Вимоги

* Результатом роботи має бути архів з наступною структурою папки: *IT-Lab5-[Student\_Name]*

*index.html*

*|\_\_ js /*

*|\_\_forEach.js*

*|\_\_ …..*

*|\_\_ getTopNRichestNames.js*

* JS-код має бути у відповідних файлах, на кожне завдання свій файл
* Код має бути робочим, відтестованим та читабельним (не забути відформатувати)

Homework

1. forEach – написати функцію яка отримує масив та функцію, як параметри та виконує дану функцію для кожного елемента у масиві. Функція має викликатися з відповідним елементом масиву як аргументом функції.

Приклад виклику:   
 forEach( [3, 5, 2], function(el){ console.log(el) } ) // -> 3 5 2

1. getTransformedArray – написати функцію яка отримує масив та функцію, як параметри та повертає новий масив з кожним відповідним елементом зміненим переданою функцією.

Приклад виклику:   
 function increment(num){ return num + 1 } // just returns incremented number  
 getTransformedArray( [1, 7, 20], increment ); // -> [ 2, 8, 21 ]

1. pluckByAttribute – написати функцію, яка отримує масив об’єктів та стрічку назви властивості об’єкту, як параметри та повертає масив значень усіх переданих у масиві об’єктів з отриманою назвою властивості.

Приклад виклику:  
 var presidents = [ { name: ‘George’, surname: ‘Kush’ } ,  
 { name: ‘Barako’, surname: ‘Obaame’ } ];  
 pluckByAttribute( presidents, ‘name’ ); // -> [‘George’, ‘Barako’]

1. cypherPhrase – написати функцію яка отримує об’єкт та стрічку, як параметри та повертає стрічку закодовану за правилами переданими у об’єкті.

Приклад виклику:  
 var charactersMap = {a: ‘o’, c: ‘d’, t: ‘g’}  
 cypherPhrase( charactersMap, ‘kitty cat’ ); // -> “kiggy dog”

1. decypherPhrase – написати функцію яка отримує об’єкт та стрічку закодовану функцією cypherPhrase, як параметри та повертає розкодовану стрічку за правилами переданими у об’єкті.

Приклад виклику:  
 var charactersMap = {a: ‘o’, c: ‘d’, t: ‘g’}  
 decypherPhrase( charactersMap, ‘kiggy dog’ ); // -> “kitty cat”