Glide Form

What is Glide Form and usage (g_form)

- The **Glide Form** is client side API, mainly used to change default behavior of the current record
- GlideForm methods are only running on the client side **(Browser).**
- The global object **g_form** is used to access **GlideForm** methods.
- These methods are used to make custom changes to the form view of records. All validation of examples was done using Client Scripts.
- **g_form** methods frequently using in **Client Scripts.**
- **g_form** methods can use in catalog client script too.

Glide Form Methods

Glide Form Methods	
setValue()	getFormElement()
getValue()	getSection()
addOption()	getTableName()
removeOption()	getUniqueValue()
addInfoMessage()	hideRelatedLists()
addErrorMessage()	isMandatory()
clearMessage()	save()
clearOptions()	submit()
disableAttachments()	setDisabled()
enableAttachments()	setDisplay()
Flash()	setMandatory()
getActionName()	setVisible()
getControl()	setReadOnly()
getElement()	setSectonDisplay()
showErrorBox()	showRelatedList()
showFieldMsg()	showRelatedLists()
addDecoration()	isNewRecord()
removeOption()	hideFieldMsg()

Glide Form

Glide Form Exercises

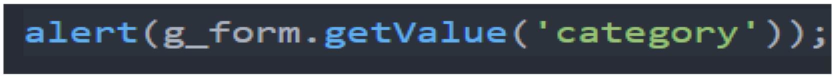
- Navigate **Incident** table
- Open one existing record
- Click on **Ctrl+Shift+j**
- Open **Java Script Executor**
- Run our code here

Ctrl+Shift+J

1. Working with **getValue()** method

This method is used to get specific field value from current form.

Alert(g_form.getValue ('category')); //Alert is used to print the out put

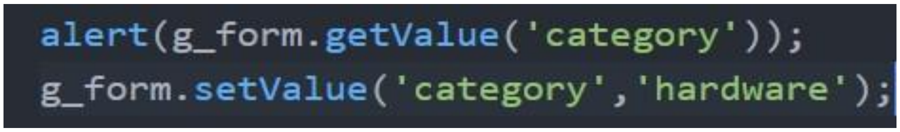


```
alert(g_form.getValue('category'));
```

2. Working with **setValue()** method

Sets the value of specific field.

alert (g_form.getValue ('category'));
g_form.setValue ('category', 'hardware');

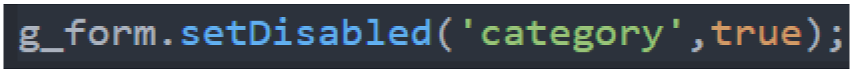


```
alert(g_form.getValue('category'));
g_form.setValue('category','hardware');
```

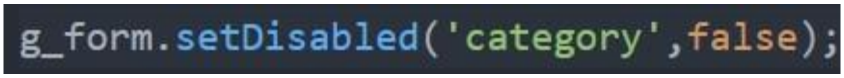
3. Working with**setDisabled()** method

This method is used to field make **Read only**

g_form.setDisabled ('category', true);



```
g_form.setDisabled('category',true);
```



```
g_form.setDisabled('category',false);
```

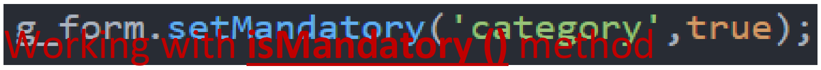
Glide Form

Note:UI Policy is best practice instead of using this method

4. Working with **setMandatory()** method

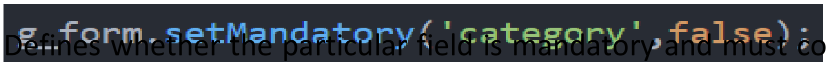
This method is used to a field make **Mandatory**

g_form.setMandatory ('category’, true);



```
g_form.setMandatory('category',true);
```

7. Working with **isMandatory ()** method



```
g_form.setMandatory('category',false);
```

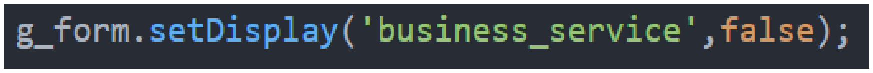
Defines whether the particular field is mandatory and must contain a value before the record can be saved or submitted display bullion **(True/False)**

