

ajabon Feb 25, 2018 8:06 PM

[Ps] PDFOpenOptionsにおけるWidthおよびHeightについて

This question has been **Answered**.

PhotoshopCC2017とCS6を取り扱っております。

標題の通りですが、PDFOpenOptionsの幅と高さを制御できず困っています。

ESTK(CC)のオブジェクトモデルビューアではGenericPDFOptionsというクラスがあるようですが...

Width、Heightどちらも説明文に DEPRECATED とあります。使えないらしいです。

当面、解像度だけで目的のピクセル数になるよう計算して、PDFを開いてから再サンプルオフで画像解像度を調整してしのいでいます。

もっといい方法はあるのでしょうか。

また、スクリプトからPDFを開く際の制御できずに開かれた幅・高さに法則性はあるのでしょうか。

以上2点、ご意見お〜待ちしております。



Correct Answer

by ruser3t30054468 on Mar 1, 2018 8:28 PM

scriptlistenerから吐き出されたログのコードを使えばWidthとHeightを制御出来ました。

理解不足なのでほぼ吐いたログそのままですけど、一応ESTKから動くことまでは確認しました。

```
var myFile = new File( "file.pdf" )
var ppi = 400;
UnitValue.baseUnit = UnitValue (1 / ppi, "in");
var wdt = UnitValue (123, "mm");
var hgt = UnitValue (506, "mm");

var idOpn = charIDToTypeID( "Opn " );
var desc4 = new ActionDescriptor();
```

```
var idAs = charIDToTypeID( "As " );
var desc5 = new ActionDescriptor();
var idNm = charIDToTypeID( "Nm " );
desc5.putString( idNm, decodeURI (myFile.name) );

var idCrop = charIDToTypeID( "Crop" );
var idcropTo = stringIDToTypeID( "cropTo" );
var idmediaBox = stringIDToTypeID( "mediaBox" );
desc5.putEnumerated( idCrop, idcropTo, idmediaBox );
//~ resolution
var idRslt = charIDToTypeID( "Rslt" );
var idRsl = charIDToTypeID( "#Rsl" );

desc5.putUnitDouble( idRslt, idRsl, ppi );
//~ ColorMode
var idMd = charIDToTypeID( "Md " );
var idClrS = charIDToTypeID( "ClrS" );
//~ var idRGBC = charIDToTypeID( "RGBC" );//RGB
//~ desc5.putEnumerated( idMd, idClrS, idRGBC );
var idECMY = charIDToTypeID( "ECMY" );//CMYK
desc5.putEnumerated( idMd, idClrS, idECMY );
//~ BitDepth
var idDpth = charIDToTypeID( "Dpth" );
desc5.putInteger( idDpth, 8 );
//~ AntiAlias
var idAntA = charIDToTypeID( "AntA" );
desc5.putBoolean( idAntA, true );
//~ Width と Height
var idWdth = charIDToTypeID( "Wdth" );
var idPxl = charIDToTypeID( "#Pxl" );
desc5.putUnitDouble( idWdth, idPxl, width.as ("px") );
var idHght = charIDToTypeID( "Hght" );
var idPxl = charIDToTypeID( "#Pxl" );
desc5.putUnitDouble( idHght, idPxl, height.as ("px") );
//~ ConstrainProportions 獣王妃
var idCnsP = charIDToTypeID( "CnsP" );
desc5.putBoolean( idCnsP, false );

var idsuppressWarnings = stringIDToTypeID( "suppressWarnings" );
desc5.putBoolean( idsuppressWarnings, false );
```

```
var idRvrs = charIDToTypeID( "Rvrs" );
desc5.putBoolean( idRvrs, false );

var idfsel = charIDToTypeID( "fsel" );
var idpdfSelection = stringIDToTypeID( "pdfSelection" );
var idpage = stringIDToTypeID( "page" );
desc5.putEnumerated( idfsel, idpdfSelection, idpage );

var idPgNm = charIDToTypeID( "PgNm" );
desc5.putInteger( idPgNm, 1 );

var idPDFG = charIDToTypeID( "PDFG" );
desc4.putObject( idAs, idPDFG, desc5 );
var idnull = charIDToTypeID( "null" );
desc4.putPath( idnull, myFile );
var idDocI = charIDToTypeID( "DocI" );
desc4.putInteger( idDocI, 196 );
executeAction( idOpn, desc4, DialogModes.NO );
```

