# Exercice sl

Le programme sl affiche des animations visant à corriger les utilisateurs qui saisissent accidentellement sl au lieu de ls. SL est l'abréviation de Steam Locomotive.

**Synopsis** …  
**>> sl [-alFe]**

**Options** …

|  |  |
| --- | --- |
| Option | Description |
| -a | Un accident semble se produire |
| -l | Affichage d’un petit train |
| -F | Affichage d’un train volant |
| -e | Permet d'interrompre par CTRL+C |

## Construction de l’image

On commence par la création d’un Dockerfile afin de construire une nouvelle image …  
**>> cd /var/tmp**  
**>> mkdir --verbose train**  
**>> cd train**  
**>> nano | vim dockerfile-cmd**  
**FROM alpine**  
**RUN apk update && apk add sl  
CMD sl**

On utilise la commande **docker image build** afin de créer la nouvelle image en utilisant le commutateur **--tag** pour la nommer train-sl‑cmd …  
**>> docker image build --file dockerfile-cmd --tag train-sl-cmd .**

On utilise la commande **docker image history** afin d’afficher les différents calques de l'image …  
**>> docker image history train-sl-cmd**

Pour démarrer un nouveau conteneur …  
**>> docker container run \  
 --interactive --tty --rm \  
 train-sl-cmd**

Pour modifier la commande (CMD) pour un nouveau conteneur …  
**>> docker container run \  
 --interactive --tty --rm \  
 train-sl-cmd \  
 sl -F**

Pour modifier la commande (CMD) pour un nouveau conteneur (une seconde fois) …  
**>> docker container run \  
 --interactive --tty --rm \  
 train-sl-cmd \  
 sl -l**

On commence par la création d’un Dockerfile afin de construire une nouvelle image …  
**>> nano | vim dockerfile-entrypoint**  
**FROM alpine**  
**RUN apk update && apk add sl  
ENTRYPOINT sl**

On utilise la commande **docker image build** afin de créer la nouvelle image en utilisant le commutateur **--tag** pour la nommer train‑entrypoint …  
**>> docker image build --file dockerfile-entrypoint --tag train-sl-entrypoint .**

On utilise la commande **docker image history** afin d’afficher les différents calques de l'image …  
**>> docker image history train-sl-entrypoint**

Pour démarrer un nouveau conteneur …  
**>> docker container run \  
 --interactive --tty --rm \  
 train-sl-entrypoint**

Pour modifier la commande (CMD) pour un nouveau conteneur …  
**>> docker container run \  
 --interactive --tty --rm \  
 train-sl-entrypoint \  
 sl -F**

Pour modifier la commande (CMD) pour un nouveau conteneur (une seconde fois) …  
**>> docker container run \  
 --interactive --tty --rm \  
 train-sl-entrypoint \  
 sl -l**

# Exercice cowsay

Le programme cowsay/cowthink permet d’afficher un texte humoristique.

**Synopsis** …  
**>> cowsay [-e <Chaine>] [-f <Fichier>] [-h] [-l] [-n] [-T <Langue>] [-W <Colonne>] [-bdgpstwy]**

**Options** …

|  |  |  |
| --- | --- | --- |
| Option | Description | Exemple |
| -e | Modification des yeux de la vache | **>> cowsay -e @@ <Message>** |
| -f | Modification de l’image | **>> cowsay -f tux <Message> >> cowsay -f eyes <Message>** |
| -l | Affichage de la liste des images | **>> cowsay -l** |
|  | Combinaison de cowsay avec echo | **>> echo "I need water!" |   cowsay -f dragon-and-cow** |
|  | **États d’esprit** **…** Paronaïa (-p) Mort – *dead* (-d) Borg (-b) Branché – *wired* (-w)  … -y -t -g -s | **>> cowsay -p <Message> >> cowsay -d <Message> >> cowsay -b <Message> >> cowsay -w <Message> …** |

Pour afficher la liste des images disponibles pour cowsay …  
**>> cowsay -l**ou  
**>> ls /usr/share/cowsay/cows/**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Images | | | | | | | |
| apt | cower | elephant | gnu | luke-koala | pony-smaller | stimpy | unipony |
| bud-frogs | daemon | elephant-in-snake | hellokitty | mech-and-cow.cow ren | suse | unipony-smaller |  |
| bunny | default. | eyes | kangaroo | milk | sheep | three-eyes | vader |
| calvin | dragon-and-cow | flaming-sheep | kiss | moofasa | skeleton | turkey.cow vader-koala |  |
| cheese | dragon | fox | koala | moose | snowman | turtle | www |
| cock | duck | ghostbusters | kosh | pony | stegosaurus | tux |  |

