

# CLASE 1

## Merge de Listas

# El problema de Merge

La operación de Merge consiste en generar una nueva estructura de datos (arreglos, listas) ordenada a partir de la mezcla de dos o mas estructuras de datos previamente ordenadas.

Las estructuras que se combinan guardan el mismo orden lógico interno (por ejemplo datos ordenados alfabéticamente)

# Merge

Un caso de la vida real

Pepe y Moni viven juntos

# Merge

Un caso de la vida real



Pepe y Moni tienen una biblioteca con 4 estantes.

Cada estante está ordenado alfabéticamente, por título

# Merge

Un caso de la vida real

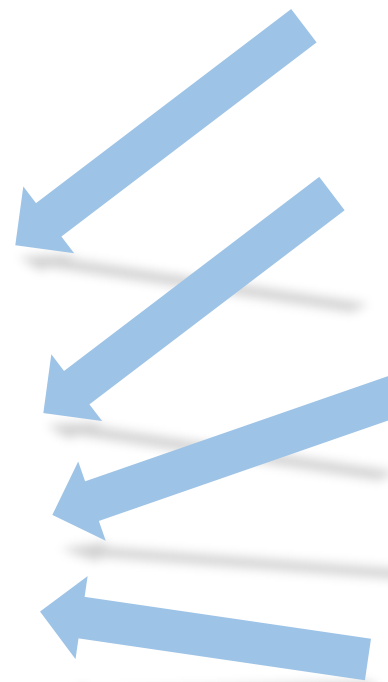


Quieren pasar todos los libros a un solo estante también ordenado alfabéticamente, por título



# Merge

Un caso de la vida real



# Merge

Un caso de la vida real

Pepe se encargará de ir  
colocando los libros en el  
nuevo estante



Moni se encargará de ir  
pasándole los libros a  
Pepe



Estante  
1

La Broma  
Platero y  
Yo  
Rayuela

Estante  
2

Cronicas  
marcianas  
Rayuela  
Salud y  
Sociedad

Estante  
3

Edad  
Prohibida  
Un mundo  
feliz  
Salud y  
Sociedad  
Viven

Estante  
4

Edad  
Prohibida  
Mr Vertigo



Comienzan a trabajar...

