조건이없고 엄청 간단한 출력 1 2 3 을 저렇게 길게 쓰게되었는데 조건이 여러가지고 한번에 여러 개의 매개변수를 함수로 받고 한다면 한눈에 보기도 힘들어지고 정작 만드는사람도 만들기 힘들고 해맬것임

아래의 코드 중 어떤 것이 순수 함수이며 어떤 것이 비순수 함수인지 서술하시오.

```
var count = 0;

function increase(n) {
  return ++n;
}

count = increase(count);
console.log(count);

count = increase(count);
console.log(count);
```

매개변수로 n을 받아서
n을 증가시키기에
비순수다

함수내부에 있는
count변수를 사용하여
증가시키기에 입력에만
의존하기에 순수함수다

```
var count = 0;

function increase() {
  return ++count;
}

count = increase(count);
console.log(count);

count = increase(count);
console.log(count);
```

commonJS와 ES6의
차이점을 조사해보자.

commonJS는 Node.js환경에서 사용된다고 하며
ES6은 웹브라우저 환경에서도 사용가능하며 최근
개발도구들에서도 지원된다고 함

ES6은 정적분석이 가능하며 commonJS는 동적으로
한다고함 ???

Common은 동기적으로 모듈을 로드 ES6은
비동기적으로 로드
그렇기에 ES6은 안끝나도 코드실행이 계속
Common은 모듈이 필요할때까지 코드실행이 차단 ??

callback 지옥을 해결하기
위한 예방법 혹은 대체법
찾아오기..

대체법으로 promise를 사용한다고 한다는데
Promise를 사용하면 비동기적으로 실행되는 순서를
명확하게 파악할수있다
중첩된 콜백함수를 피할수있다
콜백 함수대신에 함수반환이 가능해져 비동기코드를
더효율적으로 다룰수있게해줌

function를 사용해서 120줄 이상 어떤 function는
리턴값 주고, callback이 들어가야 함. 재귀도 들어가야
함. 화살표 함수도 넣어보자.

...
현재시각 6:39 일단올립니다..