

コマンドプロンプトに慣れる Ver1.3

Seiichi Nukayama

2022 年 8 月 15 日

目次

1	コマンドプロンプトとは?	1
1.1	起動	1
1.2	設定	2
1.3	動作を試す	3
1.4	コマンドプロンプトとは?	4
2	コマンドプロンプトによるディレクトリ (フォルダ) の移動	4
3	システム環境変数の PATH への登録	5
3.1	システム環境変数の編集	6
4	いろいろな情報	11
4.1	環境変数を画面に出力する	11
4.2	画面に出力された内容を保存する	11
4.3	画面の出力を 1 ページずつにする	11
4.4	役に立つ (かもしれない) コマンド	12
4.5	IP アドレスを調べる	13
5	まとめ	13
5.1	コマンド	13
5.2	その他	13
6	バッチファイル	14
6.1	コマンドを作成する	14
6.2	バッチファイル	16

1 コマンドプロンプトとは?

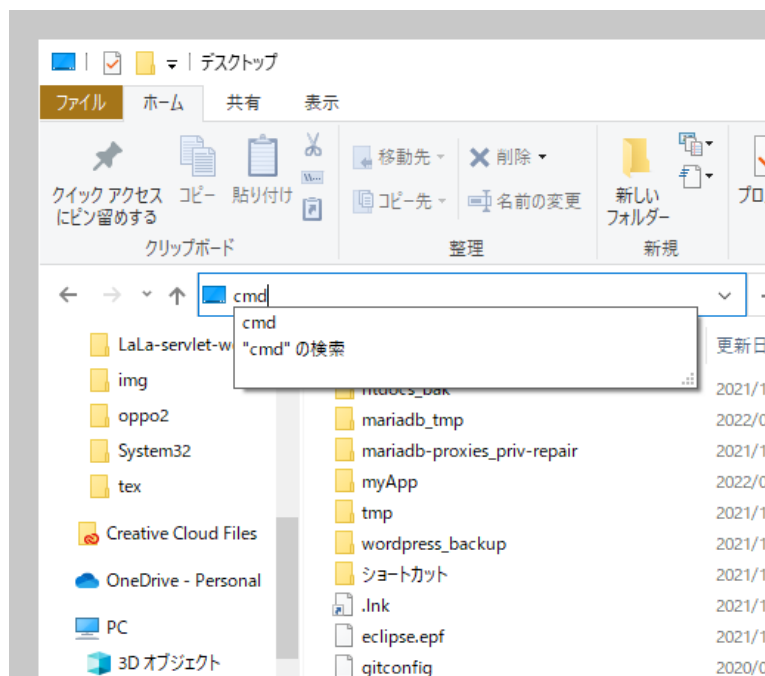
1.1 起動

“スタートボタン” – “Windows システムツール” の中に “コマンドプロンプト” はある。

右クリックして、“スタートパネルにピン留めする” かあるいは、“その他” – “タスクバーにピン留めする” にしておくとい。

しかし、ふつうは以下の方法で起動する。

- スタートボタン横の検索で “cmd” と入力して <Enter キー>
- エクスプローラの URL 欄にて “cmd” と入力して <Enter キー>

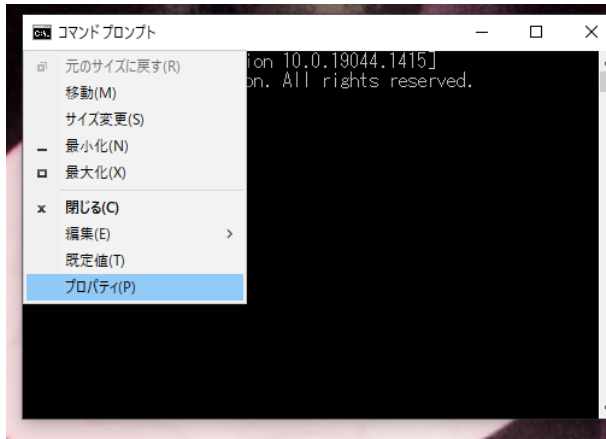


これは、エクスプローラの URL 欄に “cmd” と入力したところ。

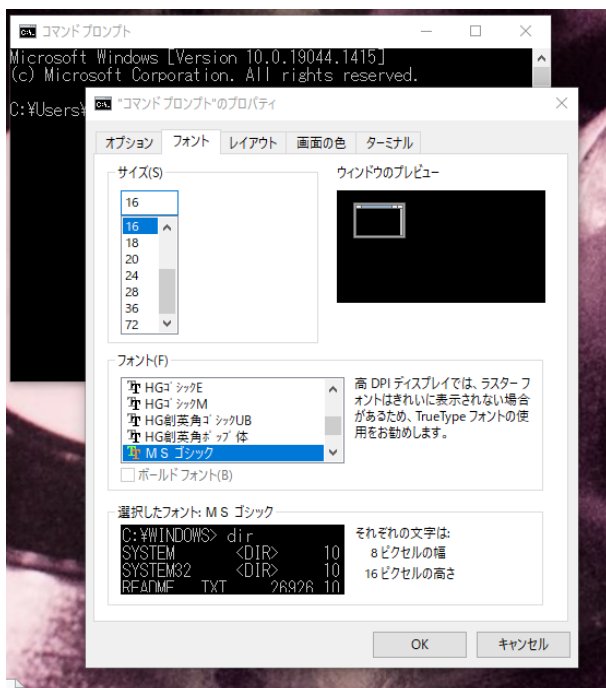
