

Apertura scheda lista classe (gwClassList)

La lista di dettaglio di classe (gwClassList) è una griglia evoluta contenente i record che fanno riferimento ad una data gwClass, con capacità di ordinare, filtrare, etc..

La lista viene aperta in un tab di tipo gwClassList, descritto [qui](#).

openGwClassListTab()

Per aprire la lista di dettaglio di classe è esposta l'apposita function openGwClassListTab().

Parametri

- **gwClassName** String, required, name of gwClass
- **title** String, optional, default ""
- **options** Object, optional, default null. Contains all Additional parameters

Parametri opzionali dentro options

- **filters** Object [], optional, default null. Allow to show only elements matching filters. This filter List, unlikely *staticFilters*, can be removed by the user through the gwClassList UI. Example:

```
[
  {
    condition: 'AND',
    columnName: 'id_item',
    attributeGwid: 1,
    operator: 'IN',
    filterType: 'INTEGER',
    value: [1, 2, 3]
  }
]
```

- **staticFilters** Object [], optional, default null. Allow to show only elements matching filters. This filter List, unlikely *filters*, can be removed by the user through the gwClassList UI. Example:

```
[
  {
    condition: 'AND',
    columnName: 'status',
    attributeGwid: 1,
    operator: '=',
    filterType: 'STRING',
    value: 'OPN'
  }
]
```

]

- **hideToolBar** Boolean, optional, default false. When true Toolbar is hidden
- **hideNewIgnorePermissions** Boolean, optional, default false. When true the 'New' button is hidden
- **hideDeleteIgnorePermissions** Boolean, optional, default false. When true the 'Delete' button is hidden
- **hideActionsDropDownButtonColumn** Boolean, optional, default false. When true the 'More Options' (AKA '3 dots') button is hidden. It overrides locally the eventually configured global gwProject setting (configured inside gwProject xml <gwClassSettings> tag)
- **hideIndirectSelectionColumn** Boolean, optional, default false. When true Indirect Selection Column in grid is hidden. It overrides locally the eventually configured global gwProject setting (configured inside gwProject xml <gwClassSettings> tag)
- **forceListedAttributes** Boolean, optional, default false. It works with 'listedAttributesNames' parameter. When true in grid are shown only the gwAttribute which name is inside 'listedAttributesNames' parameter
- **listedAttributesNames** String, optional, default null. Evaluated only when 'forceListedAttributes' is true. Is a list, comma separated, of gwAttribute name, which we want to force to compare in list (bypassing the set in gwAttribute List section in gwAdmin)
- **hiddenAttributesNamesList** String, optional, default null. Is a list, comma separated, of gwAttribute name, which we want to hide in list (the shown list is managed in gwAttribute List section in gwAdmin)
- **hiddenActionsNamesList** String, optional, default null. Is a list, comma separated, of gwAction name, which we want to hide in gwClassList Tab. This is useful when configuring many gwClassList tabs with different static filters, and it's needed to hide some actions, usually List Action, that works on incompatible (with current static filter) elements, and for this reason they have no sense in that context. This problematic can't be solved using standard DACL mechanism, and the only alternative without this parameter, will be duplicating the gwClass (and removing permissions for the specified gwActions)
- **customBeforeColumns** ArrayList<HashMap<String, String>>, optional, default null. Object [], each one containing all necessary infos to render a column. This columns are added BEFORE the base column set in gwAttribute List section in gwAdmin for the gwClass. Each column definition is made like this: { formatter: 'column_formatter', label: 'Header Label', width: 'auto' || 50, headerStyles: 'color: red;', styles: 'color: red;' } where 'formatter' is the name of the js function that generates the content (invoked with this arguments: 'item', 'rowIndex', 'cell'), 'label' is the column header text, 'width' can be an integer (px) or 'auto', 'headerStyles' e 'styles' are css rules applied respectively to header and cells.
- **customAfterColumns** ArrayList<HashMap<String, String>>, optional, default null. Object [], each one containing all necessary infos to render a column. This columns are added AFTER the base column set in gwAttribute List section in gwAdmin for the gwClass. Each column definition is made like this: { formatter: 'column_formatter', label: 'Header Label', width: 'auto' || 50, headerStyles: 'color: red;', styles: 'color: red;' }
- **onRowClickGwActionName** String, opzionale, default null. Se presente indica il nome di una azione Geoweb la quale verrà invocata (passando questi parametri [/*Integer*/ rowIndex, /*Object*/ grid]) al click della riga della griglia. Ha priorità su onRowClickFunctionName e onRowClickFunction
- **onRowClickFunctionName** String, opzionale, default null. Se presente indica il nome di una funzione js la quale verrà invocata (passando questi parametri [/*Integer*/ rowIndex, /*Object*/ grid]) al click della riga della griglia. Ha priorità su onRowClickFunction
- **onRowClickFunctionString** String, opzionale, default null. Se presente indica direttamente il codice js che verrà invocato (passando questi parametri [/*Integer*/ rowIndex, /*Object*/ grid])

al click della riga della griglia.

