

Unix / Linux - Opérateurs de base Shell

Il existe différents opérateurs pris en charge par chaque coque. Nous discuterons en détail de la coque Bourne (coque par défaut) dans ce chapitre.

Nous allons maintenant discuter des opérateurs suivants –

- Opérateurs arithmétiques
- Opérateurs relationnels
- Opérateurs booléens
- Opérateurs de cordes
- Opérateurs de test de fichiers

La coque Bourne ne disposait à l'origine d'aucun mécanisme pour effectuer des opérations arithmétiques simples mais elle utilise des programmes externes, soit **awk** ou **expr**.

L'exemple suivant montre comment ajouter deux nombres –

[Démonstration en direct](#)

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

```
val=`expr 2 + 2`  
echo "Total value : $val"
```

Le script ci-dessus générera le résultat suivant –

```
Total value : 4
```

Les points suivants doivent être pris en compte lors de l'ajout de –

- Il doit y avoir des espaces entre opérateurs et expressions. Par exemple, `2 + 2` n'est pas correct; il doit être écrit comme `2 + 2`.
- L'expression complète doit être jointe entre ‘ ‘, appelé le backtick.

Opérateurs arithmétiques

Les opérateurs arithmétiques suivants sont soutenus par Bourne Shell.

Supposons la variable **a** détient 10 et variable **b** détient 20 puis –

Afficher des exemples

Opérateur	La description	Exemple
+ (Ajout)	Ajoute des valeurs de chaque côté de l'opérateur	<code>`expr \$ a + \$ b`</code> donnera 30
- (Soustraction)	Soustrait l'opérande de la main droite de l'opérande de la main gauche	<code>`expr \$ a - \$ b`</code> donnera -10
* (Multiplication)	Multiplie les valeurs de chaque côté de l'opérateur	<code>`expr \$ a \ * \$ b`</code> donnera 200
/ (Division)	Divise l'opérande de la main gauche par l'opérande de la main droite	<code>`expr \$ b / \$ a`</code> donnera 2
% (Module)	Divise l'opérande de la main gauche par l'opérande de la main droite et retourne le reste	<code>`expr \$ b% \$ a`</code> donnera 0
= (Attribution)	Attribue l'opérande droit dans l'opérande gauche	<code>un = \$ b</code> attribuerait une valeur de b dans un
= = (Qualité)	Compare deux nombres, si les deux sont identiques, renvoie true.	<code>[\$ a = = \$ b]</code> reviendrait faux.
!= (Pas d'égalité)	Compare deux nombres, si les deux sont différents, renvoie true.	<code>[\$ a != \$ b]</code> reviendrait vrai.

Il est très important de comprendre que toutes les expressions conditionnelles doivent être à l'intérieur d'accolades carrées avec des espaces autour d'eux, par exemple `[$ a = = $ b]` est correct alors que, `[$ a = = $ b]` est incorrect.

Tous les calculs arithmétiques sont effectués à l'aide d'entiers longs.

Opérateurs relationnels

Bourne Shell prend en charge les opérateurs relationnels suivants, spécifiques aux valeurs numériques. Ces opérateurs ne fonctionnent pas pour les valeurs de chaîne, sauf si leur valeur est numérique.

Par exemple, les opérateurs suivants s'efforceront de vérifier une relation entre 10 et 20 ainsi qu'entre "10" et "20" mais pas entre "dix" et "vingt".

Supposons la variable **un** détient 10 et variable **b** détient 20 puis –

Afficher des exemples

Opérateur	La description	Exemple
-eq	Vérifie si la valeur de deux opérandes est égale ou non; si oui, alors la condition devient vraie.	[\$ a -eq \$ b] n'est pas vrai.
-ne	Vérifie si la valeur de deux opérandes est égale ou non; si les valeurs ne sont pas égales, la condition devient vraie.	[\$ a -ne \$ b] est vrai.
-gt	Vérifie si la valeur de l'opérande gauche est supérieure à la valeur de l'opérande droit; si oui, alors la condition devient vraie.	[\$ a -gt \$ b] n'est pas vrai.
-lt	Vérifie si la valeur de l'opérande gauche est inférieure à la valeur de l'opérande droit; si oui, alors la condition devient vraie.	[\$ a -lt \$ b] est vrai.
-ge	Vérifie si la valeur de l'opérande gauche est supérieure ou égale à la valeur de l'opérande droit; si oui, alors la condition devient vraie.	[\$ a -ge \$ b] n'est pas vrai.
-le	Vérifie si la valeur de l'opérande gauche est inférieure ou égale à la valeur de l'opérande droit; si oui, alors la condition devient vraie.	[\$ a -le \$ b] est vrai.

