

## Práctica guiada - Parte 2

Fecha de entrega: 24-oct-2019

Esta práctica es la segunda parte de un proyecto completo consistente en una aplicación web de gestión de tareas.

El objetivo es la implementación de un conjunto de funciones JavaScript haciendo uso de las **funciones de orden superior sobre arrays**. Todas estas funciones recibirán un array de tareas, donde cada tarea se representa mediante un objeto con tres propiedades:

- una propiedad **text** que contiene el texto de la tarea.
- una propiedad booleana **done** que indica si la tarea ha sido finalizada o no.
- una propiedad **tags**, que es un array que contiene las etiquetas de la tarea.

La propiedad **done** es opcional y, si no se indica, se considera que tiene el valor **false**.

A continuación se muestra un ejemplo de array de tareas:

```
let listaTareas = [  
  { text: "Preparar práctica AW", tags: ["AW", "practica"] },  
  { text: "Mirar fechas congreso", done: true, tags: [] },  
  { text: "Ir al supermercado", tags: ["personal"] },  
  { text: "Mudanza", done: false, tags: ["personal"] },  
];
```

Se pide implementar las siguientes funciones utilizando exclusivamente funciones de orden superior sobre arrays, excepto **forEach**. No es posible el uso de bucles para implementar estas funciones.

### Función **getToDoTasks(tasks)**

Esta función devuelve un array con los textos de aquellas tareas de la lista de tareas **tasks** que no estén finalizadas.

Por ejemplo, la llamada:

```
getToDoTasks(listaTareas)
```

devuelve el siguiente array:

```
[ 'Preparar práctica AW', 'Ir al supermercado', 'Mudanza' ]
```

### Función findByTag(tasks, tag)

Esta función devuelve un array que contiene las tareas del array **tasks** que contengan, en su lista de etiquetas, la etiqueta pasada como segundo parámetro.

Por ejemplo, la llamada:

```
findByTag(listaTareas, "personal")
```

devuelve el siguiente array:

```
[
  { text: 'Ir al supermercado', tags: [ 'personal' ] },
  { text: 'Mudanza', done: false, tags: [ 'personal' ] }
]
```

### Función findByTags(tasks, tags)

Esta función devuelve un array que contiene aquellas tareas del array **tasks** que contengan al menos una etiqueta que coincida con una de las del array **tags** pasado como segundo parámetro.

Por ejemplo, la llamada:

```
findByTags(listaTareas, ["personal", "practica"])
```

devuelve el siguiente array:

```
[
  { text: 'Preparar práctica AW', tags: [ 'AW', 'practica' ] },
  { text: 'Ir al supermercado', tags: [ 'personal' ] },
  { text: 'Mudanza', done: false, tags: [ 'personal' ] }
]
```

### Función countDone(tasks)

Esta función devuelve el número de tareas completadas en el array de tareas **tasks** pasado como parámetro.

Por ejemplo, la llamada:

```
countDone(listaTareas)
```

devuelve 1.

### Función createTask(texto)

Esta función recibe un **texto** intercalado con etiquetas, cada una de ellas formada por una serie de caracteres alfanuméricos precedidos por el signo **@**. Esta función debe devolver un objeto tarea con su array de etiquetas extraídas de la cadena texto. Por otra parte, el atributo **text** de la tarea resultante no debe contener las etiquetas de la cadena de entrada ni espacios en blanco de más.

Por ejemplo, la llamada:

```
createTask("Ir al medico @personal @salud")
```

devuelve el siguiente objeto:

```
{ text: 'Ir al médico', tags: [ 'personal', 'salud' ] }
```

La llamada:

```
createTask("@AW @practica Preparar práctica AW")
```

devuelve el siguiente objeto:

```
{ text: 'Preparar práctica AW', tags: [ 'AW', 'practica' ] }
```

La llamada:

```
createTask("Ir a @deporte entrenar")
```

devuelve el siguiente objeto:

```
{ text: 'Ir a entrenar', tags: [ 'AW', @deporte ] }
```

## Entrega de la práctica

La fecha límite para entregar el trabajo es el 24 de octubre de 2019 (el enlace para la entrega en el campus virtual se cierra a las 23:55).

En el campus se entregará un fichero **gxx.zip** (xx es el número de grupo de laboratorio) que contenga un fichero JavaScript con las funciones descritas en el enunciado. Las prácticas que no respeten el nombre de las funciones serán consideradas como no entregadas.