- **insertIndex** Integer, optional, default null. If absent the new tab will be opened at the last position
- **onStyleRowFunctionName** String, optional, default null. E' il nome della function javascript esistente nel namespace globale, invocata con questi parametri: [/dojo.grid.EnhancedGrid/ grid, /Object/ row] (disponibile dalla versione **4.6.0**)
- **onStyleRowGwActionName** String, optional, default null. E' il nome della gwAction, tipo *Liberato*, contenuta nella stessa classe. Deve contenere il solo corpo della funzione, tenendo conto che i parametri in ingresso saranno [/dojo.grid.EnhancedGrid/ grid, /Object/ row] (disponibile dalla versione **4.6.0**)
- **avoidColumnWidthAuto** Boolean, optional, default false. Avoid the 'auto' width setting upon column when related attributes are without and explicit 'listWidth' (grid equally divide the remaining space among all with 'auto'. So grid is no allowed to increase it's width, and to show an horizontal scrollbar. This is problematic with many attributes in list, that are simply not shown if the window isn't big enough) (disponibile dalla versione **4.6.10** issue #1140)
- **tabWidgetId** String, optional, default null. Quando configurato forza l'id del tab (usato in congiunzione con forceGridId e gridId per aprire più tab della stessa lista) (**dalla 4.7.2** issue #1392)
- **forceGridId** Boolean, optional, default false. Quando true abilita la valutazione di gridId
- **gridId** String, optional, default null. Da usare in congiunzione con forceGridId true. Quando configurato forza l'id della grid (da usare in congiunzione con forceGridId e tabWidgetId per aprire più tab della stessa lista)

Parametri opzionali contenuti in options [Deprecati]

- **hideAttributesNamesList** String, opzionale, default null. @Deprecated, replaced by hiddenAttributesNamesList

Esempi

Esempi con codice minimale:

```
var gwClassName = 'gwd_resource';
openGwClassListTab(gwClassName);
```

```
var gwClassName = 'gwd_resource';
var title = 'Risorse Operative'; //optional
openGwClassListTab(gwClassName, title);
```

Esempio generale:

```
var gwClassName = 'gw_class_name';
var title = 'title';
var options = {
  //..
};
var tabWidget = openGwClassListTab(gwClassName, title , options );
```

Esempio generale con utilizzo di tutti i parametri. Utile per copiare velocemente uno scheletro con la

struttura di base per poi eventualmente commentare i parametri non utilizzati:

```
var gwClassName = 'gwd_resource';
var title = 'Risorse Operative'; //optional
var options = { //optional
  filters: [
    {condition: 'AND', columnName: 'id_contract', operator: 'IN',
filterType: 'INTEGER', value: ['123', '456']}
  ],
  staticFilters: [
    {condition: 'AND', columnName: 'type_contract', operator: '=',
filterType: 'STRING', value: 'Forfait'}
  ]
  hideToolBar: false,
  hideNewIgnorePermissions: false,
  hideDeleteIgnorePermissions: false,
  hideActionsDropDownButtonColumn: false,
  hideIndirectSelectionColumn: false,
  forceListedAttributes: true,
  listedAttributesNames: 'attr_name_1,attr_name_2,attr_name_3',
  hiddenAttributesNamesList: 'attr_name_4,attr_name_5',
  hiddenActionsNamesList: 'action_name_1,action_name_2',
  customBeforeColumns: [
    { formatter: 'column_formatter', label: 'Header Label', width:
'auto' || 50, headerStyles: 'color: red;', styles: 'color: red;' }
  ],
  customAfterColumns: [
    { formatter: 'column_formatter', label: 'Header Label', width:
'auto' || 50, headerStyles: 'color: red;', styles: 'color: red;' }
  ],
  onRowClickGwActionName: 'onRowClick_gwAction_name',
  onRowClickFunctionName: 'onRowClick_name_in_js_global_name_space',
  onRowClickFunctionString: 'if(rowIndex==0)alert(\'click on row 0\');',
  onStyleRowFunctionName: 'onStyleRow_name_in_js_global_name_space',
  onStyleRowGwActionName: 'onStyleRow_gwAction_name',
};
var tabWidget = openGwClassListTab(gwClassName, title , options);
```