黒いウィンドウが表示される。これがコマンドプロンプト。

1.2 設定

ウィンドウ左上のアイコンをクリックすると“プロパティ”という項目がある。それを選択すると、設定画面になる。



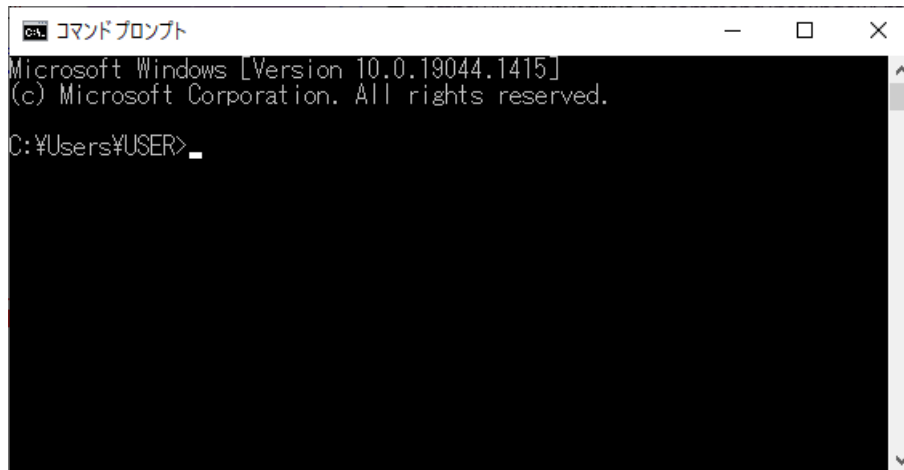
これは、プロパティを選択しているところ。



これは、プロパティでフォントを選択しているところ。

1.3 動作を試す

画面には、以下のような文字が表示されている。



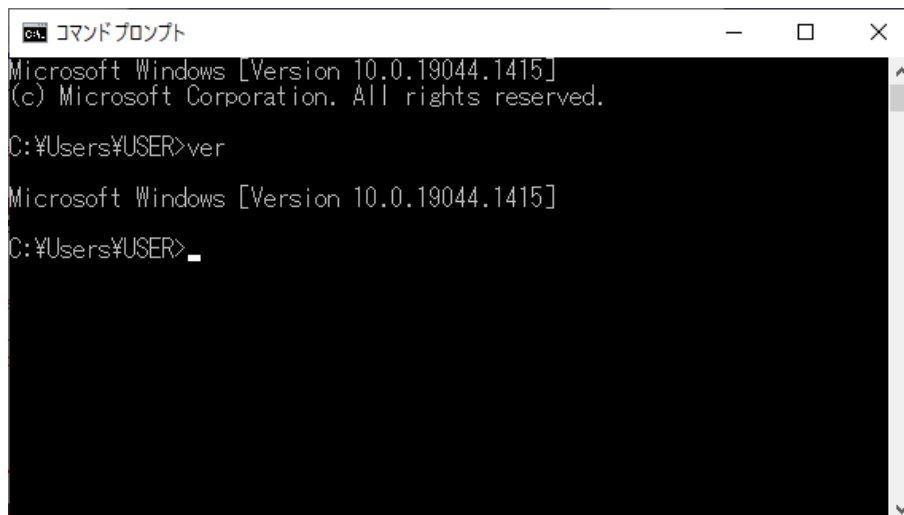
```
Microsoft Windows [Version 10.0.19044.1415]
(c) Microsoft Corporation. All rights reserved.

C:\Users\¥USER>
```

`C:\Users\User>` はプロンプトといい、現在の位置をユーザーに示している。また、> という文字列に続けて文字を入力できることを示している。

試しに以下のコマンドを入力して <Enter キー> を押してみる。

`ver`



```
Microsoft Windows [Version 10.0.19044.1415]
(c) Microsoft Corporation. All rights reserved.

C:\Users\¥USER>ver

Microsoft Windows [Version 10.0.19044.1415]
C:\Users\¥USER>
```

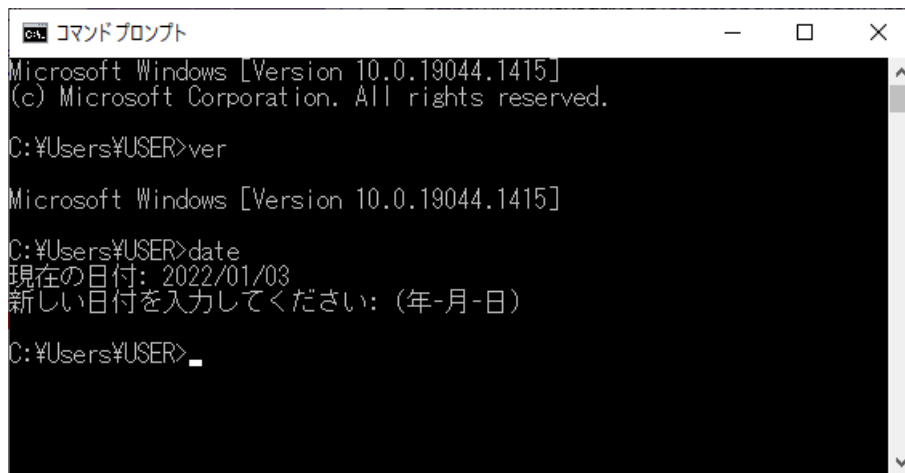
Windows のバージョンが表示される。

今度は以下のコマンドを入力する。(<Enter キー> も)

