

Practical work on web architecture

At the end of this practical work, you will :

- Gain more experience with **docker**
- Discover **docker-compose**
- Deploy a **Nginx** server and add sites to it.
- Configure SSL

Question 1

Install a dockerized wordpress using **docker-compose**.

1. Go in a working directory
2. Download wordpress using `curl https://wordpress.org/latest.tar.gz | tar -xvzf -`
3. Edit `wordpress/wp-config.php` to match your needs (see annex 1)
4. Edit a Dockerfile file to build your image according to annex 3.
5. Create a `docker-compose.yml` file (see annex 2)
6. Run `docker-compose up`
7. Access **wordpress** on your port 8000. Finish your instalation.

Annex 1 :

```
<?php
define('DB_NAME', 'wordpress');
define('DB_USER', 'root');
define('DB_PASSWORD', '');
define('DB_HOST', 'db:3306');
define('DB_CHARSET', 'utf8');
define('DB_COLLATE', '');

define('AUTH_KEY',      'put your unique phrase here');
define('SECURE_AUTH_KEY', 'put your unique phrase here');
define('LOGGED_IN_KEY',  'put your unique phrase here');
define('NONCE_KEY',     'put your unique phrase here');
define('AUTH_SALT',     'put your unique phrase here');
define('SECURE_AUTH_SALT', 'put your unique phrase here');
define('LOGGED_IN_SALT', 'put your unique phrase here');
define('NONCE_SALT',    'put your unique phrase here');

$table_prefix = 'wp_';
define('WPLANG', '');
define('WP_DEBUG', false);

if ( !defined('ABSPATH') )
    define('ABSPATH', dirname(__FILE__) . '/');

require_once(ABSPATH . 'wp-settings.php');
?>
```

Annex 2 :

```
version: '2'
services:
  web:
    build: .
    command: php -S 0.0.0.0:8000 -t /code/wordpress/
    ports:
      - "8000:8000"
    depends_on:
```

```
- db
volumes:
- ./code
db:
image: orchardup/mysql
environment:
MYSQL_DATABASE: wordpress
```

Annex 3 :

```
FROM orchardup/php5
ADD ./code
```

(This question is inspired from [this docker compose example \(https://docs.docker.com/compose/wordpress/\)](https://docs.docker.com/compose/wordpress/))

Question 2

Put wordpress behing a NGinx reverse proxy

1. Create a NGinx image based on *monsantoco/min-jessie* image using a Dockerfile. You can have a look to what was done for an Apache server last week.
2. Create a site definition in sites-enabled for your wordpress website (using a domain name, for instance example.com) and add it to your image.
3. Edit your */etc/hosts* file so that your domain name (eg : example.com) points to 127.0.0.1
4. Add NGinx to your docker-compose infrastructure.
 - It should be linked with **web**
 - **web** should not be exposed any more
 - You should build the **nginx** image each time

Question 3

Best practices.

1. NGinx should listen on 443 port and support SSL.
2. Use https://hub.docker.com/_/mysql/ instead of the image specified above, as it is deprecated.
3. MySQL databases files should be located on a volume.

Question 4

Using question 5 of docke/apache practical work.

You might use the [correction \(https://github.com/Open-Up/openup02_06/tree/master/correction/question_5\)](https://github.com/Open-Up/openup02_06/tree/master/correction/question_5).

Add a site using NGinx for your custom Apache page.