Esempio con imposizione di un singolo filtro statico:

```
var gwClassName = 'gwd_resource';
var title = 'Risorse Operative'; //optional
var options = { //optional
  forceReplaceTab: true,
  staticFilters: [
    {condition: 'AND', columnName: 'type_contract', operator: '=',
filterType: 'STRING', value: ['Forfait']}
  ]
};
openGwClassListTab(gwClassName, title , options);
```

Esempio che utilizza la piu generare API javascript openTab(). Modalità d'uso supportata, ma @Deprecated. Utilizzare una delle modalità più specifiche descritte sopra.

```
var tabWidgetType = 'gwClassList';
var identifier = 'resources_list';
var tabWidgetId = createTabId(tabWidgetType, identifier);
var tabWidgetTitle = 'Risorse Operative';

var parametersMap = {
  forceReplaceTab: true,
  className: 'gwd_resource',
  staticFilters: [
    {condition:'AND', columnName:'type_contract',
      operator:'=', filterType:'STRING', value: ['Forfait']}
  ]
};
var tabWidget = openTab(tabWidgetId, tabWidgetType, tabWidgetTitle,
parametersMap);
```

Azione che apre un tab gwClassList, date delle options. e che permette di aprirne successivamente un'altra (magari con altri filtri nelle options) senza generare errori (**dalla 4.7.2**)

```
var gwClassName = 'gwClassName ';
var title = 'Custom title'; //optional

var gridId = gwClassName+'_'+new Date().getTime();
var tabWidgetType = 'gwClassList';
var tabWidgetId = tabWidgetType+'_'+gridId;

var options = { //optional
  //others options..
  tabWidgetId: tabWidgetId,
  forceGridId: true,
  gridId: gridId
};
var tabWidget = openGwClassListTab(gwClassName, title , options);
```

Azione che apre un tab gwClassList, date delle options. e che permette di aprirne successivamente un'altra (magari con altri filtri nelle options) senza generare errori (**pre 4.7.2**)

```
var title = null; //TODO
//options already defined..

var tabWidgetType = 'gwClassList';

var gridId = gwClassName+'_'+new Date().getTime();

var tabWidgetType = 'gwClassList';
//var tabWidgetId = createTabId(tabWidgetType, gwClassName);
var tabWidgetId = tabWidgetType+'_'+gridId;
```

```
if(options.targetContainerId==null) options.targetContainerId = tabWidgetId;
options.gwClassName = gwClassName;

options.forceGridId = true;
options.gridId = gridId;

var tab = openTab(tabWidgetId, tabWidgetType, title, options);
```

openGwClassListFloatingPane()

Apri la lista di dettaglio di classe su un floatingPane.

Parametri

- **gwClassName** String, required, name of gwClass
- **contentPaneRef** String|domNode|dijit/layout/ContentPane, required
- **options** Object, optional, default {}. Contains all parameters allowed by openGwClassListTab(), and
 - **props** Object, map containing all parameters allowed by dijit/layout/ContentPane

Esempi

```
var gwClassName = 'gw_class_name';
var options = {
  //all parameters allowed by openGwClassListTab()
  title: 'GwClassList title',
  props: //all parameters allowed by openFloatingPane()
  {
    floatingPaneId: 'my_id',
    title: 'GwClassList title',
    modal: true,
    useCookie: true,
    dimensions: {w: 800, h: 600}
  }
};
var floatingPane = openGwClassListFloatingPane(gwClassName, options);
```

openGwClassListInsideContentPane ()

Apri la lista di dettaglio di classe in un qualsiasi contentPane.

Parametri

- **gwClassName** String, required, name of gwClass
- **contentPaneRef** String|domNode|dijit/layout/ContentPane, required. Può essere indifferentemente un dojo ContentPane, il suo id od il suo nodo dom.
- **options** Object, optional, default {}. Contains all parameters allowed by openGwClassListTab(), and
 - **props** Object, map containing all parameters allowed by dijit/layout/ContentPane

Esempi

```
var gwClassName = 'gw_class_name';
var contentPaneId = new ContentPane().id;
var options = {
  //all parameters allowed by openGwClassListTab()
  props: //all parameters allowed by dijit/layout/ContentPane
  {
  }
};
var contentPane = openGwClassListInsideContentPane(gwClassName,
contentPaneId, options);
```

From:
<https://wiki.geowebframework.com/> - **GeowebFramework**

Permanent link:
https://wiki.geowebframework.com/doku.php?id=custom:api_js_opengwclasslisttab

Last update: **2025/02/11 11:29**

