

【UNIX】基礎

基礎編

■UNIX の概要

- ▶ ※ UNIX はサーバー用によく使われる
- ▶ ※ UNIX は無料の互換OS がたくさん登場し広まった
- ▶ ※ 互換OS の1つ Linux はいくつもの企業・団体によって独自の設定や形式で配布されたものがあり、それらの総称を Distribution という。
- ▶ ※ UNIX 系のOS（macOS 含む）はすべて POSIX という規格で統一されているため、基本的なコマンドは共通している。

■基本

- ▶ 長いコマンドで改行したい

■シェルのショートカットキー

- ▶ コマンドを実行せずに新しいプロンプトに移る
- ▶ 今書いているコマンドを消す
- ▶ 上に書いてきたのを消す
- ▶ ファイル、ディレクトリ名を途中まで書いて補完
- ▶ 以前のコマンドを使う

■ディレクトリの構成

- ▶ ☆ 各ディレクトリの説明
- ▶ ☆ 絶対PATH と 相対PATH

■こうしたいならこのコマンド

基本操作

- ▶ カレントディを表示
- ▶ 他のディに移る
- ▶ 出力

コマンド実行全般

- ▶ 管理者権限で実行
- ▶ 出力せずファアを上書き

【UNIX】基礎

基礎編

■UNIX の概要

- ▶ ※ UNIX はサーバー用によく使われる
- ▶ ※ UNIX は無料の互換OS がたくさん登場し広まった
- ▶ ※ 互換OS の1つ Linux はいくつもの企業・団体によって独自の設定や形式で配布されたものがあり、それらの総称を Distribution という。
- ▶ ※ UNIX 系のOS（macOS 含む）はすべて POSIX という規格で統一されているため、基本的なコマンドは共通している。

■基本

- ▶ 長いコマンドで改行したい \ を入力後 Enter キーで改行可能

■シェルのショートカットキー

- ▶ コマンドを実行せずに新しいプロンプトに移る {Ctrl + C}
- ▶ 今書いているコマンドを消す {Ctrl + U}
- ▶ 上に書いてきたのを消す {Ctrl + L} ※ \$ clear でもいい
- ▶ ファイル、ディレクトリ名を途中まで書いて補完 {Tab}
- ▶ 以前のコマンドを使う {!} や {!}

■ディレクトリの構成

- ▶ ☆ 各ディレクトリの説明
- ▶ ☆ 絶対PATH と 相対PATH

■こうしたいならこのコマンド

基本操作

- ▶ カレントディを表示 \$ pwd ※ print working directory の略
- ▶ 他のディに移る \$ cd dirPath ※ change directory の略
- ▶ 出力 \$ echo 'str'

コマンド実行全般

- ▶ 管理者権限で実行 \$ sudo commands※ ※ ! も使える。
- ▶ 出力せずファアを上書き \$ commands > filePath : リダイレクション

▶ 出力せずファに追記	
▶ コマの引数にファを	
▶ 出力せず別コマ引数に	
▶ コマを立て続けに	
▶ ※ <code>{o,o,o}</code> <code>{o..o}</code> でfor文のようにできる（ブレース展開）。	

ディレクトリの中身を確認

▶ ディの中身を確認	
▶ 隠しファ含めて "	
▶ タイプ識別子付きで "	
▶ 1件1行で "	

ファイルやディレクトリの操作

▶ 更新日時を更新する	
▶ 新規作成	
▶ 複製	
▶ 移動	
▶ 名前の変更	
▶ 削除	
▶ ショートカットを作成	

ファイルの中身を確認

▶ 一気にすべて確認	
▶ ページャ	
▶ ページャ（検索可）	
▶ 行数や単語数を確認	
▶ 先頭や末尾の幾行か	
▶ 検索して該当行を抽出	

ファイルを編集（viエディター）

▶ ※ ファイルを上書きや追記するくらいなら、viエディターを使わずリダイレクションを使えばいい。	
▶ 起動	

▶ 出力せずファに追記	<code>\$ commands >> filePath</code>	:	"
▶ コマの引数にファを	<code>\$ commands < filePath</code>	:	"
▶ 出力せず別コマ引数に	<code>\$ commands anotherCommands</code>	:	パイプ
▶ コマを立て続けに	<code>\$ commands1 && commands2</code>		
▶ ※ <code>{o,o,o}</code> <code>{o..o}</code> でfor文のようにできる（ブレース展開）。			

ディレクトリの中身を確認

▶ ディの中身を確認	<code>\$ ls dirPath</code>	:	list	※ <code>dirPath</code> 省略ならカレディ
▶ 隠しファ含めて "	<code>\$ ls -a dirPath</code>			
▶ タイプ識別子付きで "	<code>\$ ls -F dirPath</code>	※	<code></code> <code>*</code> <code>/</code> <code>=</code> <code>></code> <code>@</code> <code> </code>	のどれかが末尾につく
▶ 1件1行で "	<code>\$ ls -l dirPath</code>			

ファイルやディレクトリの操作


▶ 更新日時を更新する	<code>\$ touch filePath</code>	※ファイルがない場合自動作成
▶ 新規作成	<code>\$ touch filePath \$ mkdir childDirName \$ mkdir -p dirPath</code>	
▶ 複製	<code>\$ cp filePath newFilePath \$ cp -r dirPath newDirPath</code>	
▶ 移動	<code>\$ mv Path destinedDirPath</code>	
▶ 名前の変更	<code>\$ mv Path newPath</code>	
▶ 削除	<code>\$ rm filePath</code> や <code>\$ rmdir dirPath</code>	※中身が空でないと削除不可
▶ ショートカットを作成	<code>\$ ln -s dirPath name</code>	※シンボリックリンクという

