

pdf_output

Method Summary

`byte[]` [#combinePDFDocuments\(array\)](#)
Combine multiple PDF docs into one.

`byte[]` [#combineProtectedPDFDocuments\(array, passwords\)](#)
Combine multiple protected PDF docs into one.

`byte[]` [#convertPDFFormToPDFDocument\(pdf_form, name_value_array\)](#)
Convert a PDF form to a PDF document.

`byte[]` [#convertProtectedPDFFormToPDFDocument\(pdf_form, pdf_password, name_value_array\)](#)
Convert a protected PDF form to a PDF document.

`byte[]` [#endMetaPrintJob\(\)](#)
Ends a previously started meta print job.

`byte[]` [#getPDFPrinter\(\[filename\]\)](#)
Returns a PDF printer that can be used in print calls.

`Number` [#getPagesPrinted\(\)](#)
Returns the number of pages printed by the last print call done in the context of a meta print job.

`Number` [#getTotalPagesPrinted\(\)](#)
Returns the total number of pages printed in the context of a meta print job.

`Number` [#insertFontDirectory\(\)](#)
Add a directory that should be searched for fonts.

`Boolean` [#startMetaPrintJob\(\)](#)
Used for printing multiple things into the same PDF document.

Method Details

`combinePDFDocuments`

`byte[]` **combinePDFDocuments**(array)
Combine multiple PDF docs into one.

Parameters

array

Returns

`byte[]`

Sample

```
//combine multiple pdf docs into one.  
pdf_blob_column = combinePDFDocuments(new Array(pdf_blob1,pdf_blob2,pdf_blob3));
```

`combineProtectedPDFDocuments`

`byte[]` **combineProtectedPDFDocuments**(array, passwords)
Combine multiple protected PDF docs into one.

Parameters

array

passwords

Returns

`byte[]`

Sample

```
//combine multiple protected pdf docs into one.  
pdf_blob_column = combineProtectedPDFDocuments(new Array(pdf_blob1,pdf_blob2,pdf_blob3), new Array  
(pdf_blob1_pass,pdf_blob2_pass,pdf_blob3_pass));
```

`convertPDFFormToPDFDocument`

`byte[]` **convertPDFFormToPDFDocument**(pdf_form, name_value_array)
Convert a PDF form to a PDF document.

Parameters

pdf_form

name_value_array

Returns

`byte[]`

Sample

```
var pdfform = plugins.file.readFile('c:/temp/1040a-form.pdf');
//var field_values = plugins.file.readFile('c:/temp/1040a-data.fdf');//read adobe fdf values or
var field_values = new Array();//construct field values
field_values[0] = 'f1-1=John C.J.'
field_values[1] = 'f1-2=Longlasting'
var result_pdf_doc = plugins.pdf_output.convertPDFFormToPDFDocument(pdfform, field_values)
if (result_pdf_doc != null)
{
    plugins.file.writeFile('c:/temp/1040a-flatten.pdf', result_pdf_doc)
}
```

convertProtectedPDFFormToPDFDocument

byte[] **convertProtectedPDFFormToPDFDocument**(pdf_form, pdf_password, name_value_array)

Convert a protected PDF form to a PDF document.

Parameters

pdf_form

pdf_password

name_value_array

Returns

byte[]

Sample

```
var pdfform = plugins.file.readFile('c:/temp/1040a-form.pdf');
//var field_values = plugins.file.readFile('c:/temp/1040a-data.fdf');//read adobe fdf values or
var field_values = new Array();//construct field values
field_values[0] = 'f1-1=John C.J.'
field_values[1] = 'f1-2=Longlasting'
var result_pdf_doc = plugins.pdf_output.convertProtectedPDFFormToPDFDocument(pdfform, 'pdf_password',
field_values)
if (result_pdf_doc != null)
{
    plugins.file.writeFile('c:/temp/1040a-flatten.pdf', result_pdf_doc)
}
```

endMetaPrintJob

byte[] **endMetaPrintJob**()

Ends a previously started meta print job. For meta print jobs that were stored in memory, not in a file on disk, also returns the content of the generated PDF document.

Returns

byte[]

Sample

```
//to print multiple forms to one pdf document (on file system).
var success = plugins.pdf_output.startMetaPrintJob('c:/temp/out.pdf')
if (success)
{
forms.form_one.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form one printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
forms.form_two.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form two printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
}
application.output('total printed pages: ' + plugins.pdf_output.getTotalPagesPrinted());
plugins.pdf_output.endMetaPrintJob()

//to print multiple forms to one pdf document (to store in dataprovider).
var success = plugins.pdf_output.startMetaPrintJob()
if (success)
{
forms.form_one.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form one printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
forms.form_two.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form two printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
}
application.output('total printed pages: ' + plugins.pdf_output.getTotalPagesPrinted());
mediaDataProvider = plugins.pdf_output.endMetaPrintJob()
```

[getPDFPrinter](#)

getPDFPrinter([filename])

Returns a PDF printer that can be used in print calls. If a file name is provided, then a PDF printer that generates a PDF into the specified file is returned. If no argument is provided, then the PDF printer corresponding to the last started meta print job is returned.

