

LinuC レベル1 Version10.0 技術解説無料セミナー

2020/11/29 開催

主題 1.03
GNUとUnixのコマンド

本日の講師



株式会社アイエスエイ (LPI-Japanアカデミック認定校)
木村 淳矢(Kimura Junya)

■講師自己紹介



#LinuC学習中

木村淳矢 (Kimura Junya)

株式会社アイエスエイ所属

株式会社アイエスエイはISAPCスクールを関東で13校を展開するトレーニングプロバイダーです。

ISAはLPI-Japanアカデミック認定校になっています。

ISAの新宿エルタワー校は、Pearson VUEの公認テストセンターになっています。



■LinuCとは

クラウド時代の即戦力エンジニアであることを証明するLinux技術者認定資格

- ✓現場で「今」求められている新しい技術要素に対応
 - オンプレミス／仮想化を問わず様々な環境下でのサーバー構築
 - 他社とのコラボレーションの前提となるオープンソースへの理解
 - システムの多様化に対応できるアーキテクチャへの知見
- ✓全面的に見直した、今、身につけておくべき技術範囲を網羅

今となっては使わない技術やコマンドの削除、アップデート、新領域の取り込み
- ✓Linuxの範疇だけにとどまらない領域までカバー

セキュリティや監視など、ITエンジニアであれば必須の領域もカバー



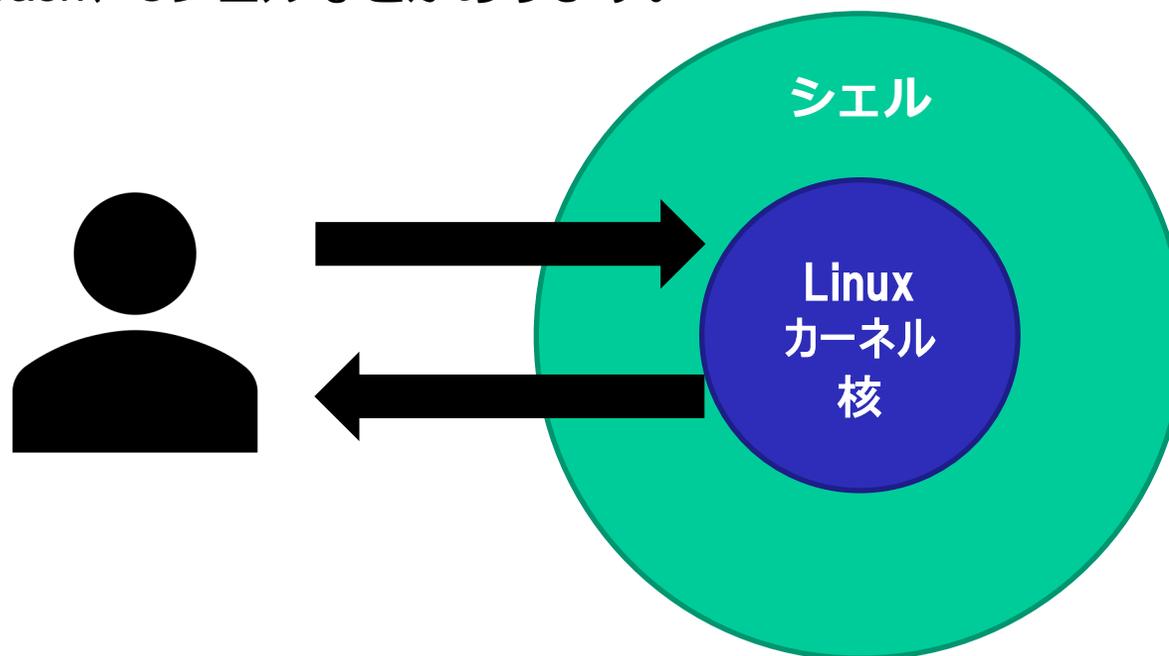
■Version10.0と従来の出題範囲の比較

テーマ	Version 10.0	従来
LinuC-1	仮想技術 <ul style="list-style-type: none"> ・仮想マシン／コンテナの概念 ・クラウドセキュリティの基礎 	← (Version10.0で新設)
	オープンソースの文化 <ul style="list-style-type: none"> ・オープンソースの定義や特徴 ・コミュニティやエコシステムへの貢献 	← (Version10.0で新設)
	その他 <p style="text-align: center;">→ (Version10.0で削除)</p>	アクセシビリティ、ディスククォータ、プリンタの管理、SQLデータ管理、他
LinuC-2	仮想化技術 <ul style="list-style-type: none"> ・仮想マシンの実行と管理(KVM) ・コンテナの仕組みとDockerの導入 	← (Version10.0で新設)
	システムアーキテクチャ <ul style="list-style-type: none"> ・クラウドサービス上のシステム構成 ・高可用システム、スケーラビリティ、他 	← (Version10.0で新設)
	その他 <ul style="list-style-type: none"> ・統合監視ツール(zabbix) ・自動化ツール(Ansible) 	← (Version10.0で出題範囲に含む)
	その他 <p style="text-align: center;">→ (Version10.0で削除)</p>	RAID、記憶装置へのアクセス方、FTPサーバーの保護、他

