

LINUX - LES ENTRÉES-SORTIES ACCÈS AUX VARIABLES D'ENVIRONNEMENT



Affecter des valeurs aux variables d'environnement

- On utilise une expression d'affectation pour créer une variable

Exemple :

```
# TAILLE=80
```

- Une variable peut être exportée pour devenir variable d'environnement =>**
visible dans les sous-processus

Exemple :

```
# export TAILLE
```

- La portée d'une variable est locale
Une variable d'environnement est visible dans le processus où elle a été créée et ses sous-processus



Consulter les valeurs des variables d'environnement

- La commande `env` permet de visualiser toutes les variables d'environnement
- Exemple :

```
# env  
TAILLE=80  
PWD=/home/vagrant  
LANG=fr_FR.utf8  
SHLVL=1  
HOME=/home/vagrant  
LOGNAME=vagrant  
#
```



Consulter les valeurs des variables d'environnement

- D'autres commandes sont possibles :
Exemple :

```
# printenv
```

```
# printenv TAILLE
```

```
# echo $TAILLE
```



Consulter les valeurs des variables d'environnement

- A partir d'un programme, on dispose de la fonction `getenv()`

```
# include <stdlib.h>
```

```
char *getenv(char *nom)
```

La fonction **getenv** renvoie un pointeur sur la valeur de la variable d'environnement **nom**

getenv retourne NULL si la variable n'existe pas



Exemple d'utilisation

```
#include <stdio.h>
#include <stdlib.h>

main() {
    char *chaine;
    chaine = getenv("TAILLE");
    if (chaine != NULL) {
        printf("Taille : %s\n",chaine);
    }
}
```



Exemple d'utilisation



Affecter des valeurs aux variables d'environnement

- A partir d'un programme, on dispose de la fonction `setenv()`

```
# include <stdlib.h>
```

```
int setenv(char *nom, char *val, int modif)
```

La fonction **setenv** ajoute la variable **nom** dans l'environnement, en lui affectant la valeur **val**, si **nom** n'existe pas encore

Si **nom** existe déjà, alors sa valeur est modifiée si **modif** est non nul

Si **modif** vaut zéro, la valeur de **nom** n'est pas changée

setenv retourne -1 si elle échoue, 0 si elle réussit



Exemple d'utilisation

```
#include <stdio.h>
#include <stdlib.h>

main() {
    char *chaine;

    setenv("VOLUME", "360 m3", 1);

    chaine = getenv("VOLUME");
    if (chaine != NULL) {
        printf("Volume : %s\n",chaine);
    }
}
```



Exemple d'utilisation

Rappel :

Une variable d'environnement
n'est visible que dans le
processus où elle a été créée



Supprimer une variable d'environnement

- A partir d'un programme, on dispose de la fonction unsetenv()

```
# include <stdlib.h>
```

```
int unsetenv(char *nom)
```

La fonction **unsetenv** détruit la variable **nom** dans l'environnement

unsetenv retourne -1 si elle échoue,
0 si elle réussit