

textxport

 Apr 07, 2024 13:24

Supported Clients

SmartClient WebClient NGClient

Property Summary

Boolean exportEnabled

Enable the export feature of this plugin.

Boolean importEnabled

Enable the import feature of this plugin.

Methods Summary

TabExporter	<code>createExporter(foundSet, separator, exportHeader)</code>	Create exporter for easier export set up.
String	<code>textExport(foundSet, dataProviderIds)</code>	Export to text 'separated value' data (*.
String	<code>textExport(foundSet, dataProviderIds, separator)</code>	Export to text 'separated value' data (*.
String	<code>textExport(foundSet, dataProviderIds, separator, exportHeader)</code>	Export to text 'separated value' data (*.

Property Details

exportEnabled

Enable the export feature of this plugin.

Returns

Boolean

Supported Clients

SmartClient

Sample

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

importEnabled

Enable the import feature of this plugin.

Returns

Boolean

Supported Clients

SmartClient

Sample

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

Methods Details

createExporter(foundSet, separator, exportHeader)

Create exporter for easier export set up. Can either use this method (for more complex exports) or textExport (...) API

Parameters

JSFoundSet	foundSet	the foundset to export with
String	separator	the separator of the data
Boolean	exportHeader	export a header

Returns

TabExporter

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
//export with ';' separator and no header
var exporter = plugins.textxport.createExporter(forms.form1.foundset,';',false);
```

textExport(foundSet, dataProviderIds)

Export to text 'separated value' data (*.tab/*.csv)

Parameters

JSFoundSet	foundSet	the foundset to export with
Array	dataProviderIds	the ids of the dataproviders

Returns

String

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
//export with default separator(tab) and no header
var dataToBeWritten = plugins.textxport.textExport(forms.form1.foundset,['id','name']);
```

textExport(foundSet, dataProviderIds, separator)

Export to text 'separated value' data (*.tab/*.csv)

Parameters

JSFoundSet	foundSet	the foundset to export with
Array	dataProviderIds	the ids of the dataproviders
String	separator	the separator of the data

Returns

String

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
//export with ';' separator and no header
var dataToBeWritten = plugins.textxport.textExport(forms.form1.foundset,['id','name'],';');
```

textExport(foundSet, dataProviderIds, separator, exportHeader)

Export to text 'separated value' data (*.tab/*.csv)

Parameters

JSFoundSet	foundSet	the foundset to export with
Array	dataProviderIds	the ids of the dataproviders
String	separator	the separator of the data
Boolean	exportHeader	true for exporting with the table header, false for not

Returns

String

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
//export with ';' separator and header
var dataToBeWritten = plugins.textxport.textExport(forms.form1.foundset,['id','name'],';',true);
```