Note:UI Policy is best practice instead of using this method

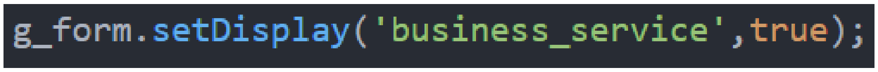
5. Working with **setDisplay()** method

This method is used to **Hide** and **Un hide** specific fields

g_form.setDisplay ('business_service', false);



```
g_form.setDisplay('business_service',false);
```



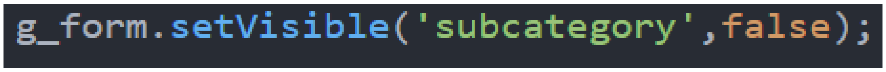
```
g_form.setDisplay('business_service',true);
```

Note:UI Policy is best practice instead of using this method

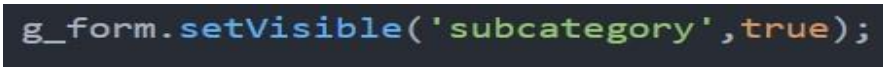
6. Working with setVisible () method

This method is usedto**Hide** field and maintain the space

g_form.setVisible ('subcategory’, false);



```
g_form.setVisible('subcategory',false);
```



```
g_form.setVisible('subcategory',true);
```

Glide Form

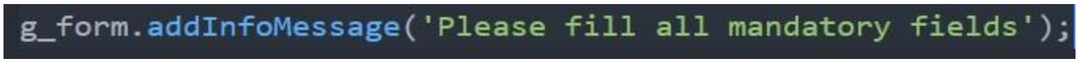
g_form.isMandatory ('category');

```
g_form.isMandatory('category');
```

8. Working with **addInfoMessage()** method

Add information message on top of the form

g_form.addInfoMessage (‘please fill all mandatory fields');



```
g_form.addInfoMessage('Please fill all mandatory fields');
```

9.Working with **addErrorMessage()** method

Add an error message on top of the form

g_form.addErrorMessage ('Fill mandatory fields');

```
g_form.addErrorMessage('Fill mandatory fields');
```

10. Working with **disableAttchments ()** method

Prevents the userattachment icon being hide

g_form.disableAttachments ();

```
g_form.disableAttachments();
```

11. Working with **enableAttchments ()** method

Allows customer file attachments to be added. Shows the paper clip icon.

g_form.enableAttachments ();

```
g_form.enableAttachments();
```

12. Working with **clearMessages ()** method

Removes all **informational** and **error** messages from the top of the form. **g_form.addInfoMessage ()** and **g_form.addErrorMessage ().**

g_form.clearMessages ();

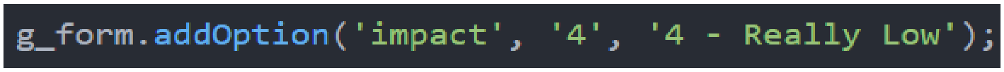
```
g_form.clearMessages();
```

Glide Form

13. Working with **addOption ()** method

Add a new choice to respective field depends on requirement

g_form.addOption ('priority', '6', '6 - Really Low');

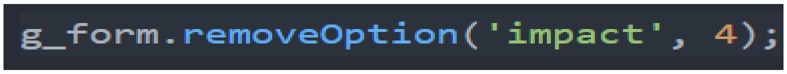


```
g_form.addOption('impact', '4', '4 - Really Low');
```

14. Working with **removeOption ()** method

Remove a choice from respective field depends on requirement.

g_form.removeOption ('impact', 4);

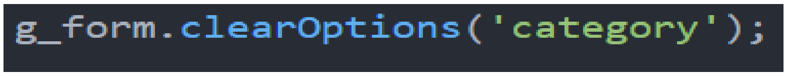


```
g_form.removeOption('impact', 4);
```

15. Working with **clearOption ()** method

Removes all options from the choice list.

g_form.clearOptions ('category');

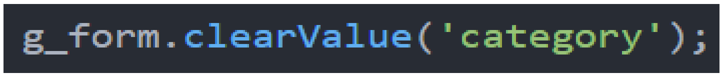


```
g_form.clearOptions('category');
```

