

Node.js 起手式

Webber Wu

註明：

今天提到的部分很基本,有經驗的人可能會覺得「超級無聊」

自我介紹

Webber (吳承翰)

愛吃東西，但現在禁止餵食
程式中迷途的小羊



環境需求

- Mac OSX
- Linux
- Window 請用 Koding



寫 code 用 mac 是一種潮流

對不起 偏離主題了..., 趕快拉回來！！

今日提到的部分

- nvm
- npm
- express generator
- jade
- bootstrap

Node Version Manager

Current version: v0.12.2

 Windows Installer node-v0.12.2-x86.msi	 Macintosh Installer node-v0.12.2.pkg	 Source Code node-v0.12.2.tar.gz
Windows Installer (.msi)	32-bit	64-bit
Windows Binary (.exe)	32-bit	64-bit
Mac OS X Installer (.pkg)	Universal	
Mac OS X Binaries (.tar.gz)	32-bit	64-bit
Linux Binaries (.tar.gz)	32-bit	64-bit
SunOS Binaries (.tar.gz)	32-bit	64-bit
Source Code	node-v0.12.2.tar.gz	

相信剛開始學習的人，應該都看過這個畫面

每次升級 Node 版本，就要重新安裝一次.....

nvm 幫你做好 Node 的版本控管

並且快速切換到不同的版本

安裝 nvm

- curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.25.1/install.sh | bash
- wget -qO- https://raw.githubusercontent.com/creationix/nvm/v0.25.1/install.sh | bash

```
codio@world-script:~/workspace$ curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.25.1/install.sh | bash
% Total    % Received % Xferd  Average Speed   Time     Time      Current
          Dload  Upload Total   Spent    Left  Speed
100  7045  100  7045    0      0  37932      0 --:--:-- --:--:-- --:--:-- 52969
=> nvm is already installed in /home/codio/.nvm, trying to update using git
=> * (detached from v0.25.1)
  master

=> Source string already in /home/codio/.bash_profile
=> Close and reopen your terminal to start using nvm
```

```
codio@world-script:~/workspace$ nvm
```

Node Version Manager

Usage:

nvm help	Show this message
nvm --version	Print out the latest released version of nvm
nvm install [-s] <version>	Download and install a <version>, [-s] from source. Uses .nvmrc if available
nvm uninstall <version>	Uninstall a version
nvm use <version>	Modify PATH to use <version>. Uses .nvmrc if available
nvm run <version> [<args>]	Run <version> with <args> as arguments. Uses .nvmrc if available for <version>
nvm current	Display currently activated version
nvm ls	List installed versions
nvm ls <version>	List versions matching a given description
nvm ls-remote	List remote versions available for install
nvm deactivate	Undo effects of `nvm` on current shell
nvm alias [<pattern>]	Show all aliases beginning with <pattern>
nvm alias <name> <version>	Set an alias named <name> pointing to <version>
nvm unalias <name>	Deletes the alias named <name>
nvm reinstall-packages <version>	Reinstall global `npm` packages contained in <version> to current version
nvm unload	Unload `nvm` from shell
nvm which [<version>]	Display path to installed node version. Uses .nvmrc if available

Example:

nvm install v0.10.32	Install a specific version number
nvm use 0.10	Use the latest available 0.10.x release
nvm run 0.10.32 app.js	Run app.js using node v0.10.32
nvm exec 0.10.32 node app.js	Run `node app.js` with the PATH pointing to node v0.10.32
nvm alias default 0.10.32	Set default node version on a shell

Note:

to remove, delete, or uninstall nvm - just remove ~/.nvm, ~/.npm, and ~/.bower folders

```
codio@world-script:~/workspace$ █
```

在 terminal 下輸入 nvm (ENTER)

```
codio@world-script:~/workspace$ nvm install v0.10
#####
# 100.0%
Now using node v0.10.38 (npm v1.4.28)
codio@world-script:~/workspace$ █
```

nvm install v0.10

會幫你安裝 node v0.10 最後的版本

```
codio@world-script:~/workspace$ nvm alias default stable  
default -> stable (→ v0.10.38)  
codio@world-script:~/workspace$ █
```

nvm alias default stable

設定 node 的 default 版本為 stable

最後讓我們來試試看
node 究竟安裝完成了沒

```
codio@world-script:~/workspace$ node
> console.log("Node.js install!!")
Node.js install!!
undefined
> █
```

