

誰にでも使えるメール講座

第1回

「Mailは&、mailxは?」

安岡孝一

yasuoka : root さん、root さん。
 root : 何だい?
 yasuoka : mbox の中にいくつかメールが入っちゃったんですけど、これって sed とか
 を使わないと、取り出せないんでしょうか?
 root : どういうことだい?
 yasuoka : えっと

```
~% cat mbox (ぼこ)
From ochi Thu Mar 22 21:18:24 1990
Received: by ginkaku.kudpc.kyoto-u.ac.jp (1.1/ku2.22)
       id AA00287; Thu, 22 Mar 90 21:18:23 JST
Date: Thu, 22 Mar 90 21:18:23 JST
From: ochi
Message-Id: <9003221218.AA00287@kudpc.kyoto-u.ac.jp>
To: yasuoka
Status: RO
```

Grasshopper no recipe tte wakaru?

ochi

```
From takahash Fri Mar 23 11:39:04 1990
Received: by ginkaku.kudpc.kyoto-u.ac.jp (1.1/ku2.22)
       id AA01310; Fri, 23 Mar 90 11:39:03 JST
Date: Fri, 23 Mar 90 11:39:03 JST
From: takahash
Message-Id: <9003230239.AA01310@kudpc.kyoto-u.ac.jp>
To: yasuoka, matukawa
```

Status: RO

Asatte no concert, machiawase doushimasu?

takahashi

```
From matukawa Fri Mar 23 11:44:14 1990
Received: by ginkaku.kudpc.kyoto-u.ac.jp (1.1/ku2.22)
       id AA01328; Fri, 23 Mar 90 11:44:12 JST
Date: Fri, 23 Mar 90 11:44:12 JST
From: matukawa
Message-Id: <9003230244.AA01328@kudpc.kyoto-u.ac.jp>
To: takahash, yasuoka
Status: RO
```

Asatte no concert, machiawase doushimasu?

Umeda ga iinaa.

Matsukawa

~% █

Mail を終了する時に、x を押すつもりが間違っ q を押しちゃったんで、
 全部 mbox に入っちゃったんです。これって、いちばん上のメールだけ取り
 出すには、head か何かしかないんでしょうか?

root : いや、Mail -f mbox を実行してごらん。

メールを読み書きするためのコマンドは、BSD では Mail、System V では mailx
 である。これらはプロンプトの違い (Mail は &、mailx は ?) などを除いては、
 ほぼ同等のコマンドである。

root : System V なら mailx -f mbox だけだね。
 yasuoka : Mail -f mbox ですか?
 % Mail -f mbox (ぼこ)
 Mail Version 9/4/88 Type ? for help.
 "mbox": 3 messages

```
> 1 ochi Thu Mar 22 21:18 13/296
  2 takahash Fri Mar 23 11:39 13/327
  3 matukawa Fri Mar 23 11:44 14/344
```

& ■

あ、これって、普通に Mail を実行した時と同じですか？

root : そう。Mail も mailx も -f を付けることで、読み出すファイルを指定できるんだよ。

yasuoka : そうなんですか。すると

& p1 (ぼこ)

Message 1:

```
From ochi Thu Mar 22 21:18:24 1990
Received: by ginkaku.kudpc.kyoto-u.ac.jp (1.1/ku2.22)
       id AA00287; Thu, 22 Mar 90 21:18:23 JST
Date: Thu, 22 Mar 90 21:18:23 JST
From: ochi
Message-Id: <9003221218.AA00287@kudpc.kyoto-u.ac.jp>
To: yasuoka
Status: RD
```

Grasshopper no recipe tte wakaru?

ochi

& ■

うん、思った通りだ。

root : p+ とすると、どんどん先に進めるよ。

yasuoka : え？

& p+ (ぼこ)

Message 2:

```
From takahash Fri Mar 23 11:39:04 1990
Received: by ginkaku.kudpc.kyoto-u.ac.jp (1.1/ku2.22)
       id AA01310; Fri, 23 Mar 90 11:39:03 JST
Date: Fri, 23 Mar 90 11:39:03 JST
From: takahash
```

```
Message-Id: <9003230239.AA01310@kudpc.kyoto-u.ac.jp>
To: yasuoka, matukawa
Status: RD
```

Asatte no concert, machiawase doushimasu?