16. Working with **clearValue ()** method

Remove value from specific field on the form

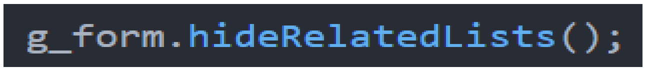
g_form.clearValue('category');



```
g_form.clearValue('category');
```

17. Working with **hideRealtedLists ()** method

This method will hide all related lists on form g_form.hideRelatedLists ();

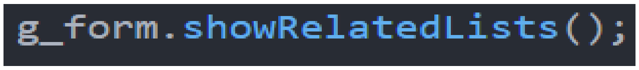


```
g_form.hideRelatedLists();
```

18. Working with **showRealtedLists ()** method

This method will show all related lists on form

g_form.showRelatedLists ();



```
g_form.showRelatedLists();
```

19. Working with **isNewRecord ()** method

If record is new true the record has never been saved.

g_form.isNewRecord ();

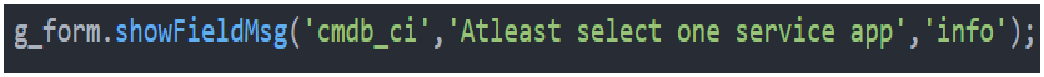
```
g_form.isNewRecord();
```

20. Working with **showFieldMsg ()** method

This method will display Informational**or Error or Warn** message under the specified form field (either a control object or the name of the field). If the control or field is off the screen, the form is scrolled to the field.

The **showErrorBox ()** method is analternate method that does not require the type parameter.

g_form.showFieldMsg ('cmdb_ci','Atleast select one service app','info');



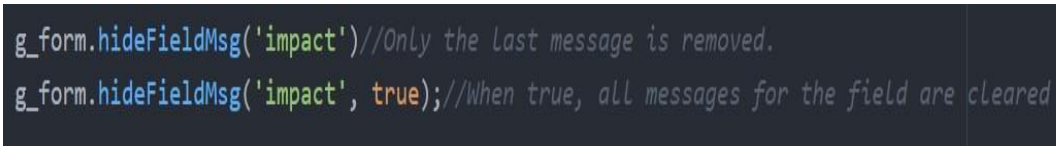
```
g_form.showFieldMsg('cmdb_ci','Atleast select one service app','info');
```

21. Working with **hideFieldMsg ()** method

This method will hides the last message placed by **showFieldMsg ().**

When true, all messages for the field are cleared. When false, only the last message is removed.

g_form.hideFieldMsg('impact')//Only the last message is removed.
g_form.hideFieldMsg('impact', true);//When true, all messages for the field are cleared

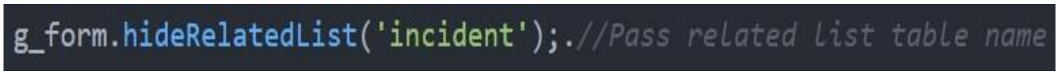


```
g_form.hideFieldMsg('impact')//Only the last message is removed.
g_form.hideFieldMsg('impact', true);//When true, all messages for the field are cleared
```

22. Working with **hideRelatedList ()** method

Hides the specified related list on the form.

g_form.hideRelatedList ('incident'); //Pass related list table name

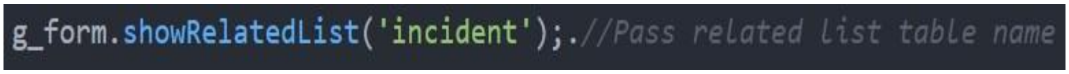


```
g_form.hideRelatedList('incident');.//Pass related list table name
```

23. Working with **showRelatedList ()** method

Hides the specified related list on the form.

