

# MyTable Doc

## columns欄位定義

屬性	類型	預設值	約束	說明
key	String		required	欄位識別鍵
header	String	column.key		名稱
displayable	Boolean	true		是否對使用者顯示，FK 應設為false
value	Function	(item) => item[column.key]		取值函式，可用於巢狀物件欄位
defaultValue	Object	null		預設值
format	Function, Array	(value) => value	格式化函式原型： (value) => value	格式化函式/格式化函式陣列
nullText	String	'..'		當值為null或空字串時的 顯示文字
sort	Function	null		自訂排序函式
sortable	Boolean	true		可否排序
thClass	String, Object, Array	null		表頭樣式類別
thStyle	Object, Array	null		表頭樣式
tdClass	String, Object, Array	null		儲存格樣式類別
tdStyle	Object, Array	null		儲存格樣式
exportable	Boolean	true		可否匯出
valueToExport	Function	(item) => item[column.key]		匯出取值函式 ex: (role) => role.permissions?.map(p => p.title).join("\n") ?? "
editable	Boolean	true		可否編輯
valueToEdit	Function	(item) => item[column.key]		編輯取值函式 ex: (role) => role.permissions?.map(p => p.id) ?? []
dataType	Function	String	String, Number, Boolean, Array, Date	資料類型
control	Object	(見control輸入控制 項資訊)		輸入控制項資訊 (見 control輸入控制項資訊)

屬性	類型	預設值	約束	說明
validators	Array	[]	驗證函式原型： (value) => true    '錯誤訊息' 或需要取得編輯項目：(value, item) => true    '錯誤訊息'	驗證函式陣列
validationMode	Object	{ greedy: false, lazy: false }		驗證模式 greedy：是否為貪婪模式(在第一個驗證函式不通過時仍繼續驗證) lazy：懶驗證模式(僅在儲存時驗證)
errors	Array	[]		錯誤訊息陣列
errorsSeparator	String	','		錯誤訊息分隔符號

## control輸入控制項資訊

屬性	類型	預設值	約束	說明
type	String	(見control.type輸入控制項類型)	'input', 'select', 'radiobuttonlist', 'checkbox', 'checkboxlist', 'textarea'	輸入控制項類型 (見control.type輸入控制項類型)
cssClass	String, Object, Array	null		輸入控制項樣式類別
style	Object, Array	null		輸入控制項樣式
dataSource	Array	[]	資料來源項目範例： [{ value: 'test1', text: '測試1' }, { value: 'test2', text: '測試2', disabled: true }]	輸入控制項資料來源(適用於select, radiobuttonlist, checkboxlist)
dataValueField	String	'value'		輸入控制項資料來源的值欄位(搭配dataSource使用)
dataTextField	String	'text'		輸入控制項資料來源的文字欄位(搭配dataSource使用)
showEmptyOption	Boolean	true		輸入控制項是否顯示空白選項(適用於select)
emptyOptionValue	Object	"		輸入控制項空白選項值(適用於select)
emptyOptionText	String	'-請選擇-'		輸入控制項空白選項文字(適用於select)

屬性	類型	預設值	約束	說明
inline	Boolean	false		輸入控制項是否為水平排列(適用於radiobuttonlist, checkboxlist)
trueValue	Boolean, String, Number	true		輸入控制項的真值(適用於checkbox)
falseValue	Boolean, String, Number	false		輸入控制項的偽值(適用於checkbox)

## control.type輸入控制項類型

輸入控制項類型	適用資料類型	預設輸入控制項資料類型
input	String, Number, Date	String, Number, Date
select	String, Number	
radiobuttonlist	String, Number	
checkbox	Boolean, String, Number	Boolean
checkboxlist	Array	Array
textarea	String	

```
// const defaultControlTypes = { [String]: 'input', [Number]: 'input', [Boolean]: 'checkbox', [Array]: 'checkboxlist', [Date]: 'input' }
```

## props

名稱及修飾詞	類型	預設值	約束	說明
items.sync	Array		required: true	資料來源
columns.sync	Array	(見columns欄位定義)	required: true validator: (columns) => columns.every((column) => column.key)	欄位定義 (見columns欄位定義)
visibleColumnKeys.sync	Array	columns.filter(col => col.displayable).map(col => col.key)		目前顯示欄位的識別鍵陣列
rowKey	String	'id'		資料列識別鍵
selected.sync	Array	[]		已勾選的資料列識別鍵
searchTerm.sync	String	"		搜尋關鍵詞
searchFilter.sync	Object	columns.filter(col => col.displayable).reduce((prev, curr) => ({...prev, [curr.key]: null}), {})		搜尋篩選欄位物件 ex: { columnKeyA: 'keyASearchTerm', columnKeyB: 'keyBSearchTerm' }