Estante nuevo



Estante  
1

La Broma

Platero y  
Yo

Rayuela

Estante  
2

Cronicas  
marcianas

Rayuela

Salud y  
Sociedad

Estante  
3

Edad  
Prohibida

Un mundo  
feliz

Salud y  
Sociedad

Viven

Estante  
4

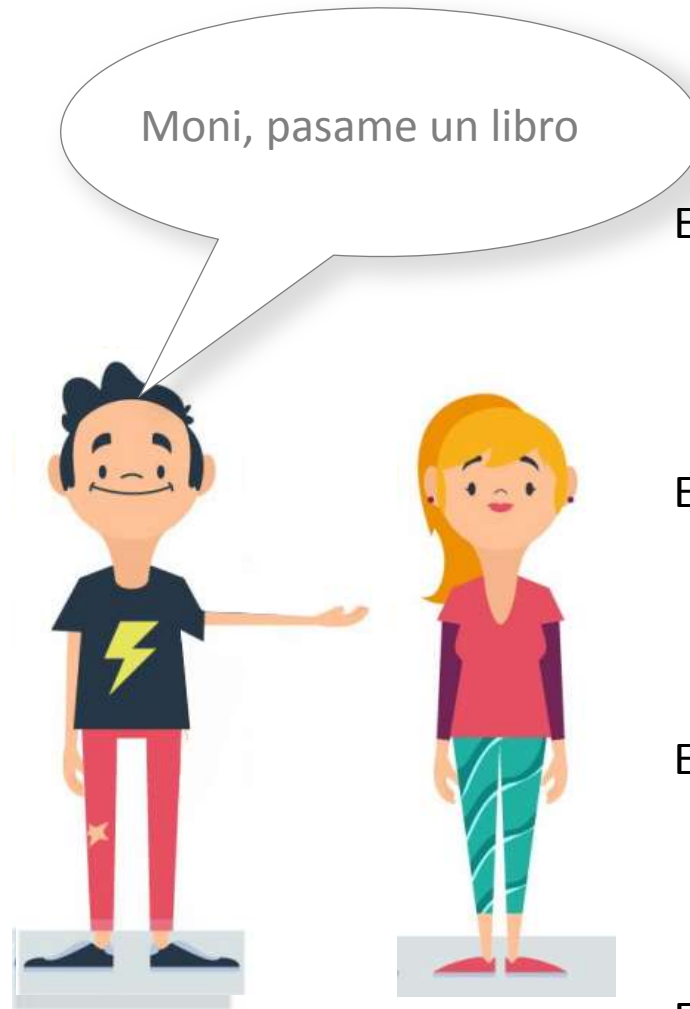
Edad  
Prohibida

Mr Vertigo

# Merge

de la vida real

Estante nuevo



Estante  
1

La Broma

Platero y  
Yo

Rayuela

Estante  
2

Cronicas  
marcianas

Rayuela

Salud y  
Sociedad

Estante  
3

Edad  
Prohibida

Un mundo  
feliz

Salud y  
Sociedad

Viven

Estante  
4

Edad  
Prohibida

Mr Vertigo

Estante nuevo



Estante  
1

La Broma  
Platero y Yo  
Rayuela

Estante  
2

Cronicas  
marcianas  
Rayuela  
Salud y  
Sociedad

Estante  
3

Edad  
Prohibida  
Un mundo  
feliz  
Salud y  
Sociedad  
Viven

Estante  
4

Edad  
Prohibida  
Mr Vertigo

Estante nuevo



Estante  
1

La Broma

Platero y  
Yo

Rayuela

Estante  
2

Cronicas  
marcianas

Rayuela

Salud y  
Sociedad

Estante  
3

Edad  
Prohibida

Un mundo  
feliz

Salud y  
Sociedad

Viven

Estante  
4

Edad  
Prohibida

Mr Vertigo

Busca el mínimo

Estante nuevo



