

Linux

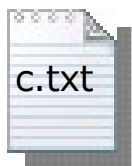
ファイルとディレクトリ

9 時間目

ファイルとディレクトリ

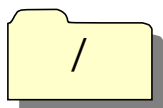
解説

ファイルとディレクトリはとても重要なワードです。
言葉だけでなく意味もしっかりと理解しておきましょう。



コンピュータで扱われるデータを**ファイル**という

→現実世界でいう「ノート」みたいなもので、
様々な情報を記録、編集、保存、実行するためのもの



ファイルなどを入れる保存場所を**ディレクトリ**という

→Windowsではファイルの保存場所のことをフォルダと呼ぶが、
Linuxではディレクトリと呼ぶ

ファイルについて

解説

LinuxでいうファイルはWindowsやMacとほぼ同じものと考えていいです。テキストやWordで作られたファイルをイメージするといいでしょう。ファイルに名前を付ける場合、Linuxでは以下のルールがあります。

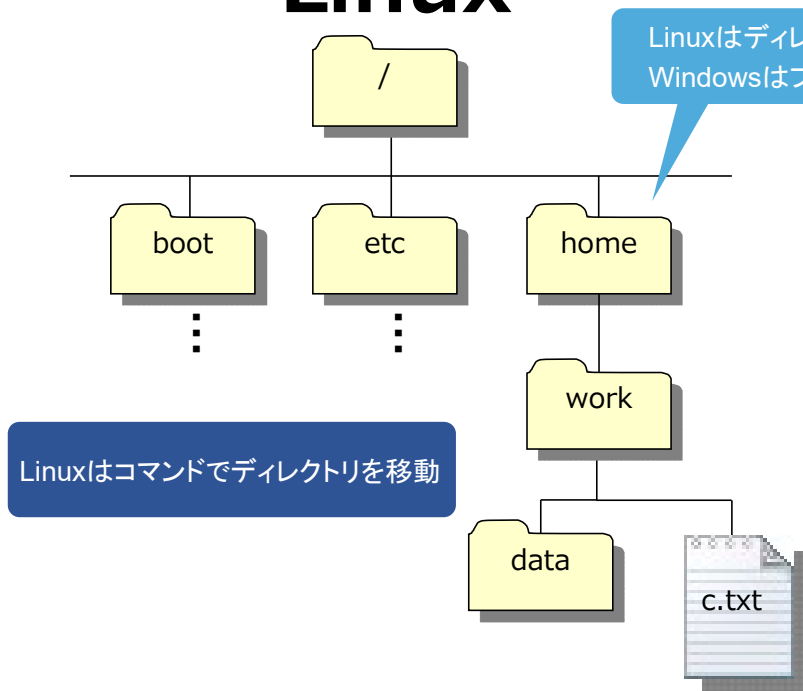
ファイル名の決まり事	例
アルファベット、数字、一部の記号（「_」 「.」 「-」）を使うことが出来る	・ Server01.log ・ data_org ・ BKUP-2017.txt
ファイル名に使ってはいけない記号がある	* ! ? / # \$ % & ¥ ' @ : ; など
アルファベットの大文字、小文字の区別がある	「est.txt」と「TEST.txt」は、Linuxでは別のファイルとして認識する
OSが日本語に対応している場合のみ、日本語で名前を付けることができる	・ 顧客情報.csv ・ 出席簿_0323.txt ・ 成績.txt
Linuxには拡張子という概念がない	—

ディレクトリについて

解説

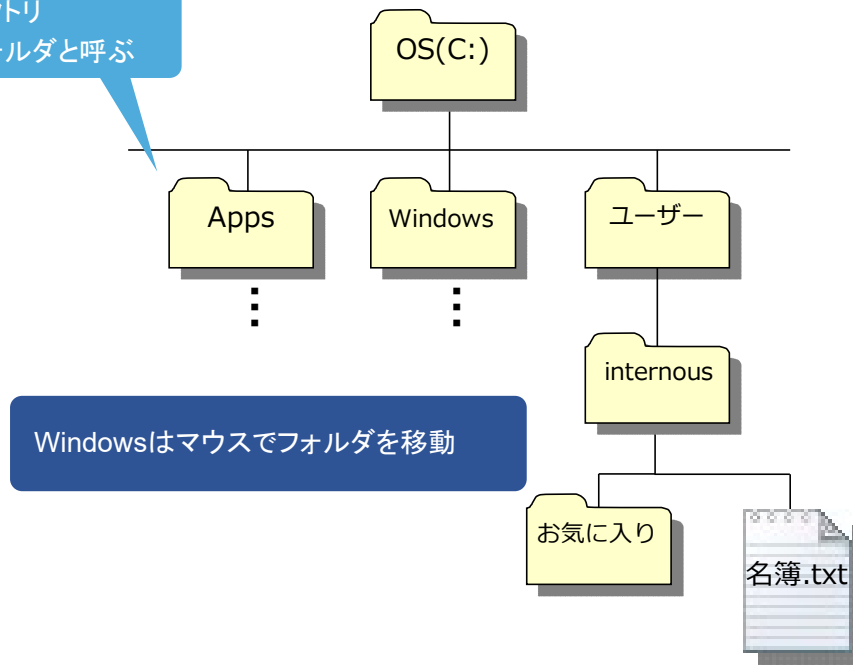
ディレクトリやフォルダは以下の図のように階層構造になっています。
Windowsと比較して見てみましょう。