名稱及修飾詞	類型	預設值	約束	說明
filtering.sync	Object	{ ignoreCase: true, logicOperation: 'or' }		篩選資訊物件 ignoreCase: 搜尋比對時忽略大小寫 logicOperation: 所有搜尋條件的邏輯運算模式(or, and)
sorting.sync	Object	{ key: '', direction: 'asc' }		排序資訊物件 key: 排序欄位屬性鍵 direction: 排序方向(asc, desc)
paging.sync	Object	{ page: 1, itemsPerPage: 100 }		分頁資訊物件 page: 頁碼 itemsPerPage: 每頁幾筆(小於等於0者不分頁)
paginator.sync	Object	{ showFirst: false, showPrev: true, showPrevGrp: true, showPages: true, showNextGrp: true, showNext: true, showLast: false, pageLinks: 5 }		分頁器資訊物件 showFirst: 顯示首頁連結 showPrev: 顯示上一頁連結 showPrevGrp: 顯示上一頁群連結 showPages: 顯示頁碼連結 showNextGrp: 顯示下一頁群連結 showNext: 顯示下一頁連結 showLast: 顯示末頁連結 pageLinks: 頁碼連結數量
itemsPerPageOptions.sync	Array	[10, 25, 50, 100]	(options) => options.every((option) => !isNaN(option))	每頁幾筆下拉選單選項
serverTotalItems	Number	undefined		伺服器資料總筆數，有提供時以伺服器端搜尋、排序、分頁(request 事件處理)
showToolBar	Boolean	true		是否顯示工具列(新增、搜尋)
showCreate	Boolean	true		是否顯示新增按鈕
showSearch	Boolean	true		是否顯示搜尋框
showMenubar	Boolean	true		是否顯示選單列(刪除勾選、匯出、每頁幾筆、顯示欄位)

名稱及修飾詞	類型	預設值	約束	說明
showDestroy	Boolean	true		是否顯示刪除勾選按鈕
showExport	Boolean	true		是否顯示匯出下拉選單
showItemsPerPage	Boolean	true		是否顯示每頁幾筆下拉選單
showVisibleColumns	Boolean	true		是否顯示顯示欄位下拉選單
showPagebar	Boolean	true		是否顯示分頁列 (分頁器、分頁資訊)
showPaginator	Boolean	true		是否顯示分頁器
showPageInfo	Boolean	true		是否顯示分頁資訊
pagebarPosition	String	'top'	'top', 'bottom'	分頁列位置
toolbarClass	String, Object, Array	undefined		工具列樣式類別
toolbarStyle	Object, Array	undefined		工具列樣式
menubarClass	String, Object, Array	undefined		選單列樣式類別
menubarStyle	Object, Array	undefined		選單列樣式
pagebarClass	String, Object, Array	undefined		分頁列樣式類別
pagebarStyle	Object, Array	undefined		分頁列樣式
exportExcel	Function	undefined		自訂匯出Excel函式
exportPdf	Function	undefined		自訂匯出Pdf函式
exportName	String	'export_data'		自訂匯出預設檔案名稱
showSelection	Boolean	true		是否顯示資料列選取方塊
showActions	Boolean	true		是否顯示資料列操作動作儲存格
showRead	Boolean	true		是否顯示資料列檢視按鈕
showUpdate	Boolean	true		是否顯示資料列編輯按鈕
showDelete	Boolean	true		是否顯示資料列刪除按鈕
emptyText	String	'查無資料！'		查無資料訊息文字
inlineEditItem.sync	Object	null		行內編輯項目