ファイルの中身を確認

▶ 一気にすべて確認	<code>\$ cat filePath</code>	
▶ ページャ	<code>\$ more filePath</code>	※{Space} で次頁へ、{Q} で終了
▶ ページャ（検索可）	<code>\$ less filePath</code>	※ " ※ <code>/</code> に続けて検索語を入力
▶ 行数や単語数を確認	<code>\$ wc filePath</code>	
▶ 先頭や末尾の幾行か	<code>\$ head -n lineCount filePath</code> か <code>\$ tail -n lineCount filePath</code>	
▶ 検索して該当行を抽出	<code>\$ grep 'str' filePath</code>	

ファイルを編集（viエディター）

▶ ※ ファイルを上書きや追記するくらいなら、viエディターを使わずリダイレクションを使えばいい。	
▶ 起動	<code>\$ vi filePath</code>

▶ ※ 起動直後はコマンドモードで、左下に  と出ている。

▶ 編集モードに	
▶ コマンドモードに戻る	
▶ 終了 (〃)	
▶ 保存して終了 (〃)	
▶ 保存せずに終了 (〃)	

ファイル実行

▶ ファイル実行	
▶ ファ名だけで実行	
▶ コマのファの場所	

ファイルやディレクトリの検索

▶ ファやディの検索	
▶ ファだけ検索	
▶ ディだけ検索	


コマンドの履歴を活用

▶ コマの履歴を見る	
▶ 履歴上の n 番目を実行	
▶ 直前のコマを実行	
▶ n 個前のコマを実行	
▶ 直前コマの最終引数	
▶ str で始まる直近のコマ	
▶ コマを検索	

すべてのユーザーやグループを確認



▶ ユーザーを確認	
▶ グループを確認	
▶ あるユの属するグ	

アクセス権限

▶ ※ 起動直後はコマンドモードで、左下に  と出ている。

▶ 編集モードに	{ }	※左下に  と出る
▶ コマンドモードに戻る	{Esc}	
▶ 終了 (〃)	:q	
▶ 保存して終了 (〃)	:wq	
▶ 保存せずに終了 (〃)	:q!	



ファイル実行

▶ ファイル実行	・ $\$ filePath$ ※ ※カレディ中のファは  に続ける必要あり ・ $\$ \$SHELL filePath$ ※ ※  に続ける必要なし
▶ ファ名だけで実行	パスを通しておき、 $\$ export PATH=dirAbsolutePath:$PATH$
▶ コマのファの場所	$\$ which commandName$

ファイルやディレクトリの検索

▶ ファやディの検索	$\$ find dirPath -name 'str'$	※ワイルドカード可
▶ ファだけ検索	$\$ find dirPath -name 'str' type -f$	※ 〃
▶ ディだけ検索	$\$ find dirPath -name 'str' type -d$	※ 〃

コマンドの履歴を活用

▶ コマの履歴を見る	$\$ history$
▶ 履歴上の n 番目を実行	$\$!n$
▶ 直前のコマを実行	$\$!!$
▶ n 個前のコマを実行	$\$!-n$
▶ 直前コマの最終引数	$\$!\$$ で使える
▶ str で始まる直近のコマ	$\$!str$ ※  とすれば実行はせず、その後  で実行
▶ コマを検索	{Ctrl + R} → str 入力 → {Ctrl + R}で次候補へ → {Enter}か{Ctrl + C}

すべてのユーザーやグループを確認

▶ ユーザーを確認	$\$ cat /etc/passwd$
▶ グループを確認	$\$ cat /etc/group$
▶ あるユの属するグ	$\$ group user$

アクセス権限

▶ アクセス権限を確認	
▶ アクセス権限を変更	
▶ Xサーバへのアク許可	

システム管理

▶ ディの所有者を確認	
▶ ディの所有者を変更	
▶ 環境変数の編集	
▶ 環境変数の値を出力	

Web

▶ HTTPアクセスをしてコンテンツを取得	
-----------------------	--

日時

▶ 日時	
▶ 今月のカレンダー	

その他

▶ コマンドの使い方を調べる	
----------------	--

■こんなときは一般的にこうしよう

▶ インストール中などでログを全く表示させない	
-------------------------	--

■注意

- ▶ ※ 基本的に ***filePath*** を指定してファを新規作成する場合、そのファの**祖先ディはすべて前もって存在していなければならない。**
- ▶ ※ オプションは複数付けられる。

▶ アクセス権限を確認	\$ ls -l <i>dirPath</i>	※ <i>dirPath</i> 省略ならカレディ
▶ アクセス権限を変更	\$ chmod <i>mode filePath</i>	
▶ Xサーバへのアク許可	\$ xhost +	※ 使ってはいけない。

システム管理

▶ ディの所有者を確認	\$ ls -ld <i>dirPath</i>	
▶ ディの所有者を変更	\$ sudo chown <i>user:group dirPath</i>	※ <i>group</i> は group でわかる
▶ 環境変数の編集	\$ export 環境変数名=値	
▶ 環境変数の値を出力	\$ echo \$環境変数名	

Web

▶ HTTPアクセスをしてコンテンツを取得	curl <i>URL</i>
-----------------------	-----------------

日時

▶ 日時	date
▶ 今月のカレンダー	cal

その他

▶ コマンドの使い方を調べる	\$ help <i>command</i> か \$ <i>com</i> --help か \$ man <i>com</i>
----------------	---

■こんなときは一般的にこうしよう

▶ インストール中などでログを全く表示させない	\$ <i>commands</i> > /dev/null 2>&1
-------------------------	-------------------------------------

■注意

- ▶ ※ 基本的に ***filePath*** を指定してファを新規作成する場合、そのファの**祖先ディはすべて前もって存在していなければならない。**
- ▶ ※ オプションは複数付けられる。