

Ping-t LinuC101

デバイスのUUIDを表示できるコマンド

- UUID : Universally Unique Identifier: 汎用一意識別子、全世界で重複が起きないように生成される一意な値
 - blkid
 - ブロックデバイス（HDDやCD-ROMなどのようにブロック単位でデータを転送するデバイス）の情報を表示する
 - lsblk
 - ブロックデバイスをツリー状に一覧表示するコマンドです。「--output」オプションで表示項目を指定する

```
lsblk --output NAME,UUID,FSTYPE
```

prコマンドで「httpd.conf」ファイルを整形する際、ヘッダに表示されるファイル名を「testfile」に変更したい。また、1ページあたり30行としたい。

```
pr -h testfile -l 30 httpd.conf
```

dpkgコマンドを使用して、「procmail_3.22-16_i386.deb」パッケージをインストールしたい。

- dpkg
 - -E : 同バージョンが既にインストールされていればインストールを行わない
 - -G : 新バージョンが既にインストールされていればインストールを行わない
 - -R : ディレクトリを再帰的に処理
 - -i(--install) : パッケージのインストール

```
dpkg -Gi procmail_3.22-16_i386.deb
dpkg -G --install procmail_3.22-16_i386.deb
```

viエディタのコマンドモードで、カーソルを1文字右へ移動するviコマンド

h	左
j	下
k	上
l	右
w	次の単語の先頭 (word)
b	現在または前の単語の先頭
e	現在または次の単語の末尾
0	行頭

^	行の最初の文字（スペースは除く）
\$	行末
Ctrl + d	半分下(down)
Ctrl + u	半分上 (up)
Ctrl + f	下(forward)
Ctrl + b	上 (backward)
行数G	指定した行への移動※1
G	最終行への移動

rpmコマンドを使用して、インストール前に「postfix-1.1.12-1.i386.rpm」ファイルからインストールされるファイルを調べたい

- rpm
 - q(--query) パッケージ名
 - 指定したパッケージがインストールされているかを紹介
 - l(--list)
 - 指定したパッケージに含まれるファイルの表示
 - p(--package)
 - 照会対象をパッケージファイルとする

シェル変数を環境変数にするコマンド

```
export LPIC
declare -x LPIC
```

yumコマンドを使用して、「php」パッケージをアンインストール

```
yum remove php
```

bzip2より圧縮率が高い圧縮形式

```
xz > bzip2 > gzip
```

「test」ディレクトリをxz形式で圧縮した「test.tar.xz」というアーカイブにしたい

```
tar cfJ test.tar.xz test
```

- c 新しいアーカイブを作成
- x アーカイブからファイルを展開
- t アーカイブの内容を一覧表示
- f アーカイブファイル名を指定
- v 処理の詳細情報を表示

```
-z gzipを通して圧縮/展開  
-j bzip2を通して圧縮/展開  
-J xzを通して圧縮/展開
```

作成したリンクがハードリンクかシンボリックリンクかを確認したい

```
ls -i  
ls -l
```

設定されている全てのシェル変数を一覧表示する

コマンド 説明

set	すべてのシェル変数と環境変数を表示 (-o でオプション設定)
declare	すべてのシェル変数と環境変数を表示
env	すべての環境変数を表示
printenv	指定の、またはすべての環境変数を表示

yumコマンドを使用して、「Emacs」パッケージグループをインストールしたい

```
yum groupinstall Emacs
```

check-update	アップデート対象のパッケージリストを表示する
update パッケージ名	指定したパッケージをアップデートする
install パッケージ名	指定したパッケージをインストールする
remove パッケージ名	指定したパッケージをアンインストールする
info	指定したパッケージの情報を表示する
list	全パッケージ情報をリスト表示する
repolist	リポジトリ一覧を表示する
search キーワード	パッケージ情報をキーワードで検索する
search all キーワード	パッケージをキーワードで検索する
groups list	パッケージグループをリスト表示する
groups install グループ	指定したグループのパ

ext3ファイルシステムを「/dev/sda2」に作成したい。

```
mkfs -t ext3 /dev/sda2  
mke2fs -j /dev/sda2  
mke2fs -t ext3 /dev/sda2
```

- mke2fs : ext2/ext3/ext4ファイルシステムを作成

- ext3の場合: 「-j」 or 「-t ext3」
- mkfs: 「-t」 オプションで、ext2/ext3/ext4/xfs/jfsなど作成

「file」ファイル内でスペースが連続している場合、スペース1つに置き換えて表示させたい

```
tr -s [:space:] < file
cat file | tr -s " "
```

[:alnum:]	アルファベットと数字
[:alpha:]	アルファベット
[:blank:]	スペースや水平タブ
[:cntrl:]	コントロール文字
[:digit:]	数字
[:graph:]	スペースを含まない表示可能な文字
[:lower:]	小文字アルファベット
[:print:]	スペースを含む表示可能な文字
[:punct:]	句読点
[:space:]	スペースや水平タブ, 垂直タブ
[:upper:]	大文字アルファベット
[:xdigit:]	16進数文字