takahashi

& ■

p番号
指定されたメッセージを出力する。番号が省略された場合は、. が指定されたものとみなす。
番号として使えるものは以下の通り。

数	「数」番目のメッセージ
.	最近アクセスしたメッセージ
+	最近アクセスしたメッセージの1つ次
-	最近アクセスしたメッセージの1つ前
^	最初のメッセージ
\$	最後のメッセージ
番号-番号	左「番号」から右「番号」までのメッセージ
*	^- \$ に同じ

yasuoka : うーむ、知らなかった。

& p+ (ぼこ)

Message 3:

```
From matukawa Fri Mar 23 11:44:14 1990
Received: by ginkaku.kudpc.kyoto-u.ac.jp (1.1/ku2.22)
       id AA01328; Fri, 23 Mar 90 11:44:12 JST
Date: Fri, 23 Mar 90 11:44:12 JST
From: matukawa
Message-Id: <9003230244.AA01328@kudpc.kyoto-u.ac.jp>
To: takahash, yasuoka
Status: RD
```

Asatte no concert, machiawase doushimasu?

```

Umeda ga iinaa.
                                Matsukawa

```

```

& p+ (ぼこ)
Referencing beyond EOF
& █

```

root : 知らなかったって、いつもはどうしてたんだい？
yasuoka : hで見ながら、数を指定してました。

```

& h (ぼこ)
  1 ochi                Thu Mar 22 21:18   13/296
  2 takahash           Fri Mar 23 11:39   13/327
>  3 matukawa         Fri Mar 23 11:44   14/344
& █

```

```

h
メッセージの一覧を、画面におさまる範囲で出力する。
z+
画面におさまる1つ次の範囲のメッセージの一覧を出力する。
z-
画面におさまる1つ前の範囲のメッセージの一覧を出力する。

```

yasuoka : この>で示されてるのが、最近アクセスしたメッセージですか？
root : そう。=でも表示されるけどね。

```

=
最近アクセスしたメッセージの番号を出力する。

```

yasuoka : =ですね。

```

& = (ぼこ)
3
& █

```

これってdとかにも適用されるんですか？

```

d番号
指定されたメッセージを削除する。番号についてはpに同じ。番号が省略された場合は、.が指定されたものとみなす。
u番号
削除したメッセージを復活する。番号についてはpに同じ。番号が省略された場合は、最近アクセスしたメッセージから+方向に削除されたメッセージをさがして、それを復活する(なければさらに-方向もさがす)。

```

root : もちろんだよ。ためしにdしてhしてuしてごらん。
yasuoka : はい。

```

& d (ぼこ)
& h (ぼこ)
  1 ochi                Thu Mar 22 21:18   13/296
>  2 takahash           Fri Mar 23 11:39   13/327
& u (ぼこ)
& h (ぼこ)
  1 ochi                Thu Mar 22 21:18   13/296
  2 takahash           Fri Mar 23 11:39   13/327
>  3 matukawa         Fri Mar 23 11:44   14/344
& █

```

もちろんdしてもxすれば削除されませんよね。

```

q
メールコマンドを終了する。削除されたメッセージは、この瞬間に実際に削除される。
x
メールコマンドを終了する。アクセス中だったファイルは変更せず、したがってメールの削除はおこなわれない。

```

root : 大丈夫だから、やってみてごらん。
yasuoka : はい。

```

& d2 (ぼこ)
& h (ぼこ)
  1 ochi                Thu Mar 22 21:18   13/296
>  3 matukawa         Fri Mar 23 11:44   14/344
& x (ぼこ)
~% █

```

大丈夫かな。

root : あ、-fの後のファイル名を省略すると、~/mboxが指定されたとみなされるよ。

yasuoka : え、そうなんですか。じゃあ、Mail -fだけでいいんですね。

```
~% Mail -f (ぼこ)
Mail Version 9/4/88 Type ? for help.
"/home/yasuoka/mbox": 3 messages
> 1 ochi Thu Mar 22 21:18 13/296
  2 takahash Fri Mar 23 11:39 13/327
  3 matukawa Fri Mar 23 11:44 14/344
& ■
```

