

HackerRank Question - Promedio

22m left

### 1. Promediando salarios

Escribe una función llamada `calcularPromedioSalarios` que reciba como parámetro un arreglo de salarios y calcule el promedio de los valores mayores a 0. La función debe retornar el promedio calculado en forma de string: "Promedio de salarios: promedio"

**Instrucciones**

1. La función `calcularPromedioSalarios` recibe un parámetro llamado `salarios`, que es un array de números enteros.
2. Dentro de la función, se debe realizar lo siguiente:
  - Si no se encuentra ningún salario mayor a 0, se debe retornar el número 0.
  - Caso contrario, retornar el promedio de todos los salarios mayores a 0.

**Ejemplo**

```
const salarios = [0, 2000, 1500, 0, 1600, 8000, 0];
```

```
calcularPromedioSalarios(salarios);
```

Language: JavaScript (Node.js) Environment Autocomplete Ready

```
1 > 'use strict';...
26
27 /*
28 * Completa la función 'calcularPromedioSalarios' a continuación.
29 *
30 * Debe retornar un STRING.
31 * Recibe un ARREGLO DE NÚMEROS como parámetro.
32 *
33 * No modifiques nada por fuera del cuerpo de la función.
34 */
35
36 function calcularPromedioSalarios(salarios) {
37   // Tu código aquí
38   var promedio = 0;
39   for (var i = 0; i < salarios.length; i++) {
40     if (salarios[i] > 0) {
41       promedio += salarios[i];
42     } if (i == salarios.length - 0) {
43       return "El promedio de los salarios es: " + promedio;
44     }
45   }
46 }
47
48
49
50
51 > function main() { ...
```

Test Results Custom Input Run Code Run Tests Submit

Line: 26 Col: 1

8:49 p.m. 23/09/2023

HackerRank Question - Control

21m left

### 2. Control de oxígeno en la nave

**Descripción de la función**

La función `verificarNivelesOxigeno` recibe como parámetro un arreglo de números llamado `niveles`, que representa los niveles de oxígeno en diferentes áreas de una nave espacial.

**Instrucciones**

La función debe realizar las siguientes acciones:

1. Verificar si algún nivel de oxígeno en el arreglo es igual o menor a 0. Si se encuentra un nivel crítico, la función debe activar la alarma de emergencia y devolver el mensaje: "Alerta de emergencia! Nivel de oxígeno crítico detectado en el área X", donde X es el índice del área con el nivel de oxígeno crítico.
2. Si algún nivel de oxígeno en el array es menor o igual a 5, la función debe devolver el mensaje: "Nivel de oxígeno bajo detectado en el área X", donde X es el índice del área con el nivel de oxígeno bajo.
3. Si todos los niveles de oxígeno son superiores a 5, la función debe devolver el mensaje: "Todos los niveles de oxígeno son estables".

Language: JavaScript (Node.js) Environment Autocomplete Ready

```
1 > 'use strict';...
26
27 /*
28 * Completa la función 'verificarNivelesOxigeno' a continuación.
29 *
30 * Debe retornar un STRING.
31 * Recibe un ARREGLO DE NÚMEROS como parámetro.
32 *
33 * No modifiques nada por fuera del cuerpo de la función.
34 */
35
36 function verificarNivelesOxigeno(nivelesOxigeno) {
37   // Tu código aquí
38   var nivelesOxigeno=0;
39 }
40
41 > function main() { ...
```

Test Results Custom Input Run Code Run Tests Submit

Line: 26 Col: 1

8:50 p.m. 23/09/2023

HackerRank Question - Clase Libro

21m left

### 3. Clase Libro

Para este ejercicio, deberás definir una clase *Libro* con sus propiedades y métodos.

Las propiedades de la clase *Libro* son:

- *titulo*
- *autor*
- *traducciones*

El constructor de la clase recibe esos tres datos por parámetro.  
Los métodos de la clase libro son:

- *getTitulo*: permite obtener el titulo del libro
- *getAutor*: permite obtener el autor del libro
- *getTraducciones*: retorna un arreglo con todos los idiomas de las traducciones
- *addTraduccion*: a partir de los parámetros recibidos, agrega un objeto al arreglo de traducciones

Ejemplo:

```
var traduccionesDeHP = [
```

```
1 > "use strict";...
25 /*
26  * Completa la clase 'Libro' a continuación.
27  *
28  * No modifiques nada por fuera del cuerpo de los métodos.
29  */
30
31 class Libro{
32   constructor(titulo, autor, traducciones){
33     // Inicializar las propiedades del libro con los valores recibidos como argumento
34     // tu código aquí
35     this.titulo= titulo
36     this.autor= autor
37     this.traducciones=traducciones
38   }
39
40   getTitulo(titulo){
41     // tu código aquí
42     this.titulo= titulo
43   }
44
45   getAutor(autor){
46     // tu código aquí
47     this.autor= autor
48   }
49
50   getTraducciones(traducciones){
51     // Retornar un arreglo con los idiomas de las traducciones
52     // tu código aquí
53     this.traducciones=traducciones
54   }
55 }
```

Test Results Custom Input

Run Code Run Tests Submit

Line: 25 Col: 1

HackerRank Question - Libros y

20m left

### 4. Libros y autores

La función *obtenerLibrosPorAutor* recibe por parámetro una lista de libros y el nombre de un autor, y debe retornar un string con las obras escritas por ese autor, separadas por ', '.