輸入 node
確認是否有進入 command 模式
接著輸入 console.log(xxx)
確認裡面的字串有被印出來, 記得加雙引號 !



u got it, babe

Node Package Manager

用來管理 Node 的 Package

有點像是 apt, brew, pip

安裝 Node 套件

```
npm install [-g] [package]
```

npm install -g gongxi

```
$ npm install -g gongxi
npm http GET https://registry.npmjs.org/gongxi
npm http 200 https://registry.npmjs.org/gongxi
npm http GET https://registry.npmjs.org/gongxi/-/gongxi-1.0.0.tgz
npm http 200 https://registry.npmjs.org/gongxi/-/gongxi-1.0.0.tgz
npm http GET https://registry.npmjs.org/commander
npm http GET https://registry.npmjs.org/lodash
npm http 200 https://registry.npmjs.org/commander
npm http 200 https://registry.npmjs.org/lodash
npm http GET https://registry.npmjs.org/lodash/-/lodash-3.8.0.tgz
npm http 200 https://registry.npmjs.org/lodash/-/lodash-3.8.0.tgz
npm http GET https://registry.npmjs.org/graceful-readlink
npm http 304 https://registry.npmjs.org/graceful-readlink
/Users/wb-pc/.nvm/v0.10.26/bin/gongxi -> /Users/wb-pc/.nvm/v0.10.26/lib/node_modules/
gongxi@1.0.0 /Users/wb-pc/.nvm/v0.10.26/lib/node_modules/gongxi
└── commander@2.8.1 (graceful-readlink@1.0.1)
    └── lodash@3.8.0
```

然後輸入 [gongxi faCai]

恩..其實就是讓你們了解Package就是這樣

刪除 Node 套件

npm uninstall [-g] [package]

```
npm uninstall -g gongxi  
unbuild gongxi@1.0.0
```

我們會利用 package.json
紀錄安裝的 Node 套件

如何產生 package.json?

npm init

```
$ npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sane defaults.
```

```
See `npm help json` for definitive documentation on these fields
and exactly what they do.
```

```
Use `npm install <pkg> --save` afterwards to install a package and
save it as a dependency in the package.json file.
```

```
Press ^C at any time to quit.
```

```
name: (ggyy) test
version: (0.0.0) 0.0.0
description: test npm init
entry point: (index.js) index.js
test command:
git repository:
keywords: hello
author: me
license: (ISC)
```

```
About to write to /Users/wb-pc/ggyy/package.json:
```

```
{
  "name": "test",
  "version": "0.0.0",
  "description": "test npm init",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "keywords": [
    "hello"
  ],
  "author": "me",
  "license": "ISC"
}
```

```
Is this ok? (yes) yes
```

接著要安裝 node 套件時
寫入 package.json

```
$ cat package.json
{
  "name": "test",
  "version": "0.0.0",
  "description": "test npm init",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [
    "hello"
  ],
  "author": "me",
  "license": "ISC"
}
```

npm install gongxi
怎麼沒有加進去？

```
$ cat package.json
{
  "name": "test",
  "version": "0.0.0",
  "description": "test npm init",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [
    "hello"
  ],
  "author": "me",
  "license": "ISC",
  "dependencies": {
    "gongxi": "^1.0.0"
  }
}
```

npm install gongxi --save

要記得加上 --save 才會存進去喔！

```
$ npm uninstall gongxi --save
unbuild gongxi@1.0.0
wb-pc@webberdeMacBook-Air ~/ggyy
$ cat package.json
{
  "name": "test",
  "version": "0.0.0",
  "description": "test npm init",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [
    "hello"
  ],
  "author": "me",
  "license": "ISC",
  "dependencies": {}
}
```

npm uninstall gongxi --save

也可以用 --save 將 package 內的紀錄移除喔

YEAH!