「configure.xz」ファイルを展開することができるコマンド

```
xz -d configure.xz          # -d, --decompress 圧縮ファイルの展開
xz -d -k configure.xz      # -k, --keep 元のファイルを削除しない
unxz configure.xz
```

xz -l は「圧縮ファイルの情報を表示する」

改行コード

- CRLF (\r\n) : Windows
- LF (\n) : Unix OS (Linux, Mac OS Xなど)
- CR (\r) : 古いMacOS (バージョン9まで)

Linux側でこのシェルスクリプトを実行すると、Windowsの改行コードCRLFのままのファイルのため、LF (\n) の直前のCR (\r) が不正な文字と解釈され、以下の通りエラーになります。

```
> od -t cx1 windows
```

改行の位置に\r\n (0d 0a) が存在することがわかります。こういった場合、CR (\r) を取り除くことでLinuxの改

行コード：LF（\n）のみにでき、正しく扱うことができます。

```
> tr -d '\r' < windows > linux
```

apt-fileコマンドを使用して、パッケージ情報を最新版に更新したい。

```
apt-file update
```

- update
 - パッケージ情報を最新版に更新
- search (or find) 検索パターン
 - パスに検索パターンが入っているファイルを含むパッケージを検索
- list (or show) パッケージ名
 - パッケージに含まれているファイルを一覧表示

BIOSかUEFIか（ファームウェアの話）

- UEFI
 - Unified Extensible Firmware Interface
 - 最新
 - ESP（EFIシステムパーティション）はUEFIシステムにおいて、ブートローダや起動に必要なドライバなどが置かれている

Btrfs（ファイルシステムの話）

- ファイルシステム
- B-tree File System: バターエフエス
- 高機能
 - マルチデバイスへの対応
 - LNM要らず
 - サブボリューム / スナップショット
 - 圧縮

正規表現について

- * 直前の文字の0回以上の繰り返し
例：1*3.txt は、13.txt、113.txt などを表す。
- . 任意の1文字の文字列
例：1..4.txt は、1234.txt、1ab4.txt などを表す。
- [] [] 内の任意の1文字の文字列
[abc]：a,b,cのどれか
[a-z]：a~zのどれか
[^abc]：a,b,c以外のどれか
例：1[23]4.txt は、124.txt または 134.txt を表す。
- ^ 行頭
例：^1 は、行頭にある 1 を表す。

```
$      行末
      例 : 1$ は、行末にある 1 を表す。
+      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
|      前後のいずれかの文字列
```

virshコマンド

- virshコマンドはXenやKVMなどの、さまざまな仮想化ソフトウェアを管理するコマンド

```
help : サブコマンドを表示

create : XML形式のファイルから新規の仮想マシンを作成し、起動

console : 仮想マシンのコンソールに接続

start : 仮想マシンの起動

shutdown : 仮想マシンの終了

destroy : 仮想マシンの強制終了

reboot : 仮想マシンの再起動

suspend : 仮想マシンの一時停止

resume : 停止中の仮想マシンの再開

list : 仮想マシンの一覧表示

dumpxml : 仮想マシンの定義ファイルをXML形式で出力
```

Linux上で、一般ユーザが「mount /mnt/usb」コマンドで特定のUSBメモリを「/mnt/usb」ディレクトリにマウントして使用できるようにしたい。rootユーザとして事前にどのような作業が必要か

- USBメモリのUUIDを確認し、マウント設定のデバイス名に記述する
 - blkid
- 「/etc/fstab」ファイルにUSBメモリのマウント設定を記述する
- mount
 - mount /mnt/usb

manコマンドのセクション

- 1 だれもが実行できるユーザコマンド
- 2 システムコール、つまり、カーネルが提供する関数
- 3 サブルーチン、つまり、ライブラリ関数
- 4 デバイス、つまり、/dev ディレクトリのスペシャルファイル
- 5 ファイルフォーマットの説明、例 /etc/passwd
- 6 ゲーム（説明不要だろうネ）