スクリプトから開いた際の幅と高さの件は、PDFをダイアログから開いた際の設定で開いていると思います。

その他、スマートオブジェクト化の有無やクロップ設定でぐちゃっとするかもです。

検証不足ですいません。

いくらかでもお役に立てればよいですが。長々とすいませんでした。

671 Views Tags :

3 HELPFUL

MOST LIKED

10 A Feb 26, 2018 12:34 AM (in response to ajabon)

Re: [Ps] PDFOpenOptionsにおけるWidthおよびHeightについて

PDFファイルを直接FileオブジェクトとしてopenしてreadInしてCropBoxやMediaBoxのバウンズを検索して抽出するというのはどうですか？

これらはpt単位で記述されているので計算も楽ですよ。

Actions ▾

Edit

Delete

Like (0)



ajabon Feb 26, 2018 7:40 PM (in response to 10 A)

Re: [Ps] PDFOpenOptionsにおけるWidthおよびHeightについて

ご提案ありがとうございます。検証が遅くなりました。
InDesignで書き出したPDFだと77行目にまとまってBounds情報が仕込まれてますね。
事務所のコピー機でスキャンしたPDFでは9行目にMediaBoxのみありました。
得意先支給のWord由来のPDFでは見つけれませんでした。。

Photoshopで開くとサイズはオリジナルまま、解像度は任意なのだとすれば、一度てきとうに開いてみて、画像のBoundsを拾って一旦閉じてまた開く、が手軽な気もしないでもないような...ピクセル数で丸められるので誤差は出そうですけど。

1 person found this helpful

Actions ▾

Edit

Delete

Report Abuse

Liked (1)



10 A Feb 26, 2018 7:48 PM (in response to ajabon)

Re: [Ps] PDFOpenOptionsにおけるWidthおよびHeightについて

残念ながらPDFのラスタライズ設定を弄れない限りは開いてバタバタする必要があるのが難点ですね。

Actions ▾

Edit

Delete

Like (0)

ruser3t30054468 Mar 1, 2018 8:28 PM (in response to ajabon)

Correct Answer Re: [Ps] PDFOpenOptionsにおけるWidthおよびHeightについて

scriptlistenerから吐き出されたログのコードを使えばWidthとHeightを制御出来ました。

理解不足なのでほぼ吐いたログそのままですけど、一応ESTKから動くことまでは確認しました。

```
var myFile = new File( "file.pdf" )
var ppi = 400;
UnitValue.baseUnit = UnitValue (1 / ppi, "in");
var wdt = UnitValue (123, "mm");
var hgt = UnitValue (506, "mm");
```

```
var idOpn = charIDToTypeID( "Opn " );
var desc4 = new ActionDescriptor();
var idAs = charIDToTypeID( "As " );
var desc5 = new ActionDescriptor();
var idNm = charIDToTypeID( "Nm " );
desc5.putString( idNm, decodeURI (myFile.name) );

var idCrop = charIDToTypeID( "Crop" );
var idcropTo = stringIDToTypeID( "cropTo" );
var idmediaBox = stringIDToTypeID( "mediaBox" );
desc5.putEnumerated( idCrop, idcropTo, idmediaBox );
//~ resolution
var idRslt = charIDToTypeID( "Rslt" );
var idRsl = charIDToTypeID( "#Rsl" );

desc5.putUnitDouble( idRslt, idRsl, ppi );
//~ ColorMode
var idMd = charIDToTypeID( "Md " );
var idClrS = charIDToTypeID( "ClrS" );
//~ var idRGBC = charIDToTypeID( "RGBC" );//RGB
//~ desc5.putEnumerated( idMd, idClrS, idRGBC );
var idECMY = charIDToTypeID( "ECMY" );//CMYK
desc5.putEnumerated( idMd, idClrS, idECMY );
//~ BitDepth
var idDpth = charIDToTypeID( "Dpth" );
desc5.putInteger( idDpth, 8 );
//~ AntiAlias
var idAntA = charIDToTypeID( "AntA" );
desc5.putBoolean( idAntA, true );
//~ Width と Height
var idWdth = charIDToTypeID( "Wdth" );
var idPxl = charIDToTypeID( "#Pxl" );
desc5.putUnitDouble( idWdth, idPxl, width.as ("px") );
var idHght = charIDToTypeID( "Hght" );
var idPxl = charIDToTypeID( "#Pxl" );
desc5.putUnitDouble( idHght, idPxl, hght.as ("px") );
//~ ConstrainProportions 獣王妃
var idCnsP = charIDToTypeID( "CnsP" );
desc5.putBoolean( idCnsP, false );
```

```
var idsuppressWarnings = stringIDToTypeID( "suppressWarnings" );
desc5.putBoolean( idsuppressWarnings, false );

var idRvrs = charIDToTypeID( "Rvrs" );
desc5.putBoolean( idRvrs, false );

var idfsel = charIDToTypeID( "fsel" );
var idpdfSelection = stringIDToTypeID( "pdfSelection" );
var idpage = stringIDToTypeID( "page" );
desc5.putEnumerated( idfsel, idpdfSelection, idpage );

var idPgNm = charIDToTypeID( "PgNm" );
desc5.putInteger( idPgNm, 1 );

var idPDFG = charIDToTypeID( "PDFG" );
desc4.putObject( idAs, idPDFG, desc5 );
var idnull = charIDToTypeID( "null" );
desc4.putPath( idnull, myFile );
var idDoc1 = charIDToTypeID( "Doc1" );
desc4.putInteger( idDoc1, 196 );
executeAction( idOpn, desc4, DialogModes.NO );
```