Estante  
1



Estante  
2



Estante  
3

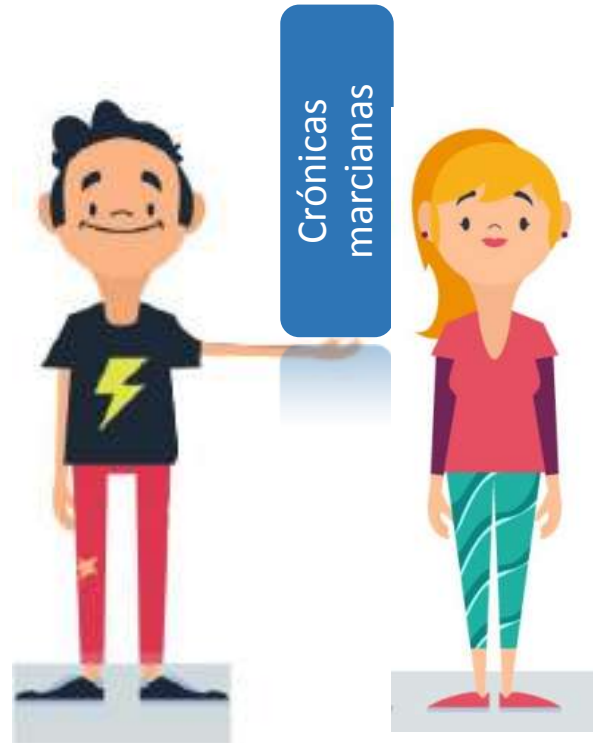


Estante  
4



Estante nuevo

Guarda el Mínimo



Estante  
1



Estante  
2



Estante  
3



Estante  
4



Crónicas  
marcianas

Estante nuevo



Estante  
1

La Broma  
Platero y  
Yo  
Rayuela

Estante  
2

Rayuela  
Salud y  
Sociedad

Estante  
3

Edad  
Prohibida  
Un mundo  
feliz  
Salud y  
Sociedad  
Viven

Estante  
4

Edad  
Prohibida  
Mr Vertigo

Crónicas  
marcianas

Estante nuevo



Estante  
1

La Broma

Platero y  
Yo

Rayuela

Estante  
2

Rayuela

Salud y  
Sociedad

Estante  
3

Edad  
Prohibida

Un mundo  
feliz

Salud y  
Sociedad

Viven

Estante  
4

Edad  
Prohibida

Mr Vertigo



Crónicas  
marcianas

Estante nuevo



Estante  
1

La Broma

Platero y  
Yo

Rayuela

Estante  
2

Rayuela

Salud y  
Sociedad

Estante  
3

Edad  
Prohibida

Un mundo  
feliz

Salud y  
Sociedad

Viven

Estante  
4

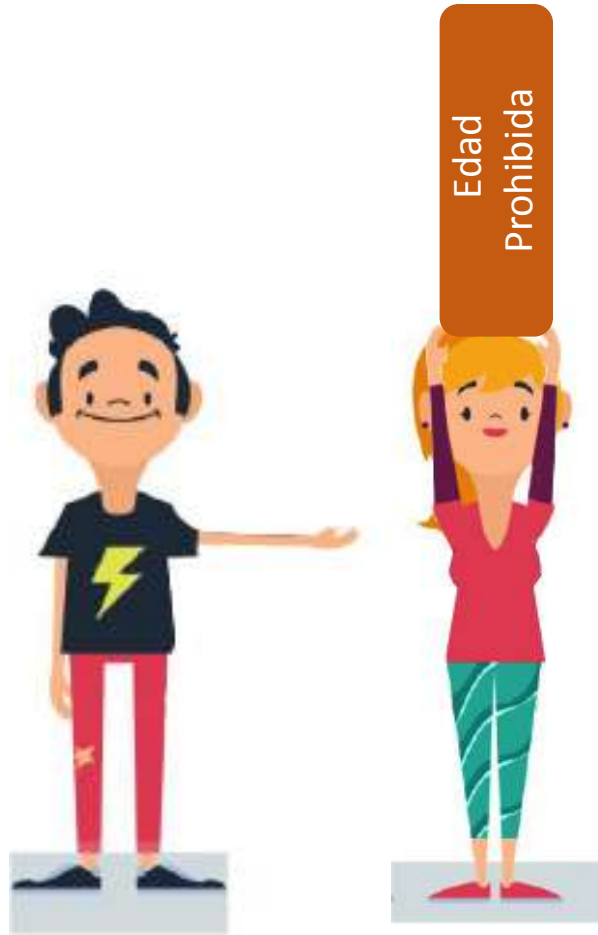
Edad  
Prohibida

Mr Vertigo

Busca el mínimo

Crónicas  
marcianas

Estante nuevo



Estante  
1

La Broma  
Platero y  
Yo  