Linux



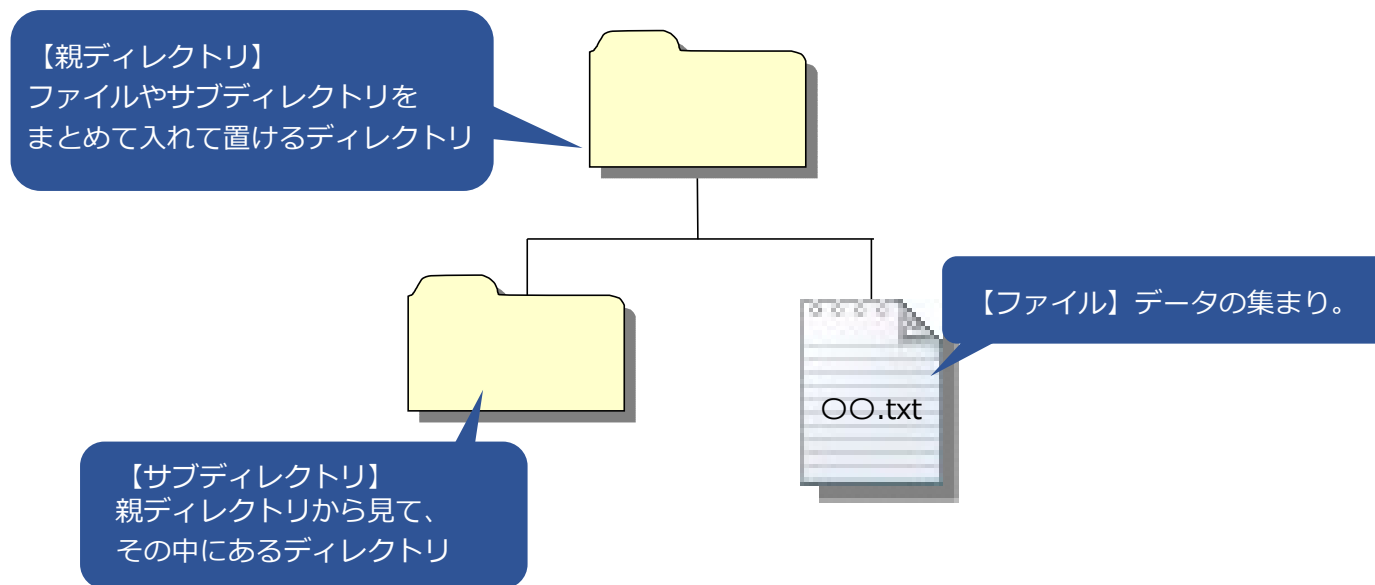
Linuxはディレクトリ
Windowsはフォルダと呼ぶ

Windows



ディレクトリについて

ディレクトリとファイルの呼び方



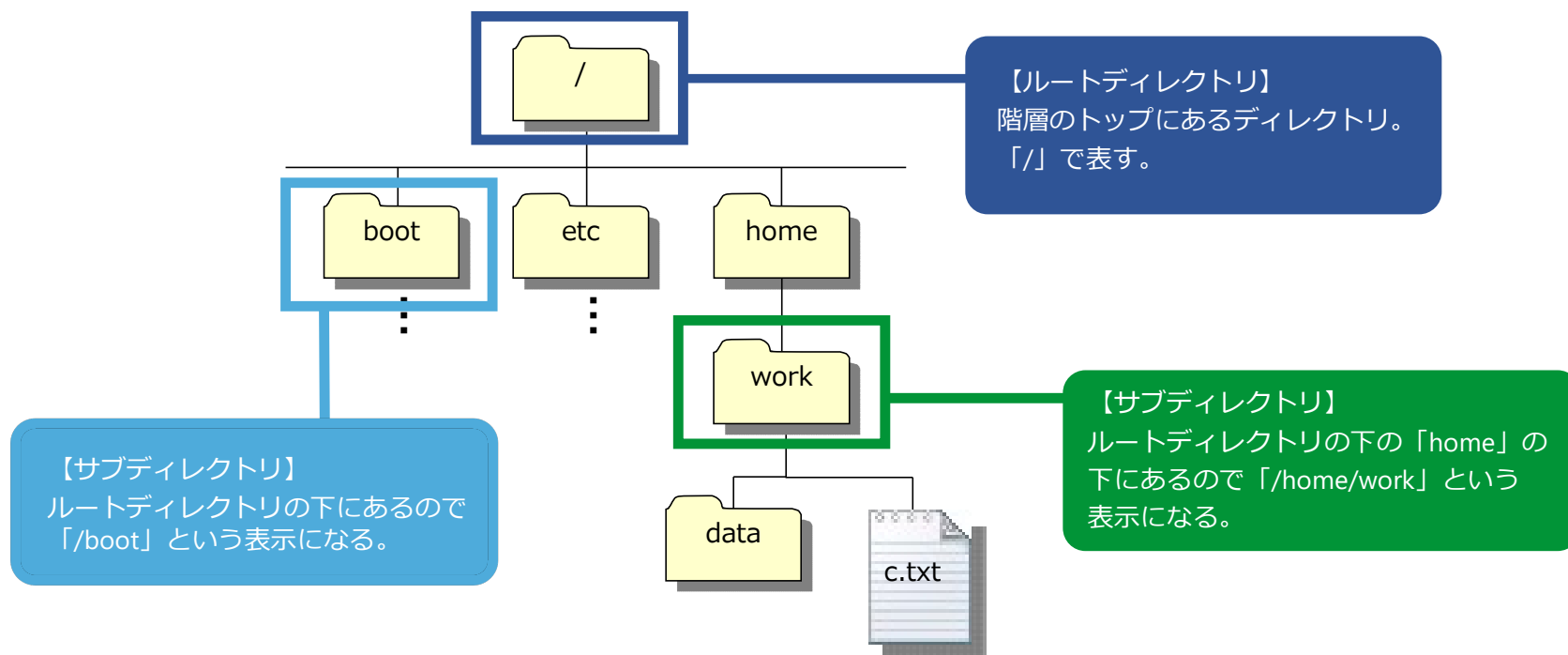
ディレクトリについて

ディレクトリの決まり事を確認する

ディレクトリの決まり事
ディレクトリの中には、ファイルやディレクトリを入れて管理ができる
ディレクトリは、ファイルと同じように名前を付けて使う
ディレクトリは中身ごと移動、削除できる
ディレクトリ名は使ってはいけない記号など、ファイル名の命名規則と同様の命名規則ある
空のディレクトリはメールなどに添付できない

