

Unix / Linux - Shell Decision Making

Dans ce chapitre, nous comprendrons la prise de décision des coquillages dans Unix. Lors de l'écriture d'un script shell, il peut y avoir une situation dans laquelle vous devez adopter un chemin à partir des deux chemins donnés. Vous devez donc utiliser des déclarations conditionnelles qui permettent à votre programme de prendre des décisions correctes et d'effectuer les bonnes actions.

Unix Shell prend en charge les déclarations conditionnelles qui sont utilisées pour effectuer différentes actions en fonction de différentes conditions. Nous allons maintenant comprendre deux déclarations de décision ici –

- Le **si ... else** déclaration
- Le **cas ... esac** déclaration

Les déclarations if ... else

Si les autres déclarations sont des déclarations de décision utiles qui peuvent être utilisées pour sélectionner une option dans un ensemble d'options donné.

Unix Shell prend en charge les formes suivantes de **si ... else** déclaration –

- **si ... déclaration fi**
- **si ... else ... déclaration fi**
- **si ... elif ... else ... fi statement**

La plupart des déclarations if vérifient les relations à l'aide des opérateurs relationnels discutés dans le chapitre précédent.

L'affaire ... déclaration d'esac

Vous pouvez en utiliser plusieurs **si ... elif** déclarations pour effectuer une branche multidirectionnelle. Cependant, ce n'est pas toujours la meilleure solution, surtout lorsque toutes les branches dépendent de la valeur d'une seule variable.

Supports Unix Shell **cas ... esac** déclaration qui gère exactement cette situation, et elle le fait plus efficacement que répété **si ... elif** déclarations.

Il n'y a qu'une seule forme de **cas ... esac** déclaration qui a été décrite en détail ici –

- **cas ... déclaration d'esac**

Le **cas ... esac** la déclaration dans le shell Unix est très similaire à la **interrupteur ... étui** déclaration que nous avons dans d'autres langages de programmation comme **C** ou **C + +** et **PERL**, etc.