

Introduzione ai metacaratteri

Nel linguaggio bash esistono alcuni **caratteri speciali** detti *metacaratteri*.

Essi sono caratteri che svolgono funzioni specifiche o specificano come un tipo di dato specificato deve essere trattato.

Quando qualcosa viene inserito nell'input della *shell*, essa deve effettuare un lavoro di **interpretazione**; è possibile che la *shell* debba:

- Capire se i comandi inseriti sono eseguibili o meno
- Capire dove iniziano e finiscono una o più stringhe o comandi
- Sostituire alcune parti del comando con altre più lunghe oppure evitare che questa cosa succeda

Il primo passo dell'interpretazione di una riga di comando da parte della *shell* sta nel riconoscimento dei *metacaratteri*, effettuando dapprima le operazioni di **espansione**.

Un esempio dell'utilizzo dei metacaratteri può essere visto tramite il comando `echo`: questo comando stampa a video tutti i caratteri scritti dopo la parola `echo` fino al **primo carattere di andata a capo**.

```
echo Hello World!
```

```
# Output: Hello World!
```

Supponiamo che vogliamo stampare il carattere `;` - questo carattere è un *metacarattere* (in particolare un **separatore di comandi**). Affinché questo carattere non venga trattato come metacarattere, esso va inserito all'interno di una stringa che viene circondata da `"`, ossia il *metacarattere di quoting*.

```
echo "Hello ; World ; !"
```

```
# Output: Hello ; World ; !
```

Utilizzo ed espansione delle variabili

La *shell* permette l'uso di **variabili**, ciascuna dotata di un *nome* e di un *valore*. Il nome è una stringa di caratteri che definisce **univocamente** una variabile (anche minuscole e maiuscole rendono due variabili differenti: `CIAO != CiAO`); il valore è anch'essa una sequenza di caratteri alfanumerici; se il valore è formato solo da numeri viene comunque trattato come una stringa di caratteri.

Le variabili possono essere di due tipi:

- Locali, se dichiarate durante l'esecuzione e che vengono perse alla chiusura del terminale in cui vengono dichiarate
- Di ambiente, variabili che vengono impostate all'avvio del sistema operativo non appena la *shell* viene messa in esecuzione per la prima volta.

Le variabili possono essere utilizzate quando vengono impartiti ordini alla *shell*: in particolare la shell **riconosce i nomi delle variabili**, se dichiarate, e cambia il contenuto dell'ordine **sostituendo al nome della variabile il suo valore associato**. Affinché questa operazione sia possibile è necessario specificare alla *shell* quando una specifica stringa è una variabile: si utilizza pertanto l'expansion `${nomevar}` (le parentesi graffe possono essere omesse se dopo all'expansion sono presenti caratteri bianchi).

Vediamo un utilizzo base delle variabili tramite il comando `echo`:

```
NUM=Ciao # Viene definita la variabile NUM, di valore Ciao.  
  
echo ${NUM}  
# Output: Ciao  
echo ${NUM}X  
# Output: CiaoX  
echo $NUM  
# Output: Ciao  
echo $NUMX  
# Output: Non viene visualizzato nulla: in questo caso la shell sta cercando  
la variabile di nome NUMX, che non è stata però dichiarata, dunque non ha  
valore.  
echo $NUM X  
# Output: Ciao X
```

Si noti come, per definire una variabile **non sono permessi spazi nè prima nè dopo il simbolo `=`**:

- `VAR=val` è un'espressione corretta: viene assegnato a `VAR` il valore `val`.
- `VAR =val` è un'espressione che la *shell* interpreta come: eseguo il comando `VAR` avendo come argomenti `=val`.
- `VAR = val` ha lo stesso identico comportamento visto sopra.
- `VAR= val` è un'espressione che la *shell* interpreta come: viene eseguito il comando `val` in un nuovo ambiente di esecuzione (vedremo più avanti), in cui viene assegnato a `VAR` il valore `(uno spazio bianco)`.

Esiste una variabile *d'ambiente* speciale chiamata **PATH**; questa contiene una sequenza di percorsi assoluti separati da `:`, all'interno dei quali sono contenuti gli eseguibili che la *shell* può eseguire specificandone semplicemente il nome.