

Bashは**Bourne Again SHell**という名前の略称です。

Bourne ShellはUNIXが登場して以来使われてきた最も基本的なシェルでその機能を拡張したものがbashです。

Shellとは貝の意味で他から影響を受けない環境を保つことからシェルという名前がついており、使用システムに存在するプログラム(コマンド)を起動するマンマシンインターフェイスでありプログラミング言語でもあります。

シェルから起動されたプログラムは、シェルの動作環境をそのまま引き継いで動作します。

この動作環境とは動作に必要な環境変数を指しています。UNIXの世界では起動されるプログラムは様々な環境変数を参照しながら動作します。

たとえば、UNIXのGUI環境はX-Serverが標準的に使われますがそこで使われる環境変数の代表的なものはDISPLAY環境変数です。この変数を見て通信するX-SERVERやX-SERVERに接続されたDISPLAYのどこに表示するかが決定されます。

CUIで代表的なものはTERM変数です。この変数は接続されている端末の種別を特定し表示の方法が定義されます。

端末自体はGUIの中でも使えるので端末を起動して

```
echo $TERM
```

と打ち込んでみましょう！

すると設定されている端末の種類が表示されます。

※環境変数の値を設定する場合は何もつけずに=で値をセットしますが参照する際は頭に\$を付けます。

## ●bashの基本的なコマンド

Bashには予約語と組込コマンドのように(Bashが自身が理解できるコマンド)と外部コマンドが存在します。外部コマンドは環境変数 PATH に指定されているディレクトリ内に存在する必要があります。

```
echo $PATH
```

で環境変数PATHの内容を確認してみましょう！

ここで注意すべき点は、PATHの中には伝統的に. は含まれません。(. はカレントディレクトリ意味します)

従ってCコンパイラ等で新しいコマンドをカレントディレクトリに作った場合 ./コマンド名 という風にしないと起動されません。

また外部コマンドか予約語や組込コマンドかを調べる方法としては

whereis コマンド名      ファイルが何処に存在するかを返してくれます。

例えば 予約語の for や case を whereis で探してみるとマニュアルの位置のみが帰ってきます

ls コマンドを調べてみると /bin/ls という風にコマンドが置かれているディレクトリと共にマニュアルファイルの位置も帰ってきます。

このコマンドで存在フォルダが帰ってくるものは外部コマンドであることがわかります。

以下によく使うコマンドと使用例を示します。

ls コマンドはファイルの情報を返してくれます。Windowsのコマンドプロンプトのdir と同じようなものです。

ls -l ではアクセス権や容量等の情報が表示されます。

ls -a では不可視属性のファイルを含む全てのファイルを表示します。UNIX系OSでは先頭に . のつくファイルは通常では表示されません。

cat コマンドはファイルの内容を表示します。

cat file1 file2 .... というように複数のファイルを指定すると指定されたすべてのファイルを表示します。

ファイルをマージする場合

```
cat file1 file2 > file3
```

という風にするとfile1とfile2のマージファイルがfile3 にできます。

ここで出てくる > はリダイレクトと呼ばれ標準出力の内容をファイルに保存します。

mkdir ディレクトリ名 で新しいディレクトリを作成します。

rmdir からのディレクトリを消去します。

空ではない場合は

rm -rf ディレクトリ名とします。rmは本来ファイルを消去するコマンドですがディレクトリも消去可能です。

ただし、このオプション付けワイルドカードでrmを使用する場合マッチングパターンには気を付ける必要があります。

cd ディレクトリを移動します。

man 名前 で名前に該当するオンラインマニュアルを表示します。

ssh は暗号化された通信でリモートアクセスを行います。一般的には

ssh user@domain でアクセスします。