## Construction de l’image

On commence par la création d’un Dockerfile afin de construire une nouvelle image …  
**>> cd /var/tmp**  
**>> mkdir --verbose cowsay**  
**>> cd cowsay**  
**>> nano | vim dockerfile-cmd**  
**FROM ubuntu:bionic  
RUN apt-get update \  
 && apt-get upgrade -y \  
 && apt-get install -y cowsay \  
 && rm -rf /var/lib/apt/lists/\*  
ENV PATH=$PATH:/usr/games/  
CMD cowsay Bonjour**

On utilise la commande **docker image build** afin de créer la nouvelle image en utilisant le commutateur **--tag** pour la nommer train-sl‑cmd …  
**>> docker image build --file dockerfile-cmd --tag cowsay-cmd.v1 .**

On utilise la commande **docker image history** afin d’afficher les différents calques de l'image …  
**>> docker image history cowsay-cmd.v1**

Pour démarrer un nouveau conteneur …  
**>> docker container run \  
 --interactive --tty --rm \  
 cowsay-cmd.v1**

Pour modifier la commande (CMD) pour un nouveau conteneur …  
**>> docker container run \  
 --interactive --tty --rm \  
 cowsay-cmd.v1 \  
 cowsay Allo**

Pour modifier la commande (CMD) pour un nouveau conteneur (une seconde fois) …  
**>> docker container run \  
 --interactive --tty --rm \  
 cowsay-cmd.v1 \  
 cowsay -b Salut**

Pour modifier la commande (CMD) pour un nouveau conteneur (une troisièeme fois) …  
**>> docker container run \  
 --interactive --tty --rm \  
 cowsay-cmd.v1 \  
 cowsay -f tux Salut**

On commence par la création d’un Dockerfile afin de construire une nouvelle image …  
**>> nano | vim dockerfile-entrypoint**  
**FROM ubuntu:bionic  
RUN apt-get update \  
 && apt-get upgrade -y \  
 && apt-get install -y cowsay \  
 && rm -rf /var/lib/apt/lists/\*  
ENV PATH=$PATH:/usr/games/  
ENTRYPOINT cowsay Bonjour**

On utilise la commande **docker image build** afin de créer la nouvelle image en utilisant le commutateur **--tag** pour la nommer train‑entrypoint …  
**>> docker image build --file dockerfile-entrypoint --tag cowsay-entrypoint.v1 .**

On utilise la commande **docker image history** afin d’afficher les différents calques de l'image …  
**>> docker image history cowsay-entrypoint.v1**

Pour démarrer un nouveau conteneur …  
**>> docker container run \  
 --interactive --tty --rm \  
 cowsay-entrypoint.v1**

Pour modifier la commande (CMD) pour un nouveau conteneur …  
**>> docker container run \  
 --interactive --tty --rm \  
 cowsay-entrypoint.v1 \  
 cowsay Allo**

Pour modifier la commande (CMD) pour un nouveau conteneur (une seconde fois) …  
**>> docker container run \  
 --interactive --tty --rm \  
 cowsay-entrypoint.v1 \  
 cowsay -b Salut**

## Variation

Il est possible de modifier le message de cowasy par une maxime.

On commence par la création d’un Dockerfile afin de construire une nouvelle image …  
**>> nano | vim dockerfile-maxime**  
**FROM ubuntu:bionic  
RUN apt-get update \  
 && apt-get upgrade -y \  
 && apt-get install -y fortune \  
 && apt-get install -y cowsay \  
 && rm -rf /var/lib/apt/lists/\*  
ENV PATH=$PATH:/usr/games/  
CMD fortune | cowsay**

On utilise la commande **docker image build** afin de créer la nouvelle image en utilisant le commutateur **--tag** pour la nommer train‑entrypoint …  
**>> docker image build --file dockerfile-maxime --tag cowsay-maxime.v1 .**

On utilise la commande **docker image history** afin d’afficher les différents calques de l'image …  
**>> docker image history cowsay-maxime.v1**

Pour démarrer un nouveau conteneur …  
**>> docker container run \  
 --interactive --tty --rm \  
 cowsay-maxime.v1**

# Encore plus d’exercices

Il est possible de répéter les mêmes exercices mais avec d’autres programmes.

**figlet**

Pour installer figlet …  
**>> sudo apt install figlet**

Pour plus d’informations sur le programme …  
**>> man figlet**

**cmatrix**

Pour installer cmatrix …  
**>> sudo apt install cmatrix**

Pour plus d’informations sur le programme …  
**>> man cmatrix**

**rig**

Pour installer rig …  
**>> sudo apt install rig**

Pour plus d’informations sur le programme …  
**>> man rig**

**bastet**

Pour installer bastet …  
**>> sudo apt install bastet**

Pour plus d’informations sur le programme …  
**>> man bastet**

# Autres jeux intéressants

* ninvaders
* pacman4console
* nsnake
* greed
* screenfetch neofetch
* noefetch