Problem 1

*const* practiceData = [

  [14, 21, 23, 64],

  [33, 46, 51, 65],

];

*/\*\**

*\* requirement one*

*\* increment each number by one and output the resultant array using a higher order method and for loop*

*\* HINT: use nesting*

*\* solution =  [[15,22,24,65], [34,47,52,66]]*

Solution 1

*const* incrementByOne = practiceData.map(*function* (*addFirst*) {

  return addFirst.map(*function* (*addSecond*) {

    return addSecond + 1;

  });

});

console.log(incrementByOne);

Using For Loop

*const* element = [[], []];

for (*let* i = 0; i < 2; i++) {

  for (*let* j = 0; j < 4; j++) {

    element[i].push(practiceData[i][j] + 1);

  }

}

console.log(element);

Problem 2

*\* requirement two*

*\* from the above array, create an array of array of even numbers using a higher order method and for loop*

*\* solution = [[14,64], [46]]*

*\*/*

Solution 2

*const* evenNumber = practiceData.map(*function* (*evenFirst*) {

  return evenFirst

    .map(*function* (*evenSecond*) {

      return evenSecond;

    })

    .filter((*evenSecond*) => evenSecond % 2 == 0);

});

console.log(evenNumber);

Solution 2

*const* element = [[], []];

for (*let* i = 0; i < 2; i++) {

  for (*let* j = 0; j < 4; j++) {

    if (practiceData[i][j] % 2 == 0) {

      element[i].push(practiceData[i][j]);

    }

  }

}

console.log(element);

Problem 3

*/ problem => return an array such that numbers at odd idex are multiplied by 2*

*// and numbers at even index are multiplied by 10*

Solution 3

*const* multiplyOddEven = array.map((*x*, *i*) => (i % 2 == 0 ? x \* 10 : x \* 2));

console.log(multiplyOddEven);

Problem 4

*const* carBrands = ["honda", "toyota", "maruti", "tata"];

*const* carModel = ["city", "innova", "alto", "nano"];

*const* carObject = {};

*//  using a higher order method, create a carObject variable such that*

*/\*\**

*\**

*\* {*

*\* honda:city,*

*\* toyota:innova,*

*\* maruti:alto,*

*\* tata:nano*

*\* }*

*\*/*

Solution 4

carBrands.forEach((*Element*, *index*) => {

  carObject[Element] = carModel[index];

});

console.log(carObject);

*const* arrays = [[1, 2, 3], [4, 5], [6]];

*const* result = arrays.reduce((*acc*, *curr*) => acc.concat(curr));

console.log(result);

Problem 5

*/\*\**

*\*  Flattening*

*Use the reduce method*

*to “flatten” an array of arrays into a single array that has all the elements of the original arrays.*

*const arrays = [[1, 2, 3], [4, 5], [6]];*

*// result =  [1, 2, 3, 4, 5, 6]*

*\**

Solution 5

*const* countries = ["Finland", "Denmark", "Sweden", "Norway", "Iceland"];

*const* upperCase = countries.map((*currElement*) => currElement.toUpperCase());

console.log(upperCase);