1, 매개변수 기본값 설정

2, 전개 연산자 = 전개구문 (\*3가지 사용법)

- 나머지 매개변수

- 배열 합칠 때

- 배열객체 데이터 복사할 때

function hello(name, message) {

  console.log(`${name}님, ${message}`)

}

hello("손민정", "반가워요");

hello("정지훈"); // 결과 > 정지훈님, undefined

 undefined가 나오지 않게 하기위해 매개변수 기본값 설정하면 됌

function hello(name, message="안녕하세요") {

  console.log(`${name}님, ${message}`)

}

★★★★★★ 전개 연산자 ★★★★★★

function addNum(...numbers) {

  let sum = 0;

  for (let el of numbers) {

    sum += el;

  }

  return sum;

}

console.log(addNum(1,2, 9))

★★★★★★★★★★ console.log(addNum(1,2, 9))

여기에 어떤 숫자가 들어가도 덧셈을 할 수 있음!!

합치는 방법 쓸때도 ... 사용할 수 있다!!!!

const animal = ["bird", "cat"];

const fruits = ["apple", "banana"];

let total = [...animal,...fruits];

total;

결과 > (4) ['bird', 'cat', 'apple', 'banana']

★★★★★★ 전개 연산자 다른거 ★★★★★★

const fruits = ["apple", "banana", "cherry"];

const favorit = fruits;

console.log(favorit)

favorit[1] = "grape";

console.log(favorit);

console.log(fruits);

배열 객체 = 객체

원시유형 자료 전달 X

참조 주소 전달  >

전개 연산자로 데려오면 원본데이터에 영향을 주지 않음!

ex) const mine = [...fruits];

const mine = [...fruits];

console.log(mine)

mine[1] = "orange"

console.log(mine)

console.log(favorit);

console.log(fruits);

===========================================

const book = {

  title : "javascript",

  pages : 789

}

새로운 프로퍼티를 추가하고싶을 때!

book.author = "민정"

키값을 띄어쓰기 하고싶을 땐?

book["author name"] = "som"

결과 > {title: 'javascript', pages: 789, author: '민정', author name: 'som'}

===========================================

function fn() {

  return "result";

}

// 키값을 함수로 불러오고싶을 때

const obj = {

  [fn()] : "함수 키"

}

===========================================

이런 경우도 있음 ..

function add(a, b) {

  return a + b;

}

const obj = {

  [`${add(10, 20)} key`] : "계산식의 키"

}

===========================================

함수로 만들수도있다!

function makeUser(name, age) {

  return {

    name : name,

    age : age

  }

}

let user1 = makeUser("민정", 29);

user1;

결과 > {name: '민정', age: 29}

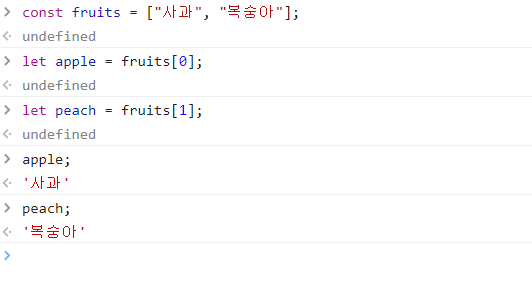
===========================================

★★★★

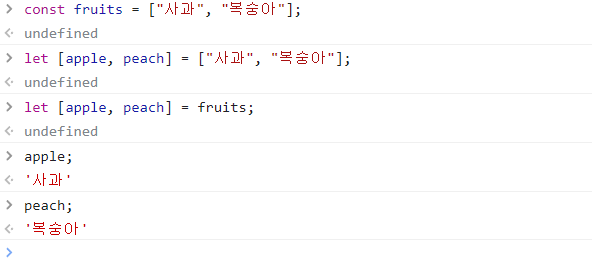
심벌 : 차별화된 객체 값을 저장하고자 할 때 사용!!!



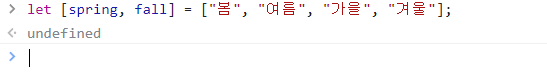
\* 구조 분해 할당 : 주어진 자료의 구조를 분해해서 변수에 할당하는 기능!



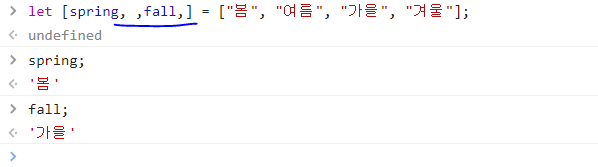
더 심플하게 할당하기



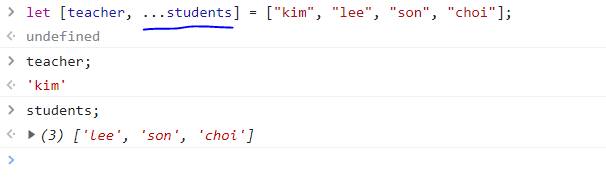
그런데 아래와 같이 매칭이 안될 땐 어떡하죠?



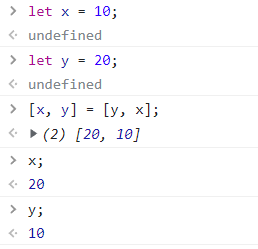
여백을 남기면 된다!



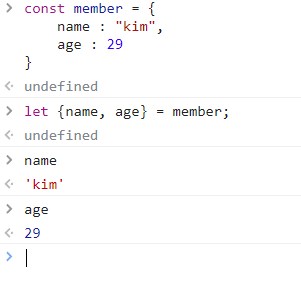
전개연산자를 사용하여 할당 할 수도 있다!



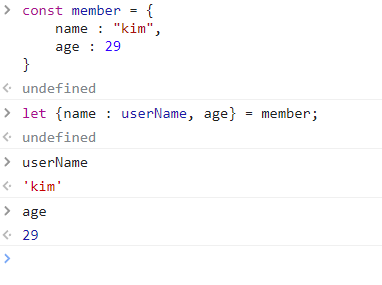
이렇게 할 수도 있음,,



객체 구조 분해 할당 방법 1



객체 구조 분해 할당 방법 2



중첩 구조 분해 할당

const student = {

  name : "홍길동",

  score : {

    history : 85,

    science : 100

  },

  friends : ["kim", "lee", "park"]

}

let {

  name,

  score :{

    history,

    science

  },

  friends : [f1, f2, f3]

} = student

결과는?

