

gwClassDetail funzioni esposte

Il gwClassDetail rappresenta il dettaglio di classe. In qualunque forma esso sia aperto (*tab*, *dialog*, *floatingPane*), l'oggetto dojo contenitore del gwClassDetail (AKA *detailContainer*) espone tutta una serie di function di utilità, che servono per interagire con il gwClassDetail o conoscerne lo stato corrente.

- [getGwWidgets\(\)](#)
- [getGwWidget\(\)](#)
- [setFormValues\(\)](#)
- [getFormValues\(\)](#)
- [getFormValuesWithGwidAsKey\(\)](#)
- [getFormValuesToShow\(\)](#)
- [createMapWithGwidAsKey\(\)](#)
- [createMapWithNameAsKey\(\)](#)
- [isSomeFormValueChanged\(\)](#)
- [hideGwActionInToolbarButton\(\)](#)
- [showGwActionInToolbarButton\(\)](#)
- [disableGwActionInToolbarButton\(\)](#)
- [enableGwActionInToolbarButton\(\)](#)

getGwWidgets()

Ritorna un array di oggetti gwWidget (array vuoto, [], nel caso niente venga trovato). Di norma la ricerca viene effettuata solo all'interno dei dettagli di classe dove *detailMode*=='*edit*'. Si può opzionalmente includere nella ricerca anche i widget contenuti dentro i dettagli di classe dove *detailMode*=='*visualization*', agendo sul flag *includeAll*.

Parametri

- **params**: Object, richiesto. Oggetto contenente le seguenti proprietà:
 - **name**: String, opzionale. Il name del gwAttribute collegato al gwWidget. Se fornito è usato come criterio di ricerca. Se omesso, verrà ritornato il primo widget disponibile.
 - **gwid**: Integer, opzionale. Il gwid del gwAttribute collegato al gwWidget. Se fornito è usato come criterio di ricerca. Se omesso, verrà ritornato il primo widget disponibile.
 - **includeAll**: Boolean, opzionale, default false. Se posto a true, includere nella ricerca anche i widget contenuti dentro form di classe dove *detailMode*=='*visualization*'.
 - **queryContext**: String|domNode, opzionale, default null. Se fornito questo verrà usato come nodo di restrizione per eseguire la query. (Usato internamente a Geoweb per il funzionamento delle schede *Procedure* del *workflow*).

Esempi

Recupera tutti i widget in edit del gwClassDetail

```
var widgets = detailContainer.getGwWidgets();
```

Recupera tutti i widget del gwClassDetail, cercando anche fra quelli contenuti in form di classe dove detailMode=='visualization'.

```
var widgets = detailContainer.getGwWidgets({includeAll: true});
```

getGwWidget()

Ritorna un singolo gwWidget (array vuoto, [], nel caso niente venga trovato). Di norma la ricerca viene effettuata solo all'interno dei dettagli di classe dove *detailMode*=='edit'. Si può opzionalmente includere nella ricerca anche i widget contenuti dentro i dettagli di classe dove *detailMode*=='visualization', agendo sul flag *includeAll*.

Parametri

- **params:** Object, richiesto. Oggetto contenente le seguenti proprietà:
 - **name:** String, opzionale. Il name del gwAttribute collegato al gwWidget. Se fornito è usato come criterio di ricerca. Se omesso, verrà ritornato il primo widget disponibile.
 - **gwid:** Integer, opzionale. Il gwid del gwAttribute collegato al gwWidget. Se fornito è usato come criterio di ricerca. Se omesso, verrà ritornato il primo widget disponibile.
 - **includeAll:** Boolean, opzionale, default false. Se posto a true, includere nella ricerca anche i widget contenuti dentro form di classe dove *detailMode*=='visualization'.
 - **queryContext:** String|domNode, opzionale, default null. Se fornito questo verrà usato come nodo di restrizione per eseguire la query. (Usato internamente a Geoweb per il funzionamento delle schede *Procedure* del *workflow*).

Esempi

Recupera il primo widget trovato in edit del gwClassDetail.

```
var widget = detailContainer.getGwWidget({name: 'gw_attribtue_name'});
```

Recupera il primo widget trovato del gwClassDetail, cercando anche fra quelli contenuti in form di classe dove detailMode=='visualization'.

```
var widget = detailContainer.getGwWidget({name: 'gw_attribtue_name', includeAll: true});
```

setFormValues()

Ritorna un singolo gwWidget (array vuoto, [], nel caso niente venga trovato). Di norma la ricerca viene effettuata solo all'interno dei dettagli di classe dove *detailMode*=='edit'. Si può opzionalmente

includere nella ricerca anche i widget contenuti dentro i dettagli di classe dove `detailMode=='visualization'`, agendo sul flag `includeAll`.

