

excelxport

This plugin allows the enabling/disabling of the Smart Client menuitems that expose the generic Excel import/export logic

Property Summary

Boolean	exportEnabled Enable the export feature of this plugin.
Boolean	importEnabled Enable the import feature of this plugin.

Method Summary

byte[]	excelExport (foundSet, dataProviderIds) Export to Excel data
byte[]	excelExport (foundSet, dataProviderIds, templateXLS) Export to Excel data
byte[]	excelExport (foundSet, dataProviderIds, templateXLS, sheetName) Export to Excel data
byte[]	excelExport (foundSet, dataProviderIds, templateXLS, sheetName, startRow) Export to Excel data
byte[]	excelExport (foundSet, dataProviderIds, templateXLS, sheetName, startRow, startColumn) Export to Excel data

Property Details

exportEnabled

Enable the export feature of this plugin.

Returns

[Boolean](#)

Sample

```
plugins.excelxport.exportEnabled = true;  
var isEnabled = plugins.excelxport.exportEnabled;
```

importEnabled

Enable the import feature of this plugin.

Returns

[Boolean](#)

Sample

```
plugins.excelxport.importEnabled = true;  
var isEnabled = plugins.excelxport.importEnabled;
```

Method Details

excelExport

byte[] [excelExport](#) (foundSet, dataProviderIds)
Export to Excel data

Parameters

{[JSFoundSet](#)} foundSet - the foundset on which to export
{[String](#)[]} dataProviderIds - the ids of the dataproviders

Returns

byte[]

Sample

```
//export in new byte array
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name']);
//export by adding to templateXLS in default (new) 'Servoy Data' worksheet
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS);
//export by adding to templateXLS, in 'mySheet' worksheet, starting at default(1/1) row/column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet');
//export by adding to templateXLS, in 'mySheet' worksheet, starting at 3rd row and 5th column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet',3,5);
```

excelExportbyte[] **excelExport** (foundSet, dataProviderIds, templateXLS)

Export to Excel data

Parameters

{[JSFoundSet](#)} foundSet - the foundset on which to export
 {[String](#)[]} dataProviderIds - the ids of the dataproviders
 {byte[]} templateXLS - the xls template to export in

Returns

byte[]

Sample

```
//export in new byte array
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name']);
//export by adding to templateXLS in default (new) 'Servoy Data' worksheet
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS);
//export by adding to templateXLS, in 'mySheet' worksheet, starting at default(1/1) row/column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet');
//export by adding to templateXLS, in 'mySheet' worksheet, starting at 3rd row and 5th column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet',3,5);
```

excelExportbyte[] **excelExport** (foundSet, dataProviderIds, templateXLS, sheetName)

Export to Excel data

Parameters

{[JSFoundSet](#)} foundSet - the foundset on which to export
 {[String](#)[]} dataProviderIds - the ids of the dataproviders
 {byte[]} templateXLS - the xls template to export in
 {[String](#)} sheetName - the name of the worksheet

Returns

byte[]

Sample

```
//export in new byte array
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name']);
//export by adding to templateXLS in default (new) 'Servoy Data' worksheet
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS);
//export by adding to templateXLS, in 'mySheet' worksheet, starting at default(1/1) row/column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet');
//export by adding to templateXLS, in 'mySheet' worksheet, starting at 3rd row and 5th column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet',3,5);
```

excelExportbyte[] **excelExport** (foundSet, dataProviderIds, templateXLS, sheetName, startRow)

Export to Excel data

Parameters

{JSFoundSet} foundSet - the foundset on which to export
 {String[]} dataProviderIds - the ids of the dataproviders
 {byte[]} templateXLS - the xls template to export in
 {String} sheetName - the name of the worksheet
 {Number} startRow - row in the foundset at which to start the export

Returns

byte[]

Sample

```
//export in new byte array
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name']);
//export by adding to templateXLS in default (new) 'Servoy Data' worksheet
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS);
//export by adding to templateXLS, in 'mySheet' worksheet, starting at default(1/1) row/column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet');
//export by adding to templateXLS, in 'mySheet' worksheet, starting at 3rd row and 5th column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet',3,5);
```

excelExport

byte[] **excelExport** (foundSet, dataProviderIds, templateXLS, sheetName, startRow, startColumn)

Export to Excel data

Parameters

{JSFoundSet} foundSet - the foundset on which to export
 {String[]} dataProviderIds - the ids of the dataproviders
 {byte[]} templateXLS - the xls template to export in
 {String} sheetName - the name of the worksheet
 {Number} startRow - row in the foundset at which to start the export
 {Number} startColumn - column in the foundset at which to start the export

Returns

byte[]

Sample

```
//export in new byte array
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name']);
//export by adding to templateXLS in default (new) 'Servoy Data' worksheet
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS);
//export by adding to templateXLS, in 'mySheet' worksheet, starting at default(1/1) row/column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet');
//export by adding to templateXLS, in 'mySheet' worksheet, starting at 3rd row and 5th column
var bytes = plugins.excelxport.excelExport(forms.form1.foundset, ['id','name'],templateXLS, 'mySheet',3,5);
```