

Manual de usuario

1.-Principios básicos del funcionamiento:

El funcionamiento principal es la conversión de un archivo Excel json, el algoritmo en primera instancia recorrerá solo el ROW 1 (primera fila) y se moverá de columna en columna hasta el final,

El rendimiento no se ve afectado por esto ya que solo leerá las celdas utilizadas por el usuario, las celdas nunca modificadas serán totalmente ignoradas.

2.-Tipos de elementos en el Excel:

→Array (elementos, objetos).

→Elementos

→Objetos,

2.1-Arrays:

Estos elementos se colorearán en azul cielo al ser reconocidos por el algoritmo.

→Condiciones para crear un arreglo.←

→Nombre del arreglo en negritas.

→Celda derecha e inferior vacías.

Los elementos del arreglo serán especificados en la columna diagonal inferior derecha del mismo; el algoritmo leerá desde ese punto hasta abajo.

Mod1		Mod1	
	tema1		tema1
	tema2		tema2

El algoritmo leerá todas las celdas hacia abajo hasta llegar al final, si existe otro arreglo anidado debajo, este detendrá su registro de datos para evitar tomar datos pertenecientes al arreglo inferior.

Modulos		
	Mod1	
		tema1
		tema2
	Mod2	
		tema1
		tema2

en gris las celdas recorridas en las que no encuentro nada

Como se observa en la ilustración anterior incluso se pueden anidar arreglos dentro de arreglos de manera infinita gracias a la recursividad y además no recorrida la celda debajo de "tema2" porque reconoció que había otro arreglo debajo (El algoritmo generara automáticamente un arreglo con la ruta del archivo).

```

"Modulos":[
  {"Mod1":[
    "tema1",
    "tema2"
  ]},
  {"Rut3":[
    "temas/tema1.html", "temas/tema2.html"
  ]},
  {"Rut4":[
    "temas/tema3.html", "temas/tema4.html"
  ]}
]

```

El json resultante de esto seria un arreglo anidando dos arreglos, ambos del tipo elemento.

→Existen 2 tipos de arreglos←

→Arreglos de elementos

→Strings

→Ints

→Arreglos de objetos

→Arreglos de elementos←

Estos arreglos se dividen a su vez en arreglos de String o de Int, no existe gran diferencia entre ambos ya que el algoritmo detecta si se trata de un numero o una cadena y lo convertirá como tal.

Mod1	
	tema1
	tema2

Trak	
0	
0	
0	
0	

La conversión seria de la siguiente manera

```
"Modulos":[
  {"Mod1":[
    "tema1",
    "tema2"
  ]},
  {"Mod1":[]}
```

```
"Trak":[
  0,
  0,
  0,
  0
]
```

→Arreglos de objetos←

Estos se declaran exactamente igual a los anteriores, con la diferencia de que contendrán objetos alineados de manera horizontal, al colocar más objetos debajo serán tomados para el arreglo también.

Evaluaciones					
	ID	MaxIntentos	IntentoActual	CalMinima	CalActual
	1	3	0	80	0
	ID	MaxIntentos	IntentoActual	CalMinima	CalActual
	2	3	0	80	0
	ID	MaxIntentos	IntentoActual	CalMinima	CalActual
	3	3	0	80	0

El json resultante de esto seria

```
"Evaluaciones":[
  {"ID":1,"MaxIntentos":3,"IntentoActual":0,"CalMinima":80,"CalActual":0},
  {"ID":2,"MaxIntentos":3,"IntentoActual":0,"CalMinima":80,"CalActual":0},
  {"ID":3,"MaxIntentos":3,"IntentoActual":0,"CalMinima":80,"CalActual":0}
],
```

2.2.-Elementos

Los elementos son simplemente los elementos de un arreglo, se colorean en naranja cuando son reconocidos por el algoritmo y existen en dos tipos, los string y los int, el algoritmo es completamente capaz de diferenciar entre ambos y convertir todo correctamente a JSON.

Mod1	
	tema1
	tema2

Trak	
0	
0	
0	
0	

2.3.-objetos

Los objetos son llamados así cuando una serie de elementos de tipo var, están dentro de un arreglo, la diferencia es muy simple

Var→

Es simplemente el nombre de la variable y su valor

NombreCurso
nombre

```
"NombreCurso": "nombre"
```

Objeto→

Un objeto es el conjunto de Vars dentro de un arreglo, la razón por la cual son llamados objetos se encuentra en el JSON ya que se genera dentro de dos llaves {} lo que le da el status de objeto.

Evaluaciones					
	ID	MaxIntentos	IntentoActual	CalMinima	CalActual
	1	3	0	80	0

```
{"ID":1,"MaxIntentos":3,"IntentoActual":0,"CalMinima":80,"CalActual":0},
```

3.-Consejos para el correcto funcionamiento:

En base a la distribución de la información en las celdas el algoritmo leerá la información y la convertirá de acuerdo a las ordenes que se le dieron.

Modulos			Evaluaciones						NombreCurso	
	Mod1			ID	MaxIntentos	IntentoActual	CalMinima	CalActual	nombre	
		tema1		1	3	0	80	0		
		tema2								
	Mod2									
		tema1								
		tema2								

→Requisitos←

→la cadena principal del json se distribuye sobre la primera fila del Excel siempre.

Modulos			Evaluaciones						NombreCurso	
	Mod1			ID	MaxIntentos	IntentoActual	CalMinima	CalActual	nombre	
		tema1		1	3	0	80	0		
		tema2								
	Mod2									
		tema1								
		tema2								

→Nombres de arrays y objetos siempre van en **negritas**.

Evaluaciones						NombreCurso
	ID	MaxIntentos	IntentoActual	CalMinima	CalActual	nombre
	1	3	0	80	0	

- Existen 2 tipos de arrays, es preferible no combinarlos entre si, pero de ser así lo ideal sería que la sección de objetos este en la parte superior del arreglo y la sección de elementos en la parte inferior (de acuerdo a la distribución que se le da en el Excel).

Bien (No recomendado pero aceptable)→

[illegible]

Mal (Nada recomendado)→

Modulos			Evaluaciones						NombreCurso
	Mod1			ID	MaxIntentos	IntentoActual	CalMinima	CalActual	nombre
		tema1		1	3	0	80	0	
		tema2							
	Mod2			0					
		tema1		0					
		tema2		0					
				0					
		ID	MaxIntentos	IntentoActual	CalMinima	CalActual			
		1	3	0	80	0			
		ID	MaxIntentos	IntentoActual	CalMinima	CalActual			
		1	3	0	80	0			

Realmente no debería haber problema al hacer esto, pero en algunos casos el algoritmo se puede confundir y crear un json incorrecto, por eso es recomendable hacerlo como en la primera opción mostrada y nunca como en la segunda.

→ Datos del desarrollo del proyecto ←

Idea Original → Adrián Ruvalcaba García

Programador → Jesús Martín Gómez Salazar