`date`

現在の日付が表示されて、次の行で “新しい日付を入力してください: (年-月-日)” と表示されて、入力が促される。

何も入力せず、そのまま <Enter キー> を押せばよい。



1.4 コマンドプロンプトとは？

コマンドプロンプトというのは、この黒い画面に文字 (コマンド) を入力してコンピュータからの返答を得るというものである。

つまり、コンピュータとの「対話」である。コンピュータと「チャット」しているようなものである。コンピュータには二つの種類のアプリケーションがある。

- GUI アプリケーション — マウスで操作するアプリケーション
- CUI アプリケーション — キーボード入力で操作するアプリケーション

GUI — Graphical User Interface

CUI — Character User Interface

“WORD” などのアプリは GUI である。大半のアプリが GUI である。

しかし、CUI アプリも数多くある。特にプログラミング言語 (PHP、Java など) は CUI である。

しかし、CUI だと使いづらいので、Java では Eclipse などの統合開発環境 (IDE) を使った開発が行なわれている。

先程の “ver” というのは、ひとつのアプリケーションで、“date” というのもアプリケーションである。

“help” と入力すれば、コマンド (アプリ) の一覧が表示される。中にはシステムの改変を行うコマンドもあるので、表示されたコマンドを不用意に実行してはいけない。

2 コマンドプロンプトによるディレクトリ (フォルダ) の移動

コマンドプロンプトには「現在の位置」が表示される。「現在どの位置にいるのか」を理解する必要がある。スタート・メニューからコマンドプロンプトを起動した場合、

`C:\Users\¥user` と表示される。この場所を **ホームディレクトリ** あるいは **ホームフォルダ** という。

また、この “C:\Users\¥user” を **パス** という。

この場所で `dir` とすると、この場所にあるファイルやディレクトリ (フォルダ) の一覧が表示される。

Desktop — デスクトップ
Documents — ドキュメント
Downloads — ダウンロード
Music — ミュージック
Pictures — ピクチャ
Videos — ビデオ

`dir /w`

とすると、横に広げてディレクトリ (フォルダ) の状態を一覧できる。

たとえば、デスクトップに移動するには、`cd desktop` とする。

`dir` とすると、たくさんのファイルやフォルダがあるので、画面の上に過ぎ去ってしまう。そこで、以下のようになる。

`dir /p`

すると、1 画面分ごとに表示される。そのとき、一番上には、以下のように表示される。

```
コマンドプロンプト - dir /p
ドライブ C のボリューム ラベルがありません。
ボリューム シリアル番号は 0AC7-0A39 です

C:\Users\¥USER のディレクトリ

2022/01/03  20:52  <DIR>          .
2022/01/03  20:52  <DIR>          ..
2020/04/12  23:26  <DIR>          .android
2020/04/11  16:06  <DIR>          .AndroidStudio3.6
2022/01/03  17:25      8,224 .bash_history
2021/10/06  19:56  <DIR>          .cache
2021/10/17  15:30  <DIR>          .codetoegether
2021/10/18  18:17  <DIR>          .config
2021/10/17  15:29     106 .ctsystem
2021/12/14  17:38  <DIR>          .docker
続行するには何かキーを押してください . . .
```

この `.` は、「ここ」をあらわす。

`..` は、「ひとつ上」をあらわす。

だから、“`./memo.txt`” とすると、このディレクトリ (フォルダ) にある `memo.txt` のことになる。(memo.txt が存在するとして)

また、“`../some.txt`” とすると、ひとつ上のディレクトリ (フォルダ) にある `some.txt` ということになる。(some.txt が存在するとして)

さらに `cd ..` とすると、ひとつ上のディレクトリ (フォルダ) に移動できる。

3 システム環境変数の PATH への登録

テラパッドを起動するには、デスクトップのテラパッド・アイコンをダブルクリックするか、メニューから起動するとかしなければならない。

先ほどのコマンドプロンプトにて、`terapad` <Enter> としても起動できない。

しかし、`terapad.exe` (テラパッド本体) のあるフォルダに移動し、

```
> cd "c:\¥Program Files (x86)\¥terapad"
```

「Program Files (x86)」の中に半角空白が入っているのに注意する。昔はファイル名やフォルダ名に半角空白は使えなかった。しかし、現在の Windows では半角空白を使える。ただし、今回のようにコマンドの中で使うためには、フォルダ名全体を"(ダブルクォーテーション)で囲む必要がある。でないと、コマンドの切れ目との区別ができない。

