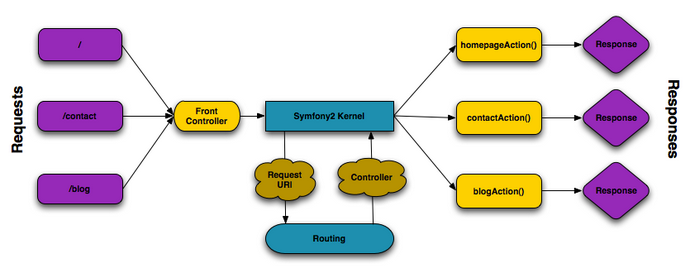
**Symfony framework:**

The book

You can use just some component not the whole framework



The Symfony Application Flow

Controller 🡪 method

Name must use –controller . for example: NoteController

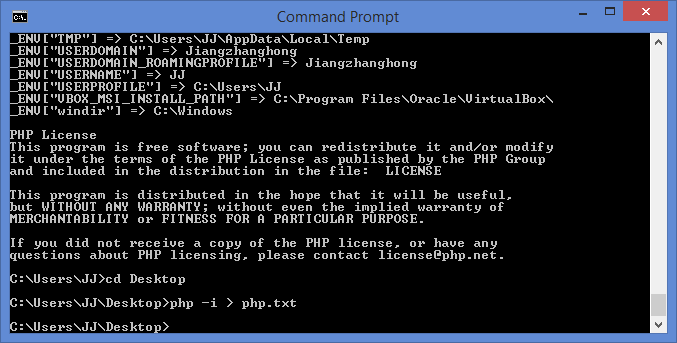
**Installation and configuration:**

A: install php

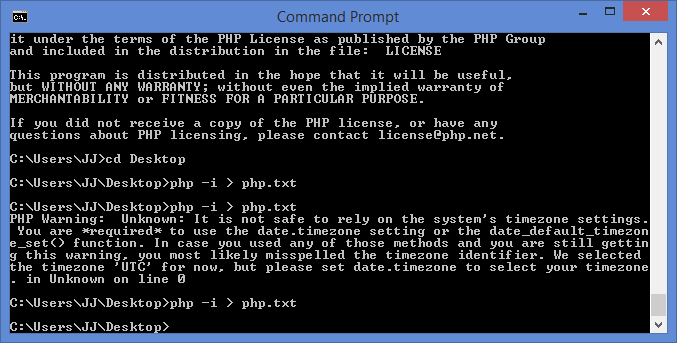
1. Download the php to c:/
2. Open command line (cmd.exe) and go to c drive (cd ..). then, run php –i (can configure)
3. To configure php, go to this pc and choose property 🡪 advance setting 🡪 environment variables, find path and edit, “….; c:/php”
4. Command line: php –i (to prove the php is running)

cd Desktop

php –i > php.txt



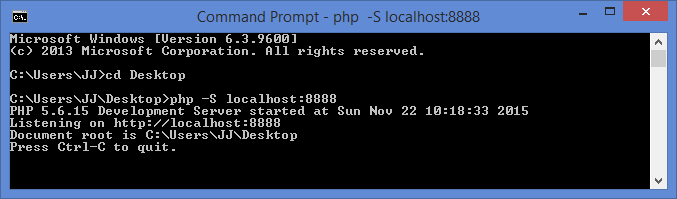
1. Now, Configuration File (php.ini) Path is not correct. Then, copy the file of php.ini-development and change its name to php.ini.
2. Run “php –i > php.txt” again and find the next problem which is about the timezone. The solution is add the timezone in php.ini



B. install SQLite

1. Create a new file (info.php)

2. Run the server: go to command line:



3. open the browser: localhost:8888/info.php and check the PDO status. If not, need to install SQLite