YOU GOT IT RIGHT! :D

express generator

快速產生 express 的基本架構

```
npm install express-generator -g
```

express myapp

建立我的第一個express app 叫做 myapp

```
$ express myapp

  create : myapp
  create : myapp/package.json
  create : myapp/app.js
  create : myapp/public
  create : myapp/routes
  create : myapp/routes/index.js
  create : myapp/routes/users.js
  create : myapp/public/javascripts
  create : myapp/views
  create : myapp/views/index.jade
  create : myapp/views/layout.jade
  create : myapp/views/error.jade
  create : myapp/public/images
  create : myapp/public/stylesheets
  create : myapp/public/stylesheets/style.css
  create : myapp/bin
  create : myapp/bin/www

install dependencies:
$ cd myapp && npm install

run the app:
$ DEBUG=myapp:* ./bin/www
```

```
.─ app.js
└── bin
    └── www
─ package.json
─ public
    ├── images
    ├── javascripts
    ├── stylesheets
        └── style.css
─ routes
    ├── index.js
    └── users.js
─ views
    ├── error.jade
    ├── index.jade
    └── layout.jade
```

7 directories, 9 files

express app 的資料結構

連 package 都幫你寫好摟！ 好棒棒

```
cd myapp  
npm install
```

進入資料夾後安裝 package 內的套件

npm start

啟動你的 app

```
↳ $ npm start  
> myapp@0.0.0 start /Users/wb-pc/ggyy/myapp  
> node ./bin/www
```

不要懷疑！這就是已經啟動摟～

在瀏覽器上輸入網址: <http://localhost:3000>



Express

Welcome to Express

我知道常常會有人不知
道伺服器已經啟動了

```
51 app.use(function(err, req, res, next) {  
52   res.status(err.status || 500);  
53   res.render('error', {  
54     message: err.message,  
55     error: {}  
56   });  
57 });  
58  
59 process.env.PORT = 9000;  
60 console.log("[Start] listening, port", process.env.PORT)  
61  
62 module.exports = app;  
63
```

讓我們加入一些 js code 改善它吧 !

打開 app.js 這個檔案拉到最後，並加入以上的 code

存檔後重新啟動

```
process.env.PORT = 9000;  
console.log("[Start] listening, port", process.env.PORT)
```

- 加入上面的 code 後存檔, 接著重新啟動(**ctrl+c** 是中斷)

```
> myapp@0.0.0 start /Users/wb-pc/hackon/myapp
> node ./bin/www

GET / 304 305.667 ms --
GET /stylesheets/style.css 304 3.090 ms --
HEAD / 200 72.169 ms - 1216
GET / 200 45.938 ms - 1216
GET /stylesheets/style.css 304 1.007 ms --
GET / 304 25.208 ms --
GET /stylesheets/style.css 304 0.859 ms --
HEAD / 200 20.632 ms - 1216
GET / 304 28.146 ms - -216
GET /stylesheets/style.css 304 1.048 ms --
HEAD / 200 21.763 ms - 1216
GET / 200 18.582 ms - 1216
^C
wb-pc@webberdeMacBook-Air ~ ~/hackon/myapp <master*>
$ npm start

> myapp@0.0.0 start /Users/wb-pc/hackon/myapp
> node ./bin/www

[Start] listening, port 9000
```

如果改寫成功你會看到我們寫的那行被啟動了
[Start] listening, port 9000

因為我剛剛改了 port, 所以這次要打開的是 localhost:9000 喔
當然 port 只要沒有跟其他的程式衝到都是可以更改唷！



Express

Welcome to Express

YO DAWG

UGOT IT

讓我們來加些東西吧

不過...在此之前先瞭解一下 **jade** 吧

jade

express 預設的 template engine

jade

- Jade 是一套用來撰寫 HTML 樣板的簡潔語言。
- 產生 HTML
- 支援動態程式碼
- 支援可重用性 (DRY)

