

Тестовое задание:

- Реализация метода **Array.groupBy**

Добавить к глобальному объекту **Array** метод **groupBy**, который позволяет сгруппировать элементы массива по результату выполнения функции полученной в качестве аргумента.

Пример:

```
[1,2,3,2,4,1,5,1,6].groupBy()  
{  
  1: [1, 1, 1],  
  2: [2, 2],  
  3: [3],  
  4: [4],  
  5: [5],  
  6: [6]  
}  
  
[1,2,3,2,4,1,5,1,6].groupBy(function(val) { return val % 3; } )  
{  
  0: [3, 6],  
  1: [1, 4, 1, 1],  
  2: [2, 2, 5]  
}
```

Примечание: элементами массива могут быть любые типы данных.

- Реализовать функцию **quaddouble(num1,num2)**

Реализовать функцию, которая принимает в качестве аргументов числа **num1** и **num2**. Функция должна возвращать **true** если в **num1** есть 4 подряд идущих одинаковых цифры и 2 тех же, подряд идущих, цифры в **num2**, в ином случае функция должна вернуть **false**.

Пример:

```
quaddouble(45568411115, 11223344) //true;  
quaddouble(11112344445, 442253) //true;  
quaddouble(12222345, 123452) //false;  
quaddouble(12345, 12345) //false;
```

- Реализовать функцию **lispHero(inputStr)**

Реализовать функцию **lispHero**, которая будет проверять корректность открытия/закрытия скобок в строке **inputStr** и возвращать **true** если строка корректна, в ином функция должна возвращать **false**.

Строка **inputStr** состоит только из скобок {}, [] и ().

Пример:

```
lispHero('()') //true
lispHero('[]') //false
lispHero('{}[]()') //true
lispHero('([{}])') //true
lispHero('()({}){()}[]') //false
```