

Práctica 1: Inicio

Abrir en el entorno de desarrollo el archivo `index.html` correspondiente a la práctica 1 y completar el código. El archivo ya contiene parte del código y el objetivo es editar los comentarios que comiencen por la palabra “TODO”.

Es importante mantener el resto del código sin modificar, así como los nombres de las funciones, clases CSS o IDs de los elementos.

Tareas:

- 1) **Añadir en los estilos los que se indica en los comentarios.**

Por ejemplo :

```
h1 {  
    background-color: #2a5599;  
    /* TODO 1  
       Poner el texto en blanco y añadir un padding de 5px  
       2 líneas de código  
    */  
    color: ;  
    padding: ;  
}
```

- 2) **Crear una variable para almacenar datos.**

Normalmente se necesita una variable para almacenar los datos cargados al principio de la aplicación. Para hacer esto añadir una variable global que almacene los datos que necesitamos dentro del tag `<script>`.

```
<script>  
    // TODO 3 Añadir una variable global llamada almacen  
    let almacen = ;
```

- 3) **Cargar los datos.**

Para cargar los datos hay que implementar la función `cargarDatos()` que es la responsable de ello.

```

function cargarDatos() {
    let promise = ;//TODO 4: Añadir el código para cargar el archivo
"rutasAereas.csv" | 1 línea
    return promise.then( rutas => {
        almacen.rutas = ;//TODO 5: Guardar las rutas en la variable
almacen
        return almacen;
    })
}

```

Los datos cargados genera un objeto JSON que es una lista de rutas (23041), cada una de ellas con los siguientes campos:

1. ID:"1"
2. AirlineCountry:"United States"
3. AirlineID:"24"
4. AirlineName:"American Airlines"
5. DestAirport:"Charlotte Douglas International Airport"
6. DestAirportID:"3876"
7. DestCity:"Charlotte"
8. DestCode:"CLT"
9. DestCountry:"United States"
10. DestLatitude:"35.2140007"
11. DestLongitude:"-80.94309998"
12. SourceAirport:"Lehigh Valley International Airport"
13. SourceAirportCode:"ABE"
14. SourceAirportID:"4355"
15. SourceCity:"Allentown"
16. SourceCountry:"United States"
17. SourceLatitude:"40.65209961"
18. SourceLongitude:"-75.44080353"

4) Preparar los datos.

El archivo cargado contiene un registro por cada ruta y lo que buscamos es visualizar la lista de compañías aéreas junto con el número de rutas de cada compañía. Para ello, hay una función que tiene como entrada la lista de rutas y como salida una lista de cada

```

function agruparPorCompania ( datos ) {
    //Recorrer todas las rutas y generar un diccionario cuyas claves
    sean los ID de las compañías
    // y los valores la información de la misma.
    let resultado = datos.reduce( (resultado, d) => {
        let compania = resultado[d.AirlineID] || {
            "ID": d.AirlineID,
            "Nombre": d.AirlineName,
            "Rutas": 0
        }

        compania.Rutas ; //TODO 6: Incrementar el contador de número
        de rutas por compañía.

        resultado[compania.ID] = ; //TODO 7: Almacenar la información
        en el diccionario de con el id como clave.

        return resultado;
    }, {})

    //Convertir el diccionario obtenido en una lista (array).
    resultado = Object.keys(resultado).map( clave =>
    resultado[clave])

    resultado = resultado.sort ( (c1,c2) => {
        return d3.descending ;
    }); //TODO 8: Ordenar los datos en orden descendente de rutas.

    return resultado;
}

```

El resultado de esta función debería ser una lista de 19 compañías (se puede ver en la consola del navegador) ordenadas de más a menos rutas.

```

(19) [{...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}]
1. 0:{ID: "4296", Nombre: "Ryanair", Rutas: 2482}
2. 1:{ID: "24", Nombre: "American Airlines", Rutas: 2340}
3. 2:{ID: "2009", Nombre: "Delta Air Lines", Rutas: 1977}
4. 3:{ID: "5265", Nombre: "US Airways", Rutas: 1947}

```

5) Ejecutar el código

Ahora ya se puede ejecutar las funciones que hemos creado. Para ello, hay que llamar las funciones de esta manera.

```
function mostrarDatos() {  
  
    almacen.companias = agruparPorCompania ( almacen.rutas );  
  
    console.log (almacen.companias );  
}  
  
cargarDatos().then( mostrarDatos );
```

Fin de la práctica

El resultado final debería ser algo similar a la siguiente imagen.

Rutas Aéreas

Compañías

Aeropuertos

| | |
|--|--|
| | |
|--|--|

Vistos los datos, qué compañía aérea es la que más rutas