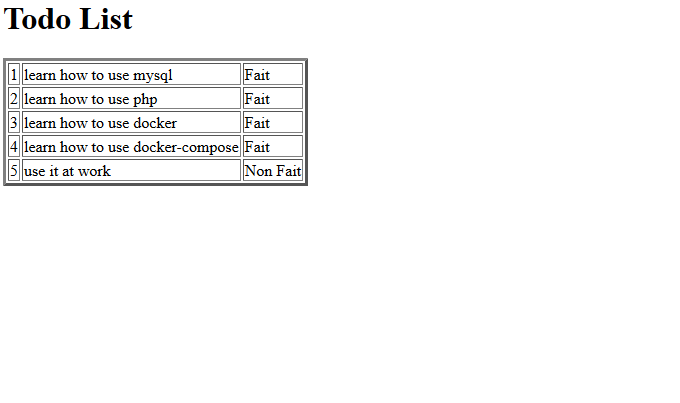
Orchestration d’un environnement Docker

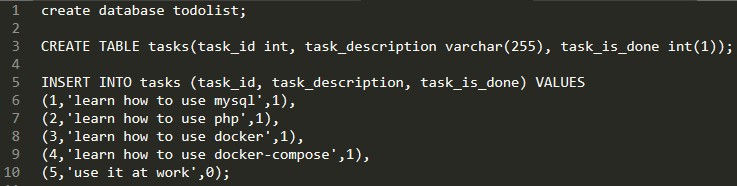
Nous avons décidé de créer un environnement Apache2 + PHP7.2 + MySql en utilisant Docker.

Nous avons donc créé un site Web en local. Ce site web appelé Todolist qui permet de lister des tâches dans un tableau, une tâche possède un ID, une description et un état (fait/pas fait).

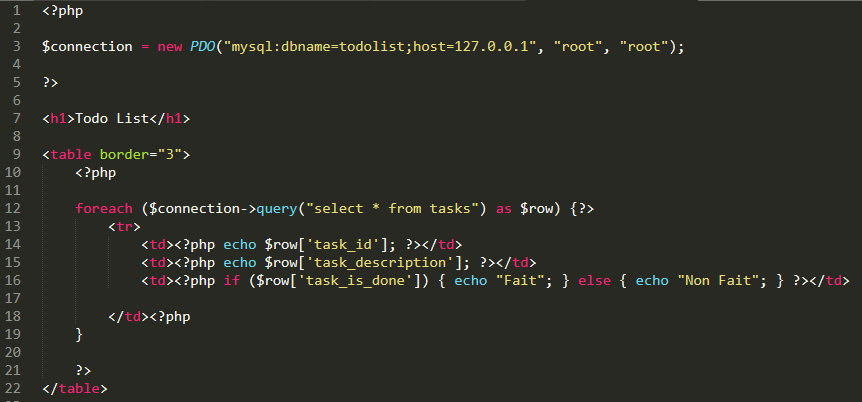
Voici le visuel de notre site :



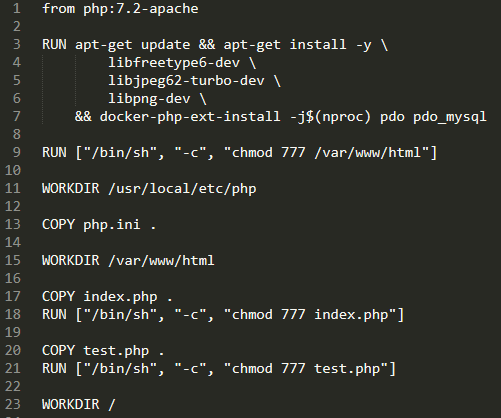
Nous avons créé le script de création de la table :



Et ensuite on a développé le site Web en local. Nous avons utilisé l’extension pdo\_mysql pour accéder à la base de données. Voici le code source ci-dessous :



Pour créer notre image Docker, il nous a fallu écrire le Dockerfile, il s’agit d’un fichier qui permet de créer l’image et de faire des actions dessus :



from php :7-2-apache

Cette commande dit que cette image aura comme base une image avec les composants php7.2 et apache

RUN apt-get update && apt-get install -y \

libfreetype6-dev \

libjpeg62-turbo-dev \

libpng-dev \

&& docker-php-ext-install -j$(nproc) pdo pdo\_mysql

Cette commande fait exécuter sur le container une mise à jour des sources d’installations.

Elle installe ensuite 3 librairies de développement permettant d’installer des extensions php tels que pdo\_mysql. Ensuite nous installons avec la commande docker-php-ext-install les 2 extensions pdo et pdo\_mysql pour accéder à la base de données.

RUN ["/bin/sh", "-c", "chmod 777 /var/www/html"]

On met le dossier /var/www/html en accès libre

WORKDIR /usr/local/etc/php

COPY php.ini .

On se place sur le dossier /usr/local/etc/php et on copie le fichier de configuration php.ini

WORKDIR /var/www/html

COPY index.php .

RUN ["/bin/sh", "-c", "chmod 777 index.php"]

COPY test.php .

RUN ["/bin/sh", "-c", "chmod 777 test.php"]

WORKDIR /

On se place sur le dossier des documents web et on copie le fichier index.php et test.php, le fichier index.php contient la page Todolist et le fichier test.php contient une page phpinfo()

Ces fichiers sont ensuite mis en accès libre.

Installation :

Pour installer cet outil, il faut utiliser une machine avec une distribution Linux car Docker ne supporte pas une version basique de Windows, il vous faut la version Pro de Windows.

Il faut récupérer le Dockerfile et les fichiers sources sur notre dépôt GitHub, grâce à la commande :

git clone https://github.com/FrancoisBasset/Todolist

Ensuite il faut se placer sur le dossier où le dépôt a été cloné et écrire cette commande :

docker build . -t todolist

Puis lancer l’image :

docker run todolist