■ シェル

Linuxでコマンドでコンピューターに命令を送りますが、そのコマンドの処理をするプログラムをシェル (Shell) と呼びます。シェルには様々な種類があります。

Bourneシェル、bash、Cシェルなどがあります。





■ 変数

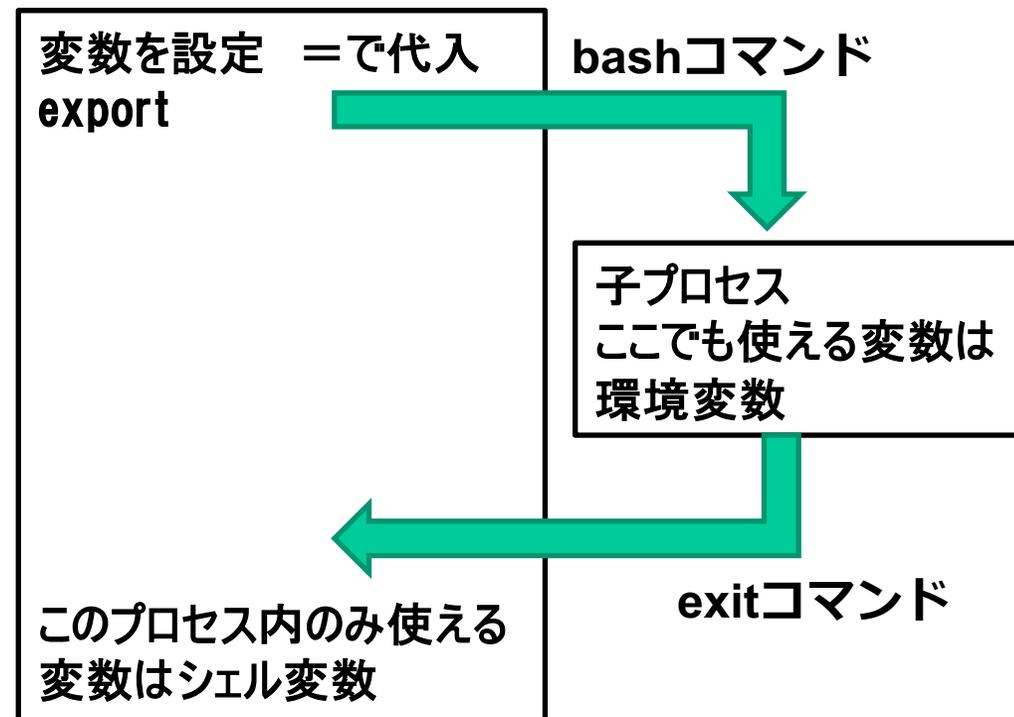
Linuxでコマンド履歴やホームディレクトリ、ユーザの指定した値など様々な情報は変数という箱のようなものに格納することができます。

変数にはスコープと呼ばれる変数を使える範囲があります。

そのシェルのみで使える変数をシェル**変数**と呼びます
そのシェルを超えて使用できる変数を**環境変数**と呼びます

環境変数は、シェル変数をexportコマンドで指定することで作成することができます。

なお、子プロセスはbashコマンド、
子プロセスを閉じるにはexitコマンドを使用します。





■ 引用符

Linuxで文字列を扱うには引用符で囲みます。その引用符には3つの種類があります。

- ‘ (シングルクォーテーション)
- “ (ダブルクォーテーション)
- ` (バッククォーテーション)

クォーテーションの種類	解説
‘	すべて文字列とみなされる
“	変数は展開される
`	コマンド結果が展開される



■ パイプ

パイプ (|) はコマンドの実行結果を別のコマンドに引き渡します。

■ 演算子

コマンド	解説
コマンド1 コマンド2	コマンド1を実行した結果をコマンド2に受け渡す
コマンド1 コマンド2	コマンド1が正常終了しなかったらコマンド2を実行する
コマンド1 && コマンド2	コマンド1が正常終了したらコマンド2を実行する
コマンド1 ; コマンド2	コマンド1を実行してからコマンド2を実行する