Rayuela

Estante  
2

Rayuela  
Salud y  
Sociedad

Estante  
3

Edad Prohibida  
Un mundo  
feliz  
Salud y  
Sociedad  
Viven

Estante  
4

Mr Vertigo

Crónicas  
marcianas

Estante nuevo

Guarda el Mínimo



Estante  
1

La Broma

Platero y  
Yo

Rayuela

Estante  
2

Rayuela

Salud y  
Sociedad

Estante  
3

Edad  
Prohibida

Un mundo  
feliz

Salud y  
Sociedad

Viven

Estante  
4

Mr Vertigo



Estante nuevo



Estante  
1



Estante  
2

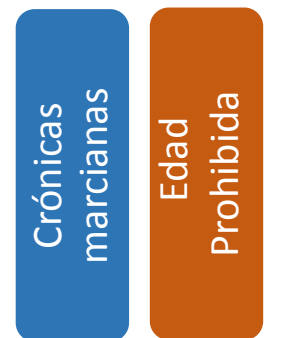


Estante  
3



Estante  
4





Estante nuevo



Estante 1



Estante 2

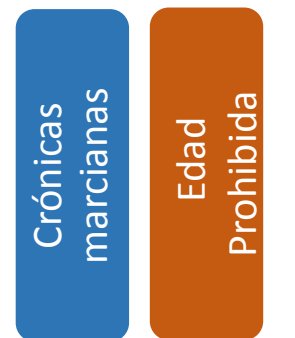


Estante 3



Estante 4





Estante nuevo



Estante  
1

La Broma

Platero y Yo

Rayuela

Estante  
2

Rayuela

Salud y Sociedad

Estante  
3

Edad Prohibida

Un mundo feliz

Salud y Sociedad

Viven

Estante  
4

Mr Vertigo

Busca el mínimo



Estante nuevo



Estante  
1



Estante  
2

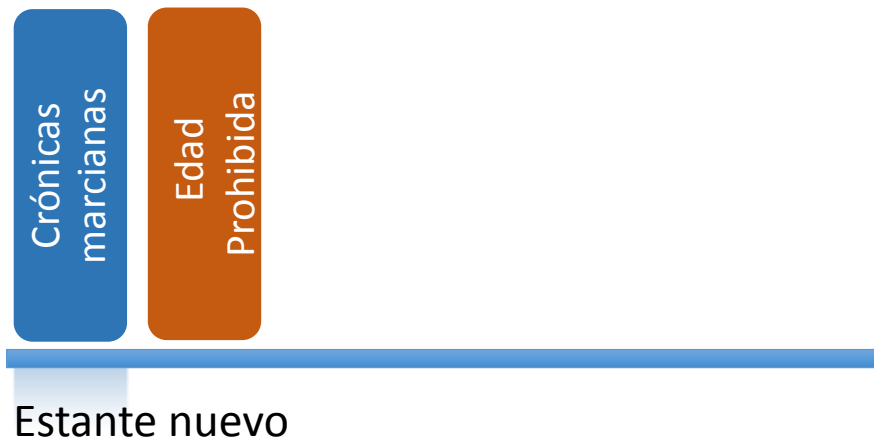


Estante  
3

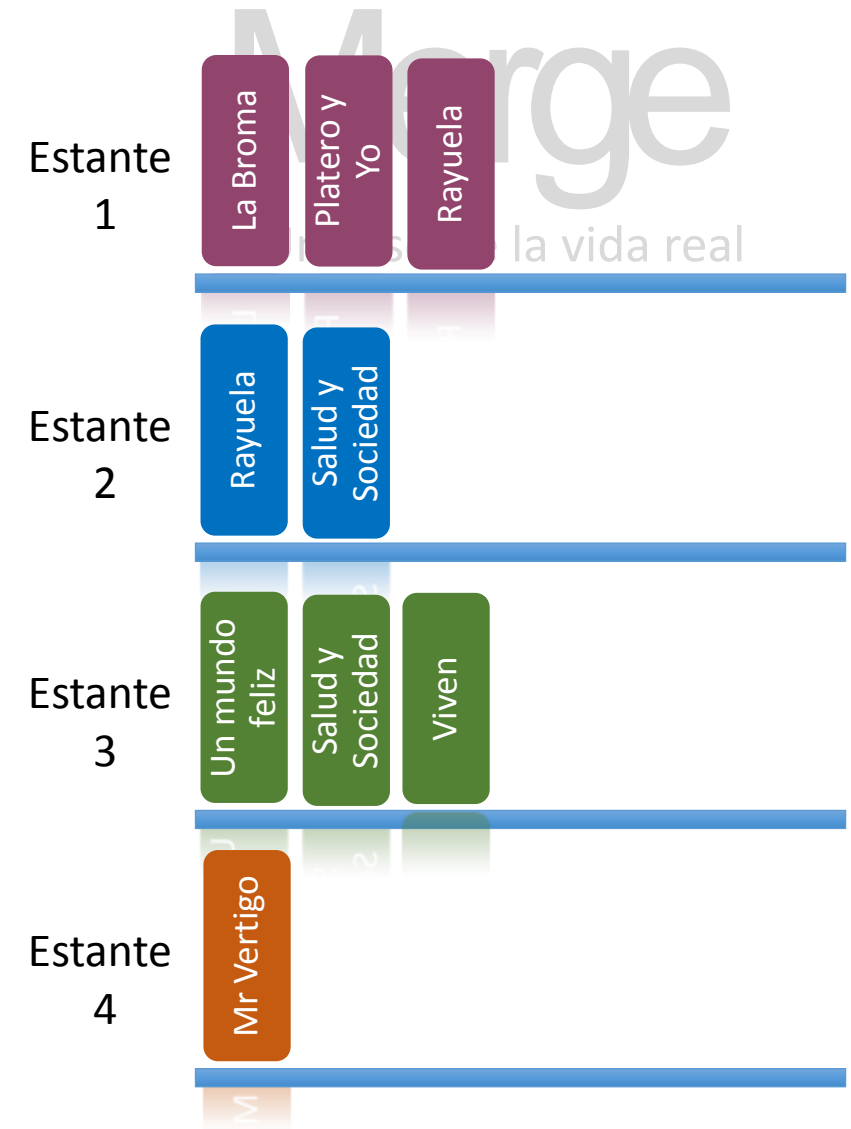


Estante  
4

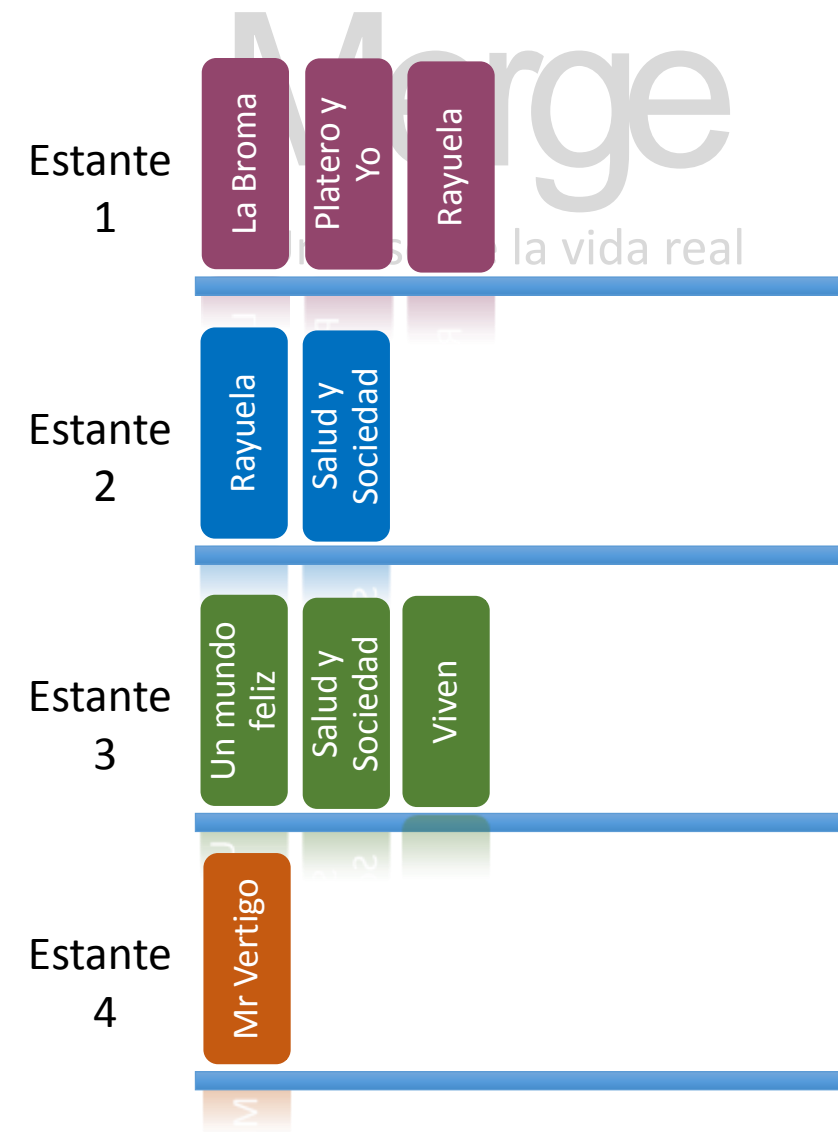
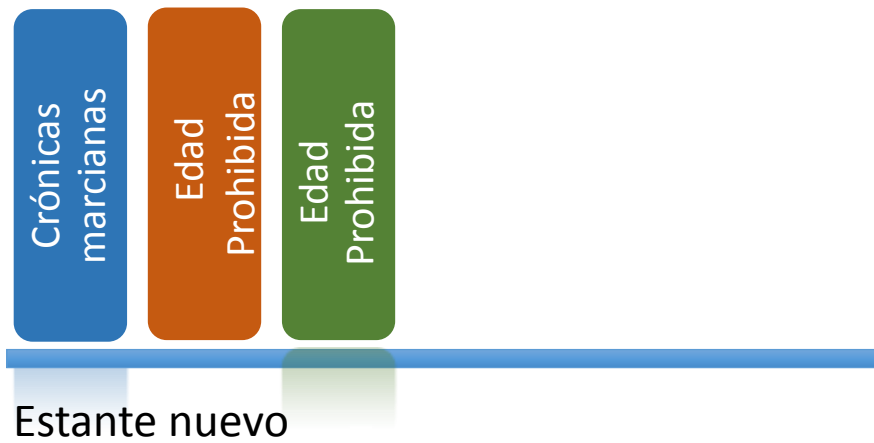


