
Subject: interpretazione errata variabile lunga
Posted by [Lutrin](#) on Wed, 18 May 2011 19:03:07 GMT
[View Forum Message](#) <> [Reply to Message](#)

in uno script per graphicsmagick, voglio salvare le immagini (scansioni a doppia facciata) con la stessa qualità che hanno gli originali

per far ciò, ho pensato di creare la variabile *quality*

```
quality="gm identify -verbose `ls *.jpg | head -n1` | grep Quality | cut -d: -f2"
```

che legge (con identify) il fattore di compressione jpeg dalla prima immagine della serie (`ls *.jpg | head -n1`), poi individua la riga con Quality e infine isola il valore (cut -d: -f2)

ma testandola con un echo

```
quality="gm identify -verbose `ls *.jpg | head -n1` | grep Quality | cut -d: -f2" ; echo $quality
```

ottengo

```
gm identify -verbose img-000.jpg | grep Quality | cut -d: -f2
```

invece del valore che ho, se uso l'insieme di comandi senza inserirlo in una variabile

```
# gm identify -verbose `ls *.jpg | head -n1` | grep Quality | cut -d: -f2  
68
```

quindi è evidente che sbagli qualcosa

--

Puppy Linux wiki: <http://dokupuppylinux.tk>

Puppy Linux Forum: <http://www.italianpuppy.org/>

Windows me genuit, Ubuntu rapuere / tenet nunc Puppy Linux...

Subject: Re: interpretazione errata variabile lunga
Posted by [Marco](#) on Wed, 18 May 2011 19:06:31 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Wed, 18 May 2011 19:03:07 GMT
Lutrin <elicona@olympo.it> wrote:

> in uno script per graphicsmagick, voglio salvare le immagini (scansioni a
> doppia facciata) con la stessa qualità che hanno gli originali

>

> per far ciò, ho pensato di creare la variabile *quality*

>

> quality="gm identify -verbose `ls *.jpg | head -n1` | grep Quality | cut
> -d: -f2"
>
> che legge (con identify) il fattore di compressione jpeg dalla prima
> immagine della serie (`ls *.jpg | head -n1`), poi individua la riga con
> Quality e infine isola il valore (cut -d: -f2)
>
> ma testandola con un echo
>
> quality="gm identify -verbose `ls *.jpg | head -n1` | grep Quality | cut
> -d: -f2" ; echo \$quality
>
> ottengo
>
> gm identify -verbose img-000.jpg | grep Quality | cut -d: -f2
>
> invece del valore che ho, se uso l'insieme di comandi senza inserirlo
> in una variabile
>
> # gm identify -verbose `ls *.jpg | head -n1` | grep Quality | cut -d: -f2
> 68
>
> quindi Ã" evidente che sbaglia qualcosa

```
quality=$(gm identify -verbose $(ls *.jpg | head -n1) | grep Quality | cut  
-d: -f2)
```

(tutto sulla stessa linea)

sorvolo sul resto, tanto sicuramente arrivera' qualcun altro a fartelo notare.

--

Cio' che si vede, e'.

Subject: Re: interpretazione errata variabile lunga
Posted by [Lutrin](#) on Wed, 18 May 2011 19:32:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Wed, 18 May 2011 20:06:31 +0100, Marco ci disse:

> quality=\$(gm identify -verbose \$(ls *.jpg | head -n1) | grep Quality |
> cut -d: -f2)
>
> (tutto sulla stessa linea)
>
> sorvolo sul resto, tanto sicuramente arrivera' qualcun altro a fartelo

> notare.

[...]

riconosco la mia insipienza (grande) e cercherÃ² di fare meglio in futuro

--

Puppy Linux wiki: <http://dokupuppylinux.tk>

Puppy Linux Forum: <http://www.italianpuppy.org/>

Windows me genuit, Ubuntu rapuere / tenet nunc Puppy Linux...

Subject: Re: interpretazione errata variabile lunga

Posted by [mallin.shetland](#) on Thu, 19 May 2011 11:46:07 GMT

[View Forum Message](#) <> [Reply to Message](#)

AddÃ¬ mercoledìÃ¬ 18 maggio 2011 21:03 Lutrin scrisse:

> per far ciÃ², ho pensato di creare la variabile *quality*

> ...

> quindi Ã² evidente che sbagli qualcosa

Non postare la tua soluzione. Dici cosa vuoi fare.

AddÃ¬ mercoledìÃ¬ 18 maggio 2011 21:06 Marco scrisse:

> sorvolo sul resto, tanto sicuramente arrivera' qualcun altro a fartelo

> notare.

Infatti Ã² arrivato chi farÃ notare il resto. PerÃ² l'unico errore Ã² postare un pezzo di codice senza dire cosa vuoi fare in realtÃ .. Ad esempio.

```
quality=$(gm identify -verbose $(ls *.jpg | head -n1) |  
grep Quality | cut -d: -f2)
```

che sarebbe meglio implementare come:

```
quality=$(gm identify -verbose "$(find . -maxdepth 1 -name '*.jpg' -print0 |  
sort -z |  
awk 'BEGIN {RS="\0"}; {print ; exit}' ) " |  
grep Quality |  
cut -d: -f2-)
```

Cosa dovrebbe fare? Esiste un altro strumento per ottenere lo stesso risultato? Il pezzo di codice:

```
ls *.jpg | head -n1
```

Che scopo ha? Cos'altro dovrebbe selezionare il primo file secondo l'ordinamento alfanumerico (se non avesse un bug); a cosa ti serve farlo? Vuoi fare veramente questo?

E cos'altro via.