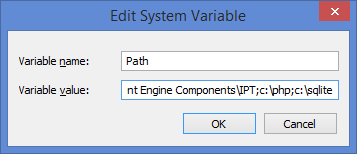
4. install SQLite

a. go to the homepage: [www.sqlite.org](http://www.sqlite.org) and download file

**Precompiled Binaries for Windows:** [sqlite-shell-win32-x86-3090200.zip](https://www.sqlite.org/2015/sqlite-shell-win32-x86-3090200.zip)

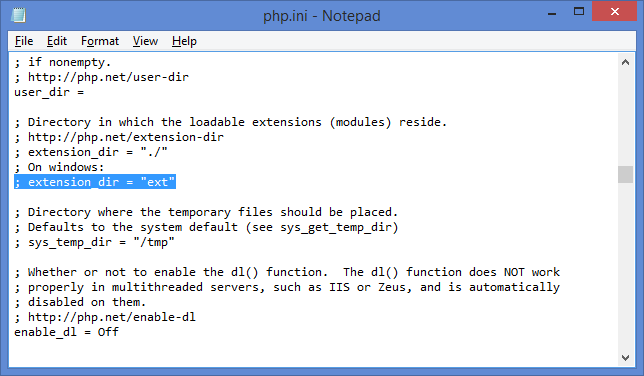
b. move sqlite3 file to c: drive

c. add the path to this file

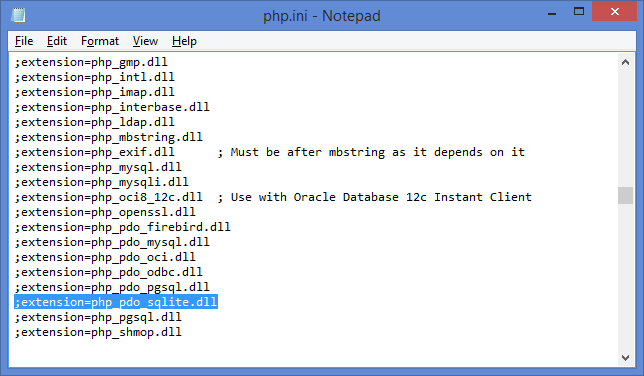


d. go to php.ini and run this sqlite:

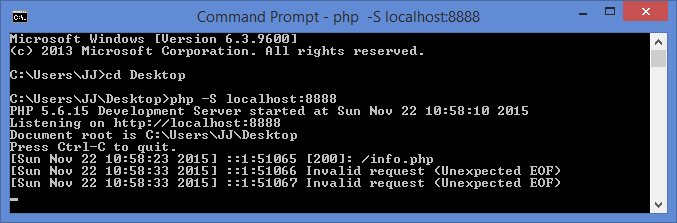
* uncommand “; extension\_dir = "ext"” 🡪(change to) “extension\_dir = "ext"”



Uncommand “;extension=php\_pdo\_sqlite.dll” 🡪(change to) “extension=php\_pdo\_sqlite.dll”



e. check:





