

# Nano

## Nano : raccourcis clavier

CTRL + A : Permet d'aller au début de la ligne

CTRL + E : Permet d'aller à la fin de la ligne

CTRL + Y : Permet de remonter de page en page

CTRL + V : Permet de descendre de page en page

CTRL + \_ : Permet de se rendre au n° de ligne indiqué

CTRL + C : Permet de savoir à quel n° de ligne / colonne / caractère se trouve votre curseur

CTRL + W : Permet de faire une recherche

CTRL + D : Permet de supprimer un caractère

CTRL + K : Permet de supprimer une ligne complète (comme la commande dd sous Vi)

CTRL + O : Permet de sauvegarder votre fichier

CTRL + X : Permet de quitter nano

CTRL + G : Affiche l'aide

### copier / couper / coller

CTRL + K pour la couper, suivi d'un CTRL + U pour la coller. Maintenant que cette ligne est en mémoire, vous pouvez aller la coller ailleurs.

Plusieurs lignes : faites un CTRL + ^ et déplacez les flèches du clavier pour surligner le texte qui vous intéresse. Ensuite un petit CTRL + K, CTRL + U et vous pourrez recopier ce passage ailleurs avec un autre CTRL + U.

## Coloration syntaxique

```
~/ .nanorc
```

Pour installer une coloration syntaxique (pour le C, Dockerfile, CSS, Git, Go, HTML, JSON, Markdown, nginx, PHP, Python, etc, ), télécharger le fichier xxx.nanorc de votre langage et le placer dans le

dossier ~/.nano/ de votre utilisateur, puis l'inclure dans la configuration de Nano (~/.nanorc) via cette ligne :

```
include "~/.nano/yaml.nanorc"
```

Le git où trouver ces fichiers de coloration syntaxique : [Fichiers de coloration syntaxiques de nano](#)

Script d'installation (télécharge et active toutes les colorations syntaxiques):

```
curl https://raw.githubusercontent.com/scopatz/nanorc/master/install.sh | sh
```

### En images Soulignage Pour du php

```
# Clause de la requete
$sqlClause = ' FROM `'.$db_prefix.'users ORDER BY username';

# Test de la requete
$sql = 'SELECT COUNT(*)' . $sqlClause ;
if ($res = $pdo->query($sql)) {

    # Test du nombre de lignes qui correspond à la requête SELECT COUNT
    if ($res->fetchColumn() > 0) {

        # Des résultats sont retournés , effectue la requête SELECT et travaille sur les résultats
        $sql = 'SELECT * ' . $sqlClause ;
        $nbUsers = 0;
        foreach ($pdo->query($sql) as $row) { # Affichage des résultats
            $nbUsers++;
            # ENVOI DE L'EMAIL
            $mailTo = $row['username'] . ' <' . $row['email'] . '>';
            $subject = 'Sujets hebdomadaires (' . $nbMsg . ') du for
```

Aide Quitter    ^O Écrire    ^W Chercher    ^K Couper    ^J Justifier    ^C Pos. cur.    M-U Annuler  
 Lire fich.    ^R Lire fich.    ^\ Remplacer    ^U Coller    ^T Orthograp.    ^\_ Aller ligne    M-E Refaire

### Pour du bash

```
# TEST DE LA PRESENCE DES 24 IMAGES AVEC BON NOMMAGE
for file in `ls -l $imgStock`
do
    # Analyse du nom du fichier. Séparation nom / extension
    fullfilename=$(basename $file)
    extension=${fullfilename##*.}
    filename=${fullfilename%.*}

    # Test : image 00 pour 00, à 23 pour 23 H...
    if [ "$$cmtImg" -eq "$filename" ] ; then
        echo "Fichier Nom : OK"
    else
        echo "Problème nom de fichier. L'image doit être au format HH.exten
        echo "Exemple : 05.png pour l'image de 5 heures"
        exit
    fi

    # Test de l'extension.
    if [ $extension = $defExtension ] ; then
        echo "Extension OK : $extension / $defExtension"
    else
        echo "Probleme extension."
        echo "Mettre des images au format $defExtension"
```

^G Aide    ^O Écrire    ^W Chercher    ^K Couper    ^J Justifier  
 ^X Quitter    ^R Lire fich.    ^\ Remplacer    ^U Coller    ^T Orthograp.

## Pour du python

```
def lancement_jeu(self, widget):
    """
    Fonction initialisant le jeu
    :param widget:
    :return:
    """
    self.gestion_bouton('on')

    # Chargement fichier par défaut ou du fichier sélectionné ?
    try:
        self.file
    except NameError:
        print("Chargement du fichier de défaut.")
        self.file = 'autre-liste-francais.txt'
    else:
        print("sure, it was defined : ", self.file)

    # Quelques variables
    self.jeu_actif = True           # Si nouvelle partie (booléen)
    self.nb_echecs = 0             # Compte le nombre d'échecs (integer)
    self.image_pendu.set_from_file('/usr/share/pendu-peda/images/' + self.niveau + '/pendu_0.gif') # Image

    # Chargement du fichier et choix aléatoire d'un mot
    fichier = open(str(self.repData) + '/' + str(self.file), "r")
    liste_mots = fichier.readlines()

Aide      ^O Écrire      ^W Chercher    ^R Couper      ^J Justifier   ^C Pos. cur.   M-U Annuler
Quitter   ^R Lire fich.  ^\ Remplacer  ^U Coller     ^T Orthograp. ^_ Aller ligne M-E Refaire
```

From:

<https://cbiot.fr/dokuwiki/> - **Cyrille BIOT**

Permanent link:

<https://cbiot.fr/dokuwiki/php-bash:nano?rev=1583572792>

Last update: **2020/03/07 10:19**