Subject: Re: interpretazione errata variabile lunga
Posted by [Fulvio](#) on Thu, 19 May 2011 12:26:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Lutrin wrote:

```
> `ls *.jpg | head -n1`
```

Deprecated

```
$(ls *.jpg | head -n1) #e' la moda d' oggi
```

--

Archlinux on (uname -a)

F

Subject: Re: interpretazione errata variabile lunga
Posted by [Lutrin](#) on Thu, 19 May 2011 12:44:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Thu, 19 May 2011 13:46:07 +0200, mallin.shetland ci disse:

```
> Cosa dovrebbe fare?
```

```
[...]
```

leggere il fattore di compressione jpeg e creare un nuovo file (tagliando a met' l'immagine a doppia facciata con il resto dello script) con lo stesso fattore di compressione

```
> ls *.jpg | head -n1
```

```
> Che scopo ha? Cos'altro dovrebbe selezionare il primo file secondo
```

```
> l'ordinamento alfanumerico (se non avesse un bug); a cosa ti serve farlo?
```

```
[...]
```

mi occorre per conoscere il fattore di compressione di una serie di immagini che estraggo da un pdf, siccome tutte hanno lo stesso fattore di compressione, basta sapere il primo per poi applicarlo a tutte le altre

questi comandi fanno parte di uno script che:

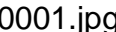
taglia a met' le scansioni jpeg a doppia facciata, crea una cartella in

cui riporle e quindi ce le sposta come pagine singole tagliate dopo avervi posposto le lettere A e B per un corretto ordinamento (A = dispari B = pari)

le coordinate di ritaglio sono calcolate automaticamente dividendo a metà la larghezza della prima immagine (uguale a quella di tutte le altre successive) usando la variabile \$first con l'impiego di identify (lo script Ã per graphicsmagick, ma basta togliere *gm* e funziona anche con imagemagick*)

eccolo qui: (versione su pastebin: <http://pastebin.com/iSL8JGy9>)

```
#!/bin/sh
if [ ! -e even-odd ]; then mkdir even-odd; fi
first="`ls -1 *.jpg | head -n1`"
let "halfwidth=`gm identify -format '%w \n' "$first"/2"
width="`gm identify -format '%w \n' "$first`"
height="`gm identify -format '%h \n' "$first`"
quality="$(gm identify -verbose $(ls *.jpg | head -n1) | grep Quality |
cut -d: -f2)"
for FILE in *.jpg ; do gm convert -quality $quality -crop
"$halfwidth"x"$height"+0+0 "$FILE" "$FILE-A.jpg" ; mv `ls *.jpg | grep A`
even-odd ; gm convert -quality $quality -crop
"$width"x"$height"+"$halfwidth"+0 "$FILE" "$FILE-B.jpg" && mv `ls *.jpg |
grep B` even-odd
done ;
```

ho ancora qualche problema poichÃ© data una immagine a doppia facciata il risultato Ã

img-0001.jpg-A.jpg
img-0001.jpg-B.jpg

e non correttamente:

img-0001-A.jpg
img-0001-B.jpg

qui c'Ã una immagine scansionata a doppia facciata per eventuali test:
<http://ifile.it/reja8yz>

--

Puppy Linux wiki: <http://dokupuppylinux.tk>

Puppy Linux Forum: <http://www.italianpuppy.org/>

Windows me genuit, Ubuntu rapuere / tenet nunc Puppy Linux...

Subject: Re: interpretazione errata variabile lunga

AddÃ¬ giovedì 19 maggio 2011 14:44 Lutrin scrisse:

> mi occorre per conoscere il fattore di compressione di una serie di
> immagini che estraggo da un pdf, siccome tutte hanno lo stesso fattore di
> compressione, ...

AltolÃ¬ ! stai facendo una supposizione pericolosa. Comunque se
vuoi proprio fidarti:

```
for un_file_a_caso in *.jpg ; do break; done
```

```
echo "$un_file_a_caso"
```

Ricorsa: usare l'output di ``ls" in uno script Ã¬ MALE!

```
> #!/bin/sh
```

```
> if [ ! -e even-odd ]; then mkdir even-odd; fi
```

Usare una directory fissa Ã¬ pericoloso; dovresti usare una
directory casuale o passare la directory come parametro

```
> first="`ls -1 *.jpg | head -n1`"
```

Usando il mio suggerimento:

```
for first in "$@" ; do break; done
```

e passi i file come parametri oppure:

```
filenames='*.jpg'
```

```
for first in "$filenames" ; do break; done
```

```
> let "halfwidth=`gm identify -format '%w \n' "$first"/2"
```

```
> width="`gm identify -format '%w \n' "$first`"
```

```
> height="`gm identify -format '%h \n' "$first`"
```

```
> quality="$(gm identify -verbose $(ls *.jpg | head -n1) | grep Quality |
```

```
> cut -d: -f2)"
```

PerchÃ© non hai usato il nome del file salvato?

```
quality="$(gm identify -verbose "$first" | grep Quality | cut -d: -f2)"
```

al piÃ¹ dovrai mettere un escape alle virgolette.

```
> for FILE in *.jpg ; do gm convert -quality $quality -crop  
> ...  
> done
```

Di nuovo se passi i file come argomento:

```
for FILE in "$@" ....
```

altrimenti:

```
for FILE in $namefiles
```

```
> ho ancora qualche problema poichÃ© data una immagine a doppia facciata  
> img-0001.jpg il risultato Ã¨:  
>  
> img-0001.jpg-A.jpg  
> img-0001.jpg-B.jpg
```

dovresti dare una occhiata a basename o, se stai usando la bash a:
\${parameter/pattern/string}

```
> qui c'Ã¨ una immagine scansionata a doppia facciata per eventuali test:  
> http://ifile.it/reja8yz
```

Forse il debug dovresti farlo tu