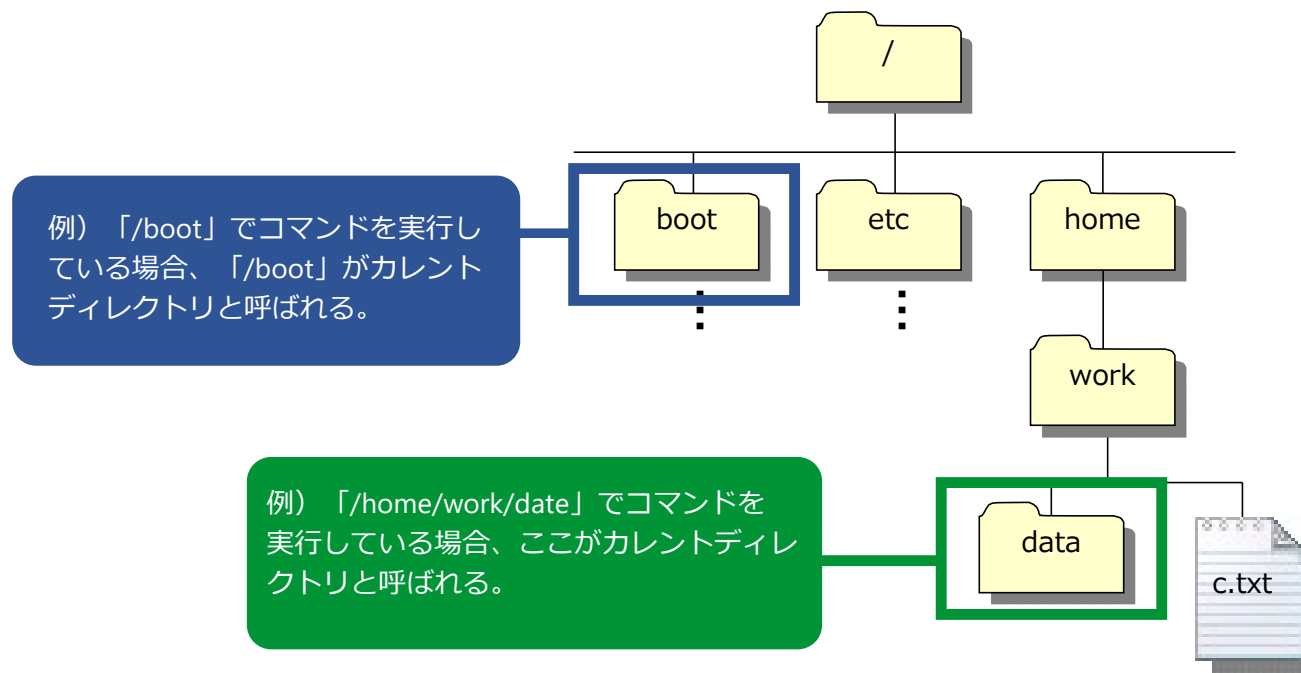
ルートディレクトリについて

ルートディレクトリとはすべてのファイルやディレクトリの一番上の階層にあるディレクトリのこと



カレントディレクトリについて

カレントディレクトリとは現在ユーザーが
コマンドを実行しているディレクトリ(場所)のこと

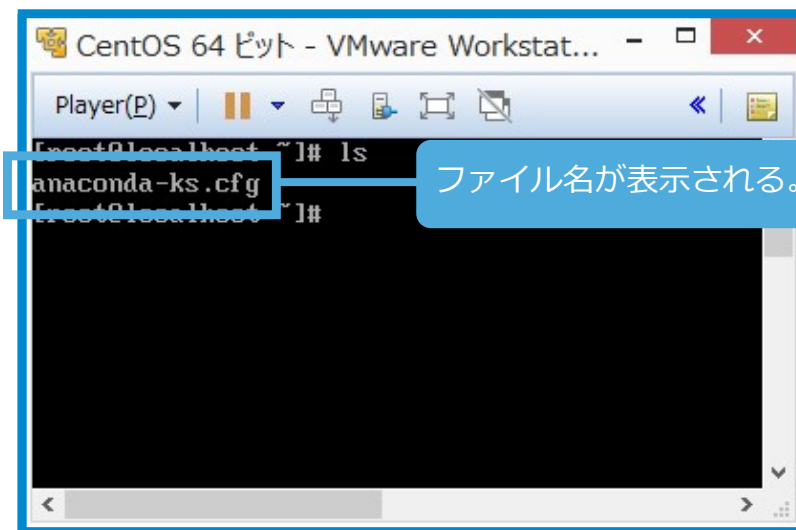
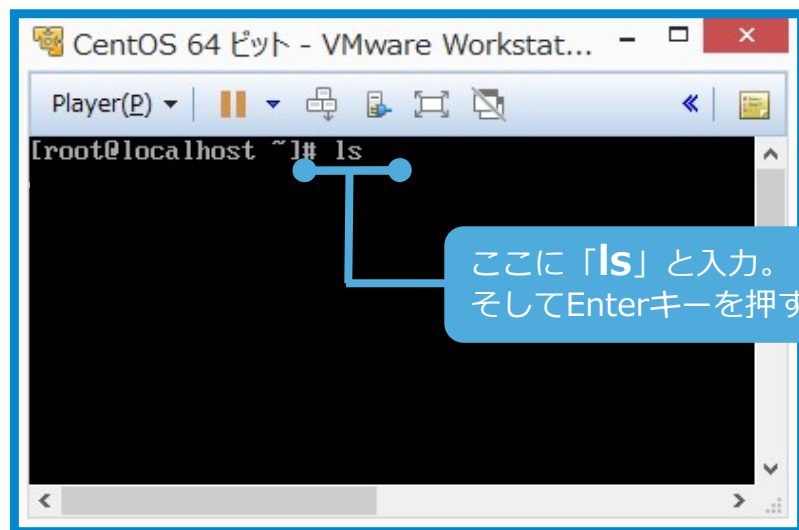


ファイルを表示する 「ls」 コマンド

解説

ファイルの命名規則が分かったところで、まずはファイルを表示してみましょう。
ファイルの表示は「ls」コマンドを使います。今は、何のファイルも作成していないのでインストール時に自動で生成された「anaconda-ks.cfg」というファイルが1つだけ表示されます。

3 ls ファイルやディレクトリのリストを表示する



ファイルを作成する 「touch」 コマンド

解説

ファイル名の命名規則に従ってファイルを新規作成してみましょう。ファイルを新規作成する方法はいくつかありますが、今回は「touch」コマンドを使います。新規作成するファイル名は「test_01.txt」とします。

4 touch **ファイル名** ファイルを作成する

CentOS 64 ビット - VMware Workstat...

Player(P) | [Icons] [Buttons]

```
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]# touch test_01.txt
```

ここに「**touch**」とファイル名「**test_01.txt**」と入力。
そしてEnterキーを押す。

ここに「**touch**」とファイル名
「**test_01.txt**」と入力。
そしてEnterキーを押す。

CentOS 64 ビット - VMware Workstat...

Player(P) | [Icons: Full Screen, Settings, Print, Copy, Paste, Undo, Redo, Find, Help]

```
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]# touch test_01.txt
[root@localhost ~]#
```

