

JSVariable

 Apr 04, 2024 06:27

Supported Clients

SmartClient WebClient NGClient MobileClient

Constants Summary		
Number	DATETIME	Constant to be used when the type of a variable needs to be specified.
Number	INTEGER	Constant to be used when the type of a variable needs to be specified.
Number	MEDIA	Constant to be used when the type of a variable needs to be specified.
Number	NUMBER	Constant to be used when the type of a variable needs to be specified.
Number	TEXT	Constant to be used when the type of a variable needs to be specified.

Property Summary		
String	defaultValue	The default value of the variable.
String	name	The name of the variable.
Number	variableType	The type of the variable.

Methods Summary	
String	getScopeName() Get scope name
UUID	getUUID() Returns the UUID of the variable

Constants Details

DATETIME
Constant to be used when the type of a variable needs to be specified.

Returns
Number

Supported Clients
SmartClient,WebClient,NGClient,MobileClient

Sample

```
var dateVar = solutionModel.newGlobalVariable('globals', 'gDate', JSVariable.DATETIME);
dateVar.defaultValue = 'now';
application.output(scopes.globals.gDate); // Prints the current date and time.
```

INTEGER
Constant to be used when the type of a variable needs to be specified.

Returns
Number

Supported Clients
SmartClient,WebClient,NGClient,MobileClient

Sample

```
var intVar = solutionModel.newGlobalVariable('globals', 'gInt', JSVariable.INTEGER);
intVar.defaultValue = 997;
application.output(scopes.globals.gInt); // Prints 997
```

MEDIA
Constant to be used when the type of a variable needs to be specified.

Returns[Number](#)**Supported Clients**

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var mediaVar = solutionModel.newGlobalVariable('globals', 'gMedia', JSVariable.MEDIA);
mediaVar.defaultValue = 'new Array(1, 2, 3, 4)';
application.output(scopes.globals.gMedia); // Prints out the array with four elements.
```

NUMBER

Constant to be used when the type of a variable needs to be specified.

Returns[Number](#)**Supported Clients**

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var numberVar = solutionModel.newGlobalVariable('globals', 'gNumber', JSVariable.NUMBER);
numberVar.defaultValue = 192.334;
application.output(scopes.globals.gNumber); // Prints 192.334
```

TEXT

Constant to be used when the type of a variable needs to be specified.

Returns[Number](#)**Supported Clients**

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var txtVar = solutionModel.newGlobalVariable('globals', 'gText', JSVariable.TEXT);
txtVar.defaultValue = "some text"; // Use two pairs of quotes if you want to assign a String as default value.
application.output(scopes.globals.gText); // Prints 'some text' (without quotes).
```

Property Details**defaultValue**

The default value of the variable.

It is interpreted as a JS expression.

For form variables ,setting this property requires the form instances to be destroyed (history.remove("formName")).

If you want to use a default value for a newly created variable create the variable using the 3 parameter version newVariable(name,type,defaultValue).

For INTEGER variables it can be an integer constant, like 10 for example.
 For NUMBER variables it can be a real constant, like 22.41. For DATETIME variables it can be "now", or a JS expression like "new Date()". For TEXT variables it can be any string surrounded with quotes, like "'some text'".

Returns[String](#)**Supported Clients**

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var intVar = solutionModel.newGlobalVariable('globals', 'gInt', JSVariable.INTEGER);
intVar.defaultValue = 997;
application.output(scopes.globals.gInt); // Prints 997
var numberVar = solutionModel.newGlobalVariable('globals', 'gNumber', JSVariable.NUMBER);
numberVar.defaultValue = 192.334;
application.output(scopes.globals.gNumber); // Prints 192.334
var dateVar = solutionModel.newGlobalVariable('globals', 'gDate', JSVariable.DATETIME);
dateVar.defaultValue = 'now';
application.output(scopes.globals.gDate); // Prints the current date and time.
var txtVar = solutionModel.newGlobalVariable('globals', 'gText', JSVariable.TEXT);
txtVar.defaultValue = "some text"; // Use two pairs of quotes if you want to assign a String as default value.
application.output(scopes.globals.gText); // Prints 'some text' (without quotes).
var mediaVar = solutionModel.newGlobalVariable('globals', 'gMedia', JSVariable.MEDIA);
mediaVar.defaultValue = 'new Array(1, 2, 3, 4)';
application.output(scopes.globals.gMedia); // Prints out the array with four elements.
```

name

The name of the variable.

Returns

[String](#)

Supported Clients

SmartClient, WebClient, NGClient, MobileClient

Sample

```
var gVar = solutionModel.newGlobalVariable('globals', 'gtext', JSVariable.TEXT);
gVar.name = 'anotherName';
gVar.defaultValue = 'default text';
// The next two lines will print the same output.
application.output(scopes.globals[gVar.name]);
application.output(scopes.globals.anotherName);
```

variableType

The type of the variable. Can be one of: TEXT, INTEGER, NUMBER, DATETIME or MEDIA.

Returns

[Number](#)

Supported Clients

SmartClient, WebClient, NGClient, MobileClient

Sample

```
var g = solutionModel.newGlobalVariable('globals', 'gtext', JSVariable.TEXT);
scopes.globals.gtext = 'some text';
g.variableType = JSVariable.DATETIME;
scopes.globals.gtext = 'another text'; // This will raise an error now, because the variable is not longer of
type text.
```

Methods Details**getScopeName()**

Get scope name

Returns

[String](#)

Supported Clients

SmartClient, WebClient, NGClient, MobileClient

Sample

```
var globalVariables = solutionModel.getGlobalVariables();
for (var i in globalVariables)
    application.output(globalVariables[i].name + ' is defined in scope ' + globalVariables[i].
getScopeName());
```

getUUID()

Returns the UUID of the variable

Returns

UUID

Supported Clients

SmartClient,WebClient,NGClient,MobileClient

Sample

```
var dateVar = solutionModel.newGlobalVariable('globals', 'gDate', JSVariable.DATETIME);
application.output(dateVar.getUUID().toString());
```