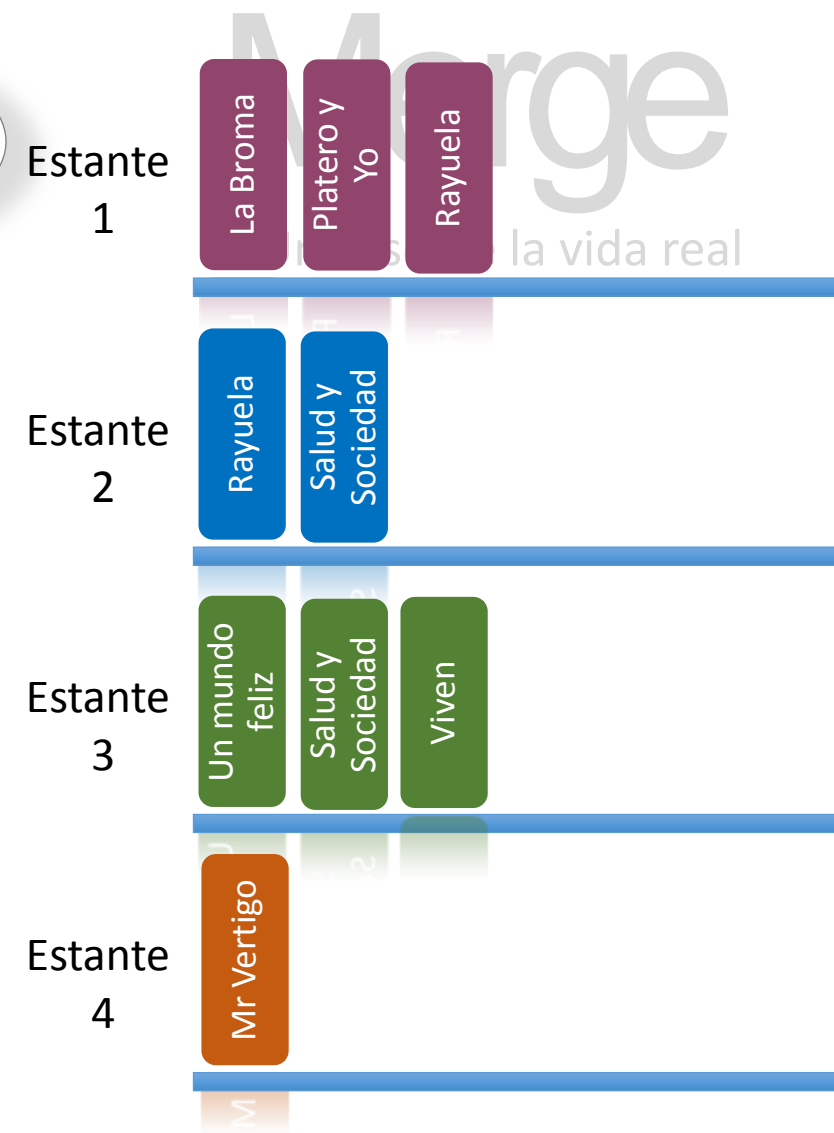
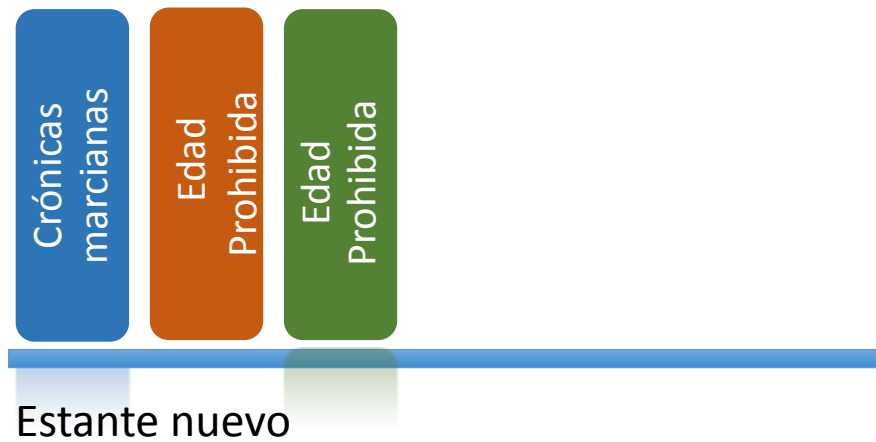


Guarda el Mínimo









A large teal abstract shape, resembling a stylized banner or a folded piece of paper, is positioned in the upper center of the slide. It has a long horizontal top edge and a jagged, folded bottom edge. The text "Siguiendo los mismos pasos..." is written in white on the upper part of this shape.

Siguiendo los mismos pasos...