ここでは何も表示されません。

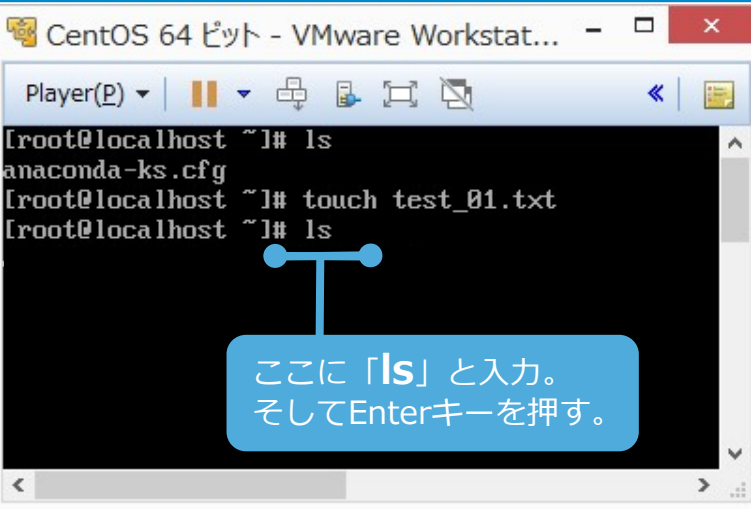
ここでは何も表示されません。

ファイルを表示する 「ls」 コマンド

解説

新規作成したファイル「test_01.txt」が正しく作成されたのか確認してみましょう。
もう一度「ls」コマンドを実行してファイルを表示します。

3 ls ファイルやディレクトリのリストを表示する

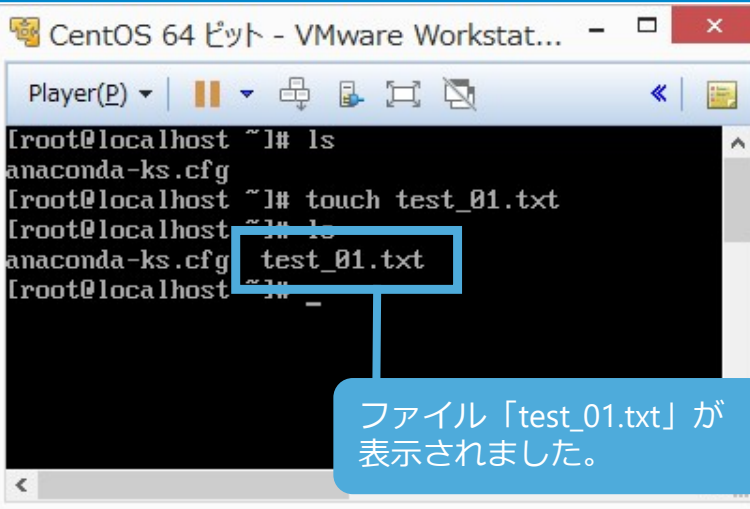


CentOS 64 ビット - VMware Workstat...

Player(P) | [Icons] | [Navigation]

```
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]# touch test_01.txt
[root@localhost ~]# ls
```

ここに「ls」と入力。
そしてEnterキーを押す。



CentOS 64 ビット - VMware Workstat...

Player(P) | [Icons] | [Navigation]

```
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]# touch test_01.txt
[root@localhost ~]# ls
anaconda-ks.cfg  test_01.txt
[root@localhost ~]#
```

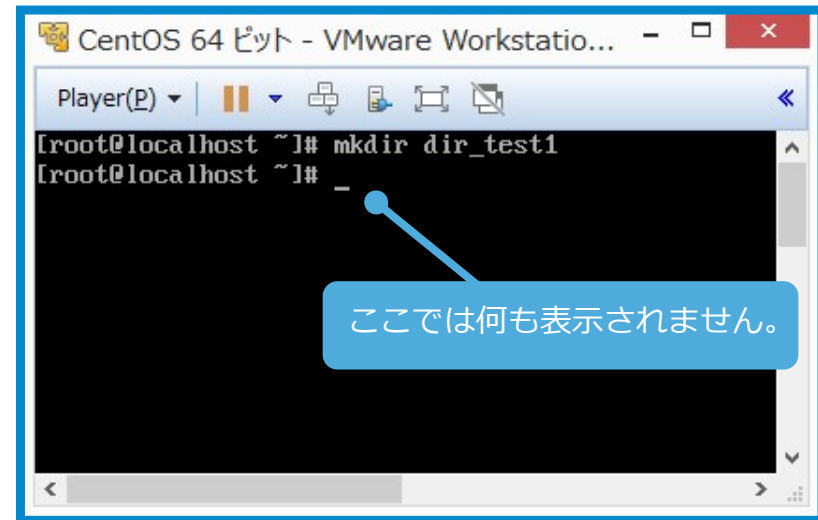
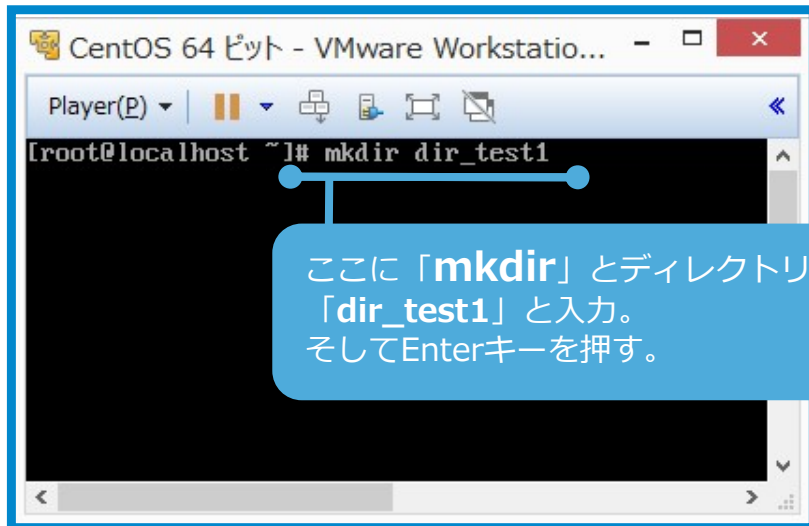
ファイル「test_01.txt」が
表示されました。

ディレクトリを作成する 「mkdir」 コマンド

解説

ディレクトリを作ってみましょう。ディレクトリの新規作成は「mkdir」コマンドを使います。
ディレクトリ名は「dir_test1」とします。

5 mkdir ディレクトリ名 ディレクトリを作成する

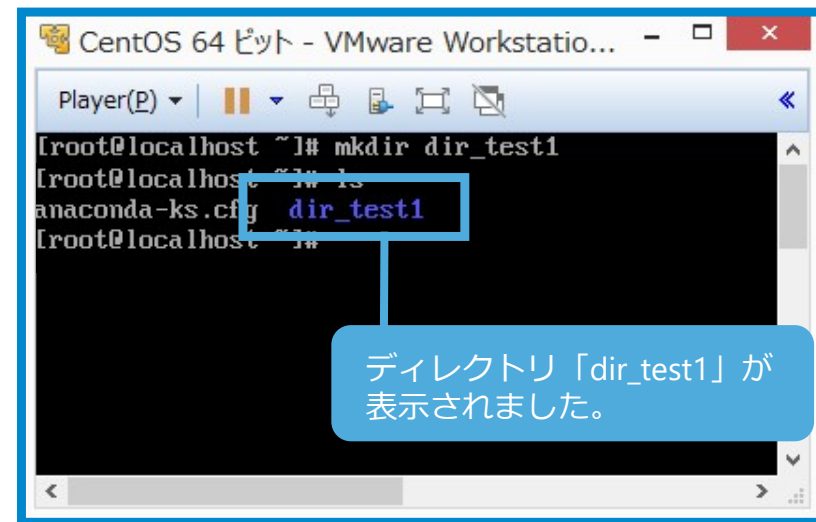
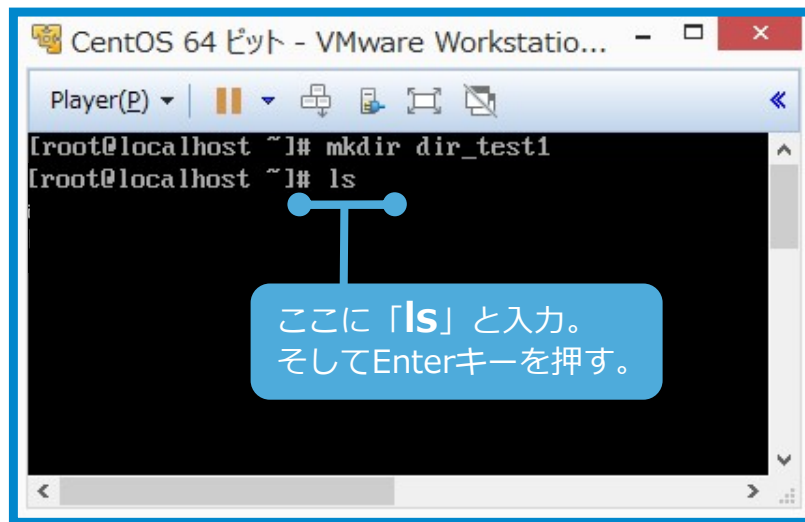