そのフォルダで

```
> terapad <Enter>
```

とすると、テラパッドを起動できる。

また、

```
> "c:\Program Files (x86)\terapad\terapad"
```

としてもテラパッドを起動できる。

この"テラパッド本体"のある場所を PC に覚えさせると、どこからでもコマンドプロンプトから起動できるようになる。

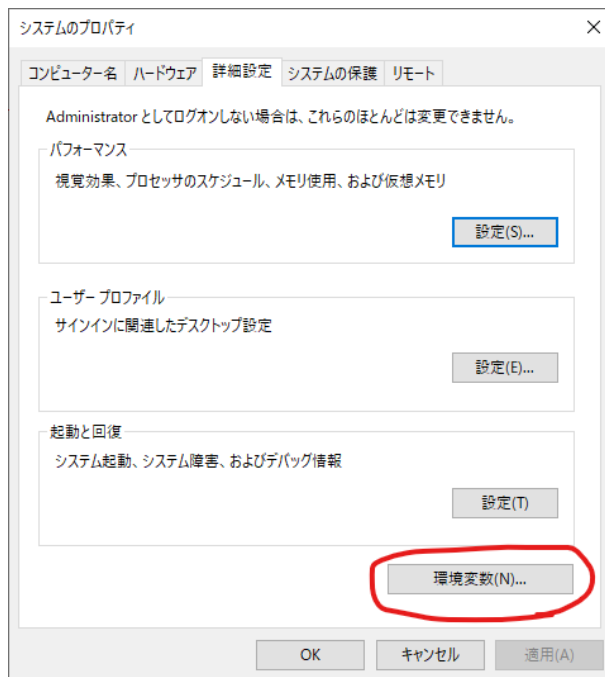
Windows には"システム環境変数"という仕組みがあり、そこに"PATH"という変数が用意されていて、その変数に、"terapad"のフォルダを登録すると、このコンピュータのどこからでも terapad を呼び出すことができる。

3.1 システム環境変数の編集

スタートボタンの横の検索に"システム環境変数"と入力すると、"システム環境変数の編集"という文字が現れるので、それをクリックする。

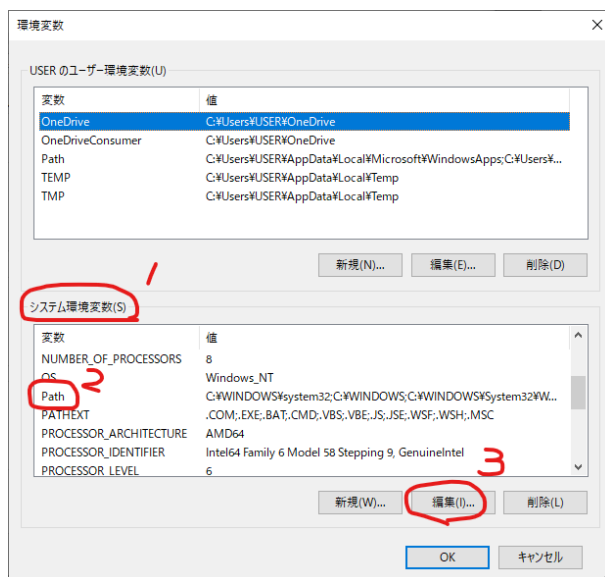


"スタートボタン"を右クリックして「システム」を選択 — 関連設定「システムの詳細設定」でもこのウィンドウを開くことができる。

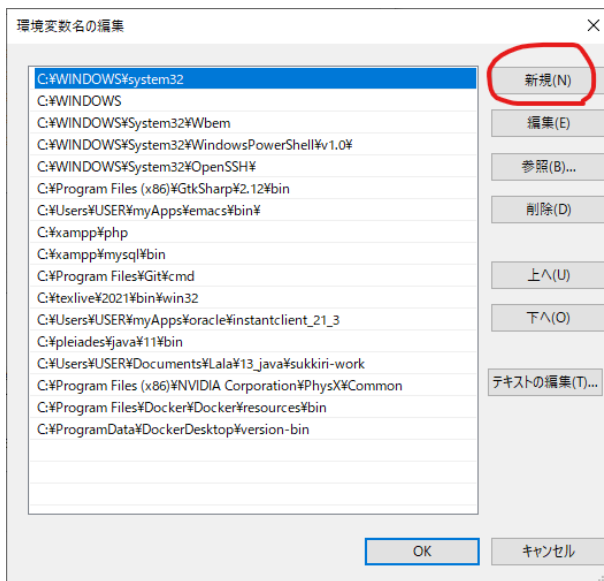


開いたウィンドウで、“環境変数”をクリックする。

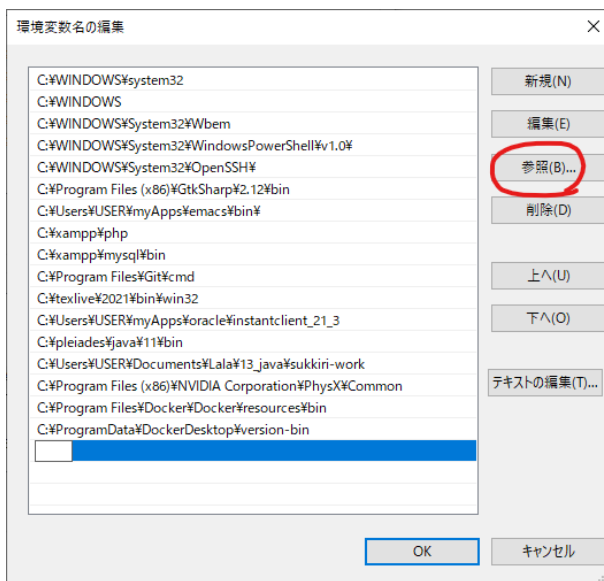
環境変数のダイアログが開くので、“システム環境変数”の“Path”を選択し、“編集”をクリックする。



