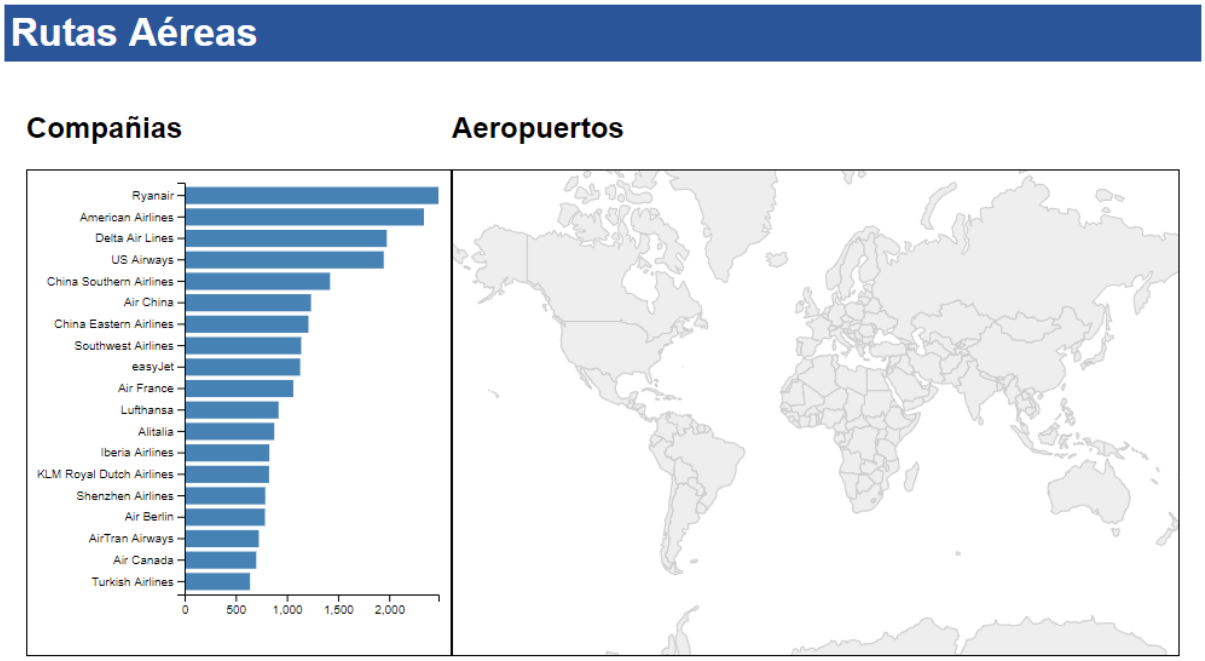
Práctica 3: Mapa Base

Abrir en el entorno de desarrollo el archivo index.html correspondiente a la práctica 2 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.

En esta práctica el objetivo es añadir un mapamundi a la aplicación. De momento este mapa solo contiene las fronteras de los países.

Al final de la práctica debería obtener algo similar a esto:



Tareas:

1) **Carga de la información del mapa**.
   Para dibujar el mapa hay que cargar un archivo GeoJSON con las especificaciones de los países y sus fronteras. El archivo se llama countries.geo.json y se encuentra en la carpeta mapas.

   Para ello hay que modificar la función de cargarDatos para que cargue también los datos del mapa.

```
function cargarDatos() {
        return Promise.all([
                d3.csv("/datos/rutasAereas.csv"),
                d3.json("/mapas/countries.geo.json")
            ]).then( datasets => {
                console.log('Datos leidos');
                almacen.rutas = datasets[0];
                almacen.geoJSON = datasets[1];

            return almacen;
        });
    }
```

Ahora se usa Promise.all para cargar múltiples archivos y la variable datasets contendrá los archivos en el mismo orden en el que se pasan.

2) Configuración del mapa.
De la misma forma que en el caso del diagrama de barras, hay una función para gestionar el tamaño del mapa denominada "*configuracionMapa*".

```
    function configuracionMapa () {
        let ancho = 600;
        let alto = 400;
        let contenedor = ; //TODO 1: seleccionar el svg con id Mapa
        contenedor.attr("width",)     //TODO 2: asignar anchura y altura
del contenedor.
                  .attr("height",);

        return {ancho, alto, contenedor};
    }
```

3) Proyección.
En el siguiente paso hay que generar una proyección y para ello hay una función que se encargará de ello. Será necesario trasladar la proyección para asegurarse que el centro del mapa corresponde con el centro del svg. La proyección también se guardará en el almacén.

```
function obtenerProyeccionMapa ( configuracion ) {
        let {ancho, alto} = configuracion;
        let proyeccion = ; //TODO 3: Crear proyeccion de tipo Mercator.
        proyeccion.scale(97)
                    .translate([ancho / 2, alto / 2 + 20])

        almacen.proyeccionMapa = proyeccion;

        return proyeccion;
    }
```

4) Dibujar el mapa.
   Ahora hay que dibujar el mapa base y la función encargada de ello es "*dibujaMapaBase*". Tiene como parámetros el contenedor, los datos GeoJSON (o lo que es lo mismo, el contorno de los países a dibujar) y la proyección.

```
function dibujarMapaBase (contenedor , paises , proyeccion ){
        let path = ; //TODO 4: crear un generador geoPath y asignarle su
proyeccion

        contenedor.selectAll("path")
            .data(countries)
            .enter()
                .append("g")
                .append("path")
                    .attr("d",  ) //TODO 5: usar el generador path para
dibujar cada país
                    .attr("stroke", "#ccc")
                    .attr("fill", "#eee");

    }
```

5) Llamar a las funciones.
   Por último, hay una función que se encarga de llamar a las funciones anteriores y pasar los parámetros. Esa función es "*dibujarMapa*".

```
function dibujarMapa ( geoJSON ) {
        let configuracion = configuracionMapa();
        let proyeccion = obtenerProyeccionMapa(configuracion)
        dibujarMapaBase ( configuracion.contenedor , geoJSON.features ,
proyeccion );
    }
```

También hay que añadir una llamada a "*dibujarMapa*" desde "*mostrarDatos*".

```
function mostrarDatos() {

    almacen.companias = agruparPorCompania ( almacen.rutas );

    //console.log (almacen.companias );

    dibujarDiagramaCompanias (almacen.companias);

    dibujarMapa ( almacen.geoJSON );

}
```