Parameters

[filename]

Sample

```
//to print current record without printdialog to pdf file in temp dir.
controller.print(true,false,plugins.pdf_output.getPDFPrinter('c:/temp/out.pdf'));
```

[getPagesPrinted](#)

getPagesPrinted()

Returns the number of pages printed by the last print call done in the context of a meta print job.

Returns

[Number](#)

Sample

```
//to print multiple forms to one pdf document (on file system).
var success = plugins.pdf_output.startMetaPrintJob('c:/temp/out.pdf')
if (success)
{
forms.form_one.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form one printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
forms.form_two.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form two printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
}
application.output('total printed pages: ' + plugins.pdf_output.getTotalPagesPrinted());
plugins.pdf_output.endMetaPrintJob()

//to print multiple forms to one pdf document (to store in dataprovider).
var success = plugins.pdf_output.startMetaPrintJob()
if (success)
{
forms.form_one.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form one printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
forms.form_two.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form two printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
}
application.output('total printed pages: ' + plugins.pdf_output.getTotalPagesPrinted());
mediaDataProvider = plugins.pdf_output.endMetaPrintJob()
```

[getTotalPagesPrinted](#)

Number [getTotalPagesPrinted\(\)](#)

Returns the total number of pages printed in the context of a meta print job. Call this method before ending the meta print job.

Returns

[Number](#)

Sample

```
//to print multiple forms to one pdf document (on file system).
var success = plugins.pdf_output.startMetaPrintJob('c:/temp/out.pdf')
if (success)
{
forms.form_one.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form one printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
forms.form_two.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form two printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
}
application.output('total printed pages: ' + plugins.pdf_output.getTotalPagesPrinted());
plugins.pdf_output.endMetaPrintJob()

//to print multiple forms to one pdf document (to store in dataprovider).
var success = plugins.pdf_output.startMetaPrintJob()
if (success)
{
forms.form_one.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form one printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
forms.form_two.controller.print(false,false,plugins.pdf_output.getPDFPrinter());
application.output('form two printed ' + plugins.pdf_output.getPagesPrinted() + ' pages. ');
}
application.output('total printed pages: ' + plugins.pdf_output.getTotalPagesPrinted());
mediaDataProvider = plugins.pdf_output.endMetaPrintJob()
```

[insertFontDirectory](#)

Number [insertFontDirectory\(\)](#)

Add a directory that should be searched for fonts. Call this only in the context of an active meta print job.

Returns

[Number](#)

Sample

```
//Insert font directories for font embedding.  
//You must create an MetaPrintJob before using it.  
plugins.pdf_output.insertFontDirectory('c:/Windows/Fonts');  
plugins.pdf_output.insertFontDirectory('c:/WinNT/Fonts');  
plugins.pdf_output.insertFontDirectory('/Library/Fonts');
```

[startMetaPrintJob](#)

Boolean [startMetaPrintJob\(\)](#)

Used for printing multiple things into the same PDF document. Starts a meta print job and all print calls made before ending the meta print job will be done into the same PDF document. If a file name is specified, then the PDF document is generated into that file. If no argument is specified, then the PDF document is stored in memory and can be retrieved when ending the meta print job and can be saved into a dataprovider, for example.

Returns

[Boolean](#)

Sample

```
//to print multiple forms to one pdf document (on file system).  
var success = plugins.pdf_output.startMetaPrintJob('c:/temp/out.pdf')  
if (success)  
{  
  forms.form_one.controller.print(false,false,plugins.pdf_output.getPDFPrinter());  
  application.output('form one printed ' + plugins.pdf_output.getPagesPrinted() + ' pages.');
```

```
  forms.form_two.controller.print(false,false,plugins.pdf_output.getPDFPrinter());  
  application.output('form two printed ' + plugins.pdf_output.getPagesPrinted() + ' pages.');
```

```
}  
application.output('total printed pages: ' + plugins.pdf_output.getTotalPagesPrinted());  
plugins.pdf_output.endMetaPrintJob()  
  
//to print multiple forms to one pdf document (to store in dataprovider).  
var success = plugins.pdf_output.startMetaPrintJob()  
if (success)  
{  
  forms.form_one.controller.print(false,false,plugins.pdf_output.getPDFPrinter());  
  application.output('form one printed ' + plugins.pdf_output.getPagesPrinted() + ' pages.');
```

```
  forms.form_two.controller.print(false,false,plugins.pdf_output.getPDFPrinter());  
  application.output('form two printed ' + plugins.pdf_output.getPagesPrinted() + ' pages.');
```

```
}  
application.output('total printed pages: ' + plugins.pdf_output.getTotalPagesPrinted());  
mediaDataProvider = plugins.pdf_output.endMetaPrintJob()
```