Recursion

1. Write a recursive function to determine whether all digits of the number are odd or not.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4211133 | false |
| 7791 | true |
| 5 | true |

1. Given an array of numbers. Write a recursive function to find its minimal positive element. (if such element does not exist, return -1)․

|  |  |
| --- | --- |
| **Input** | **Output** |
| [56, -9, 87, -23, 0, -105, 55, 1] | 0 |
| [45, -9, 15, 5, -78] | 5 |
| [-5, -9, -111, -1000, -7] | -1 |

1. Given an array of nested arrays. Write a recursive function that flattens it. (Hint create function that concats arrays).

|  |  |
| --- | --- |
| **Input** | **Output** |
| [1, [3, 4, [1, 2]], 10] | [1, 3, 4, 1, 2, 10] |
| [14, [1, [[[3, []]], 1], 0] | [14, 1, 3, 1, 0] |

1. Given a number. Write a function that calculates its sum of the digits and if that sum has more than 1 digit find the sum of digits of that number. Repeat that process if needed and return the result.

|  |  |
| --- | --- |
| **Input** | **Output** |
| 14 | 5 |
| 29 | 2 |
| 999999999999 | 9 |

Objects

1. Given an object. Write a function that creates a deep copy of it.

|  |  |
| --- | --- |
| **Input** | **Output** |
| var a = { a: ‘1’, b: { a: 2} };  var b = deepCopy(a);  a.b.a = 123;  console.log(b.b.a !== 123); | true |

1. Write a function to get the length of an object.

|  |  |
| --- | --- |
| **Input** | **Output** |
| {  name: "David Nolan",  class: "VI",  rollno: 12  }; | 3 |

1. \* Given the list of the following readers:  
   [  
    { book: "Catcher in the Rye", readStatus: true, percent: 40},  
    { book: "Animal Farm", readStatus: true, percent: 20},  
    { book: "Solaris", readStatus: false, percent: 90 },  
    { book: "The Fall", readStatus: true, percent: 50 },  
    { book: "White Nights", readStatus: false, percent: 60 } ,  
    { book: "After Dark", readStatus: true, percent: 70 }  
   ];  
   Output the books sorted by the percent in descending order which readStatus is true.