■ リダイレクト

標準入力の入力元や標準出力、標準エラー出力の出力先を変更する機能。「>」（リダイレクタ）を使う。

■ リダイレクタ

コマンド	解説	整数値	内容	既定値
>	標準出力の出力先を変更する	0	標準入力	キーボード
>>	標準出力の出力先を変更する(追記)	1	標準出力	モニタ
<	標準入力の入力元を変更する	2	標準エラー出力	モニタ
2>	標準エラー出力の出力先を変更する			
2>&1	標準エラー出力を標準出力と同じ出力先にする			



■ 使用ファイル

次のデモからは下記の2つのファイルを使います。

score.txt

Meg	100	eng
Aya	85	math
Nana	90	science
Kana	95	computer
Meg	100	eng

grade.txt

1	Meg	A
2	Aya	C
3	Nana	B
4	Kana	A
5	Meg	A



■ テキスト処理

Linuxではテキストを処理するコマンドがたくさんあります。これらのコマンドをうまく組み合わせることで目的の処理が行えます。

■ テキスト処理のコマンド

コマンド	解説	オプション
cat	ファイルの内容の表示	-n, --number (行番号の表示)
head	ファイルの先頭の表示	-n 行数, -行数(行数の指定)
tail	ファイルの末尾の表示	-n 行数, -行数(行数の指定)、-f, --follow (追記を表示)
cut	ファイルからタブで区切られたフィールドの切り出し	-f, --fields 切り出し位置の指定 -c, --characters 文字数(文字位置の指定) -d, --delimiter 区切り文字(区切り文字の指定)
paste	行を水平方向に連結	-d 区切り文字(区切り文字の指定)
join	共通フィールドを連結	-j フィールド(連結フィールドの指定)

■ テキスト処理

Linuxではテキストを処理するコマンドがたくさんあります。これらのコマンドをうまく組み合わせることで目的の処理が行えます。

■ テキスト処理のコマンド

コマンド	解説	オプション
tr	文字列の変換や削除	-d, --delete (削除)、[:alpha:] (英字)、[:lower:] (小文字) [:upper:] (大文字)、[:digit:] (数字)、[:space:] (スペース)
sort	フィールドの並び替え	-k, --key (キーの指定) -r, --reverse (降順にする)、-n, --numeric-sort (数字として扱う)
split	ファイルの分割	-行数 (指定された行ごとに分割する)
uniq	重複行の削除	-d, --repeated (重複行のみ表示) -u, --unique (重複していない行のみ表示)



■ 正規表現

正規表現もテキストにフィルタをかけられます。目的の文字列パターンを表示して検索などに使えます。

■ 主な正規表現

メタキャラクタ	解説
.(ピリオド)	任意の1文字
*	直前の文字の繰り返し
[]	[]内の文字のいずれか1文字
^	行頭
\$	行末
\ (バックスラッシュ)	エスケープレター



■ grepコマンド

ファイルやテキストの中を正規表現を使って検索するコマンド

■ 主なオプション

オプション	解説
-c, --count	パターンがマッチした行数の表示
-l, --ignore-case	大文字、小文字の区別をしない
-v, --invert-match	パターンにマッチしない行の表示
-E	拡張正規表現の利用 +(直前の文字の1回以上の繰り返し) ?(直前の文字の0回もしくは1回の繰り返し) (左右いずれかにマッチする) {n}(直前の文字のn回の繰り返し) {n,m}(直前の文字のn回からm回の間の繰り返し)



■ sedコマンド

テキストファイルに対して編集を行い、編集した結果を表示する

sed オプション コマンド ファイル

sed オプション -e コマンド1 -e コマンド2 ファイル

■ 主なオプションとコマンド

オプション	解説
-f	コマンドが書かれたスクリプトファイルを指定する sed オプション -f スクリプトファイル 対象ファイル

コマンド	解説
d	マッチした行を削除する
s	最初の対象を置換
g	すべての対象を置換
y	複数の置換



■ vi

viエディターはテキストファイルを編集するエディターです。LinuxではVimとEmacsが良く使われています。

■ 主な色々なコマンド、モード

コマンド	解説
i	カーソルの前にテキストを入力
a	カーソルの後にテキストを入力
o	下行に空白を挿入し、その行にテキストを入力
h	左へ移動
l	右へ移動
k	上へ移動
j	下へ移動

コマンド	解説
0(ゼロ)	行の先頭へ移動
\$	行の末尾へ移動
gg	ファイルの先頭行へ移動
G	ファイルの最終行へ移動
nG	n行目に移動



■ 主な色々なコマンド、モード

コマンド	解説
:q	viの終了
:w	ファイルの上書き保存
:q!	保存せずにviを終了

コマンド	解説
x	文字の削除
dd	行の削除
yy	コピー
p	下に貼り付け

コマンド	解説
/	下に検索
?	上に検索
n	次を検索
N	上に次を検索
:set nu	行番号の表示
:set nonu	行番号の非表示



ご清聴ありがとうございました