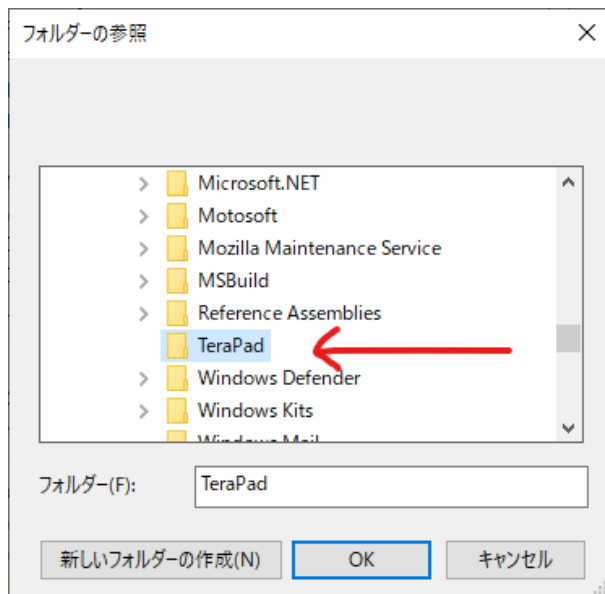
環境変数名の編集ダイアログが開くので、“新規”を選択する。



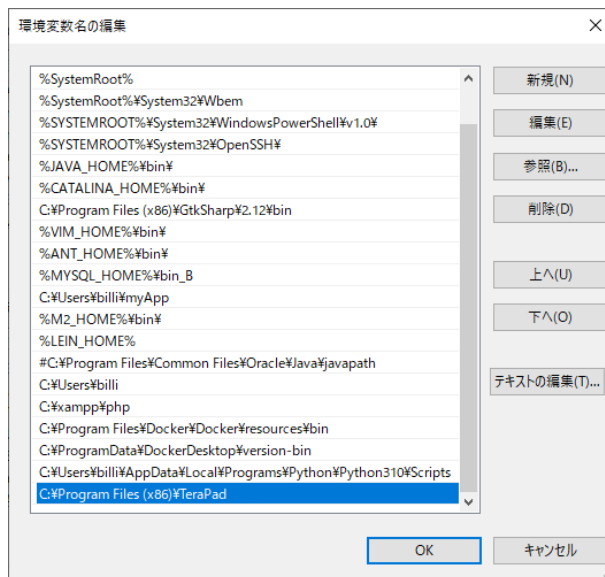
入力欄ができる。“参照”をクリックする。



開いたダイアログで C:\Program Files (x86)\terapad を選択して “OK”。



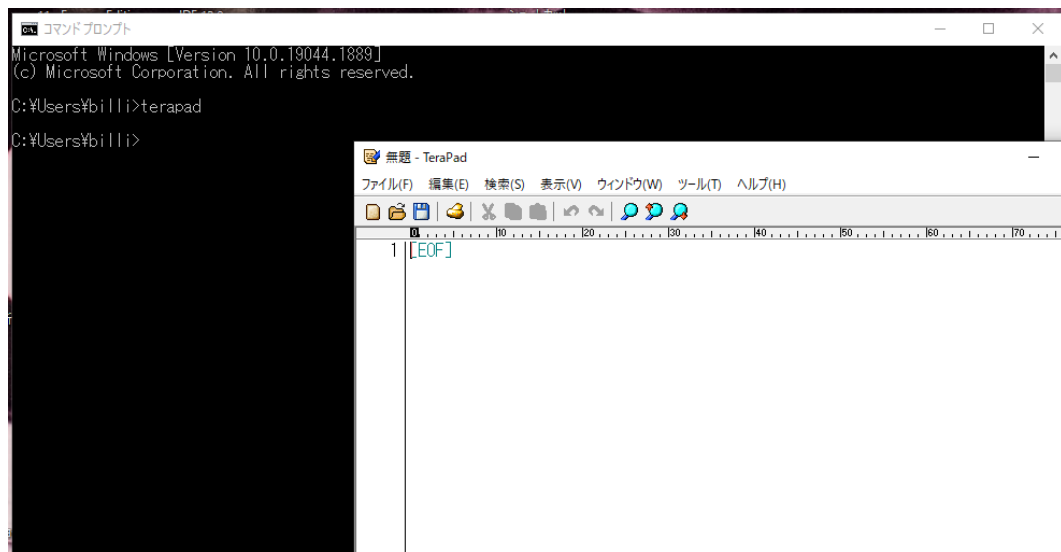
“C:¥Program Files (x86)¥TeraPad” が環境変数“Path” に登録された。