Estante 1

Platero y Yo

Rayuela

Estante 2

Rayuela

Salud y Sociedad

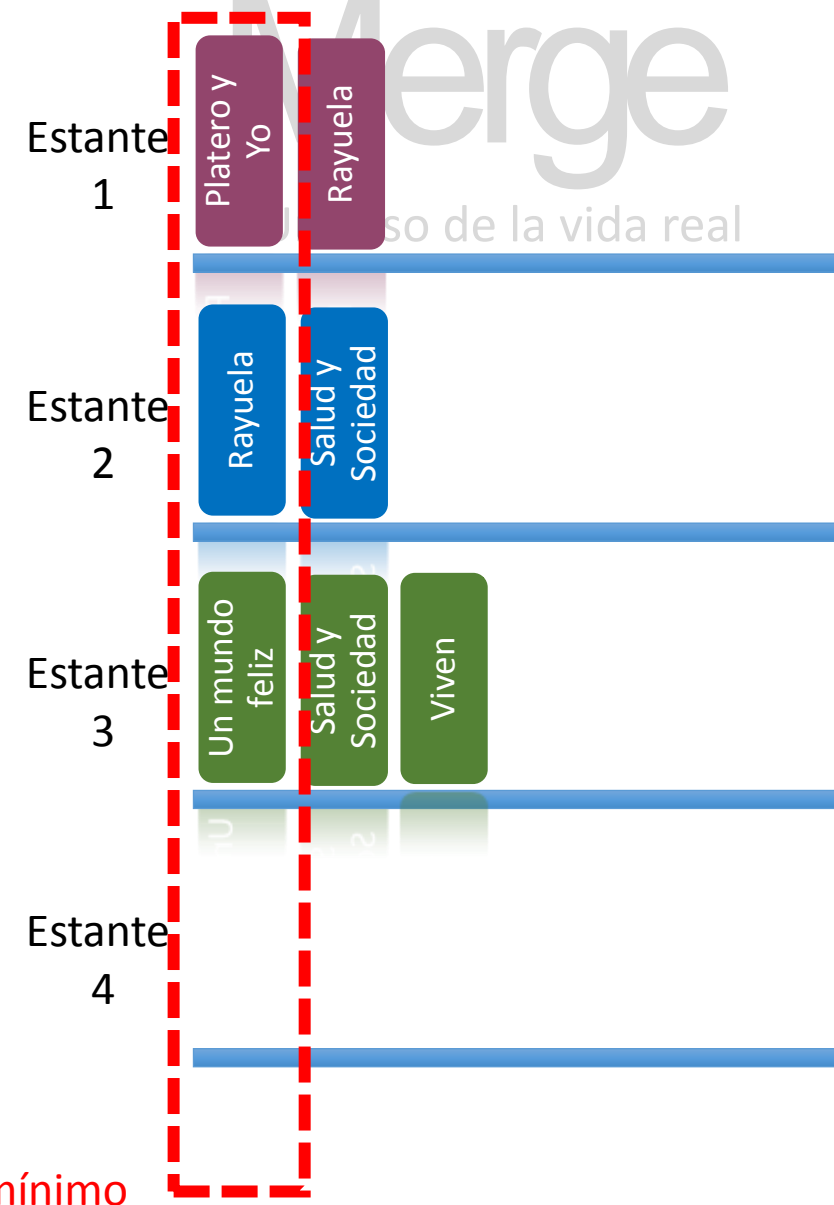
Estante 3

Un mundo feliz

Salud y Sociedad

Viven

Estante 4

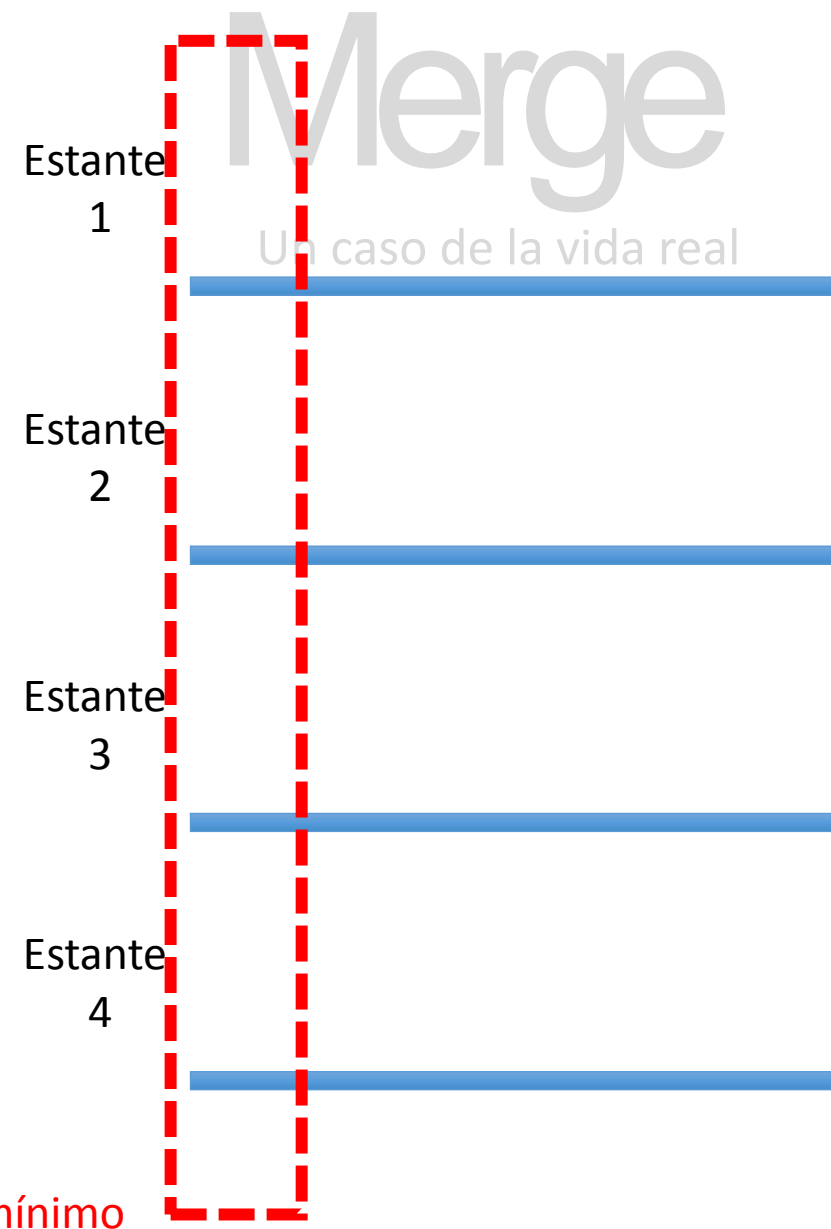


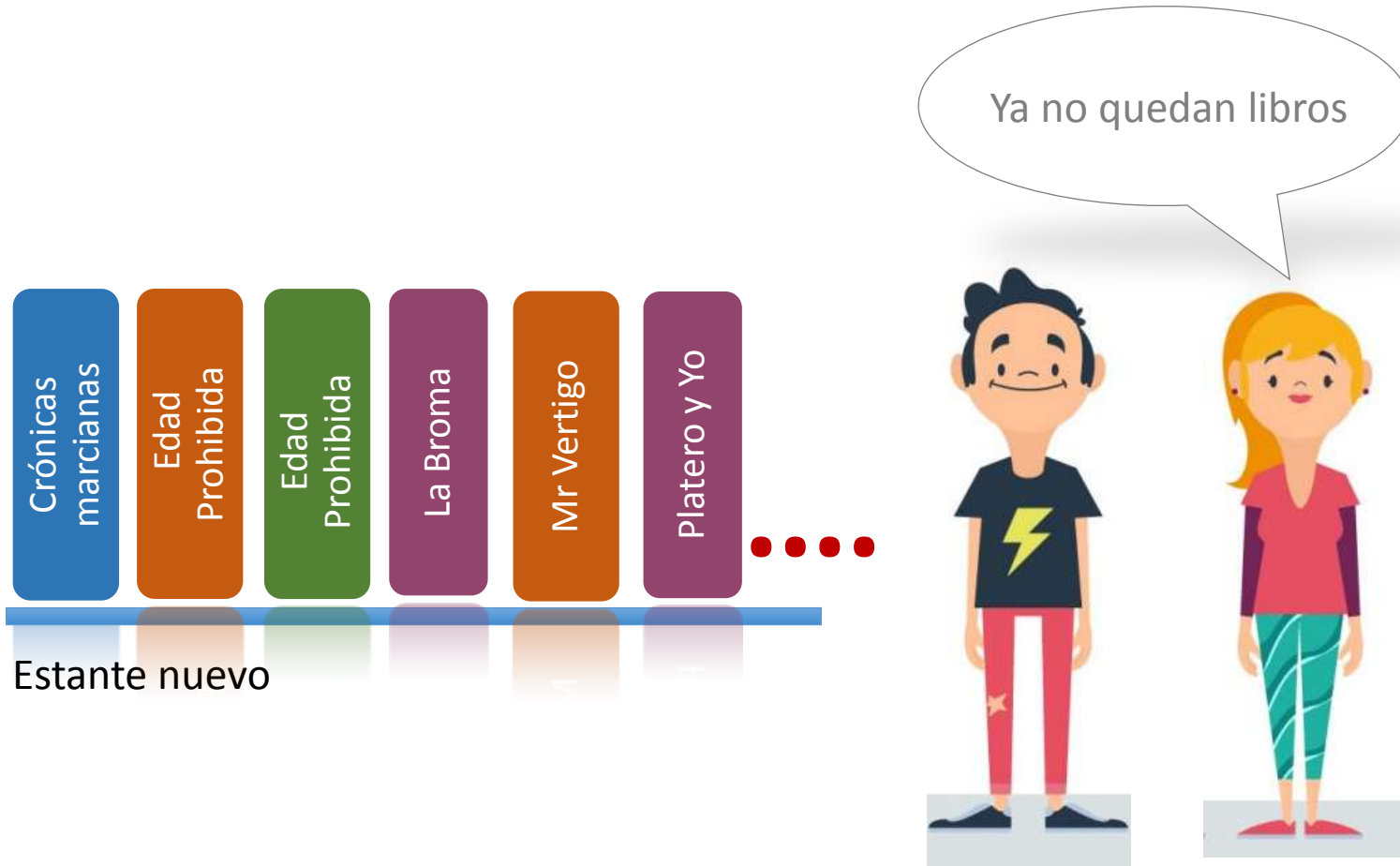
A large teal abstract shape, resembling a stylized banner or a folded piece of paper, is positioned in the upper center of the slide. It has a long horizontal top edge and a jagged, folded bottom edge. The text "Siguiendo los mismos pasos..." is written in white on the upper part of this shape.

Siguiendo los mismos pasos...



Moni, pásame un libro





# Merge

Un caso de la vida real

Estante  
1

Estante  
2

Estante  
3

Estante  
4



# Merge

Un caso de la vida real



Todos los libros  
quedaron  
ordenados  
alfabéticamente  
en el nuevo  
estante



¿Cuándo se  
termina la  
tarea?