C. install Symfony

1. Create a new folder and enter this folder, install the symphony staller

Command line: mkdir final

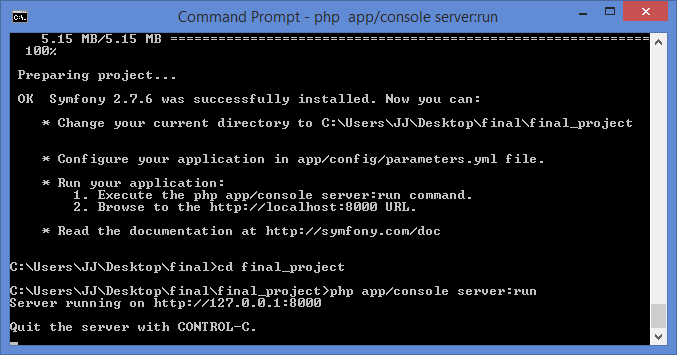
cd final

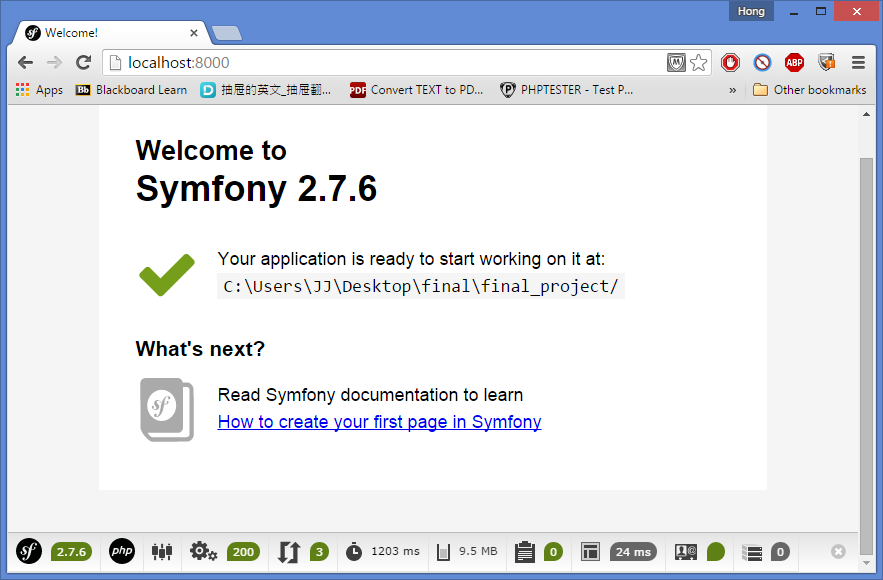
php -r "file\_put\_contents('symfony', file\_get\_contents('http://symfony.com/installer'));"

php sympony new final\_project

D. symphony:

1. Run the server: php app/console server:run





2. Create the first controller: Create your First Page in Symfony.pdf

LuckyController.php:

*// src/AppBundle/Controller/LuckyController.php*

namespace AppBundle\Controller;

use Symfony\Bundle\FrameworkBundle\Controller\Controller;

use Sensio\Bundle\FrameworkExtraBundle\Configuration\Route;

use Symfony\Component\HttpFoundation\Response;

class LuckyController extends Controller

{

*/\*\**

*\* @Route("/lucky/number")*

*\*/*

public function numberAction()

{

$number = rand(0, 100);

return new Response(

'<html><body>Lucky number: '.$number.'</body></html>'

);

}

}

Lucky is controller and number is action

Localhost:8000/lucky/number

Templet: number.html.twig

[Twig](http://symfony.com/doc/current/book/page_creation.html)

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | *{# app/Resources/views/lucky/number.html.twig #}*  {% extends 'base.html.twig' %}  {% block body %}  <h1>Lucky Numbers: {{ luckyNumberList }}</h1>  {% endblock %} |

It's time to explore and demystify the files in your project. You've already worked inside the two most important directories:

app/

Contains things like configuration and templates. Basically, anything that is not PHP code goes here.

src/

Your PHP code lives here.

99% of the time, you'll be working in src/ (PHP files) or app/ (everything else). As you get more advanced, you'll learn what can be done inside each of these.

The app/ directory also holds a few other things, like the cache directory app/cache/, the logs directory app/logs/ and app/AppKernel.php, which you'll use to enable new bundles (and one of avery short list of PHP files in app/).

The src/ directory has just one directory - src/AppBundle - and everything lives inside of it. A bundle is like a "plugin" and you can [find open source bundles](http://knpbundles.com/) and install them into your project. But even your code lives in a bundle - typically AppBundle (though there's nothing special aboutAppBundle). To find out more about bundles and why you might create multiple bundles (hint: sharing code between projects), see the [Bundles](http://symfony.com/doc/current/book/bundles.html) chapter.

So what about the other directories in the project?

vendor/

Vendor (i.e. third-party) libraries and bundles are downloaded here by the [Composer](https://getcomposer.org/) package manager.

web/

This is the document root for the project and contains any publicly accessible files, like CSS, images and the Symfony front controllers that execute the app (app\_dev.php and app.php).