## events

事件	處理函式原型	說明
request	<pre>@request="{ { searchTerm, searchFilter, filtering, sorting, paging } } =&gt; onRequest({ searchTerm, searchFilter, filtering, sorting, paging })"</pre> <p>事件參數： searchTerm: 搜尋關鍵詞 searchFilter: 搜尋篩選欄位物件 filtering: 篩選資訊物件 sorting: 排序資訊物件 paging: 分頁資訊物件</p>	<p>觸發伺服器請求時發出 範例：</p> <pre>async onRequest({ searchTerm, searchFilter, filtering, sorting, paging }) {   console.log('onRequest');   const { data } = await list({ params: { searchTerm, searchFilter, filteringIgnoreCase: filtering.ignoreCase, sortingKey: sorting.key, sortingDirection: sorting.direction, pagingPage: paging.page, pagingItemsPerPage: paging.itemsPerPage } });   //console.log(data);   this.serverTotalItems = data.serverTotalItems;   this.items = data.items;   //this.searchTerm = searchTerm;   //this.searchFilter = searchFilter;   //this.filtering = filtering;   //this.sorting = sorting;   //this.paging = paging; }</pre>
create-item	<pre>@create-item="(item, itemIndex) =&gt; onCreateItem(item, itemIndex)"</pre> <p>事件參數： item: 資料來源項目 itemIndex: 新增資料來源的索引位置</p>	<p>於新增資料來源項目時觸發 ex:</p> <pre>async onCreateItem(item, index) {   this.items.splice(index, 0, item);   this.inlineEditItem = Object.assign({}, item); }</pre>
destroy-items	<pre>@destroy-items="(selected) =&gt; onDestroyItems(selected)"</pre> <p>事件參數： selected: 已勾選的資料列識別鍵</p>	於刪除勾選時觸發
change-page	<pre>@change-page="(page) =&gt; onChangePage(page)"</pre> <p>事件參數： page: 頁碼</p>	於分頁換頁時觸發
validate-item-failed	<pre>@validate-item-failed="{ { errors, item } } =&gt; onValidateItemFailed({ errors, item })"</pre> <p>事件參數： { errors, item } errors: 錯誤資訊物件({ key: (欄位識別鍵), errors: (錯誤訊息陣列), errMsg: (以分隔符號串接之錯誤訊息) }) item: 行內編輯項目</p>	於驗證行內編輯項目失敗時觸發
save-create-item	<pre>@save-create-item="(item) =&gt; onSaveCreateItem(item)"</pre> <p>事件參數： item: 行內編輯項目</p>	於儲存新增行內編輯項目時觸發(已通過驗證)
save-update-item	<pre>@save-update-item="(item) =&gt; onSaveUpdateItem(item)"</pre> <p>事件參數： item: 行內編輯項目</p>	於儲存編輯行內編輯項目時觸發(已通過驗證)

事件	處理函式原型	說明
cancel-item	@cancel-item="(item) => onCancelItem(item)" 事件參數： item: 行內編輯項目	於取消行內編輯項目時觸發
read-item	@read-item="(item) => onReadItem(item)" 事件參數： item: 資料來源項目	於檢視資料來源項目時觸發
update-item	@update-item="(item) => onUpdateItem(item)" 事件參數： item: 資料來源項目	於編輯資料來源項目時觸發
delete-item	@delete-item="(item) => onDeleteItem(item)" 事件參數： item: 資料來源項目	於刪除資料來源項目時觸發

## scoped slots

名稱	slotProps	說明
toolbar-left-before	vmData: \$data	工具列左側前方的插槽
toolbar-left-after	vmData: \$data	工具列左側後方的插槽
toolbar-right-before	vmData: \$data	工具列右側前方的插槽
toolbar-right-after	vmData: \$data	工具列右側後方的插槽
menubar-left-before	vmData: \$data	選單列左側前方的插槽
menubar-left-after	vmData: \$data	選單列左側後方的插槽
menubar-right-before	vmData: \$data	選單列右側前方的插槽
menubar-right-after	vmData: \$data	選單列右側後方的插槽
pagebar-left-before	vmData: \$data	分頁列左側前方的插槽
pagebar-left-after	vmData: \$data	分頁列左側後方的插槽
pagebar-right-before	vmData: \$data	分頁列右側前方的插槽
pagebar-right-after	vmData: \$data	分頁列右側後方的插槽
edit-[columnKey]	item: 行內編輯項目 column: 欄位定義	欄位的編輯模式插槽
item-[columnKey]	item: 資料來源項目 column: 欄位定義	欄位的顯示模式插槽

## public methods

方法函式原型	參數	回傳值	說明
getExportParams		匯出參數 { exportItems, exportColumns }	取得匯出參數 exportItems: 匯出資料來源 exportColumns: 匯出欄位({ key, header })
doExportExcel			匯出Excel
doExportPdf			匯出Pdf

方法函式原型	參數	回傳值	說明
getProcessedItems		已處理資料來源	取得已處理資料來源(已篩選、已排序)
getPagedItems		已處理並分頁後的資料來源	已處理並分頁後的資料來源(已篩選、已排序、已分頁)
validateInlineEditItem		是否通過驗證	驗證行內編輯項目
clearErrors			清除錯誤訊息
getErrors		錯誤資訊物件 { key, errors, errMsg } key: (欄位識別鍵) errors: (錯誤訊息陣列) errMsg: (以分隔符號串接之錯誤訊息)	取得錯誤資訊物件