

CX's Memo (Windows関連)

[Latest] [Edit] [Config] [Help] | [Group] [Home] [PV:12249]

2006 | [04](#) |

2009 | [05](#) | [06](#) | [09](#) |

2010 | [02](#) |

[<\[Windows 7\]\[評価版\]](#) [\[Windows 7 Enterpris...](#)

2010-02-03

■ Windowsコマンドプロンプト基礎文法最速マスター

Windowsコマンドプロンプトの文法一覧です。他の言語をある程度知っている人はこれを読めばコマンドプロンプトの基礎をマスターしてバッチファイルを書くことができるようになります。簡易リファレンスとしても利用できます。

1. 基礎

メッセージ表示

```
ECHO Hello, world.
```

コメント

```
REM コメント
```

変数の宣言

SET コマンドで変数の設定を行います。変数名と「=」の間は空白を入れず詰めて入力するようにしましょう(空白を含んだ変数名が設定されてしまう為)

```
SET 変数名=[文字列]
SET /A 変数名=[数式]
```

変数を参照する場合は、%変数名% と「%」で囲む必要があります。

```
ECHO %変数名%
```

バッチファイルの実行

バッチファイルを実行するにはコマンドラインで次のようにします。

```
script.bat [Enter]
```

または

```
script.cmd [Enter]
```

.BAT (Windows バッチファイル) と .CMD (Windows コマンド スクリプト) の大きな違いはありません。

CX's Bookmark / Windows

7 (87)

- [「Windows 7」の管理機能](#)
- [窓の杜 - \[NEWS\]MS、Win](#)
- [【PC Watch】Windows 7を搭](#)
- [【山田祥平のWindows 7カ](#)
- [窓の杜 - \[NEWS\]MS、機](#)
- [窓の杜 - \[NEWS\]物理](#)
- [【PC Watch】Microsoftのヒ](#)
- [窓の杜 - \[NEWS\]マイ](#)
- [【山田祥平のWindows 7カ](#)
- [【PC Watch】続・意外に](#)

CX's Bookmark / MS (982)

- [詳細初リーク! マイ](#)
- [マイクロソフト勉強会](#)
- [PDC 09資料で見る、2010](#)
- [窓の杜 - \[REVIEW\]IE6/7/](#)
- [【PC Watch】マイクロソ](#)
- [\[速報\]Internet Explorer](#)
- [Internet Explorer 9が今晚](#)
- [\[PDC09\]マイクロソフ](#)
- [【PC Watch】Microsoftのソ](#)
- [\[速報\]マイクロソフ](#)

CX's Bookmark

- [Windowsコマンドプロン](#)
- [基礎文法最速マスター](#)
- [はてなブログラミン](#)
- [Bash基礎文法最速マス](#)
- [UWSC 基礎文法最速マス](#)
- [VBA基礎文法最速マスタ](#)
- [VBScript 基礎文法最速マ](#)
- [Windows Script Wiki - エデ](#)
- [詳細初リーク! マイ](#)
- [ウィルコム、会社更生](#)

CX's Antenna

- 02/04 07:41
[\[impress\] 窓の杜](#)
- 02/04 07:03
[\[InfoQ\] InfoQ Japan](#)
- 02/04 06:55
[\[ITNAVI\] セキュリティ](#)
- 02/04 06:55
[\[@IT\]](#)
- 02/04 06:30
[\[impress\] PC](#)
- 02/04 06:28
[\[impress\] やじうまWatch](#)
- 02/04 06:28
[\[impress\] INTERNET](#)
- 02/04 06:27

同じスクリプトに同じ名前を付けた場合は .BAT が優先されます。優先順位は SET PATHEXT にて確認できます。

```
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
```

出力結果をファイルに書き出すにはリダイレクトを使います。「>」は上書きで「>>」は追記になります。

```
script.bat > file.txt [Enter]
```

2. 数値

数値の表現

SET コマンドの /A オプションを用いることで変数に数値を代入することができます。なお、代入できるのは整数(32bit 値)のみで、小数は代入できません。

```
SET /A num=-2147483647 ... OK
SET /A num=0 ... OK
SET /A num=123 ... OK
SET /A num=2147483647 ... OK
SET /A num=2147483648 ... NG : 「無効な数字です。数値は 32 ビットで表記され
SET /A num=1.23 ... NG : 「演算子がありません。」
```

数値は、16進数、8進数表記での設定も可能です。

```
SET /A num=0x12 ... OK : 16進数 (値は28)
SET /A num=022 ... OK : 8進数 (値は28)
SET /A num=09 ... NG : 「無効な数字です。数値定数は 10 進 (17 桁)。
```

四則演算

四則演算です。

```
SET /A num=1+1 ... 2
SET /A num=1-1 ... 0
SET /A num=1*2 ... 2
SET /A num=1/2 ... 0 : 小数は切り捨て (商)
SET /A num=1/0 ... NG : 0 除算エラーです。
```

