

Ejercicio

Conversión CSV

Se requiere convertir archivos CSV de manera genérica, para lo cual se necesita una función en *elixir* que reciba por parámetros:

- un string con toda la data contenida en un archivo CSV (incluyendo la línea de encabezados),
- un mapa con las opciones de conversión para cada uno de los headers que hay que hay que capturar del CSV, si un encabezado no esta en el mapa, no hay que capturar dicho campo.
- una lista de opciones para realizar la conversión; por ejemplo, el separador entre cada uno de los campos del archivo CSV que por defecto es la coma (,) pero puede ser otro campo como el tabulador o el punto y coma (;).

El resultado debe ser una tupla de la forma:

- `{:ok, respuesta}`, si no hubo errores de procesamiento; **respuesta** es una lista de mapas con cada una de las listas procesadas ó
- `{:error, errores}`, donde **errores** es una lista:
 - errores en la conversión de un campo debe tener al menos:
 - * el número de la línea de la fila donde ocurrió el error
 - * el nombre del campo donde ocurrió el error
 - * el valor del campo
 - * una reseña (puede ser un átomo) al tipo de conversión que se trato de aplicar
 - otro tipo de errores
 - * el número de la línea de la fila donde ocurrió el error
 - * la lista de encabezados de los campos
 - * la lista de campos reconocidos
 - * una reseña (puede ser un átomo) al tipo de error que ocurrió

Los tipos de datos requeridos son **string**, **integer**, **float** y **boolean**; pero puede implementar mas si lo desea.

Ejemplos

CSV

A continuación se presenta un string que contiene la data de un CSV de ejemplo:

```
"System ID,School,Phone,Address,City,State,Zip,Type,Principal,Website,Other,Image,Active\r\n"
```

Ejemplo de Mapa de definiciones 1 A continuación se presenta un ejemplo de un mapa para reconocer el CSV contenido en el string anterior:

```
%{  
  "Active" => :string,
```

```

    "Image" => :string,
    "System ID" => :string
  }

```

Ejemplo resultado al procesar con el mapa de definiciones 1 A continuacion se presenta el resultado esperado para el proceso del CSV anterior:

```

{:ok,
 [
   %{
     "Active" => "TRUE",
     "Image" => "/cms/lib/TC00000006/centricity/Domain/4866/Clip1.jpg",
     "System ID" => "569"
   },
   %{
     "Active" => "TRUE",
     "Image" => "/cms/lib/TC00000006/centricity/Domain/4866/SWDocCoverLogo.jpg",
     "System ID" => "570"
   },
   %{
     "Active" => "TRUE",
     "Image" => "/cms/lib/TC00000006/centricity/Domain/4866/a_bug_icon.png",
     "System ID" => "571"
   },
   %{
     "Active" => "TRUE",
     "Image" => "/cms/lib/TC00000006/centricity/Domain/8774/Lion2.jpg",
     "System ID" => "572"
   }
 ]}

```

Ejemplo de Mapa de definiciones 2 A continuacion de presenta un ejemplo de un mapa para reconocer el CSV contenido en el string anterior:

```

%{
  "Active" => :boolean,
  "Phone" => :string,
  "School" => :string,
  "State" => :string,
  "System ID" => :integer
}

```

Ejemplo resultado al procesar con el mapa de definiciones 2 A continuacion se presenta el resultado esperado para el proceso del CSV anterior:

```

{:ok,
 [

```

```

%{
  "Active" => true,
  "Phone" => "814-555-1212",
  "School" => "Happy Valley Elementary School",
  "State" => "PA",
  "System ID" => 569
},
%{
  "Active" => true,
  "Phone" => "814-555-1212",
  "School" => "Happy Valley High School",
  "State" => "PA",
  "System ID" => 570
},
%{
  "Active" => true,
  "Phone" => "814-555-1212",
  "School" => "Happy Valley Middle School",
  "State" => "PA",
  "System ID" => 571
},
%{
  "Active" => true,
  "Phone" => "814-555-1212",
  "School" => "Happy Valley School District",
  "State" => "PA",
  "System ID" => 572
}
}]

```