

Function 一覧

SMPS_Exec	1
SMPS_ExecSync	2
SMPS_Get("LastError")	2
SMPS_Get("Result")	2
SMPS_Get("PSThreadState")	2
SMPS_Lock	3
SMPS_Version	3

SMPS_Exec

※フィールド | タイプ : 計算 での使用は、非推奨です。
 スクリプトで、「変数を設定」や「フィールド設定」で使用して下さい。

目的

PowerShell スクリプトを実行

構文

SMPS_Exec(Script)

引数

PowerShell スクリプト

戻り値のデータタイプ

テキスト

説明

SMPS_Exec(Script)を実行すると FileMaker は、30 秒間は、PowerShell Script の終了を待ちます。PowerShell Script の処理が 30 秒以上かかっている FileMaker にコントロールが戻りますが、PowerShell Script の処理は、継続されます。PowerShell Script の状況は、SMPS_Get("PSThreadState")で確認できます。

例 :

「フィールド:Script」に記述した PowerShell を実行。

フィールド:Script 値 : (ファイル名を取得)

Get-ChildItem "C:¥Program Files (x86)¥FileMaker¥FileMaker Pro 13"

変数を設定 [\$Result; 値:SMPS_Exec(SMPS::Script)]

戻り値 :

FileMaker Pro.exe

FileMaker Pro.exe.config

FMEngine.dll

FMP Acknowledgements.pdf

FMP お読みください.pdf

FMP ライセンス.pdf

.....

例 :

スクリプトに直接 PowerShell を記述。

変数を設定 [\$Result; 値:SMPS_Exec("Get-ChildItem ¥"C:¥Program Files (x86)¥FileMaker¥FileMaker Pro 13¥"")]

SMPS_ExecSync

※フィールド | タイプ : 計算 での使用は、非推奨です。
スクリプトで、「変数を設定」や「フィールド設定」で使用して下さい。

目的
PowerShell スクリプトを実行。

構文
SMPS_ExecSync(Script)

引数
PowerShell スクリプト

戻り値のデータタイプ
テキスト

説明
SMPS_ExecSync(Script) を実行すると FileMaker は、PowerShell Script が終了を待ちます。
終了しない PowerShell Script を実行した場合、FileMaker は、待機状態から戻りません。

例：
コンピューター名を取得。
変数を設定 [\$Result; 値: SMPS_ExecSync("\$Env:COMPUTERNAME")]

エラーを取得したい場合：
以下を PowerShell コードの前後に追加

```
$ERROR.Clear();  
$ERROR[0];
```

例：
変数を設定 [\$Result; 値: SMPS_ExecSync("\$ERROR.Clear(); \$Env:COMPUTERNAME; \$ERROR[0];")]

SMPS_Get("LastError")

SMPS_Exe(Script) を実行した場合のエラーを返します。

正常終了時の戻り値
0

SMPS_Get("Result")

SMPS_Exe(Script) を実行した場合に使用。

SMPS_EXE の戻り値を SMPS_Get("Result") で取得することが出来ます。
SMPS_EXE 実行後、30 秒が過ぎ、タイムアウトした場合、SMPS_EXE の戻り値が空欄になります。
SMPS_Get("PSThreadState") で、Stopped(PowerShell の処理が完了しスレッドが停止している。) を確認後、
SMPS_Get("Result") で値を取得します。

SMPS_Get("Result") は、
SMPS_EXE の処理が 30 秒以上かかる場合に使用します。

SMPS_Get("PSThreadState")

SMPS_Exe(Script) を実行した場合に使用。

PowerShell の実行は、別スレッドで行われています。

SMPS_Get("PSThreadState") は、スレッドの状態を返します。

SMPS_EXE を実行した場合、タイムアウトは、30 秒です。
PowerShell の処理が終わらず、30 秒以上時間がたった場合、FileMaker にコントロールが戻ります。
SMPS_Get("PSThreadState") で処理の状態を取得し、処理の完了を判断します。

戻り値：
Running | 実行中。
Stopped | PowerShell の処理が完了しスレッドが停止している。
WaitSleepJoin | スレッドがブロックされている。

SMPS_Lock

SMPS_Exe(Script) を実行する場合に使用。

SMPS_Lock(int)

ScriptMakerPS v1 以降

構文

SMPS_Lock(0 or 1)

引数

0 or 1

1 = プラグインの使用開始を宣言する。

0 = プラグインの使用終了を宣言する。

戻り値

0 or タイムスタンプ

引数(1)の場合、
別のプロセスで既にプラグインの使用開始宣言がされている場合、
その時のタイムスタンプ
プラグインの使用開始宣言がされていない場合、
0

引数(0)の場合、

0

説明

FileMaker Server スケジュールで **【ScriptMakerPS】** を使用する場合の排他制御の為に使用します。

※FileMaker Pro(クライアント)で **【ScriptMakerPS】** を利用の場合は、特に「SMPS_Lock(int)」関数を設定する必要は、ありません。

※FileMaker Server スケジュールから実行する場合、「ScriptMakerPS」を使用したスクリプトの実行が、重ならない様に設定すると共に「SMPS_Lock(int)」関数で明示的に排他制御を行なって下さい。

※注意!

「SMPS_Lock(int)」関数は、プラグインの使用開始・終了のフラグを立てているだけで、この関数自体で排他制御をしているわけではありません。

排他制御は、FileMaker のスクリプトで行なって下さい。

例 :

#プラグインの使用開始を宣言する。
変数を設定 [\$Lock; 値:SMPS_Lock(1)]

既にプラグインの使用開始宣言がされている場合、
その時のタイムスタンプ 例 : 2016/12/27 17:18:30 が返ります。

プラグインの使用開始宣言がされていない場合、「0」 が返ります。

戻り値が「0」の場合、処理をすすめる。

戻り値が タイムスタンプ 例 : 2016/12/27 17:18:30 (戻り値が「0」でない)の場合、処理を停止するように FileMaker スクリプトを設定して下さい。

#プラグインの使用終了を宣言する。
変数を設定 [\$Lock; 値:SMPS_Lock(0)]

SMPS_Version

True:Registered:ScriptMakerPS Version 2.0.0.2

True:+OK 2018/12/31 23:59:59:ScriptMakerPS Version 2.0.0.2

False::ScriptMakerPS Version 2.0.0.2