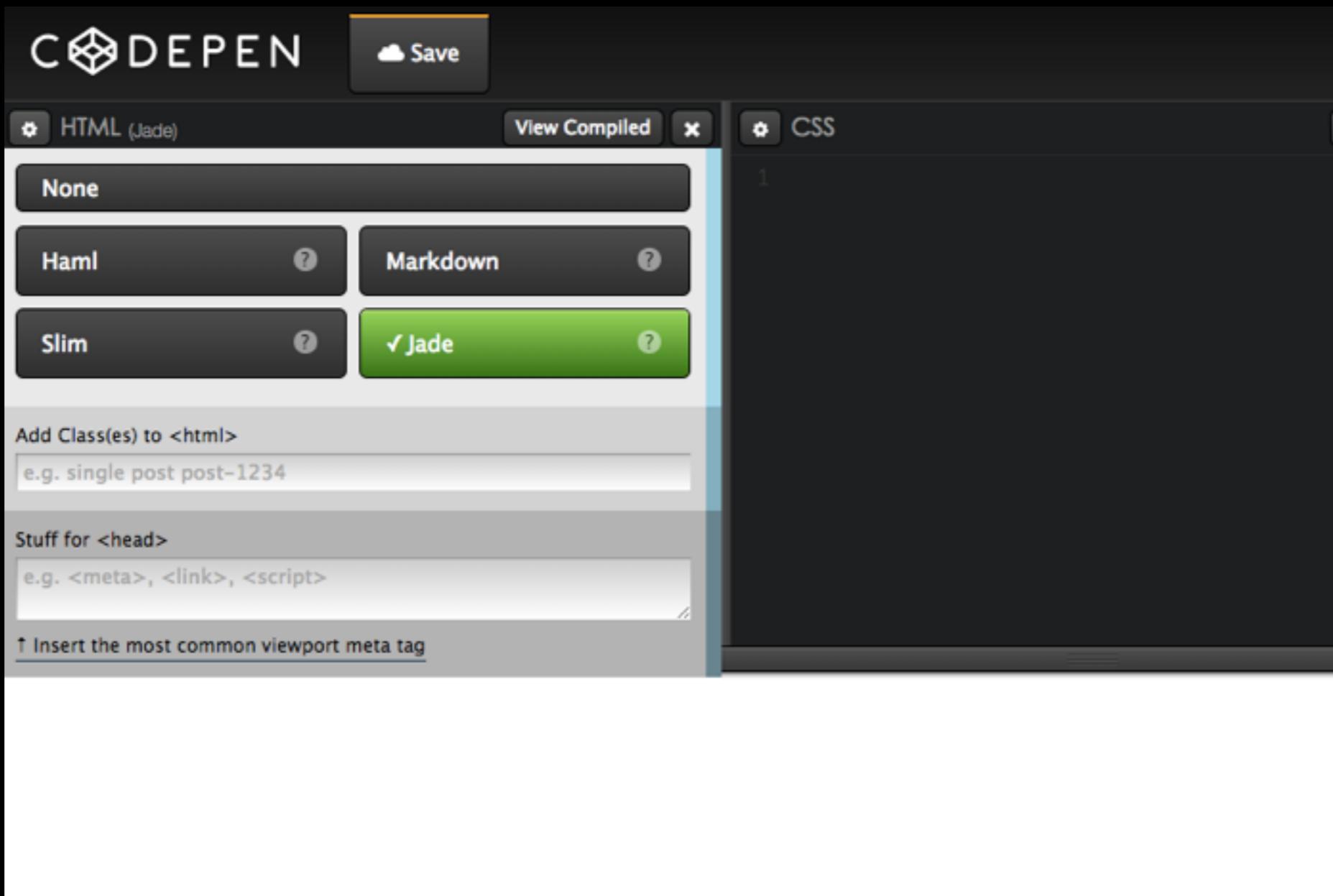
```
1 Jade 語法：  
2  
3 html  
4   head  
5     title= This is Jade Example  
6   body  
7     Hello World  
8  
9 結果：  
10  
11 <html>  
12   <head>  
13     <title>This is Jade Example</title>  
14   </head>  
15   <body>Hello World</body>  
16 </html>
```

Jade 語法 - 用簡單的語法快速生成 HTML

(使用縮排描述巢狀結構, 如同 Python)

讓我們來體驗一下 jade 吧！

codepen: <http://codepen.io/pen/>



點選 HTML 左上角的設定按鈕
並且將 jade 打勾

```
1 //-- Jade
2
3 //-- class 的寫法
4 button.btn-success success
5
6 //-- id的寫法
7 #home.section
8
9 //-- 其他Attributes的寫法
10 img.img-circle(src="http://graph.facebook.com/bird23074035/picture?width=180")
11
```

```
1 <!-- HTML結果 -->
2
3 <!-- class 的寫法 -->
4 <button class="btn-succes">success</button>
5
6 <!-- id的寫法 -->
7 <div id="home" class="section"></div>
8
9 <!-- 其他Attributes的寫法 -->
10 
```