スクリプトから開いた際の幅と高さの件は、PDFをダイアログから開いた際の設定で開いていると思います。

その他、スマートオブジェクト化の有無やクロップ設定でぐちゃっとするかもです。

検証不足ですいません。

いくらかでもお役に立てればよいですが。長々とすいませんでした。

3 people found this helpful

Actions ▾

 Edit

 Delete

 Report Abuse

 Liked (3)



ajabon Mar 1, 2018 9:59 PM (in response to ruser3t30054468)

Re: [Ps] PDFOpenOptionsにおけるWidthおよびHeightについて

ありがとうございます、目的を果たせました。

ScriptListener常套手段なのにPDFOpenOptionsに捉われすぎてすっかり忘れておりました。。 [ruser3t30054468](#)

**10 A** Mar 1, 2018 10:39 PM (in response to ruser3t30054468)

ACP

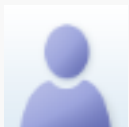
Re: [Ps] PDFOpenOptionsにおけるWidthおよびHeightについて

わたしも別件（この近所にあるスレ）でスクリプトリスナー使ってやるとUIのプロパティセットできるんじゃないかって書いてました。やはりPSの場合はアクションディスクリプタが要になりますね。

これってアクションディスクリプタ定義してそれにアクションリファレンスやなんやかんやと設定して実行させるという流れです。ほとんどメニュー操作と同じ事を出来るようになっていますが、ややこしいのがtypeIDという数字を設定してあげないといけないのです。これを把握するのは至難の業なので人間が見ても（比較的）理解しやすいcharIDというのが紐付けされています。それで所々でこいつらを変換するためのcharIDToTypeIDというメソッドが入ってくる事で限りなく見通しが悪くなります。この辺を押さえた上で見て頂けると分かりやすく成るかもしれません。これらの対比表はPhotoshopSDKのヘッダファイルを参照するのが一番良いかと思われます。

...しかし「獣王妃」ってなんだろう

Actions ▾

 Edit Delete Like (0)**ruser3t30054468** Mar 2, 2018 12:12 AM (in response to 10 A)

Re: [Ps] PDFOpenOptionsにおけるWidthおよびHeightについて

自分がわかればいいやと思って放置したタイプミスをそのままコピペしてしまいました恥ずかしい！

4-charIDについての文書が見つからず、RGBとCMYKのカラーモード変更もリスナーに吐かせなおしてました。CMYKが素直に"CMYK"じゃないあたりが難解で。

きれいに書き直せるとよかったんですが、知識も技術も足りてないもので、すいません。

お二方とものサイトには随分お世話になっておりますので、お役に立てたようであれば幸いです。

Actions ▾

 Edit Delete Report Abuse Like (0)



ACP

Re: [Ps] PDFOpenOptionsにおけるWidthおよびHeightについて

charIDとtypeIDについてですが、Phototshop SDKの中のサンプルコード内
pluginsdk/photoshopapi/photoshop/PITerminology.h
common/include/OLEConsts.h
この2点のヘッダファイルが参考になるでしょう。

3 people found this helpful

Actions ▾

 Edit

 Delete

 Like (1)