

# Проходження практики

Мета - Вивчити всі технології, щоб могли створити свій сайт/блог

## Домашка 1:

### 1. Встановити VirtualBox і Extension Pack

Все можна знайти на офіційному сайті.

Створіть віртуальну машину з назвою ubuntu-server-16.

Достатньо одного ядра і 512 МБ оперативної пам'яті.

Жорсткий диск має бути динамічним на ~30 ГБ.

### 2. Встановити Ubuntu Server 16.04 (32 bit)

Мова англійська, ім'я користувача/пароль ubuntu/ubuntu, локація Україна, таймзона України.

Додатково треба встановити openssh-server (можна вибрати на одному з етапів конфігурації).

```
sudo apt install openssh-server
```

Після завершення встановлення потрібно запустити оновлення і перезавантажитись.

### 3. Налаштувати port-forwarding

Робиться в налаштуваннях віртуальної машини, в пункті налаштування мережі. Форварднути потрібно порт 80 (HTTP) і 22 (SSH) для всіх адрес (далі так буде легше переглядати завдання). Через специфіку Linux/macOS форвардити порти з номерами меншими за 1024 не можна. Потрібно вибрати інші. Типово можна вибрати 8080 для HTTP і 8022 для SSH і перенаправити їх на стандартні на віртуальній машині.

Налаштування:

- для Windows

Хост 80 -> Віртуальна 80

Хост 22 -> Віртуальна 22

- для Linux/macOS

Хост 8080 -> Віртуальна 80

Хост 8022 -> Віртуальна 22

Колонки IP-адреса залишити порожніми.

#### 4. Підключитись до віртуальної машини через SSH

При стандартних налаштуваннях:

Користувач: ubuntu

Пароль: ubuntu

Для Windows використовувати PuTTY.

Для Linux/macOS -- стандартний ssh клієнт.

Приклад (з використанням нестандартного порта):

```
ssh -p 8022 ubuntu@localhost
```

Далі все потрібно робити через SSH клієнт.

На даному етапі можна зробити повний бекап віртуалки, так як ймовірність поламати щось далі доволі висока.

#### 5. Налаштувати спільну папку між хост системою і віртуальною машиною

(хост система - Ваша основна ОС, не віртуалка)

Треба встановити Guest Additions.

- Щоб встановити Guest Additions, у вікні віртуальної машини натиснути  
Пристрої -> Встановити гостьові доповнення
- Встановити додаткові необхідні пакети  

```
sudo apt-get install build-essential linux-headers-$(uname -r)
```
- Змонтувати /dev/cdrom в /media/cdrom  

```
mount /dev/cdrom /media/cdrom
```
- Зайти в /media/cdrom і запустити встановлення  

```
sudo ./VBoxLinuxAdditions.run
```
- Спробувати змонтувати папку командою  

```
mount -t vboxsf Share/ Share/
```

На хост машині десь створити папку Workspace. Це буде наша робоча папка, там будуть всі інші папки і файли. Там створити папку Exercises. Вибрати її, як спільну для нашої віртуалки, галочку автоматичного монтування **не ставити**. На віртуальній машині створити аналогічну папку в домашній директорії (якщо нічого не змінювалось при встановленні, то це буде -- /home/ubuntu/Exercises).

Налаштувати автоматичне монтування при завантаженні віртуалки. Для цього:

- в `/etc/modules` додати `vboxsf`. Це потрібно для того, щоб модуль монтування **VirtualBox** завантажився швидше за `fstab`.
- Додати в `/etc/fstab` стрічку для автоматичного монтування папки:

```
Exercises    /home/ubuntu/Exercises vboxsf defaults    0    0
```

Посилання:

<http://askubuntu.com/questions/252853/how-to-mount-a-virtualbox-shared-folder-at-startup>

## 6. Встановити NGINX, PHP, MySQL і зробити базові налаштування.

Де потрібно, можна використовувати IP адресу локального хоста 127.0.0.1 або localhost.

7. Сконфігурувати nginx згідно [статті](#). (Змінити шлях до публічної папки - root, придумати ім'я сервера - server\_name і т.д.).

Посилання:

<https://www.digitalocean.com/community/tutorials/how-to-install-linux-nginx-mysql-php-lemp-stack-in-ubuntu-16-04>

Додатково:

