

UNIX 入門・実習内容

実習 1 : MacOSX の UNIX 環境を確認

Dock メニューから「ターミナル」を起動する。

(ターミナルの在処は、アプリケーション/ユーティリティ)

実習 2 : ディレクトリの中身を見る (ls)

ターミナルのウィンドウをクリックする。

ホームディレクトリ上で **ls** と入力してリターンキーを押す (=実行する)。

続いて下記もそれぞれ実行する。

ls ..	ホームディレクトリの一つ上のディレクトリの中身を見る
ls /	ルートディレクトリの中身を見る

実習 3 : 隠しファイル (ドットファイル) の表示 (ls -a)

ホームディレクトリ上で **ls -a** を実行する

実習 4 : ファイル名の補完

ホームディレクトリ上で **ls da** と入力してタブキーを押してファイル名の補完を試してみる。

同様に、

```
ls data/HN/sprot/143 と入力してタブキーを押して動作を確認する。
ls data/H <tab>
ls data/HN/s <tab>
ls data/HN/sprot/143 <tab>
```

実習 5 : コマンド履歴

Control + p (Control キーと p を同時に押す) を何度か入力してみる。

また、Control + n を何度か入力してみる。

リターンを押すとコマンドが再実行される。

実習 6 : ディレクトリを移動する (cd)

現在のディレクトリ上で **pwd** と入力し、現在のディレクトリを確認する。

ディレクトリ data/HN に移動して **pwd** を入力する。また、**ls** を実行する。

```
pwd
cd data/HN
pwd
ls
```

さらに **sprot** に移動し、**pwd** で現在のディレクトリを確認する。

```
cd sprot
```

pwd

(/Users/nibb/data/HN/sprot と表示される)

実習 7 : ワイルドカード

カレントディレクトリは /Users/nibb/data/HN/sprot

このディレクトリ上で、下記コマンドを実行してみる。

```
ls *.fasta
ls *_HUMAN*
ls 1A2?_HUMAN.fasta
ls 1A2[1-5]*.fasta
ls 1A25_HUMAN.{fasta,phylip}
```

また、**ls *** も試してみる。

実習 8 : ファイルの内容を一括表示する (cat)

カレントディレクトリは /Users/nibb/data/HN/sprot

このディレクトリ上で、下記のコマンドを実行してみる。

```
cat 1A25_HUMAN.fasta
cat *.fasta
```

実習 9 : ファイルの部分表示 (head, tail)

カレントディレクトリは /Users/nibb/data/HN/sprot

このディレクトリ上で、下記コマンドを実行してみる。

```
head 1A25_HUMAN.sprot
tail -20 1A25_HUMAN.sprot
cat 1A25_HUMAN.fasta
tail -n +2 1A25_HUMAN.fasta
```

実習 10 : ファイルの内容を見る (less)

カレントディレクトリは /Users/nibb/data/HN/sprot

このディレクトリ上にある 1433B_HUMAN.sprot の内容を less で見る。

```
less 1433B_HUMAN.sprot
```

文字列検索も行ってみる (例えば binding を検索 (less 内で **/binding** と入力))。

q を押して less を終了。

参考 : less の詳しい操作法は、less を立ち上げた状態で「h」 と打つと確認できる。

実習 11 : ディレクトリの作成と削除 (mkdir, rmdir)

cd でホームディレクトリに戻って、実習用のディレクトリ unixtest を作成

```
cd
```

```
mkdir unixtest
```

作成した `unixtest` ディレクトリに移動し、`pwd` で `/Users/nibb/unixtest` と表示されることを確認する。

```
cd unixtest
```

```
pwd
```

実習 1 2 : ファイルのコピー (cp)

カレントディレクトリは `/Users/nibb/unixtest`

`~/data/HN/sprot/1433B_HUMAN.sprot` をカレントディレクトリ (`.`) にコピーする。

```
cp ~/data/HN/sprot/1433B_HUMAN.sprot .
```

できた `1433B_HUMAN.sprot` を `copyfile` にコピーする (ファイルからファイルへのコピー)

```
cp 1433B_HUMAN.sprot copyfile
```

新たなディレクトリ `Fasta` を作成し、`~/data/HN/sprot/`以下の `.fasta` ファイルだけを `Fasta` ディレクトリにコピーする。その際、ワイルドカードを使う。

```
mkdir Fasta
```

```
cp ~/data/HN/sprot/*.fasta Fasta
```

実習 1 3 : ファイル名の変更 (移動) (mv)

カレントディレクトリは `/Users/nibb/unixtest`

`copyfile` を `newfile` に名称変更する。

```
mv copyfile newfile
```

新たにディレクトリ `Human` をつくり、`Fasta` 以下の `HUMAN` のファイルだけを `Human` ディレクトリ に移動する。その際、ワイルドカードを使う。

```
mkdir Human
```

```
mv Fasta/*HUMAN* Human
```

参考 : 実際にワイルドカードを使う際は、事前に `ls` で対応するファイルが正しく指定されていることを確認した方がよい。

実習 1 4 : シンボリックリンク (別名) の作成 (ln -s)