ディレクトリを表示する 「ls」 コマンド

解説

新規作成したディレクトリ「dir_test1」が正しく作成されたのか確認してみましょう。
もう一度「ls」コマンドを実行してディレクトリとファイルを表示します。

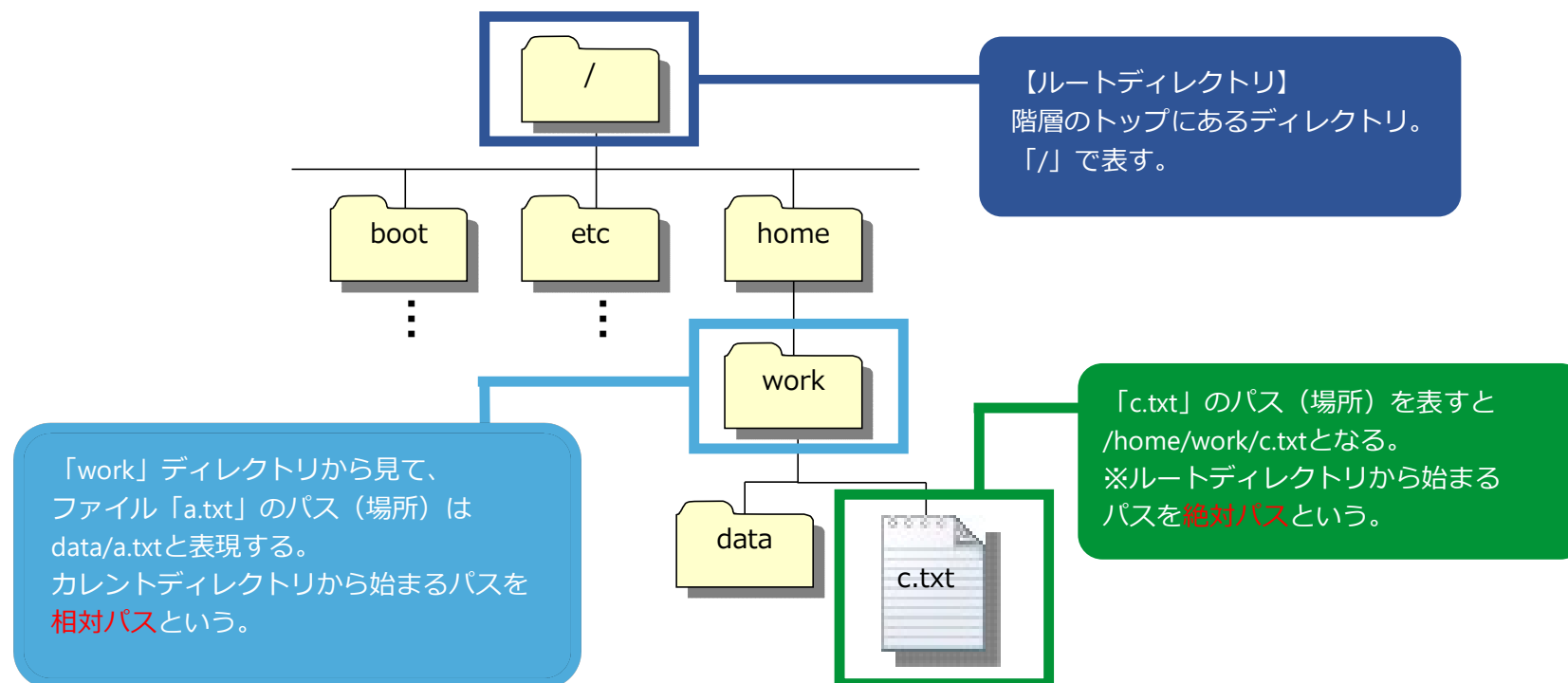
3 ls ファイルやディレクトリのリストを表示する



パス(場所)について

解説

ディレクトリの階層構造学びましたが、特定のディレクトリやファイルがどこに在るかを識別する仕組みが必要になります。それが**パス**です。パスはディレクトリやファイルの場所を示す住所のようなもので、パスで階層の境目を表現する場合、Windowsでは「¥」を使いますが、Linuxでは「/」を使います。



カレントディレクトリを確認する 「pwd」 コマンド

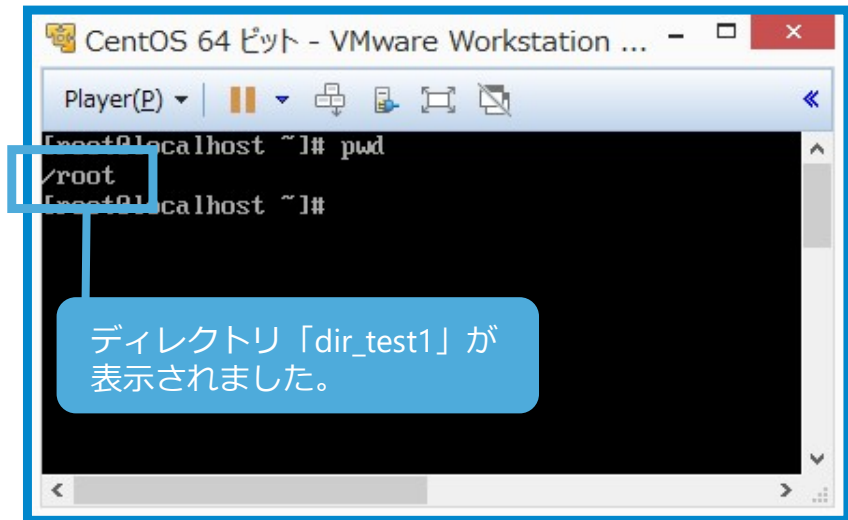
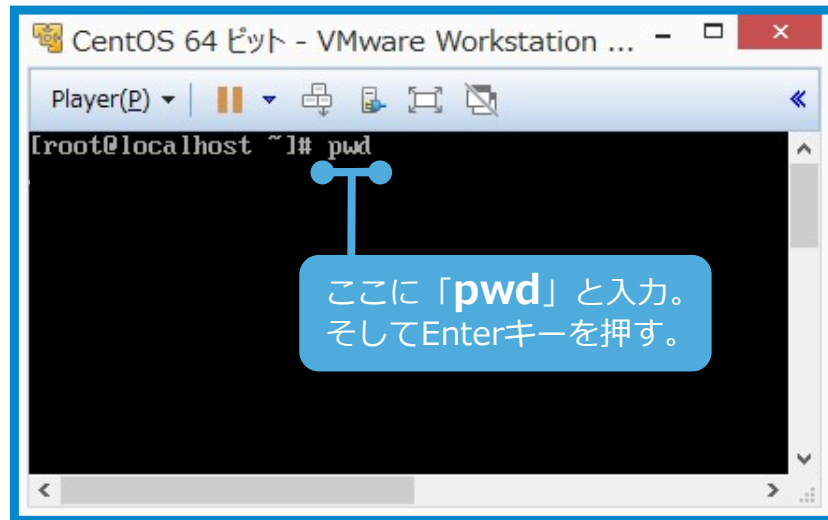
解説

