**Rest / Spread Operator Exercises**

1. const filterOutOdds = (...nums) => nums.filter(num => num % 2 === 0);
2. const findMin = (...args) => Math.min(...args);
3. const mergeObjects = (obj1, obj2) => ({ …obj1, ...obj2 });
4. const doubleAndReturnArgs = (arr, ...args) => [...arr, ...args.map(arg => arg \* 2)];
5. Solution:
   1. const removeRandom = (items) => {

const randomIndex = Math.floor(Math.random() \* items.length);

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

}

* 1. const extend = (array1, array2) => [...array1, ...array2];
  2. const addKeyVal = (obj, key, val) => {

const newObj = { …obj };

newObj[key] = val;

return newObj;

}

* 1. const removeKey = (obj, key) => {

const newObj = { ...obj };

delete newObj[key];

return newObj;

}

* 1. const combine = (obj1, obj2) => ({ ...obj1, ...obj2 });
  2. const update = (obj, key, val) => {

const newObj = { ...obj };

newObj[key] = val;

return newObj;

}