Estante  
1

Estante  
2

Estante  
3

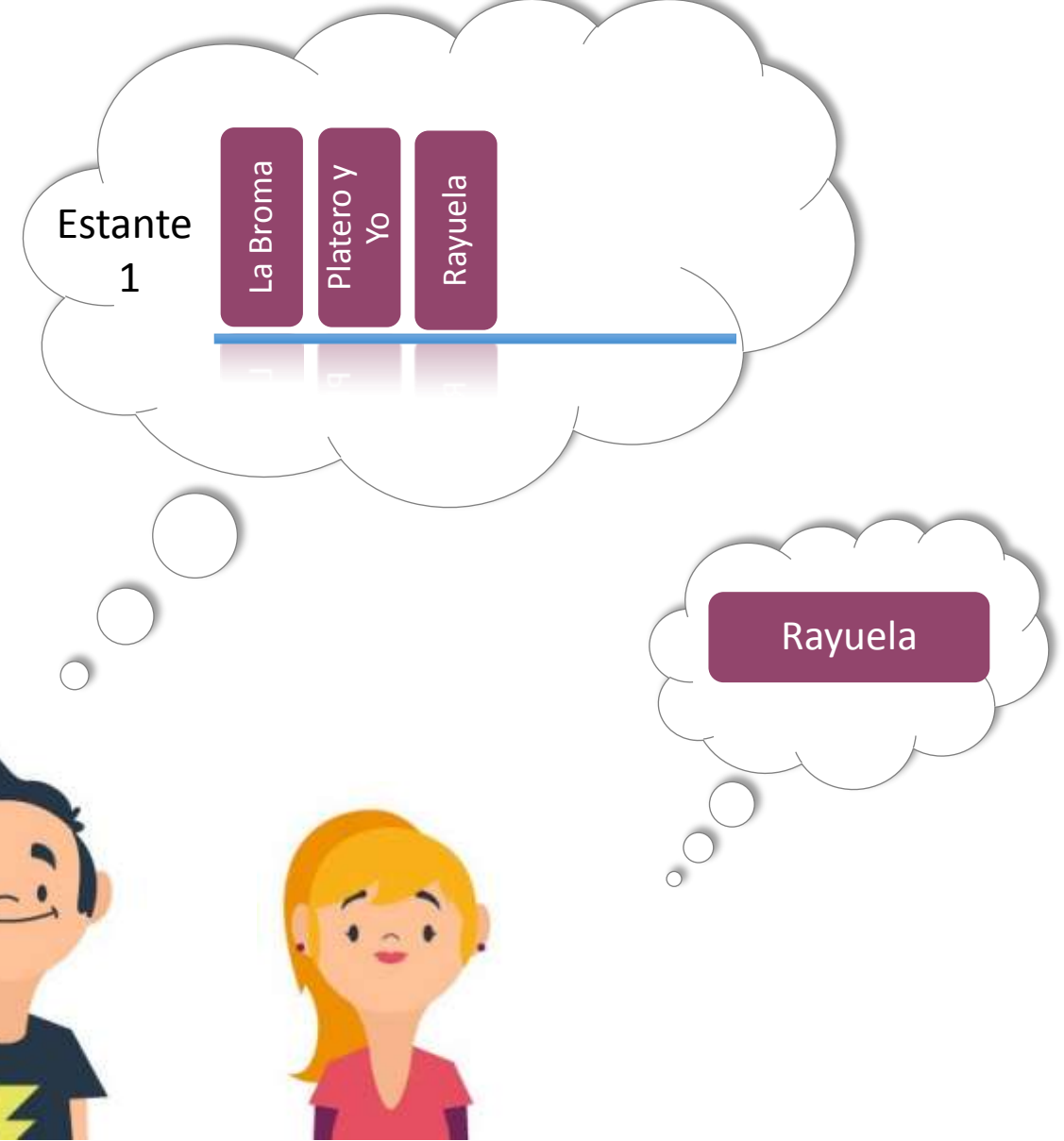
Estante  
4

Si pensamos en la representación  
de los datos ...

Libro → Elemento de la Lista

Estante → Lista

Biblioteca (4 estantes) → Vector de Listas



Si pensamos en el proceso de acomodar los libros ordenados alfabéticamente por título en un único estante (una nueva lista)...

```
MoniBuscaMinimo  
mientras (libro mínimo es válido)  
    PepeGuardaMinimo  
MoniBuscaMinimo
```



## PepeGuardaMinimo

es tan solo un agregar atrás en la lista nueva

```
MoniBuscaMinimo  
mientras (libro mínimo es válido)  
    PepeGuardaMinimo  
    MoniBuscaMinimo
```



**MoniBuscaMinimo**  
requiere un poco mas de  
análisis ...

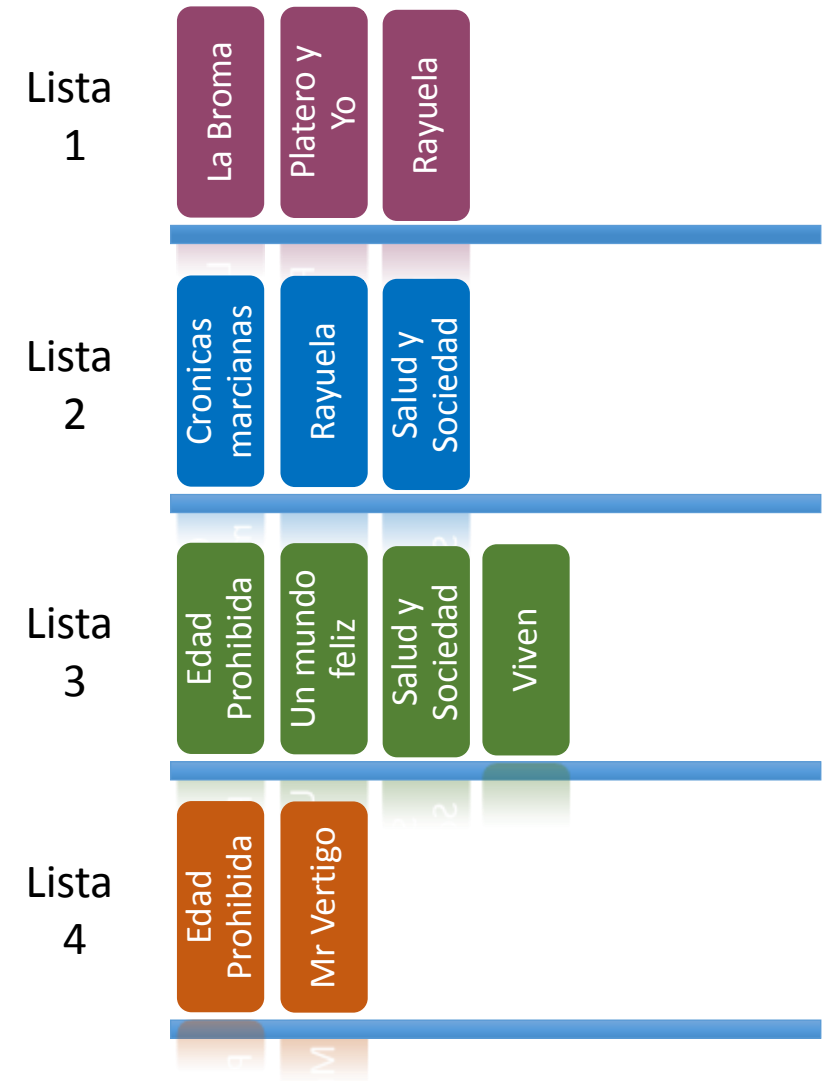
```
MoniBuscaMinimo  
mientras (libro mínimo es válido)  
    PepeGuardaMinimo  
MoniBuscaMinimo
```



**MoniBuscaMinimo** requiere un poco mas de análisis ...

Moni debe devolver el libro con título “menor” (de acuerdo al orden alfabético), para que Pepe lo agregue a la lista nueva.

- Como las listas de libros están ordenadas alfabéticamente por título, sólo debe mirar el primer elemento de cada lista, compararlos y determinar el menor entre estos.



**MoniBuscaMínimo** requiere un poco más de análisis ...

Moni debe devolver el libro con título “menor” (de acuerdo al orden alfabético), para que Pepe lo agregue a la lista nueva.

- Como las listas de libros están ordenadas alfabéticamente por título, sólo debe mirar el primer elemento de cada lista, compararlos y determinar el menor entre estos.