Ejemplo:

```
const libros = [
  { titulo: "Cien años de soledad",
    autor: "Gabriel García Márquez" },
  { titulo: "1984", autor: "George Orwell" },
  { titulo: "Un mundo feliz", autor:
    "Aldous Huxley" },
  { titulo: "Crónica de una muerte
    anunciada", autor: "Gabriel García
    Márquez" }
];

const autor = "George Orwell"

obtenerLibrosPorAutor(libros, autor)
```

debería retornar:

```
'1984'
```

```
1 > "use strict";...
26 /*
27  * Completa la función obtenerLibrosPorAutor a continuación.
28  *
29  * La función debería retornar un STRING.
30  * La función acepta un ARREGLO DE OBJETOS y un STRING como parámetro
31  *
32  * No modifiques nada por fuera del cuerpo de esta función.
33  */
34
35 function obtenerLibrosPorAutor(libros, autor) {
36   // Tu código aquí
37   var librosPorAutor = [];
38   for (var i = 0; i < libros.length; i++) {
39     if (libros[i].autor === autor.split(",")) {
40       librosPorAutor.push(libros[i]);
41     }
42   }
43   return librosPorAutor;
44 }
45
46 > function main() { ...
```

Test Results Custom Input

Run Code Run Tests Submit

Line: 26 Col: 1

HackerRank Question - Bienvenido Sr. | 20m left

### 5. Bienvenido Sr

En esta ocasión trabajarás con objetos.

Deberás escribir una función que, a partir de un objeto que representa una persona, retorne un mensaje de bienvenida a la misma.

- Si el objeto tiene las propiedades **nombre**, **apellido** e **invitado (en true)** debe retornar "Nombre Apellido, un gusto tenerlo nuevamente! Bienvenido".
- Si el objeto tiene las propiedades **invitado (en true)** y **apellido** debe retornar "Bienvenido Sr. Apellido".
- Si el objeto tiene las propiedades **invitado (en true)** y **nombre** debe retornar "Hola Nombre, tu mesa está lista".
- En caso de que la propiedad **invitado** sea **false** o no la tenga, debe retornar "Disculpe señor, no está invitado a la fiesta".

Ejemplo:

```
let persona = {
  nombre: "Lionel",
  apellido: "Messi"
}
```

Language: JavaScript (Node.js) | Environment | Autocomplete Ready

```
1 > 'use strict';...
28
29
30 /*
31 * Completa la función 'bienvenidoSr' a continuación.
32 *
33 * La función debe retornar un STRING.
34 * La función recibe como parámetro un OBJETO 'persona'.
35 * No modifiques nada por fuera del cuerpo de la función.
36 */
37
38 function bienvenidoSr(persona) {
39   //Tu código aquí
40   var persona = persona;
41   var intado = [];
42
43   for (var i = 0; i < persona.length; i++) {
44     if (persona[i] > 0) {
45       this.nombre = persona;
46       this.apellido = persona;
47       this.intado = true;
48       return this.persona+"Un gusto tenerlo nuevamente! Bienvenido";
49     }if(i == persona.length - 0) {
50       this.intado = true;
51       this.apellido = persona;
52       return "Bienvenido Sr. " + this.apellido;
53     }else {
54       this.intado = false;
55     }
56   }
57 }
```

Test Results Custom Input Run Code Run Tests Submit

Line: 28 Col: 1

8:51 p.m. 23/09/2023

HackerRank Question - Obtener | 20m left

### 6. Obtener iniciales

Escribe una función que recibirá por parámetro un string con nombres. Cada uno de estos tendrá su nombre de pila y su apellido, los cuales estarán separados por comas (","), Deberás retornar las iniciales del nombre y apellido de cada uno.

Ejemplo:

```
obtenerIniciales("Juan Perez, María García, Pedro Sanchez")
```

Debería retornar:

```
'JP, MG, PS'
```

Descripción de la función

La función **obtenerIniciales** debe retornar un **string**.

La función **obtenerIniciales** recibe como parámetro(s):

- **nombres**: string con nombre de pila y apellido separados por coma (",")

Language: JavaScript (Node.js) | Environment | Autocomplete Ready

```
1 > 'use strict';...
26
27 /*
28 * Complete la función 'obtenerIniciales' a continuación.
29 *
30 * Se espera que la función devuelva un STRING.
31 * La función recibe un STRING como parámetro.
32 */
33
34 function obtenerIniciales(nombres) {
35   // Tu código aquí
36   var nombres = nombres.split(",");
37   var iniciales = "";
38   return iniciales.plot;
39 }
40
41 > function main() { ...
```

Test Results Custom Input Run Code Run Tests Submit

Line: 26 Col: 1

8:51 p.m. 23/09/2023