

Unixの基本コマンド (注) はスペース

mkdir dirA	ディレクトリを作成する
ls	カレントディレクトリの情報を表示する
ls dirA	dirAの中身を表示する
ls -a	隠しファイルを含むすべてのファイルを表示する
ls -l	ファイルの詳細を表示する (ファイル形式, 作成日, 実行時刻, アクセス権等) アクセス権の記述法 d[rwx][rwx][rwx] (順に[自分][グループ][全員]の権限を示す) (r...読み出し, w...書き込み, x...実行, -...不可)
cd dirA	dirAに移動する
cd ..	ひとつ上のディレクトリに移動する
cd	ホームディレクトリに移動する
cd -	一つ前のディレクトリに戻る
cd /	ルートディレクトリに移動する
pwd	カレントディレクトリの場所を絶対パスで表示する
touch fileA	fileA (空のファイル) を作成する。
mv fileA fileB	fileAの名前をfileBに変更する (名前の変更)
mv fileB dirA	fileAをdirAの中へ移動する (ファイルの移動)
mv dirA dirB	dirAの名前をdirBに変更する (名前の変更)
cp fileA dirA	fileAをdirAの中へコピーする
cp fileA fileB	fileAをfileBにコピーする
cp -r dirA dirB	dirAをdirBにコピーする
rm fileA	fileAを削除する (注) Unixでは、ゴミ箱はなく、削除すると復元できないので、注意すること。
rm -i fileA	fileAを削除するかどうかを尋ね、Yesなら消す。rm -i test.*のようにワイルドカードと組み合わせて使う。
rm -r dirA	ディレクトリを (中身も含む) 削除する
exit	シェルを終了する

<役立つキー>

~/ ホームディレクトリ
./ カレントディレクトリ
* "全部"を意味する

<ターミナルのショートカットキー>

Ctrl + l 画面をクリアする
tab ディレクトリ名等の入力を補完する

シェルスクリプト (bash)

シェルスクリプトは、一連の処理を行うコマンドをファイルに保存したものです。そのファイルを実行することで、毎回、コマンドを打つ手間が省け、また間違いを防ぐことができます。

演習用のフォルダへ移動します。

```
> cd MS_summer_school2021/unix_training
```

練習 1 : WinSCPで、MS_summer_school2021/unix_training/script1へ移動し、aa.shをダブルクリックすると、エディターが開きます。

#!/bin/bash	←	bash スクリプトのおまじない
echo "before touch"	←	"before touch"を画面に出力する
ls	←	最初の状態をls
touch bb	←	空のファイルbbを生成
echo "after touch"	←	"after touch"を画面に出力する
ls	←	最後の状態をls

PuTTYのターミナルを開き、同じディレクトリへ移動し、以下のコマンドを実行します。

<pre>> cd MS_summer_school2021/unix_training/script1</pre>	
<pre>> ./aa.sh</pre>	
before touch	
aa.sh	← 最初のlsによる表示。
after touch	
aa.sh bb	← 2回目のlsによる表示。bbが生成している。

練習2：自分でスクリプトを作ってみよう。

WinSCPで、新規→ファイルを選び、ファイル名をcc.shと入力すると、エディターが開きます。

```
#!/bin/bash  
  
pwd
```

← 1行目におまじないを書き、その下にコマンドを書く。

左上の保存ボタンを押すと、cc.shがサーバに作成されます。PuTTYのターミナルで同じディレクトリーへ移動し、以下のコマンドを実行してみよう。

```
> chmod +x cc.sh  
> ./cc.sh  
/home1/uxx/MS_summer_school2021/unix_tutorial/script1
```

← cc.shに実行属性を与える。

← pwdコマンドと同じ結果が得られる。

練習 3 : Gaussian16を実行してみよう。

WinSCPで、script2ディレクトリーへ移動すると、rhodamine1.comとrun.shがあります。rhodamine1.comはGaussian16のインプットで、run.shは実行スクリプトです。ダブルクリックすることで、中身を見ることができます。rhodamine1.comの内容は後ほど説明します。run.shは以下のようになっています。

```
#!/bin/bash

module load gaussian

export GAUSS_SCRDIR=/scr/$USER/job.$$

mkdir -p $GAUSS_SCRDIR

MOL=rhodamine1

(time g16 < ${MOL}.com) > ${MOL}.out 2>&1

formchk ${MOL}.chk

rm -r $GAUSS_SCRDIR
```

- ← bash スクリプトのおまじない
- ← gaussianに必要な設定をロード（このサーバ特有のコマンドです）
- ← 環境変数の設定。Gaussianが一時ファイルを書き出すスクラッチディレクトリの場所を設定。
- ← “-p”オプションで、ディレクトリ階層を一気に作る。
- ← MOLに値を代入。以下では、\${MOL} はrhodamine1を指す。
- ← Gaussian16の実行。rhodamine1.comから入力し、rhodamine1.outが出力になる。
- ← formchkはGaussian16のユーティリティプログラム。checkpointファイル（rhodamine1.chk）から、formatted checkpointファイルを生成。
- ← スクラッチディレクトリを削除。

Gaussian16を実行するには、PuTTYターミナルで以下のようにします。

```
> cd script2
> ./run.sh &
```

← “&”をつけるとジョブがバックグラウンドで流れます。

“&”をつけ忘れると、ジョブが終わるまで（約15分）、コマンドプロンプトが返ってきません。対処法は、

```
> ./run.sh
> bg
```

← Ctrl+zでジョブが一時停止

← バックグラウンドでジョブが再開

あるいは、ジョブを切っても良ければ、

```
> ./run.sh
```

← Ctrl+cでジョブが切れる

となります。ジョブの状態は、jobsや、jobstコマンドで確認することができます。

```
> jobs
[1]+  Running    ./run.sh &
```

← [1]がジョブID

```
> jobst
```

```
-----
03:25:56 up 12:27,  3 users,  load average: 15.84, 13.17, 6.94
-----
```

```
      PID    MEM(KB) CPU%    ELAPSED      TIME    COMMAND
[ peach01 ]
uxx      24439  13171036  1582      08:59   02:22:09  1502.exe
```

← 24439がプロセスID

流れているジョブを切りたいとき。

```
> kill %1
```

← ジョブIDで切る。

```
> kill 24439
```

← プロセスIDで切る。

rhodamine1.outはアスキー形式のアウトプットで、ダブルクリックすることで中身を見れます。しかし、非常に分量が多いため、grepコマンドによるキーワードの抜き出しが有効です。

```
> grep "SCF Done" rhodamine1.out
```

```
SCF Done:  E(RB3LYP) = -1106.22100154      A.U. after   16 cycles
SCF Done:  E(RB3LYP) = -1106.23648347      A.U. after   16 cycles
SCF Done:  E(RB3LYP) = -1106.23910627      A.U. after   16 cycles
SCF Done:  E(RB3LYP) = -1106.24376086      A.U. after   16 cycles
SCF Done:  E(RB3LYP) = -1106.24376580      A.U. after   13 cycles
```

← "SCF Done"がヒットする行が出力される。

容量の大きいファイルをダウンロードするとき、ファイルを圧縮・アーカイブすると容量が減ります。

```
zip -r data.zip rhodamine1.*
  adding: rhodamine1.chk (deflated 26%)
  adding: rhodamine1.com (deflated 67%)
  adding: rhodamine1.fchk (deflated 61%)
  adding: rhodamine1.out (deflated 71%)
```

data.zipが生成するので、これをWinSCPでダウンロードします。ダウンロードしたファイルを右クリック→すべて展開、を選ぶことで、解凍されます。