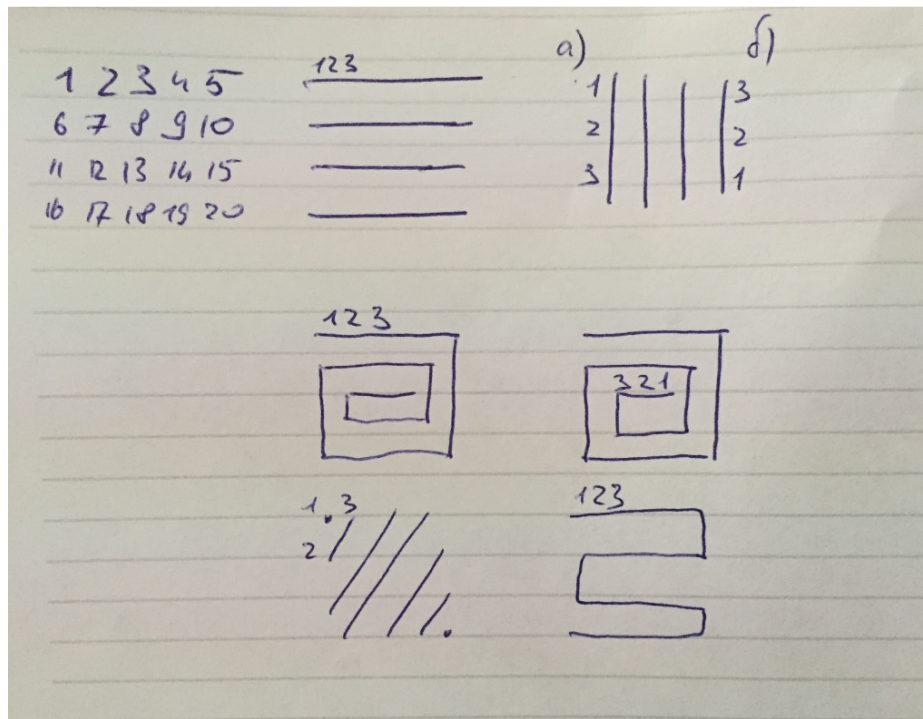
- Console (<https://www.codecademy.com/learn/learn-the-command-line>)

## Домашка 2:

- Стандарт PSR-1, PSR-2, PSR-4 (<http://www.php-fig.org/psr/>)
- PHP (Курс: <https://www.codecademy.com/learn/php>, Читиво: <http://www.phptherightway.com/>)

## Домашка 3:

- Composer (<https://habrahabr.ru/post/145946/>, <https://getcomposer.org/> (документація));
- Посортувати масив (з Composer)



Детальніше:

Реалізувати алгоритми "сортування" масивів, які будуть розставляти елементи у відповідному порядку (див. рисунок).

1 - Елементи розташовуються в горизонтальному порядку.

Приклад:

1 2 3

4 5 6

7 8 9

2. Вертикальний порядок.

1 4 7

2 5 8

3 6 9

3. Равлик.

1 2 3

8 9 4

7 6 5

4. Зворотній равлик.

9 8 7

2 1 6

3 4 5

#### 5. Діагональ.

1 3 6  
2 5 8  
4 7 9

#### 6. Змійка.

1 2 3  
6 5 4  
7 8 9

Під час реалізації мають бути дотримані стандарти PSR-1, PSR-2, PSR-4. Обов'язково використовувати composer autoloader. Кожен алгоритм - це окремий клас, в якому реалізований метод, який розставляє елементи масиву, який подається на вхід, у відповідному порядку в залежності від типу.

## Домашка 4:

- Git (<https://www.codecademy.com/learn/learn-git>, <https://try.github.io/levels/1/challenges/1>)
- SQL (<https://www.codecademy.com/learn/learn-sql>);
- Закинути на Gitlab

Репозиторій на Gitlab:

<https://gitlab.com/andrii.bodnar/practice-january>

Потрібно буде створити запит на приєднання до репозиторію

Кожен створить свою папку і закине туди свій код. Папки називати в форматі NameSurname.

1. Запушити свій код в цей репозиторій у свою папку.
2. Додати записування масивів в **БД**. Структуру таблиці можна придумати. Запушити на Gitlab.

## Домашка 5:

- JS/jQuery (<https://www.codecademy.com/learn/javascript>, <https://www.codecademy.com/learn/jquery>);
- Додати UI (бажано з Bootstrap)

**Додаткова ачівка** тому, хто замість jQuery використає **ReactJS**.

<https://www.codecademy.com/learn/react-101>

<https://www.codecademy.com/learn/react-102>

Дописати простий генератор випадкових чисел. Клацаєш на кнопку, генерується масив. Клацаєш на іншу кнопку, сортує і виводить.

- Перечитати і відформатувати код відповідно до стандартів (PSR-1,2).
- Налаштувати робоче середовище на заміну табуляцій на пробіли.
- Переписати масиви згідно ООП підходу і використати:
  - Абстрактний клас або інтерфейс.
  - Винести сортування кожного масива в окремий клас.
  - Виведення реалізувати в окремому класі.
- Запустити на гітлаб

## Домашка 6:

Переробити все під фреймворк Laravel. <https://laravel.com/docs/5.5/>

- Рознести різні реалізації сортування по класах
- Вивід в консоль/на сторінку і в файл
- Поправити форматування, модифікатори доступу
- Зробити метод вводу даних від користувача (форму) для вводу розмірності масива
- Реалізувати як мінімум два генератора масивів (випадкових чисел/простих чисел/чисел Фібоначчі)
- Запустити на гітлаб

## Домашка 7:

### Теоретичний кусок:

- Базовий курс по всьому  
(<https://www.youtube.com/watch?v=J45c907-GmM&list=PLSdH7dYnIGYgrWg5wsEG4v03MiJ1WSOTx>)
- Детальніший курс з наглядним використанням Composer, namespace & autoload  
(<https://www.youtube.com/watch?v=Bo8M6g5waos&list=PL5QUj3iB7EoJGDMxuzlV58uh0ybgAHlcN>)
- Магічні методи (<http://php.net/manual/en/language.oop5.magic.php>)

### Практичний кусок:

- Попрацювати над структурою проекту (повиносити сутності в окремі файли, файли погрупувати в папках)
- Вивід в окремому класі
- Переписати код з використанням композера, неймспесів і автолоада.