Il est très important de comprendre que toutes les expressions conditionnelles doivent être placées à l'intérieur de bretelles carrées avec des espaces autour d'elles. Par exemple, [\$ a < = \$ b] est correct alors que, [\$ a < = \$ b] est incorrect.

Opérateurs booléens

Les opérateurs booléens suivants sont soutenus par le Bourne Shell.

Supposons la variable **a** détient 10 et variable **b** détient 20 puis –

Afficher des exemples

Opérateur	La description	Exemple
!	C'est une négation logique. Cela inverse une véritable condition en fausse et vice versa.	[! faux] est vrai.
-o	C'est logique OU . Si l'un des opérandes est vrai, la condition devient vraie.	[\$ a -lt 20 -o \$ b -gt 100] est vrai.
-une	C'est logique ET . Si les deux opérandes sont vrais, alors la condition devient vraie sinon fausse.	[\$ a -lt 20 -a \$ b -gt 100] est faux.

Opérateurs de cordes

Les opérateurs de chaînes suivants sont pris en charge par Bourne Shell.

Supposons la variable **un** détient "abc" et variable **b** détient "efg" puis –

Afficher des exemples

Opérateur	La description	Exemple
=	Vérifie si la valeur de deux opérandes est égale ou non; si oui, alors la condition devient vraie.	[\$ a = \$ b] n'est pas vrai.
!=	Vérifie si la valeur de deux opérandes est égale ou non; si les valeurs ne sont pas égales, la condition devient vraie.	[\$ a != \$ b] est vrai.
-z	Vérifie si la taille d'opérande de chaîne donnée est nulle; s'il est de longueur nulle, il renvoie vrai.	[-z \$ a] n'est pas vrai.
-n	Vérifie si la taille d'opérande de chaîne donnée n'est pas nulle; s'il ne fait pas zéro, il revient vrai.	[-n \$ a] n'est pas faux.
str	Vérifie si str n'est pas la chaîne vide; s'il est vide, il renvoie faux.	[\$ a] n'est pas faux.

Opérateurs de test de fichiers

Nous avons quelques opérateurs qui peuvent être utilisés pour tester diverses propriétés associées à un fichier Unix.

Supposons une variable **fichier** détient un nom de fichier existant "test" dont la taille est de 100 octets et a **lire**, **écrire** et **exécuter** autorisation sur –

Afficher des exemples

Opérateur	La description	Exemple
-b fichier	Vérifie si le fichier est un fichier spécial de bloc; si oui, alors la condition devient vraie.	[-b \$ le fichier] est faux.
fichier -c	Vérifie si le fichier est un fichier spécial de caractère; si oui, alors la condition devient vraie.	[-c \$ fichier] est faux.
-d fichier	Vérifie si le fichier est un répertoire; si oui, alors la condition devient vraie.	[-d \$ fichier] n'est pas vrai.
-f fichier	Vérifie si le fichier est un fichier ordinaire par opposition à un répertoire ou un fichier spécial; si oui, alors la condition devient vraie.	[-f \$ fichier] est vrai.
fichier -g	Vérifie si le fichier a son ensemble d'ID de groupe (SGID) bit défini; si oui, la condition devient vraie.	[-g \$ fichier] est faux.
fichier -k	Vérifie si le fichier a son jeu de bits collant; si oui, alors la condition devient vraie.	[-k \$ le fichier] est faux.
fichier -p	Vérifie si le fichier est un tuyau nommé; si oui, alors la condition devient vraie.	[-p \$ fichier] est faux.
-t file	Vérifie si le descripteur de fichier est ouvert et associé à un terminal; si oui, alors la condition devient vraie.	[-t \$ fichier] est faux.
-u file	Vérifie si le fichier a son ID utilisateur défini (SUID) bit défini; si oui, la condition devient vraie.	[-u \$ fichier] est faux.
-r fichier	Vérifie si le fichier est lisible; si oui, la condition devient vraie.	[-r \$ fichier] est vrai.
-w fichier	Vérifie si le fichier est accessible en écriture; si oui, alors la condition devient vraie.	[-w \$ fichier] est vrai.
fichier -x	Vérifie si le fichier est exécutable; si oui, alors la condition devient vraie.	[-x \$ fichier] est vrai.
fichier	Vérifie si le fichier a une taille supérieure à 0; si oui, la condition devient vraie.	[-s \$ fichier] est vrai.
-e fichier	Vérifie si le fichier existe; est vrai même si le fichier est un répertoire mais existe.	[-e \$ fichier] est vrai.

Opérateurs de coquillages C

Le lien suivant vous donnera une brève idée des opérateurs de shell C –

Opérateurs de coquillages C

Opérateurs Korn Shell

Le lien suivant vous aide à comprendre les opérateurs Korn Shell –

Opérateurs Korn Shell