Parametri

- **mapWithNameAsKey**: Object, richiesto. Oggetto mappa che ha per chiavi i `name` dei `gwAttribute` e valori i valori da settare nelle form di dettaglio di classe

Esempi

```
detailContainer.setFormValues({attr_name_1: 'value1', attr_name_2: 2:  
attr_name_3: new Date()});
```

getFormValues()

Ritorna una Object javascript che ha per chiavi i `name` dei `gwAttribute` e valori i valori attualmente presenti nelle form di dettaglio di classe

Parametri

- **onlyModified**: Boolean, opzionale, default true. Se posto a true viene ritornata una mappa contenente solo i valori dei `gwWidget` che sono stati modificati

Esempi

```
var onlyModified = true;  
var formValues = detailContainer.getFormValues(onlyModified);  
var attr_1_value = formValue.attr_1_name; //if 'attr_1_name' is the name of  
a gwAttribute inside the form
```

getFormValuesWithGwidAsKey()

Ritorna una Object javascript che ha per chiavi i `gwid` dei `gwAttribute` e valori i valori attualmente presenti nelle form di dettaglio di classe

Parametri

- **onlyModified**: Boolean, opzionale, default true. Se posto a true viene ritornata una mappa contenente solo i valori dei `gwWidget` che sono stati modificati

Esempi

```
var onlyModified = true;
var formValues = detailContainer.getFormValuesWithGwidAsKey(onlyModified);
var attr_1_value = formValue.attr_1_name; //if 'attr_1_name' is the name of
a gwAttribute inside the form
```

getFormValuesToShow()

Ritorna una Object javascript che ha per chiavi i *name* dei gwAttribute e valori i *valueToShow* attualmente presenti nelle form di dettaglio di classe. In alcuni widget *valueToShow*, valore in visualizzazione, si differenzia dal *valueToStore*, valore da persistere. Negli altri widget, non c'è tale distinzione ed i due coincidono in *value*.

Parametri

- **onlyModified**: Boolean, opzionale, default true. Se posto a true viene ritornata una mappa contenente solo i valori dei gwWidget che sono stati modificati

Esempi

```
var onlyModified = true;
var formValuesToShow= detailContainer.getFormValuesToShow(onlyModified);
var attr_1_valueToShow = formValuesToShow.attr_1_name; //if 'attr_1_name' is
the name of a gwAttribute inside the form
```

createMapWithGwidAsKey()

Ritorna una Object javascript che ha per chiavi i *name* dei gwAttribute e valori gli stessi valori dell' Object javascript in ingresso (indicizzato per *gwid*).

Parametri

- **nameAsKeyMap**: Boolean, richiesto. Object javascript, mappa chiave valore, che ha per chiavi i *name* dei gwAttribute

Esempi

```
var nameAsKeyMap = {
  attr_1_name: 'value1',
```

```
    attr_2_name: 2,  
    attr_3_name: new Date()  
};  
var gwidAsKeyMap = detailContainer.createMapWithGwidAsKey(nameAsKeyMap);
```

createMapWithNameAsKey()

Ritorna una Object javascript che ha per chiavi i *gwid* dei gwAttribute e valori gli stessi valori dell' Object javascript in ingresso (indicizzato per *name*).

Parametri

- **gwidAsKeyMap**: Boolean, richiesto. Object javascript, mappa chiave valore, che ha per chiavi i *gwid* dei gwAttribute

Esempi

```
var gwidAsKeyMap = {  
  123: 'value1', //123 is attr_1_gwid  
  456: 2, //456 is attr_2_gwid  
  789: new Date() //789 is attr_3_gwid  
};  
var nameAsKeyMap = detailContainer.createMapWithGwidAsKey(gwidAsKeyMap);
```

isSomeFormValueChanged()

Ritorna true se il dettaglio corrente ha subito dei cambiamenti negli input delle form, ad opera dell'utente od automatismi, false altrimenti.

Parametri

Esempi

```
var isSomeFormValueChanged = detailContainer.isSomeFormValueChanged();
```

hideGwActionInToolbarButton()

Dato il *name* della gwAction, nasconde il button corrispondente.

Parametri

- **gwActionName**: String, richiesto. Il *name* della gwAction del gwActionButton

Esempi

```
var gwActionName = 'gw_action_name';
detailContainer.hideGwActionInToolbarButton(gwActionName);
```

showGwActionInToolbarButton()

Dato il *name* della gwAction, mostra il button corrispondente.

Parametri

- **gwActionName**: String, richiesto. Il *name* della gwAction del gwActionButton

Esempi

```
var gwActionName = 'gw_action_name';
detailContainer.showGwActionInToolbarButton(gwActionName);
```

disableGwActionInToolbarButton()

Dato il *name* della gwAction, disabilita il button corrispondente.

Parametri

- **gwActionName**: String, richiesto. Il *name* della gwAction del gwActionButton

Esempi

```
var gwActionName = 'gw_action_name';
detailContainer.disableGwActionInToolbarButton(gwActionName);
```

enableGwActionInToolbarButton()

Dato il *name* della gwAction, abilita il button corrispondente.

Parametri

- **gwActionName**: String, richiesto. Il *name* della gwAction del gwActionButton

Esempi

```
var gwActionName = 'gw_action_name';  
detailContainer.enableGwActionInToolBarButton(gwActionName);
```

From:

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

Permanent link:

https://wiki.geowebframework.com/doku.php?id=custom:api_js_gw_class_detail_exposed_fx&rev=1576599124

Last update: **2019/12/17 17:12**

