

Field Based APIs		Field Based APIs (cont)		User Objects (cont)	
Hide/Display variable	<code>g_form.setDisplay(&lt;fld&gt;,true/false);</code>	Get Integer Value of Field	<code>g_form.getIntValue(&lt;fieldname&gt;);</code>	if the current user has at least one role specified, <code>g_ole</code> false otherwise.	
Disable a field	<code>g_form.setDisable(&lt;fld&gt;,true/false);</code>	Get Decimal Value of Field	<code>g_form.getDecimalValue(&lt;fieldname&gt;);</code>		
Field ReadOnly	<code>g_form.setReadOnly(&lt;fld&gt;,true/false);</code>	<p>&lt;fld&gt; Represent the field name on which the function has to work</p> <p>value - is something what you have to set.</p> <p>true / false - true is for enabling, false is for disable</p>		<h3>Related List APIs</h3> <p>Show Related List <code>g_form.showRelatedList(&lt;re</code></p> <p>Hide Related List <code>g_form.hideRelatedList(&lt;re</code></p> <p>Show All Related Lists <code>g_form.showRelatedLists()</code></p> <p>Hide All Related Lists <code>g_form.hideRelatedLists();</code></p>	
Clear Value from Field	<code>g_form.clearValue(&lt;fld&gt;);</code>	<b>User Objects</b>		<b>Sections based API</b>	
Change Label of Field	<code>g_form.setLabelOf(&lt;fld&gt;,-newlabel);</code>	User name of the current user e.g. employee	<code>g_user.userName</code>	Show/Hide a Section	<code>g_form.setSectionD</code>
Make field mandatory	<code>g_form.setMandatory(&lt;fld&gt;,true/false);</code>	First name of the current user e.g. Joe	<code>g_user.firstName</code>	Get All Sections*	<code>g_form.getSectionN</code>
Set Value of Field	<code>g_form.setValue(&lt;fld&gt;,-value);</code>	Last name of the current user e.g. Employee	<code>g_user.lastName</code>	Get All Sections**	<code>g_form.getSections</code>
Show error and scroll to field	<code>g_form.showErrorBox(&lt;fld&gt;,-msg,true/false);</code>	sys_id of the current user e.g. 681cca-f9c0a8016400b98a06818d57c7	<code>g_user.userID</code>	Check whether section is Visible	<code>g_form.isSectionVis</code>
Add Options to Choice filed	<code>g_form.addOption(&lt;fld&gt;,-value,-label,-index);</code>	if the current user has the role specified, false otherwise. ALWAYS returns true if the user has the 'admin' role.	<code>g_user.hasRole(&lt;rolename&gt;)</code>	<p>*Returns all the sections including the invisible ones,</p> <p>**Not available on Mobile Platform</p> <p>&lt;sectionname&gt; : the section name in ServiceNow is created by "_" and trimming rest of the spaces, for ex :if a section "Section Specific API" then section name is "section_sp</p>	
Remove Options from Choice field	<code>g_form.removeOption(&lt;fld&gt;,-value);</code>	if the current user has the exact role of 'admin' role.	<code>g_user.hasRoleExactly(&lt;rolename&gt;)</code>		
Clear all Options from Choice Field	<code>g_form.clearOptions(&lt;fld&gt;);</code>				
Get Reference field Value and Attributes	<code>g_form.getReference(&lt;fld&gt;,-callbackfunction);</code>				
Flash a particular field	<code>g_form.flash(table.name,-color,-no of flashes)</code>				



### Decorations API

Add decorations(not applicable for Catalog Client Scripts)	<code>g_form.addDecoration(&lt;fld&gt;,iconname,titleforicon);</code>
Remove Decorations(not applicable for Catalog Client Scripts)	<code>g_form.removeDecoration(&lt;fld&gt;,iconname,titleforicon);</code>
LiveUpdate running on form*	<code>g_form.isLiveUpdating()</code>

\*Live update can detect any change of the data on the form from another session.

### HTML based APIs

Control a Field	<code>g_form.getControl(&lt;fldname&gt;)</code>
Get an HTML Element from Parameter*	<code>g_form.getElement(&lt;parameter&gt;)</code>
Get HTML element of Option**	<code>g_form.getOption(&lt;fieldname&gt;,value);</code>
Get HTML element of Form	<code>g_form.getFormElement()</code>
Get HTML element of HelpText	<code>g_form.getHelpTextControl(&lt;fldname&gt;);</code>

\*Not available in the Mobile and Service Portal

\*\*The method will not work on readonly fields, and if the field is not present on the form it will result in null.

### Action Specific API

Submit a form	<code>g_form.submit(actionName)</code>
Save a form	<code>g_form.save()</code>

### Best Practices Client Scripting

**Use UI policies instead of Client scripts :** For basics like making field readonly, mandatory Policies.

**Always use isLoading check when in onChange Client Scripts :** All the onChange as well, always check whether the form is loading by isLoading function

**Avoid Using GlideRecord function in Client script :** Replace the calls by GlideAjax

**Avoid DOM(manipulation of form elements) :** As all the elements might be changed in the page, avoid doing DOM

**Use g\_scratchpad to minimise server calls :** When working with lot of data write a Display function available to Client Side

**When using JQuery in Client Scripts :** Instance before london, has no possibility of isolation. London and above has a strict client mode, where methods like window, eval would only work if isolated.



By **Bibin Gokuldas** (bibingokuldas)  
[cheatography.com/bibingokuldas/](https://cheatography.com/bibingokuldas/)  
[bibingokuldas.com/](https://bibingokuldas.com/)

Published 25th October, 2018.  
 Last updated 11th August, 2020.  
 Page 2 of 2.

Sponsored by **Readable**  
 Measure your website  
<https://readable.com>