カレントディレクトリは `/Users/nibb/unixtest`

さきほど移動させたファイル `Human/1433B_HUMAN.fasta` のシンボリックリンクをカレントディレクトリ (`.`) に作成する。

```
ln -s Human/1433B_HUMAN.fasta .
```

なお、`mv` や `cp` と同様に、最後の引数がディレクトリの場合は、そのディレクトリ上にオリジナルと同じ名前のシンボリックリンクが作られる。

参考 : `-s` オプションをつけない場合は、ハードリンクと呼ばれ、オリジナルファイルを移動・消去してもリンクを通した参照が維持されるようになる。ただし、ディレクトリのハードリンクは作れないなどの制約がある。

実習 15 : ファイル情報の表示 (ls -l)

カレントディレクトリは /Users/nibb/unixtest
~/unixtest 上で以下を実行してみる

```
ls -l
ls -lt
```

実習 16 : ファイルの削除 (rm)

カレントディレクトリは /Users/nibb/unixtest
newfile を消去する。次に Fasta ディレクトリ全体を消去する。rm -rf を使う

```
rm newfile
rm -rf Fasta
```

実習 17 : マニュアルの参照 (man)

```
man ls
```

 を実行してみる。

実習 18 : 行数・単語数のカウント (wc)

カレントディレクトリは /Users/nibb/unixtest
以下のコマンドを実行してみる。

```
wc 1433B_HUMAN.fasta
wc
This is a pen.
(Control-D)
```

実習 19 : パターン検索 (grep)

カレントディレクトリは /Users/nibb/unixtest
以下のコマンドを実行してみる

```
grep GO 1433B_HUMAN.sprot
grep ^FT 1433B_HUMAN.sprot
```

実習 20 : リダイレクトの例 (出力)

カレントディレクトリは /Users/nibb/unixtest
以下のコマンドを実行して GO_count ファイルを作成し、中身を less で確認する。

```
grep GO 1433B_HUMAN.sprot
grep GO 1433B_HUMAN.sprot > GO_count
less GO_count
```

実習 21 : パイプの例

カレントディレクトリは /Users/nibb/unixtest

下記のコマンドを実行してみる。

```
grep GO 1433B_HUMAN.sprot | wc
grep ^FT 1433B_HUMAN.sprot | less
grep ^FT 1433B_HUMAN.sprot | grep HELIX | less
```

実習 2 2 : シェルスクリプト

カレントディレクトリは /Users/nibb/unixtest

~/data/HN/sprot/testpg をカレントディレクトリ (.) にコピーし中身を確認する

```
cp ~/data/HN/sprot/testpg .
less testpg
```

実行してみる。

```
./testpg
```

実行権限がなくエラーとなるため、実行権を付与する

```
chmod +x testpg
./testpg
```

また、最後に ./testpg でなく、単に testpg (パス指定なしで実行) とした場合の出力も確認しておく。

```
testpg
```

実習 2 3 : コマンドパス

```
echo $PATH
```

 を実行してコマンドパスを確認する。

実習 2 4 : コマンドパスの設定

カレントディレクトリは /Users/nibb/unixtest

~/.bash_profile の内容を閲覧し、コマンドパスに ~/bin が追加されていることを確認する。

```
less ~/.bash_profile
```

ホームディレクトリ配下に bin ディレクトリを作成し、testpg コマンドを bin ディレクトリへ移動する。最後にパスの指定なしで testpg を再度実行してみる。

```
cd
mkdir bin
mv unixtest/testpg bin
testpg
```

実習 2 5 : SAMtools のインストール

Safari を起動し、samtools で検索

SF Download Page -> samtools -> 1.2 へ移動

samtools-1.2.tar.gz2 をクリックしてダウンロード。

~/Downloads/に保存されるので、~/Downloads から ~/unixtest にファイルを移動してから展開する。

ソースディレクトリで make を実行した後、実行コマンド samtools, bcftools, vcfutils.pl を ~/bin/ にコピーする。

```
cd
mv Downloads/samtools-1.2.tar.gz2 unixtest
cd unixtest
tar xvfj samtools-1.2.tar.bz2
cd samtools-1.2
less INSTALL
make prefix=~ install
samtools
export MANPATH=$MANPATH:~/share/man
man samtools
```

備考：「INSTALL」というファイルを less で読み、インストール方法を確認している。

慣例的に「prefix=」で指定したディレクトリ直下の「bin」ディレクトリにインストールされるので、ここでは「~/bin」にインストールするために「prefix=~」を指定している。

samtools のように make 時に prefix を指定できるものは少なく、configure 時に prefix を指定することの方が圧倒的に多い。（INSTALL や、README というファイルが同梱されているので、手順を確認してからインストール作業をすると良い）。

最後から 2 番目の行は、同時に samtools のマニュアルも「~/share/man」にインストールされるので、man コマンドで見られるようにするための設定。最後の行で samtools のマニュアルを確認している。