

Exercices

Consigne : sauf mention contraire, n'utilisez pas d'orchestrateur (par exemple, ni `docker compose`, ni `k8s`).

Évitez les IA, ces exercices sont là pour faire toucher vos neurones. Cherchez dans la documentation (man pages, aide en ligne, Google Search), posez des questions quand vous êtes totalement bloqué.

Il n'est pas important de comprendre les codes d'exemple. Les messages d'erreur doivent suffire à trouver ce qui manque via un moteur de recherche.

Accéder le shell dans un conteneur

Pourquoi cette commande ne semble rien faire ?

```
docker run alpine
```

Sachant que la commande par défaut de ce conteneur est `/bin/sh`, comment faire pour conserver le shell (dans le conteneur) et l'utiliser ?

Pouvons-nous lancer cette image et l'utiliser ainsi depuis Docker Desktop ? Pourquoi ?

Utiliser des commandes dans un conteneur

En utilisant les commandes `find` et `wc`, déterminer le nombre de fichiers présents dans les images de conteneur :

- `alpine:3.19`
- `debian:buster-slim`
- `ubuntu:noble`

Utiliser l'image officielle de PHP et installer des modules complémentaires

Lancez l'image `PHP` et installez-y les composants nécessaires pour que le script suivant puisse s'exécuter sans erreur et produire un GIF rempli.

Sauvegardez ce script dans un fichier `nggyu.php` et exécutez-le dans un conteneur *ad hoc*.

```
<?php
$gif =
'https://media2.giphy.com/media/v1.Y2lkPTc5MGI3NjExbXMyajE2cG5raWI3dGZyZWdjYm91ZmdtbDg0dWwwenFxeDkyNnFyaCZlcD12MV9pbnRlcm5hbF9naWZfYnlfYWQmY3Q9ZW/Vuw9m5wXviFIQ/giphy.gif';
$gif = file_get_contents($gif)
    or die('Nope');
$size = getimagesizefromstring($gif)
    or die('Nope');
[$width, $height] = [$size[0]/2, $size[1]/2];
$gif = imagecreatefromstring($gif)
    or die('Nope');
$half = imagecreatetruecolor($width, $height)
    or die('Nope');
imagecopyresampled($half, $gif, 0, 0, 0, 0, $width, $height,
$size[0], $size[1])
    or die('Nope');
imagegif($half, 'rick.gif')
    or die('Nope');
```

Hint

Lisez la documentation de l'image officielle sur le Hub

Point bonus : afficher la nouvelle image.

Une base de données

En utilisant l'image `mariadb` ou `mysql` officielle :

1. chargez automatiquement, à la création du conteneur, le script SQL, en utilisant le mécanisme fournit par l'image choisie

Indices

- cherchez la signification de ce qu'est un *bind mount*
- lisez la documentation du conteneur pour savoir comment rendre le script disponible au conteneur
- lisez la documentation de l'option `--volume` de `docker run`

2. assurez la persistance des données de la base entre deux instantiations à l'aide d'un volume nommé (named volumes)
3. faites en sorte que la base de données soit créée avec un utilisateur dans la base de données, accessible depuis l'hôte, en utilisant la fonctionnalité de l'image choisie
4. connectez MySQL Workbench (ou le client MySQL de votre choix) à la base de données du conteneur et vérifiez que les tables et données ont bien été ajoutées

Le script à charger :

```
create database westeros;
use westeros;

create table pack
( id integer auto_increment
, name text not null
, primary key (id)
);

insert into pack (name)
values ("Summer"), ("Shaggydog"), ("Ghost"),
      ("Lady"), ("Nymeria") , ("Grey Wind");
```

Faire communiquer les conteneurs

```
mkdir public
```

```
<?php
# public/wolves.php
try
{
    $con = new mysqli
        ( $_ENV['DB_HOST']
        , $_ENV['DB_USER']
        , @$_ENV['DB_PASS'] ?: null
        , $_ENV['DB_NAME']
        , @$_ENV['DB_PORT'] ?: 3306
        );
}
catch(\Exception $e)
{
    header('Content-type: text/plain');
```

```
    echo "Exception:";
    var_dump($e);
    die();
}

$all = $con->query('select * from animaux');
$all and 0 === $all->num_rows or die(':\'(');
echo '<ul>';
foreach($all->fetch_all(MYSQLI_ASSOC) as $wolf)
    echo "<li>{$wolf['name']}</li>";
echo '</ul>';
```

1. créez (ou réutilisez) un conteneur de base de données
2. utilisez la variante Apache de l'image PHP pour publier l'exemple de code