自分（ログインしているユーザ）がどのディレクトリで作業しているのか、カレントディレクトリ(現在の場所)を確認してみましょう。確認は「pwd」コマンドを使います。

6

pwd

カレントディレクトリを確認する



ディレクトリを移動する 「cd」 コマンド

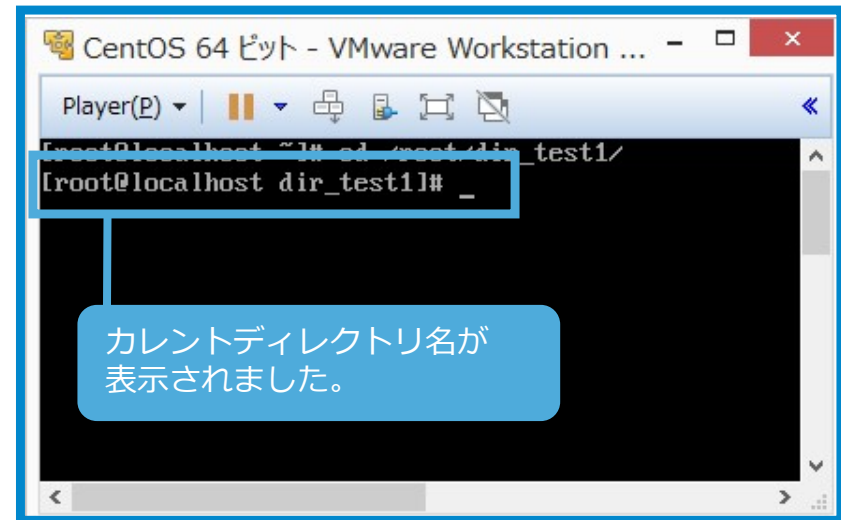
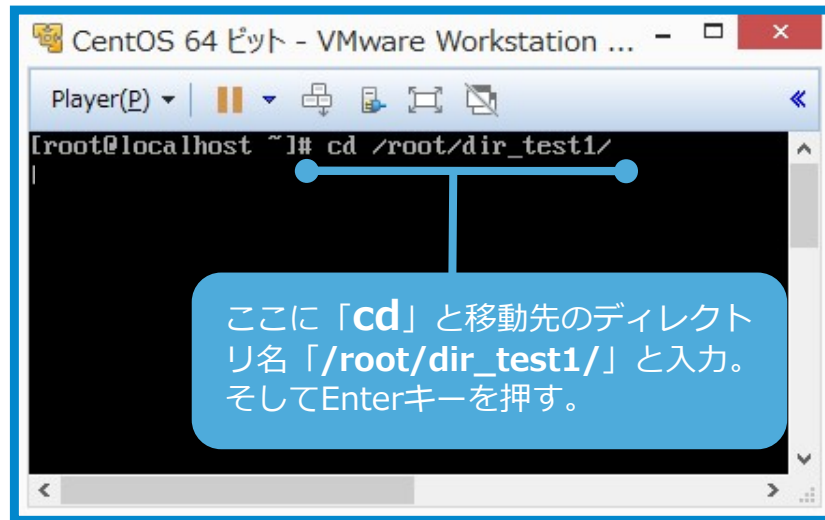
解説

カレントディレクトリから別のディレクトリに移動することにより、カレントディレクトリ位置を変更することができます。ディレクトリの移動は「cd」コマンドを使います。先ほど新規作成したディレクトリ「dir_test1」に移動して見ましょう。

7

cd ディレクトリ名

ディレクトリを移動する



カレントディレクトリを確認する 「pwd」 コマンド

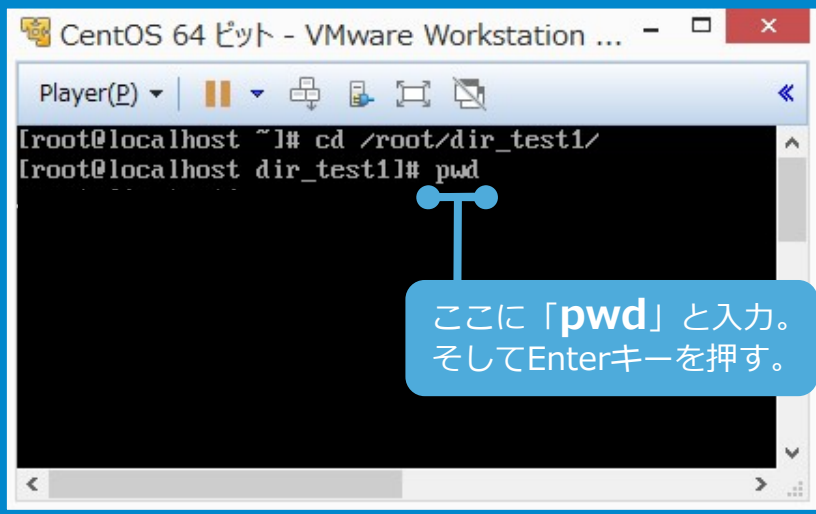
解説

もう一度自分（ログインしているユーザ）がどのディレクトリで作業しているのか、カレントディレクトリ(現在の場所)を確認してみましょう。

6

pwd

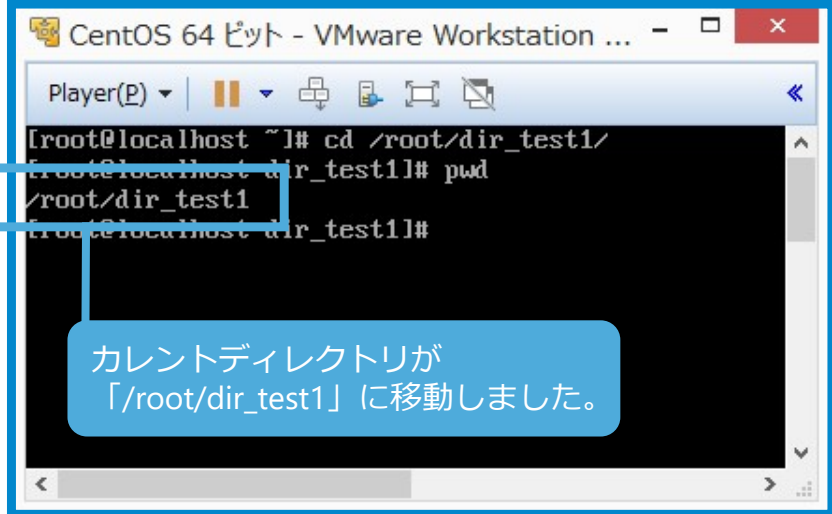
カレントディレクトリを確認する



CentOS 64 ビット - VMware Workstation ...

```
Player(P) | [Icons] <<
[root@localhost ~]# cd /root/dir_test1/
[root@localhost dir_test1]# pwd
```