あとは、“OK” をクリックしてダイアログを閉じていく。“X”(閉じる) をクリックすると、今までの操作がすべてキャンセルされるので気をつける。

このようにシステム環境変数の “Path” にアプリのある場所を登録することで、その場所にいなくても、そのアプリを実行できるようになる。

ただ、現在開いているコマンドプロンプトはいったん閉じて、再度開きなさいとこの変更は反映されない。



4 いろいろな情報

4.1 環境変数を画面に出力する

"echo" というコマンドがある。これは 画面出力したいときに使う。

たとえば、システム環境変数の "PATH" に何が登録されているかを調べるには、

```
path
```

と入力すると表示されるが、以下のようにすることもできる。

```
echo %path%
```

% ではさんで echo すると、その変数の内容を出力することができる。

"HOME" というシステム環境変数がある。これにはユーザーのホームフォルダ (ホームディレクトリ) の位置 (パス) がおさめられている。

```
echo %home%
```

とすると、その内容が表示される。また、

```
cd %home%
```

とすることで、他のディレクトリ (フォルダ) にいても、瞬時にホームディレクトリ (ホームフォルダ) に戻ってこられる。

```
set
```

と入力すると、現在の環境変数が一覧できる。また、

```
set home
```

としても、環境変数 HOME の内容を確認できる。

4.2 画面に出力された内容を保存する

画面に出力された内容を保存することもできる。正確にいうと、echo の出力先を "画面" から "ファイル" に変更することになる。

```
echo %path% > path.txt
```

とすると、環境変数 PATH の内容が path.txt に保存される。

```
dir > dir.txt
```

とすると、"dir" で表示される内容を "dir.txt" というファイルに保存できる。

ただ、このコマンドを実行している場所 (ディレクトリあるいはフォルダ) に保存される。

"C:¥Users¥user" (ホーム) で実行しているなら、その場所に保存される。

システム環境変数の一覧を見たい場合も

```
set > set.txt
```

とすると、set.txt をエディタで開き、システム環境変数の一覧を見ることができる。

4.3 画面の出力を 1 ページずつにする

たとえば `set` というコマンドを実行すると、システム環境変数の一覧が画面の上に流れていくが、

```
set | more
```

とすると、画面の高さによって、自動的に出力が停止し、画面下部に `-- more --` と表示される。

ここで、“スペースキー”を押すと、次の1画面高さ分表示される。

また、“Enter キー”を押すと、1行ずつ上にずれて表示される。

“q”を押すと、画面出力の MORE 表示から抜けることができる。

4.4 役に立つ (かもしれない) コマンド

4.4.1 ポート番号を調べる

開発をしていると、Apache などの Web サーバーを 80 番ポートで動作させたいことがある。ところが、80 番ポートを他のアプリがすでに使っていて、Apache を起動できない場合がある。そんな時、80 番ポートを使っているアプリを探すのに、以下のコマンドが役に立つ。

```
netstat -noa
```

画面上に流れていくから、

```
netstat -noa > netstat.txt
```

とすると、エディタを使ってゆっくり見ることができる。また、

```
netstat -noa | more
```

とすると、1画面高さ分ずつ見ることができる。

“PID”とはプロセス ID のことで、これは“タスクマネージャー”をみることでそのプロセス ID は何のアプリかはわかる。

あるいは、コマンドプロンプトを管理者権限で起動すると

```
netstat -nba
```

というコマンドが実行できる。これは、“PID”の代わりに、実行しているファイル名がわかる。

コマンドプロンプトを管理者権限で起動するには、スタートパネルから起動するとして、そのアイコンを右クリックし、表示されたメニューから“その他”を選択し、そのサブメニューから“管理者として実行”を選択するとよい。



4.5 IP アドレスを調べる

4.5.1 自機の IP アドレスを調べる

コンピュータの IP アドレスを調べることができる。

```
ipconfig
```

Wifi を使用しているなら、“Wireless LAN adapter Wi-Fi” の項目を見る。

“IPv4 アドレス” なら、“192.168.X.X” になっているはずである。(X はなにかの数字)

“デフォルト ゲートウェイ” は、おそらく “192.168.0.1” あるいは “192.168.1.1” になっているだろう。

4.5.2 インターネット上のサイトの IP アドレスを調べる

```
nslookup google.co.jp
```

 — グーグルの場合

調べたいドメインを引数として入力すれば、name サーバーに問い合わせしてくれる。

5 まとめ

5.1 コマンド

dir	現在のディレクトリ (フォルダ) の一覧を表示する
dir /w	dir を横並びで表示する
dir /p	dir を 1 画面ごとにページ表示する
cd	現在のパス (位置) を表示する
cd <パス>	<パス> の位置に移動する
cd ..	一つ上のディレクトリ (ホーム) に移動する
type <ファイル>	ファイルの内容を表示する
more <ファイル>	ファイルの内容を 1 画面ごとに表示する
>	出力先を画面から (たとえば) ファイルに変える。(例) dir > dir.txt
(バーティカルバー)	コマンドをつなげる。(例) dir more

5.2 その他

↑↓	ヒストリー機能 (過去のコマンド履歴を表示)
[F7]	履歴一覧。↑↓で選択、実行
TAB キー	入力補完
% 環境変数 %	環境変数の内容
cd %HOME%	ホームディレクトリ (ホームフォルダ) に移動する