余りの求め方です。

```
SET /A num=3%%2 ... 1 (余り)
```

インクリメントとデクリメント

インクリメントとデクリメントです

```
REM インクリメント
SET /A num+=1

REM デクリメント
SET /A num-=1
```

3. 文字列

文字列の表現

[\[GIGAZINE\]](#)

9. 02/04 06:26

[\[ASCII.jp\]](#)

10. 02/04 06:17

[\[ZDNet\]](#)

文字列には引用符(ダブルクォーテーション等)を指定する必要はありません。また「=」の後の空白も文字列として処理されます。

```
SET str1=abc
SET str2=de f
SET str2= g hi
```

文字列操作

文字列を結合する方法です。

```
REM 結合
SET str1=aaa
SET str2=bbb
SET str3=%str1%%str2%           ... aaabbb
```

配列に分割する機能はありません。

文字列の長さを求めるコマンドはありませんが、以下のようなバッチファイルで代用可能です。

```
REM File : GetLength.bat
REM Usage : GetLength.bat abcde
IF "%1"==" " GOTO :EOF
SET /A LENGTH=0
SET LAST=%1

:LOOP
SET /A LENGTH+=1
SET LAST=%LAST:~1%
IF NOT "%LAST%"==" " GOTO LOOP

EXIT /B %LENGTH%
```

文字列から任意の位置の場所の文字を切り出す方法です。

```
REM 切り出し
SET str1=abcd
SET str2=%str1:~0,2%           ... ab (1桁目から2文字)
```

4. 配列

配列はありません。

5. ハッシュ

ハッシュ変数はありません。

6. 制御文

IF文

IF文です。1行に書く形式とブロック形式の構文が利用できます。

```
IF 条件 処理
```

ブロック形式のIF文には()を用います。

```
IF 条件 (
    処理
)
```

条件の箇所には、以下のような条件を指定することができます。

```
IF [NOT] ERRORLEVEL 番号 コマンド
IF [NOT] 文字列1==文字列2 コマンド
IF [NOT] EXIST ファイル名 コマンド
IF 文字列1 比較演算子 文字列2 コマンド
```

比較演算子は次のいずれかになります。

```
EQU - 等しい
NEQ - 等しくない
LSS - より小さい
LEQ - 以下
GTR - より大きい
GEQ - 以上
```

IF ~ ELSE 文

```
IF 条件 (
    処理
) ELSE (
    処理
)
```

条件ループ

WHILE 文はありませんが、GOTO 文と条件式を組み合わせることで代用可能です。

```
SET /A 変数=0

:LOOP

処理

IF "%変数%"=="終了値" GOTO END

SET /A 変数+=1

GOTO LOOP

:END
```

FOR 文

```
FOR /L %%変数 IN (開始,増分,終了) DO (
    処理
)
```

7. サブルーチン

CALL コマンドでバッチファイルを呼び出すか、:ラベル名を指定します。引数は %1, %2, %3, ... としてセットされます。

サブルーチンの戻り値は EXIT /B [終了コード] を指定することで ERRORLEVEL として返却可能です。

```
CALL :SUM 1, 2
ECHO %ERRORLEVEL%

GOTO END

:SUM
SET /A num1=%1
SET /A num2=%2
SET /A total=num1+num2

EXIT /B %total%

:END
```

8. ファイル入出力

FOR コマンドの /F オプションを利用することでファイルを解析し読み込むことが可能です。

以下のサンプルはファイルを1行ずつ読み込み出力ファイルに追記するサンプルです。

```
FOR /F "tokens=*" %i IN (input.txt) DO ECHO %i>> output.txt
```

その他の基礎文法マスター

この記事は他の基本文法マスターに便乗して書いた物です。誤り等ございましたらコメント等して頂けると助かります。

- [Perl基礎文法最速マスター - Perl入門～サンプルコードによるPerl入門～](#)
- [Route 477 - Ruby基礎文法最速マスター](#)
- [PHP基礎文法最速マスター | Shin x blog](#)
- [Python基礎文法最速マスター - D++のはまり日誌](#)
- [Java基礎文法最速マスター - 何かしらの言語による記述を解析する日記](#)
- [VBA基礎文法最速マスター - 何かしらの言語による記述を解析する日記](#)
- [Brainf*ck基礎文法最速マスター - 医者立志す妻を応援する夫の日記](#)
- [Haskell基礎文法最速マスター - think and error](#)
- [Bash基礎文法最速マスター - 何かしらの言語による記述を解析する日記](#)
- [VBScript 基礎文法最速マスター - CX's VBScript Diary - VBScript グループ](#)
- [JavaScript基礎文法最速マスター - なんとなく日記](#)

[コメント](#)

[<\[Windows 7\]\[評価版\]](#) [\[Windows 7 Enterpris...](#)