ここに「pwd」と入力。
そしてEnterキーを押す。



CentOS 64 ビット - VMware Workstation ...

```
Player(P) | [Icons] <<
[root@localhost ~]# cd /root/dir_test1/
[root@localhost dir_test1]# pwd
/root/dir_test1
[root@localhost dir_test1]#
```

カレントディレクトリが
「/root/dir_test1」に移動しました。

ファイルをコピーする 「cp」 コマンド①

解説

「cp」 コマンドはファイルやディレクトリをコピーするコマンドです。
ここではファイルをコピーしていきます。本講座では2つのコピーパターンをご紹介します。

8

【パターン①】

cp [コピー元のファイル名] [コピー先のファイル名]

⇒ あるファイルをコピーする。ファイル名は別名でつける。

【パターン②】

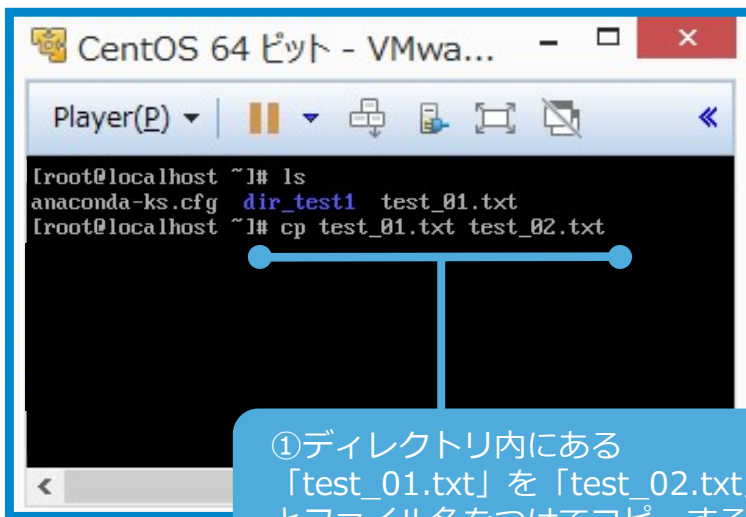
cp [コピー元のファイル名] [コピー先のディレクトリ名]

⇒ あるファイルを別ディレクトリにコピーする。ファイル名は同じままでコピー。

ファイルをコピーする 「cp」 コマンド②

【パターン①】

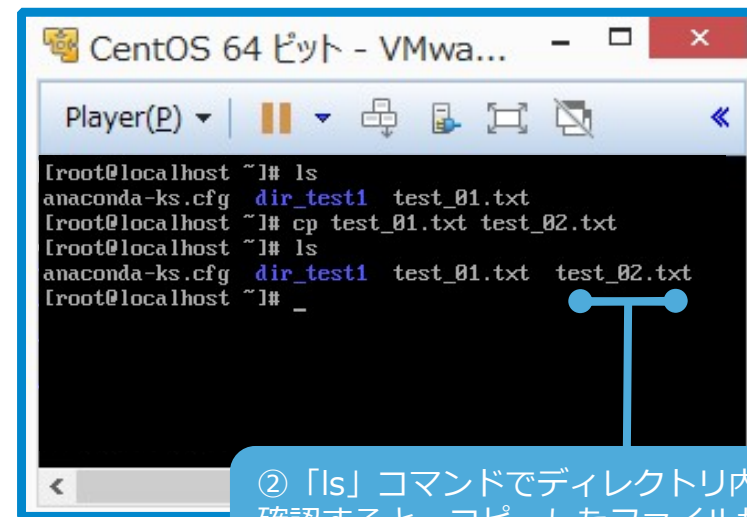
cp コピー元のファイル名 コピー先のファイル名



A terminal window titled "CentOS 64 ビット - VMwa..." showing the execution of the cp command. The prompt is [root@localhost ~]#. The first command is ls, which lists the files anaconda-ks.cfg, dir_test1, and test_01.txt. The second command is cp test_01.txt test_02.txt. A blue line with dots at both ends highlights the command and its output.

```
[root@localhost ~]# ls
anaconda-ks.cfg  dir_test1  test_01.txt
[root@localhost ~]# cp test_01.txt test_02.txt
```

①ディレクトリ内にある
「test_01.txt」を「test_02.txt」
とファイル名をつけてコピーする。



A terminal window titled "CentOS 64 ビット - VMwa..." showing the confirmation of the copied file. The prompt is [root@localhost ~]#. The first command is ls, which lists the files anaconda-ks.cfg, dir_test1, test_01.txt, and test_02.txt. The second command is cp test_01.txt test_02.txt. The third command is ls, which lists the files anaconda-ks.cfg, dir_test1, test_01.txt, and test_02.txt. The fourth command is _ (underscore). A blue line with dots at both ends highlights the command and its output.

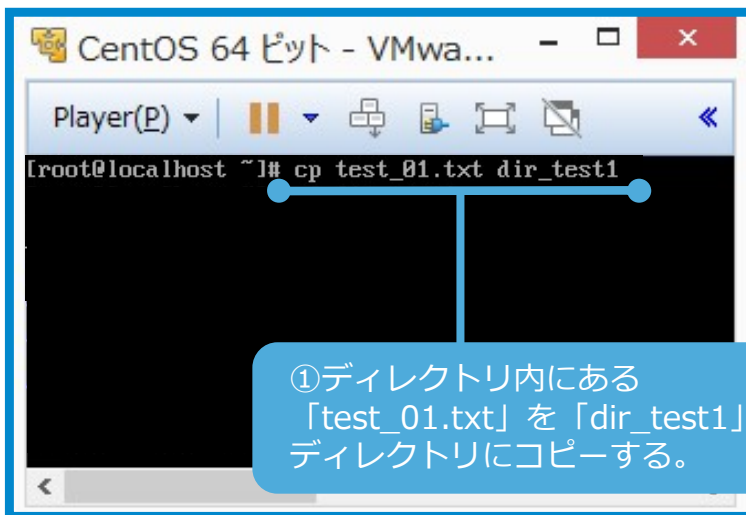
```
[root@localhost ~]# ls
anaconda-ks.cfg  dir_test1  test_01.txt
[root@localhost ~]# cp test_01.txt test_02.txt
[root@localhost ~]# ls
anaconda-ks.cfg  dir_test1  test_01.txt  test_02.txt
[root@localhost ~]# _
```