6 バッチファイル

6.1 コマンドを作成する

簡単なコマンドを作成してみる。以下のコードをテラパッドなどで入力して、“hello.bat”として保存する。
保存先は デスクトップ にしておく。

リスト 1 hello.bat

```
1 echo こんにちは
```

現在いる位置はコマンドプロンプトに表示されている。“C:¥users¥USER”である。

そこで、以下のコマンドを入力する。

```
cd Desktop <Enter キー>
```

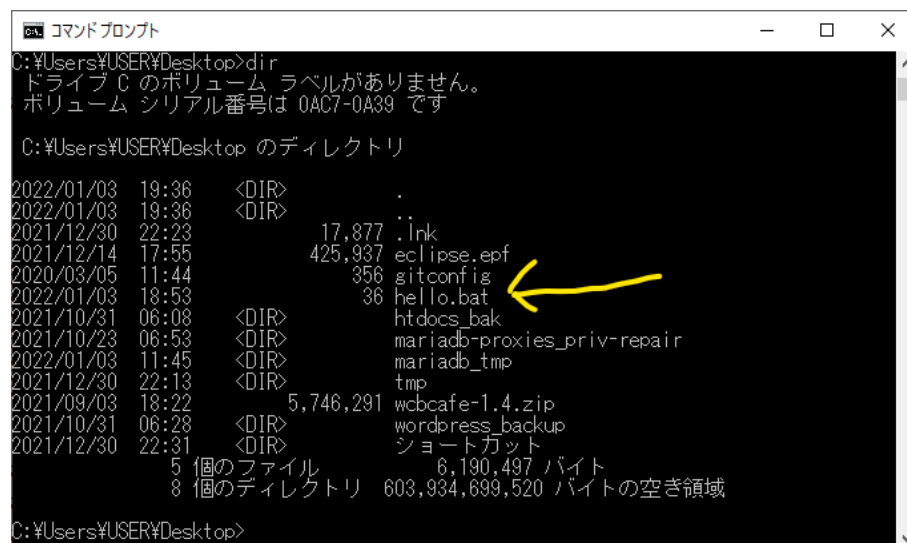
コマンドプロンプトが“C:¥users¥USER¥Desktop”となる。

(このとき、`cd De` まで入力したら、TAB キー を押すと `cd Desktop` と補完してくれる。)

以下のコマンドを入力する。

```
dir <Enter>
```

以下のように、そのフォルダにあるファイルやフォルダが表示される。



```
コマンドプロンプト
C:¥Users¥USER¥Desktop>dir
ドライブ C のボリューム ラベルがありません。
ボリューム シリアル番号は 0AC7-0A39 です

C:¥Users¥USER¥Desktop のディレクトリ

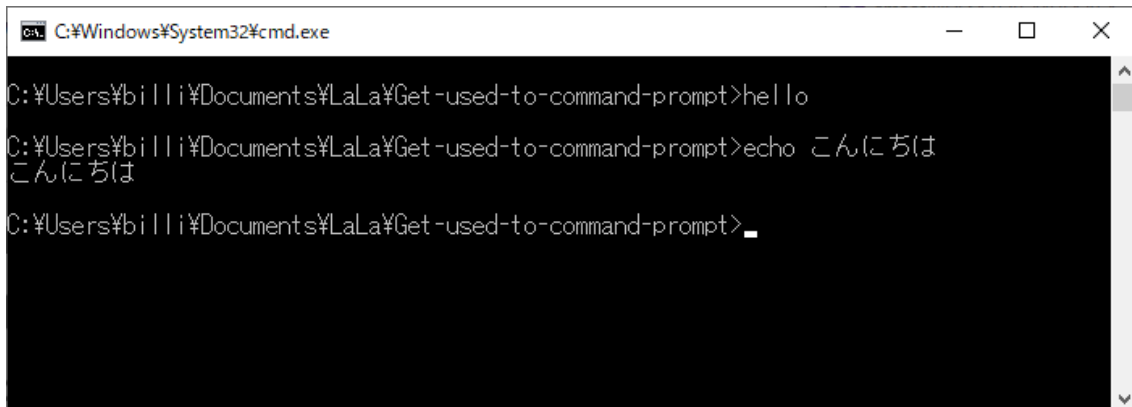
2022/01/03  19:36  <DIR>          .
2022/01/03  19:36  <DIR>          ..
2021/12/30  22:23                17,877  .lnk
2021/12/14  17:55        425,937  eclipse.epf
2020/03/05  11:44                356  gitconfig
2022/01/03  18:53                36  hello.bat
2021/10/31  06:08  <DIR>          htdocs_bak
2021/10/23  06:53  <DIR>          mariadb-proxies_priv-repair
2022/01/03  11:45  <DIR>          mariadb_tmp
2021/12/30  22:13  <DIR>          tmp
2021/09/03  18:22        5,746,291  wbcacafe-1.4.zip
2021/10/31  06:28  <DIR>          wordpress_backup
2021/12/30  22:31  <DIR>          ショートカット
                    5 個のファイル              6,190,497 バイト
                    8 個のディレクトリ 603,934,699,520 バイトの空き領域

C:¥Users¥USER¥Desktop>
```

