Rest / Spread Operator Exercises

filterOutOdds()=>{

var nums=Array.prototype.slice.call(…);

return nums.filter((num)=>{num%2===0});

}

const findMin = (...args) => Math.min(...args)

const mergeObjects(odj1,obj2)=>{…obj1,…obj2};

const doubleAndReturnArgs(arr, …arg2)=>{return arr,2\*…arg2;};

const removeRandom(items)=> {let idx = Math.floor(Math.random() \* items.length);

return [...items.slice(0, idx), ...items.slice(idx + 1)];

}

const extend = (array1, array2) => {

return [...array1, ...array2];

}

const addKeyVal = (obj, key, val) => {

let newObj = { ...obj }

newObj[key] = val;

return newObj;

}

const removeKey = (obj, key) => {

let newObj = { ...obj }

delete newObj[key]

return newObj;

}

const combine = (obj1, obj2) => {

return { ...obj1, ...obj2 };

}

const update = (obj, key, val) => {

let newObj = { ...obj }

newObj[key] = val;

return newObj;

}