- 7 その他 例： マクロパッケージや取り決めの文書
- 8 システム管理者だけが実行できるシステム管理用のツール
- 9 Linux 独自のカーネルルーチン用のドキュメンテーション
- n 新しいドキュメンテーション：よりふさわしい場所に移動されるだろう
- o 古いドキュメンテーション 猶予期間として保存されているもの
- l 独自のシステムについてのローカルなドキュメンテーション

lspciで出力される内容

lspciコマンドの実行結果から以下のような情報が読み取れます。

- ・(1)PCI識別番号 00:03.3
- ・(2)PCIデバイスの種類 Serial controller
- ・(3)ベンダー名 (ベンダーID) Intel Corporation
- ・(4)デバイス名 Mobile 4 Series Chipset AMT SOL Redirection (rev 07)
- ・(5)バスの速度(詳細表示のみ) 66MHz
- ・(6)IRQ番号(詳細表示のみ) 17
- ・(7)I/Oポートアドレス(詳細表示のみ) 1830

systemdが稼働するシステムにおいて、次回起動時にグラフィカルログインさせるようにしたい。実行すべきコマンドはどれか

```
rm -f /etc/systemd/system/default.target  
  
ln -s /lib/systemd/system/graphical.target /etc/systemd/system/default.target
```

sedコマンド「test.txt」ファイル内の全ての「a」を「A」に、また「b」を「B」に置換したい。

```
sed y/ab/AB/ test.txt  
sed -e s/a/A/g -e s/b/B/g test.txt  
  
sed [オプション] 編集コマンド [ファイル名]  
sed [オプション] -e 編集コマンド1 [-e 編集コマンド2 ...] [ファイル名]  
sed [オプション] -f スクリプト [ファイル名]
```

現在実行中のプロセスからtestユーザーが実行した特定の名前を持つプロセスIDを検索したい

```
pgrep -u test  
  
pgrep [ オプション ] 検索パターン
```

例) カレントディレクトリ以下にある14日以上アクセスの無いファイルを削除する場合

```
$ find . -atime +13 -type f -exec rm {} \;
```

デバイスが使用中のDMAチャンネルに関する情報

```
/proc/dma
```

DMAとは、CPUを介することなくメインメモリと周辺機器の間で直接的に情報転送を行う方式です。DMAチャンネルは周辺機器がDMAのコントローラ（制御装置）に対して情報転送を要求するために使用する通信経路の事

ping localhost > ping.txt」 コマンドをログアウト後もバックグラウンドで実行し続けたい

```
nohup ping localhost > ping.txt &
```

test.txt」 ファイルから、「PINGT」または「pingt」という文字列を含む行を抽出したい。

```
grep -E 'PINGT|pingt' test.txt  
egrep 'PINGT|pingt' test.txt
```

10Kバイトのサイズを指定してファイルを作成したい

```
dd if=/dev/zero of=file1 bs=10K count=1
```

ホットプラグデバイスを接続した際に、デバイスファイルを動的に作成する仕組みは

- udev

プログラム「/usr/local/bin/myapp」の現バージョンのファイルを残したまま、新しいバージョンのファイルをインストールして使いたい。

```
ls /usr/local/bin  
myapp-1.0 myapp-2.0  
  
ln -s /usr/local/bin/myapp-2.0 /usr/local/bin/myapp
```

D-Bus

- Linuxで使われるプロセス間通信機構
 - D-Bus（Desktop Bus）は、プログラム同士が情報を伝達するプロセス間通信機構のひとつです。Linuxではdbus-daemonなどがプロセス間通信の中継を行います。この機能により、例えば新しいデバイスの認識情報を他のアプリケーションに伝達し、そのアプリケーションが新しいデバイスをすぐに使えるようになるなどのシステム管理上の利便性も向上します。

IRQ

- IRQに関する情報は「/proc/interrupts」ファイルで確認できます。IRQ(Interrupt ReQuest)とは、マウスやキーボードなどの周辺機器(デバイス)からCPUへの割り込み要求のことです。IRQには0から1、2、3のように順に番号がつき、そのうちのいくつかは特定のデバイスに割り当てられています。例えば0はシステムタイマー、1はキーボードに割り当てられています。なお、IRQ番号が重複しているとハードウェアが正常に動作しないことがあります。

partedを使って、新しいハードディスクに以下の要件の通りパーティションを作成した。

- 先頭に一つの1000MBの基本パーティション
- ファイルシステムはext4
- パーティションテーブルはMBR形式

```
mklabel msdos

print

mkpart primary ext4 1 1000MB
```

Xクライアント（172.16.0.4）上で実行したプログラム「xeyes」をXサーバ（172.16.0.1）のディスプレイに表示させたい

Xクライアント上で実行したプログラムをXサーバのディスプレイに表示させるには、以下のような手順が必要です。

■Xサーバ側

xhostコマンドで、「172.16.0.4」からのアクセスを許可します。

```
$ xhost +172.16.0.4
172.16.0.4 being added to access control list
```

■Xクライアント側

環境変数DISPLAYで表示先（Xサーバ）を指定し、Xサーバに表示させたいプログラム（xeyes）を実行します。環境変数DISPLAYの書式は以下の通りです。

[ホスト名]：ディスプレイ番号

デフォルトのディスプレイを使用する場合はディスプレイ番号に0を指定します。

```
$ DISPLAY=172.16.0.1:0
$ export DISPLAY
$ xeyes &
```