g_form.showRelatedList ('incident'); //Pass related list table name



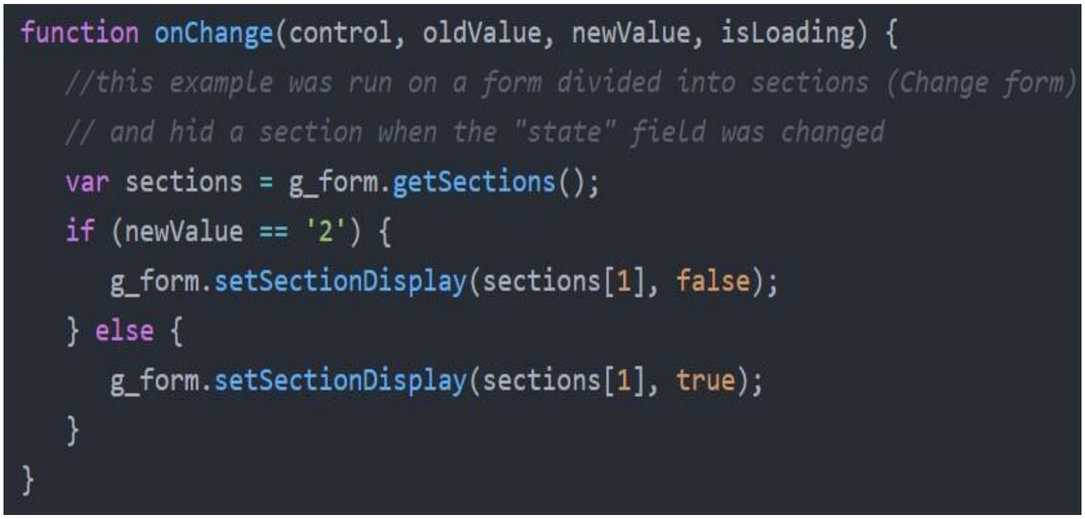
```
g_form.showRelatedList('incident');.//Pass related list table name
```

24. Working with **getSections () and setSectionDisplay ()** method

Displayan array of the form's sections, which are available.

```
functiononChange (control, oldValue, newValue, isLoading) {
  //this example was run on a form divided into sections (Change form)

  // and hid a section when the "state" field was changed

var sections = g_form.getSections();

if (newValue == '2') {

g_form.setSectionDisplay(sections[1], false);

  } else {

g_form.setSectionDisplay(sections[1], true);
```

```
  }
}
```

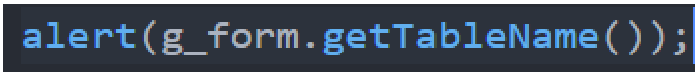


```
function onChange(control, oldValue, newValue, isLoading) {
   //this example was run on a form divided into sections (Change form)
   // and hid a section when the "state" field was changed
   var sections = g_form.getSections();
   if (newValue == '2') {
      g_form.setSectionDisplay(sections[1], false);
   } else {
      g_form.setSectionDisplay(sections[1], true);
   }
}
```

25. Working with **getTableName ()** method

Returns the name of the table to which this record belongs.

alert (g_form.getTableName ());

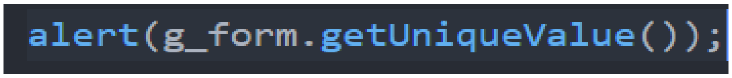


```
alert(g_form.getTableName());
```

26. Working with **getUniqueValue ()** method

It will return the sys_id of the record displayed in the form.

alert (g_form.getUniqueValue ());



```
alert(g_form.getUniqueValue());
```

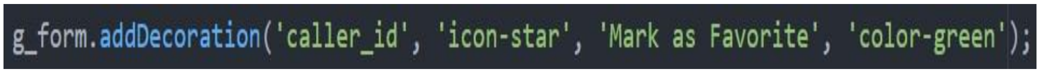
27. Working with **addDecoration ()** method

Adds an icon on a field’s label.

Adding the same item twice is prevented; however, you can add the same icon with a different title.

Glide Form

g_form.addDecoration ('caller_id', 'icon-star', 'Mark as Favorite', 'color-green');

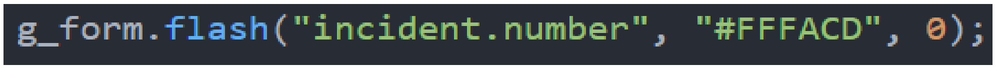


```
g_form.addDecoration('caller_id', 'icon-star', 'Mark as Favorite', 'color-green');
```

28. Working with **flash ()** method

This method is used to Flashes the specified color for a specified duration of time in the specified field.

g_form.flash (‘incident.number’, ‘#FFFACD’, 0);



```
g_form.flash("incident.number", "#FFFACD", 0);
```