その中に“hello.bat”があることを確認する。

hello.bat は以下のようにして実行できる。

```
> hello
```



```
C:\Windows\System32\cmd.exe

C:\Users\billi\Documents\LaLa\Get-used-to-command-prompt>hello

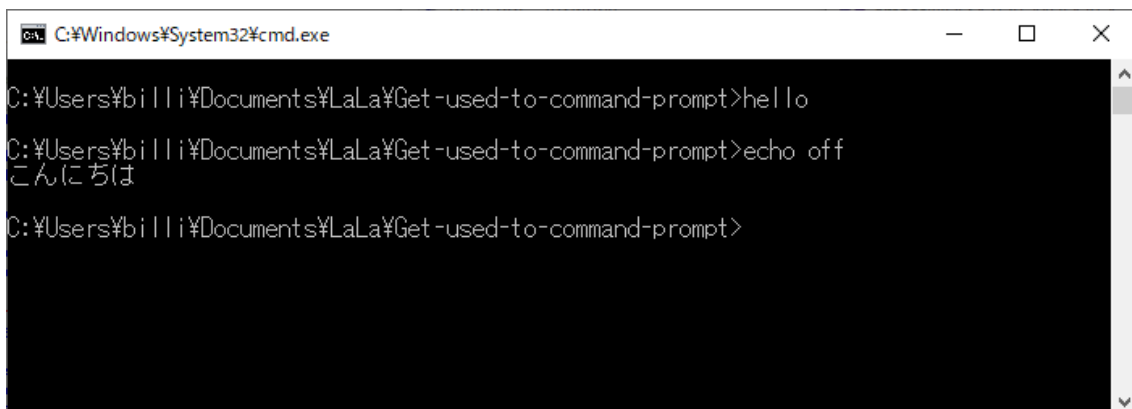
C:\Users\billi\Documents\LaLa\Get-used-to-command-prompt>echo こんにちは
こんにちは

C:\Users\billi\Documents\LaLa\Get-used-to-command-prompt>_
```

画面に `echo こんにちは` と表示されてから `こんにちは` と表示されている。`echo こんにちは` が表示されないようにするには、以下のようにする。

リスト 2 hello.bat

```
1 echo off
2 echo こんにちは
```



```
C:\Windows\System32\cmd.exe

C:\Users\billi\Documents\LaLa\Get-used-to-command-prompt>hello

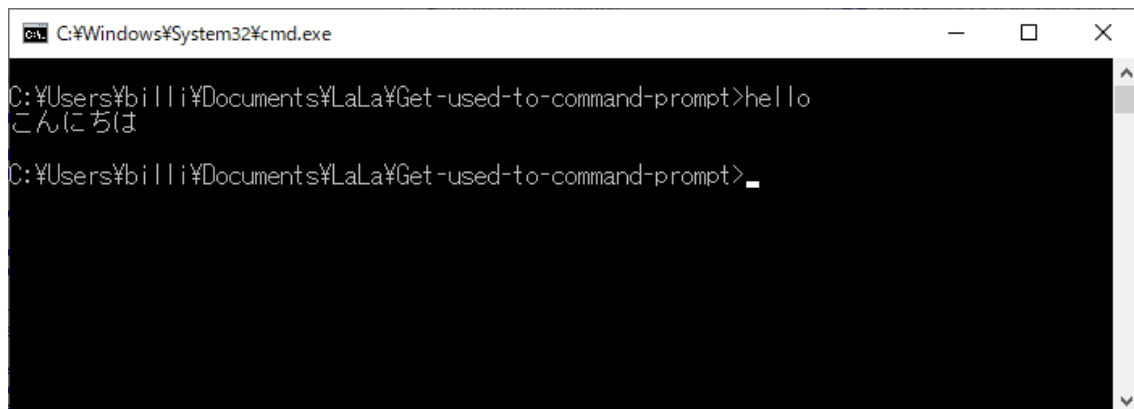
C:\Users\billi\Documents\LaLa\Get-used-to-command-prompt>echo off
echo off
こんにちは

C:\Users\billi\Documents\LaLa\Get-used-to-command-prompt>
```

`echo こんにちは` は消えたけれど、かわりに `echo off` が表示されてしまった。それも表示されないようにするには、`echo off` の先頭に "@" をつける。

リスト 3 hello.bat

```
1 @echo off
2 echo こんにちは
```


A screenshot of a Windows Command Prompt window. The title bar shows the path "C:\Windows\System32\cmd.exe". The command prompt shows the current directory as "C:\Users\billi\Documents\LaLa\Get-used-to-command-prompt". The user has entered the command "hello", and the prompt has responded with "こんにちは". The prompt is now waiting for the next command, indicated by a cursor after the ">" symbol.

```
C:\Windows\System32\cmd.exe
C:\Users\billi\Documents\LaLa\Get-used-to-command-prompt>hello
こんにちは
C:\Users\billi\Documents\LaLa\Get-used-to-command-prompt>_
```

6.2 バッチファイル

今作成した "hello.bat" はバッチファイルと呼ばれるもので、コンピュータに与えるコマンドを手順として多数記述しておいて、それらを実行させるものである。"スクリプト" とも呼ばれる。

今作成したのは簡単な手順であるが、業務で使われる場合は複雑なものとなる。

拡張子は ".bat" である。