Busca el mínimo

Lista  
1

La Broma

Platero y Yo

Rayuela

Lista  
2

Cronicas  
marcianas

Rayuela

Salud y  
Sociedad

Lista  
3

Edad  
Prohibida

Un mundo  
feliz

Salud y  
Sociedad

Viven

Lista  
4

Edad  
Prohibida

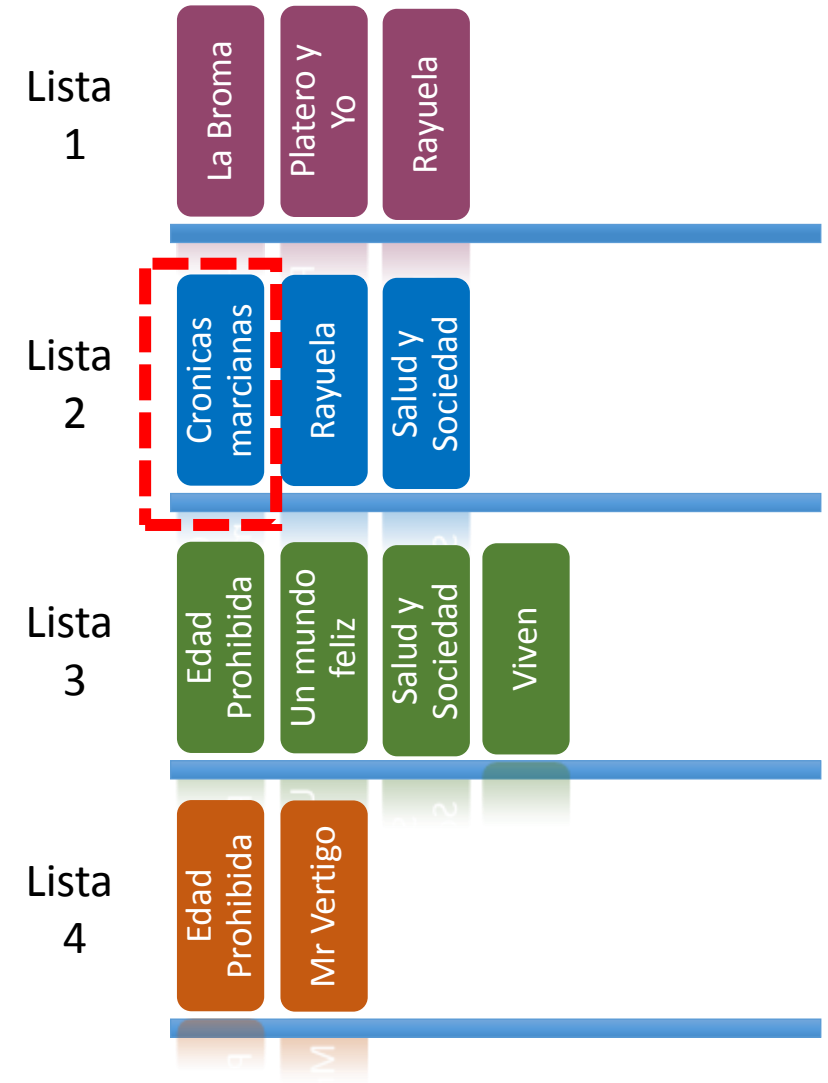
Mr Vertigo



MoniBuscaMinimo requiere un poco mas de análisis ...

Moni debe devolver el libro con título “menor” (de acuerdo al orden alfabético), para que Pepe lo agregue a la lista nueva.

En este caso, el menor entre “La Broma”, “Crónicas Marcianas”, “Edad Prohibida” y “Edad Prohibida” es **“Crónicas Marcianas”**



**MoniBuscaMinimo** requiere un poco mas de análisis ...

Moni debe devolver el libro con título “menor” (de acuerdo al orden alfabético), para que Pepe lo agregue a la lista nueva.

Obtenido el menor, lo sacará de la lista y “Rayuela” quedará como el próximo elemento de la lista que deberá compararse.

Este se quitará de la lista

Este pasará a ser el próximo elemento a comparar



Lista 1

La Broma  
Platero y Yo  
Rayuela

Lista 2

Cronicas marcianas  
Rayuela  
Salud y Sociedad

Lista 3

Edad Prohibida  
Un mundo feliz  
Salud y Sociedad  
Viven

Lista 4

Edad Prohibida  
Mr Vertigo

### MoniBuscaMinimo:

```
libroMinimo := valor_muy_alto  
for i:= 1 to total_estantes do  
    if ( estante_i noVacío ) then  
        if (libroEnEstante_i < libroMinimo ) then libroMinimo := libroEnEstante_i  
                                estanteMinimo:= i  
if (libroMinimo <> valor_muy_alto) then estante_iPasaASiguienteLibro
```



### MoniBuscaMinimo:

```
libroMinimo := valor_muy_alto  
  
for i:= 1 to total_estantes do  
    if ( estante_i noVacío ) then  
        if (libroEnEstante_i < libroMinimo ) then libroMinimo := libroEnEstante_i  
                                                    estanteMinimo:= i  
  
if (libroMinimo <> valor_muy_alto) then estante_iPasaASiguienteLibro
```



¿Cómo quedaría el arreglo cuando todos los estantes están vacíos?

¿Qué devuelve en libroMinimo si todos los estantes están vacíos?

# Actividad en máquina

## Objetivo de la actividad 1:

Realizar un **merge** de 4 listas de libros.



# Actividad en máquina

**Tareas:** Implemente el **ProgramaMerge4Listas** que:

- Invoque al módulo **CrearVectorListas** que genera un vector de 4 listas de libros, utilizando el módulo *CrearListaOrdenada* del **ProgramaListas** (clase 0). Cada libro se representa con un código ISBN entre 0 y 10 (usar función Random).
- Invoque al módulo **ImprimirVectorListas** que imprime en pantalla el contenido de cada lista del vector, utilizando el módulo *ImprimirLista* del **ProgramaListas** (clase 0).



# Actividad en máquina

**Tareas:** Implemente el **ProgramaMerge4Listas** que:

c. Invoque al módulo **Merge4** que recibe el vector generado en a) y devuelve una nueva lista con los elementos de las 4 listas. Este módulo requiere:

1. Implementar el módulo **DeterminarMinimo** que retorne el valor mínimo de un vector de listas ordenadas de forma creciente. El valor retornado se debe quitar de la lista original.
2. Implementar el módulo **AgregarAlFinal**. Para ello revisar el módulo **CrearListaAgregarAtras** del **ProgramaListas** de la clase 0.



# Actividad en máquina

## **Objetivo de la actividad 2:**

Retomar la Actividad 5 de la Clase 0 y obtener una única lista de empleados ordenada por apellido que contenga apellido, año de ingreso y categoría de cada uno de los empleados.