ほんとだ。-fの後は、どんなファイルでも指定できるんですか？

root : いや、メールの入ったファイルだけだ。mboxとか、sで書き込んだファイルとか。

s番号 ファイル名
指定されたメッセージをファイルに書き込む。番号についてはpに同じ。番号が省略された場合は、.が指定されたものとみなす。ファイル名の1文字目が+の場合には、メールフォルダの中のファイルを表す。

yasuoka : すると

```
& q (ぼこ)
~% cd Mail (ぼこ)
~/Mail% cd Ochi (ぼこ)
~/Mail/Ochi% ls (ぼこ)
from1 from2 from5 from8 to10 to4 to7
from10 from3 from6 from9 to2 to5 to8
from11 from4 from7 to1 to3 to6 to9
~/Mail/Ochi% Mail -f from1 (ぼこ)
Mail Version 9/4/88 Type ? for help.
"from1": 1 message 1 new
> 1 ochi Sat Feb 24 13:11 11/269
& p (ぼこ)
Message 1:
From ochi Sat Feb 24 13:11:12 1990
```

```
Received: by ginkaku.kudpc.kyoto-u.ac.jp (1.1/ku2.22)
      id AA00626; Sat, 24 Feb 90 13:11:10 JST
Date: Sat, 24 Feb 90 13:11:10 JST
From: ochi
Message-Id: <9002240411.AA00626@kudpc.kyoto-u.ac.jp>
To: yasuoka
Status: R
```

Konban Nomimasenka?
ochi

& ■

これは読めるんですね。

root : わざわざqしなくても、メールコマンドにもcdはあるのに。

cd ディレクトリ名
カレントディレクトリを指定されたディレクトリに変更する。ディレクトリ名が省略された場合は、ホームディレクトリが指定されたものとみなす。またディレクトリ名の1文字目が+の場合には、メールフォルダの中のディレクトリを表す。

yasuoka : ええ？

```
& cd (ぼこ)
& pwd (ぼこ)
Unknown command: "pwd"
& ■
```

でもpwdできませんけど？

root : メールコマンドの中では!pwdだよ。

!文字列
文字列をUnixコマンドとみなして実行する。文字列は複数書いてもよい。

yasuoka : !をpwdの前に書くんですか？

```
& !pwd (ぼこ)
~
```

```

!
& ■
何か~と!とが出てますけど。
root : ~はpwdの出力。!は「!文字列」の実行後に必ず出るんだよ。!whoとか
      してみてください。
yasuoka : はい。

& !who (ぼこ)
yasuoka console Mar 23 15:02
takahash ttyp0 Mar 23 10:36 (kinkaku:0.0)
!
& ■

うん、確かに。

& cd Mail/Ochi (ぼこ)
& !pwd (ぼこ)
~/Mail/Ochi
!
& ■

あれ、でもpwdにCシェルのaliasが効いてるな。

& !alias (ぼこ)
cd !! ; set prompt="`pwd`% "
chdir !! ; set prompt="`pwd`% "
his history
popd !! ; set prompt="`pwd`% "
pushd !! ; set prompt="`pwd`% "
pwd !! | sed "s/^\~/home\yasuoka/~/"
rm rm -i
whoami echo You are YASUOKA.
!
& ■

ふーん、こういうことか。ディレクトリを変更できるのはわかりましたけど、
アクセスするファイルの変更は?
root : fiだよ。

```

```

fi ファイル名
   アクセスするファイルを指定されたファイルに変更する。
   特殊なファイル名として、以下のものが指定できる。
   %           郵便箱ファイル
   &           ~/mbox
   +ファイル名 メールフォルダの中のファイル

```

```

yasuoka : この郵便箱ファイルって?
root : 普通にMailを実行した時に、見に行くファイル。
yasuoka : 他人からのメールが入ってるアレですか?
root : そう。
yasuoka : ふーん。

& fi % (ぼこ)
No mail for yasuoka
& ■

あ、ない。

& fi & (ぼこ)
"from1" complete
"/home/yasuoka/mbox": 3 messages
>  1 ochi           Thu Mar 22 21:18   13/296
   2 takahash       Fri Mar 23 11:39   13/327
   3 matukawa       Fri Mar 23 11:44   14/344
& ■

ところで、rootさん。
root : 何だい?
yasuoka : さっきからところどころで出てくる、メールフォルダって何ですか?
root : うーん、ようするに、sとかfiとかでファイル名を指定する時に、手を抜くための方法なんだよ。でも今日はもう時間がないから、それはまた次回にしてくれるかい?
yasuoka : はい、わかりました。じゃ、次回はメールフォルダですね。

```