②「ls」コマンドでディレクトリ内を
確認すると、コピーしたファイルが
出来上がっている。

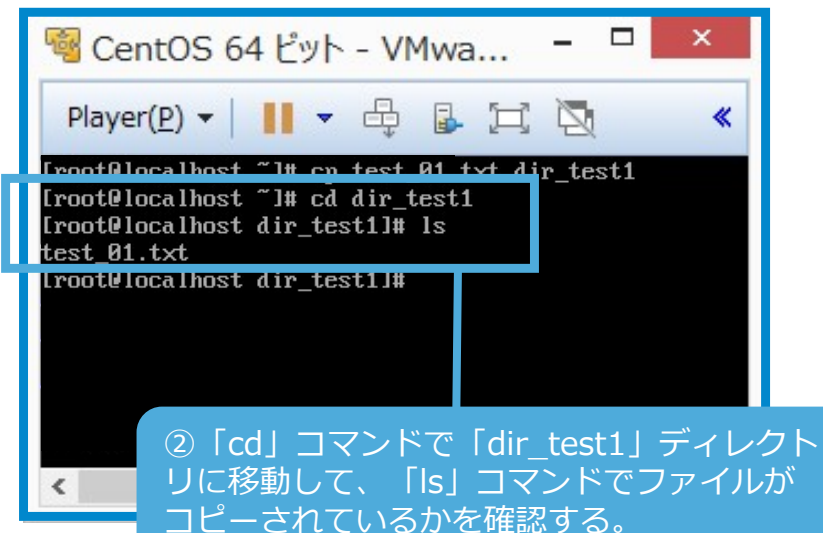
ファイルをコピーする 「cp」 コマンド③

【パターン②】

cp コピー元のファイル名 コピー先のディレクトリ名



A terminal window titled "CentOS 64 ビット - VMwa..." showing the command `cp test_01.txt dir_test1` being executed. A blue callout box points to the command with the text: ①ディレクトリ内にある「test_01.txt」を「dir_test1」ディレクトリにコピーする。



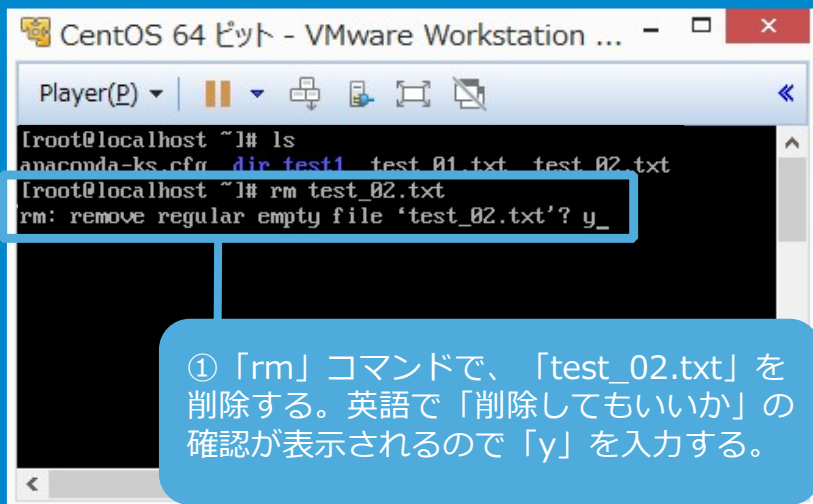
A terminal window titled "CentOS 64 ビット - VMwa..." showing the sequence of commands: `cp test_01.txt dir_test1`, `cd dir_test1`, and `ls`. The output of `ls` is `test_01.txt`. A blue callout box points to the `cd` and `ls` commands with the text: ②「cd」コマンドで「dir_test1」ディレクトリに移動して、「ls」コマンドでファイルがコピーされているかを確認する。

ファイル、ディレクトリを削除する 「rm」 コマンド

解説

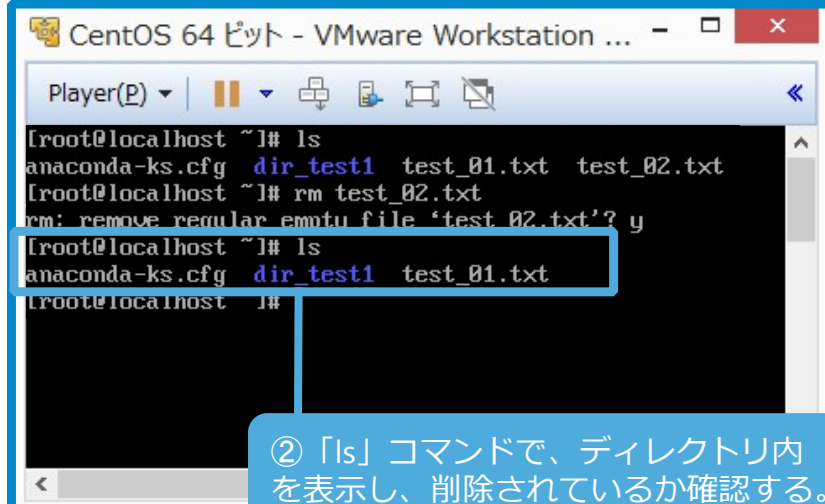
「rm」 コマンドはファイルとディレクトリ(中にファイルがあるディレクトリのみ)を削除するコマンドです。中にファイルが入っていないディレクトリを削除するときは「mkdir」 コマンドを使用しますので、覚えておきましょう。

9 rm ファイル名 ファイル/ディレクトリを削除する



```
CentOS 64 ビット - VMware Workstation ...
Player(P)
[root@localhost ~]# ls
anaconda-ks.cfg  dir_test1  test_01.txt  test_02.txt
[root@localhost ~]# rm test_02.txt
rm: remove regular empty file 'test_02.txt'? y_
```

① 「rm」 コマンドで、「test_02.txt」を削除する。英語で「削除してもいいか」の確認が表示されるので「y」を入力する。



```
CentOS 64 ビット - VMware Workstation ...
Player(P)
[root@localhost ~]# ls
anaconda-ks.cfg  dir_test1  test_01.txt
[root@localhost ~]#
```

② 「ls」 コマンドで、ディレクトリ内を表示し、削除されているか確認する。

ファイル、ディレクトリを削除する 「rm」 コマンド

解説

「rm」 コマンドにもオプションをつけて使用することが多いです。2つのオプションコマンドを紹介します。

【オプションコマンド】

rm -f 削除するファイル名

⇒最終確認をしないで、いきなりファイルを削除する。

例1) `rm -f ./test.txt`

rm -r 削除するディレクトリ名

⇒ディレクトリとその中身を1つずつ削除。毎回、確認メッセージが表示される。

例1) `rm -r ./Data_dir_01`

rm -rf 削除するディレクトリ名

⇒ディレクトリとその中身を一括で削除。確認メッセージを表示せず、すぐに削除する。

例2) `rm -rf ./Data_dir_01`