

yusuke.s Mar 14, 2018 5:25 PM

## [ID]開いたドキュメントを強制的に閉じるイベント

This question has been **Answered**.

自動化フォーラムの皆さま、こんばんは。

タイトルの通りのイベントをInDesignに仕込みたいと思っています。

```
01. #targetengine "testOpen"
02. app.addEventListener("afterOpen", function (e){
03.     if (confirm("ドキュメントを閉じてもいいですか")) e.target.parent.close(SaveOption
04. });

01. #targetengine "testOpen"
02. app.addEventListener("afterOpen", function (e){
03.     if (confirm("ドキュメントを閉じてもいいですか")) e.target.parent.close(SaveOption
04. });
```

これがうまくいきません。

Event.preventDefaultメソッドや、stopPropagationメソッドなんかも試したのですがうまくいかず。

本当にやりたいことは

- ・ドキュメントを開く前に「開いていいか聞く」
- ・開いてよければドキュメントをそのまま開く
- ・開いてはいけないうらpreventDefaultメソッドでドキュメントを開くことを阻止する

みたいな処理です。

でもEventListenerのbeforeOpenイベントを使ってしまうと、いままさに開こうとしているドキュメントをEventから受け取ることができません。当たり前ですが。なのでドキュメントを開いて（afterOpen）、開きたくなかったらそのドキュメントを閉じる、という苦肉の策で挑んだのですが、上の通りうまくいきません。

このような処理を実装したいのですが、もし解決策、ないしアイディアでもいただければ嬉しいです。

よろしくお願いします。

※Acrobatコミュニティフォーラム (Japan) から アプリケーション自動化総合コミュニティフォーラム (Japan) に移動しました[moderator]



### Correct Answer

by 10 A on Mar 15, 2018 5:02 PM

これ、beforeOpenでいけますね。以下の様にするると開く前にアラートでファイル名取れます。

```
01. #targetengine session
02. if (app.eventListeners.itemByName('test')!=null){
03.     app.eventListeners.itemByName('test').remove();
04. }
05.
06. function addLsnr(){
07.     var el = app.addEventListener(
08.         "beforeOpen",
09.         function cb(e){
10.             alert(e.fullName);
11.             //e.stopPropagation();
12.             //e.preventDefault();
13.         });
14.     el.name = "test";
15. }
16.
17. addLsnr ();
```

ちなみにリスナが返すオブジェクトのプロパティはuserInteractionLevel、fullName、及びcurrentTargetとなります。  
currentTargetはアプリケーション自身、fullNameはファイルパスです。

1516 Views Tags : [indesign](#), [extend script](#), [event handler](#)

## 1 HELPFUL

10 A Mar 15, 2018 5:02 PM (in response to yusuke.s)

**Correct Answer** Re: [ID]開いたドキュメントを強制的に閉じるイベント

これ、beforeOpenでいけますね。以下の様にするると開く前にアラートでファイル名取れます。

```
01. #targetengine session
02. if (app.eventListeners.itemByName('test')!=null){
03.     app.eventListeners.itemByName('test').remove();
04. }
```



ACP



```

05.
06. function addLsnr(){
07. var el = app.addEventListener(
08. "beforeOpen",
09. function cb(e){
10. alert(e.fullName);
11. //e.stopPropagation();
12. //e.preventDefault();
13. });
14. el.name = "test";
15. }
16.
17. addLsnr ();

01. #targetengine session
02. if (app.eventListeners.itemByName('test')!=null){
03.     app.eventListeners.itemByName('test').remove();
04. }
05.
06. function addLsnr(){
07. var el = app.addEventListener(
08. "beforeOpen",
09. function cb(e){
10. alert(e.fullName);
11. //e.stopPropagation();
12. //e.preventDefault();
13. });
14. el.name = "test";
15. }
16.
17. addLsnr ();

```

ちなみにリスナが返すオブジェクトのプロパティはuserInteractionLevel、fullName、及びcurrentTargetとなります。

currentTargetはアプリケーション自身、fullNameはファイルパスです。

Actions ▾

 Edit  Delete  Like (1)

yusuke.s Mar 15, 2018 5:35 PM (in response to 10 A)

Re: [ID]開いたドキュメントを強制的に閉じるイベント

Ten Aさま、いつもありがとうございます。

event.targetがApplicationだったので取得できていないのかと思っていましたが、fullNameで開こうとしているファイルが取れるのですね。勉強になりました！

```

01. #targetengine "test"
02.
03. app.addEventListener ("beforeOpen", function (e){
04.     if (confirm (e.fullName+"\nこのファイルを開いていいですか")) return;
05.     e.preventDefault();
06. });

01. #targetengine "test"
02.
03. app.addEventListener ("beforeOpen", function (e){

```

```
04.     if (confirm (e.fullName+"\nこのファイルを開いていいですか")) return;  
05.     e.preventDefault();  
06.     });
```

ちょっと雑ですが、これでいけました。  
ありがとうございました！すっきりしました。

Actions ▾

 Edit

 Delete

 Report Abuse

 Like (0)



ACP

10 A Mar 15, 2018 6:08 PM (in response to yusuke.s)

Re: [ID]開いたドキュメントを強制的に閉じるイベント

表題に関しては

```
01. app.addEventListener("afterOpen", callBackFunc);  
01. app.addEventListener("afterOpen", callBackFunc);
```

このようにするとイベントを捉えられるはずですが、しかしながらドキュメントを開いた後にイベントが発生するためにプロパゲーションを止めることも基本の挙動のキャンセルも出来ません。before-afterの大きく違う挙動ですからしっかり押さえておきたいポイントです。

また、オブジェクトモデルに関しては世の中に様々な物が出ていますが、わたしやAdobe社がリリースしているものも含めて丸々信用するのは危険です。実際に挙動が不審に感じた場合は以下の様にオブジェクトとそのプロパティを洗い出す作業を行うことをおすすめします。

```
01. for (a in targetObject) $.writeln(a + " : " + targetObject[a]);  
01. for (a in targetObject) $.writeln(a + " : " + targetObject[a]);
```

これはPropatyExplorerのコアにも利用されているロジックです。ドキュメント化されていないオブジェクトに関しても現在の値諸共抜き出して記録できますからコーディング前の予備 調査に威力を発揮するでしょう。

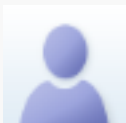
2 people found this helpful

Actions ▾

 Edit

 Delete

 Like (0)



yusuke.s Mar 15, 2018 6:33 PM (in response to 10 A)

Re: [ID]開いたドキュメントを強制的に閉じるイベント

after○○にpreventDefaultなどが使えないの、すっかり失念していました。



for in文でプロパティの洗い出し、大事ですね。

「こうならないからできない」じゃなくて、周辺状況含めて全部確認する必要性を再認識しました。

InDesignスクリプトはできる子！ それを信じてプロパティを洗い出してあげるようにします。

Actions ▾



Edit



Delete



Report Abuse



Liked (1)