

Commenter

- Objet : Les commentaires
- Niveau requis :
[débutant](#), [avisé](#)
- Commentaires : *Commenter une ligne de configuration.*
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Suivi :
 - Création par [smolski](#) le 04/10/2013
 - Testé par [smolski](#) le 04/10/2013
- Commentaires sur le forum : [Lien vers le forum concernant ce tuto](#) ¹⁾

Introduction

Les fichiers de configuration abondent de titres et d'indications supplémentaires permettant à l'administrateur de comprendre le sens d'une option facultative ou non mises en aparté dans les fichiers qu'il administre.

Pour obtenir cela, on utilise un caractère spécifique selon les contextes devant chacune des lignes qui ne doit pas être interprétée par le système. 😊

Exemple

Dans le contexte d'un [shell bash](#) par exemple le dièse²⁾ : # mis en tête de la ligne à commenter permet au système d'en ignorer le contenu tout en l'affichant textuellement. Exemple ici :

- [Commenter la ligne source d'installation restant d'origine attachée au support.](#)

Également, des titrages sont ainsi rendus possibles en cumulant quelques dièses # (leur nombre dépend uniquement de l'esthétisme fonctionnel qu'ils représentent).

Exemple : D'origine dans le fichier d'une configuration, par exemple dans le fichier `/etc/apache2/apache2.conf` ici :

```
##  
## Server-Pool Size Regulation (MPM specific)  
##  
  
# prefork MPM
```

Où le titrage écrit avec des dièses doubles est une mise en valeur de ce qui suit.
L'option commentée # `prefork MPM` est indiquée mais inactive avec le dièse placé devant.

Ou en titrage élaboré dans un script personnel :

```
#####  
## Gestion des appels simultanés
```

```
# Affiche un warning et quitte si le programme tourne, continue sinon
function verif_verrou ()
{
if [ -f "$LOCK" ];      # On regarde si le verrou existe
then
    # Auquel cas, on affiche un message d'erreur et on quitte en renvoyant le
    statut d'erreur : 1 ...
```

Pass'que des fois on s'souvient pas très bien qui fait quoi dans c'qui est fabriqué ! 😊

Ici, seules les lignes sans dièse sont lues et interprétées par le système pour l'exécution de cette fonction.

Ou encore en signature à la fin d'un script issu de la main d'un geek bien poilu mais souriant :

```
# C'est fini Nini !

# vim:set ft=sh tw=0 foldmethod=marker:
```

Et oui, y'en a des comm' ça, y faut croire que la pizza cong'lée ne rend pas morose ! 😊

Nota

Le commentaire indiquant le contexte du shell d'un script, par exemple : **#!/bin/bash** est lu par le système au lancement, mais ignoré par l'interpréteur, c'est l'appel système `exec*` qui interprète le `#!/bin/bash`

Merci de ces renseignements à **enicar** et au **captfnfab** en harmonie tous deux sur le salon irc df ! 😊

Remarque

Tous les fichiers de conf n'utilisent pas le # comme commentaire !

- Par exemple pour le `.Xdefaults` / `.xresources` c'est le point virgule ; qui fait office de commentaire
- pour samba, le point virgule ; et le dièse # sont tous deux utilisés pour commenter
- pour Abiword, c'est du xml, donc les commentaires sont indiqués entre les balises `<!--` et `-->`
- ario *idem*
- awesome, il faut mettre deux tirets -- en début de ligne (c'est du lua)
- brasero, xml aussi

Précisions du **captfnfab** là ! 😊

1)

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

2)

nommé aussi : croisillon

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/doc:programmation:commenter>



Last update: **08/03/2020 09:49**