賦予 Element Attributes 的方式：

```
1 <ul>
2   <li>item1</li>
3   <li>item2</li>
4   <li>item3</li>
5   <li>item4</li>
6   <li>item5</li>
7   <li>item6</li>
8   <li>item7</li>
9 </ul>
10
11
12 ul
13 - for (var x = 1; x < 8; x++)
14   li item #{x}
```

用 for 迴圈產生多個 HTML 標籤

```
1 - var user = { name: 'webber', email: 'webber@gmail.com' }
2 #user
3   if user.name
4     h2 Name: #{user.name}
5     h2 Email: #{user.email}
6   else
7     h1 您尚未登入喔！
8
9 //-----
10 - var user = ''
11 #user
12   if user.name
13     h2 Name: #{user.name}
14     h2 Email: #{user.email}
15   else
16     h1 您尚未登入喔！
17
```

運用簡單的邏輯達到想要的效果

```
1 <!-- HTML結果 -->
2
3 <!doctype html>
4 <html>
5   <head>
6     <title>Article Title</title>
7   </head>
8   <body>
9     <h1>My Article</h1>
10  </body>
11 </html>
```

```
1 //-- jade語法
2 //-- layout.jade
3 doctype html
4 html
5   head
6     block title
7       title Default title
8   body
9     block content
```

```
1 //-- jade語法
2 //-- index.jade
3 extends ./layout.jade
4
5 block title
6   title Article Title
7
8 block content
9   h1 My Article
```

jade - extends 模板繼承

extends 允許樣板去擴充佈局或父樣板。
它可以覆蓋某些預定義的內容區塊。

The screenshot shows a code editor with three tabs open:

- index.jade**: The main file containing the full HTML structure. It includes a head section with scripts for jQuery and app.js, and a body section with an h1 and a p tag.
- head.jade**: A partial file used in the head inclusion. It contains the doctype html, html tag, and an include directive pointing to the head.jade file.
- foot.jade**: A partial file used in the footer inclusion. It contains a #footer block with a p tag.

```
1 <!-- HTML結果 -->
2
3 <!doctype html>
4 <html>
5   <head>
6     <title>My Site</title>
7     <script src='/javascripts/jquery.js'></script>
8     <script src='/javascripts/app.js'>
9       </script>
10    </head>
11    <body>
12      <h1>My Site</h1>
13      <p>Welcome to my super lame site.</p>
14      <div id="footer">
15        <p>Copyright (c) foobar</p>
16      </div>
17    </body>
18  </html>
```

```
1 //-- jade語法
2 //-- index.jade
3 doctype html
4 html
5   include ./includes/head.jade
6   body
7     h1 My Site
8     p Welcome to my super lame site.
9     include ./includes/foot.jade
```

```
1 //-- jade語法
2 //-- includes/head.jade
3 head
4   title My Site
5   script(src='/javascripts/jquery.js')
6   script(src='/javascripts/app.js')
```

```
1 //-- includes/foot.jade
2 #footer
3   p Copyright (c) foobar
```

jade - Includes(引入)

引入你的 jade 檔案內容到其他檔案中。

大致先介紹這幾個基本用法
有興趣的人可以去 jade 發掘了。

Have fun!

YOU GOT IT

RIGHT!

Bootstrap

一個 HTML、CSS 和 JS 的 responsive 框架
簡單來說可以 **加快** 你開發網頁的速度

為什麼使用 bootstrap ?

- 容易上手
- Grid System 方便編排你的網頁
- 提供很多方便使用的元件，**加快**開發速度
- 有響應式設計(Responsive)

如何安裝？

讓我們在開啟 codepen 吧

讓我們使用 CDN 的方式

- jquery:

<https://code.jquery.com/jquery-1.11.3.min.js>

- bootstrap:

1. <https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css>

2. <https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js>

接著
就讓我們邊寫邊學吧！

這是一個 bootstrap 的基本模板

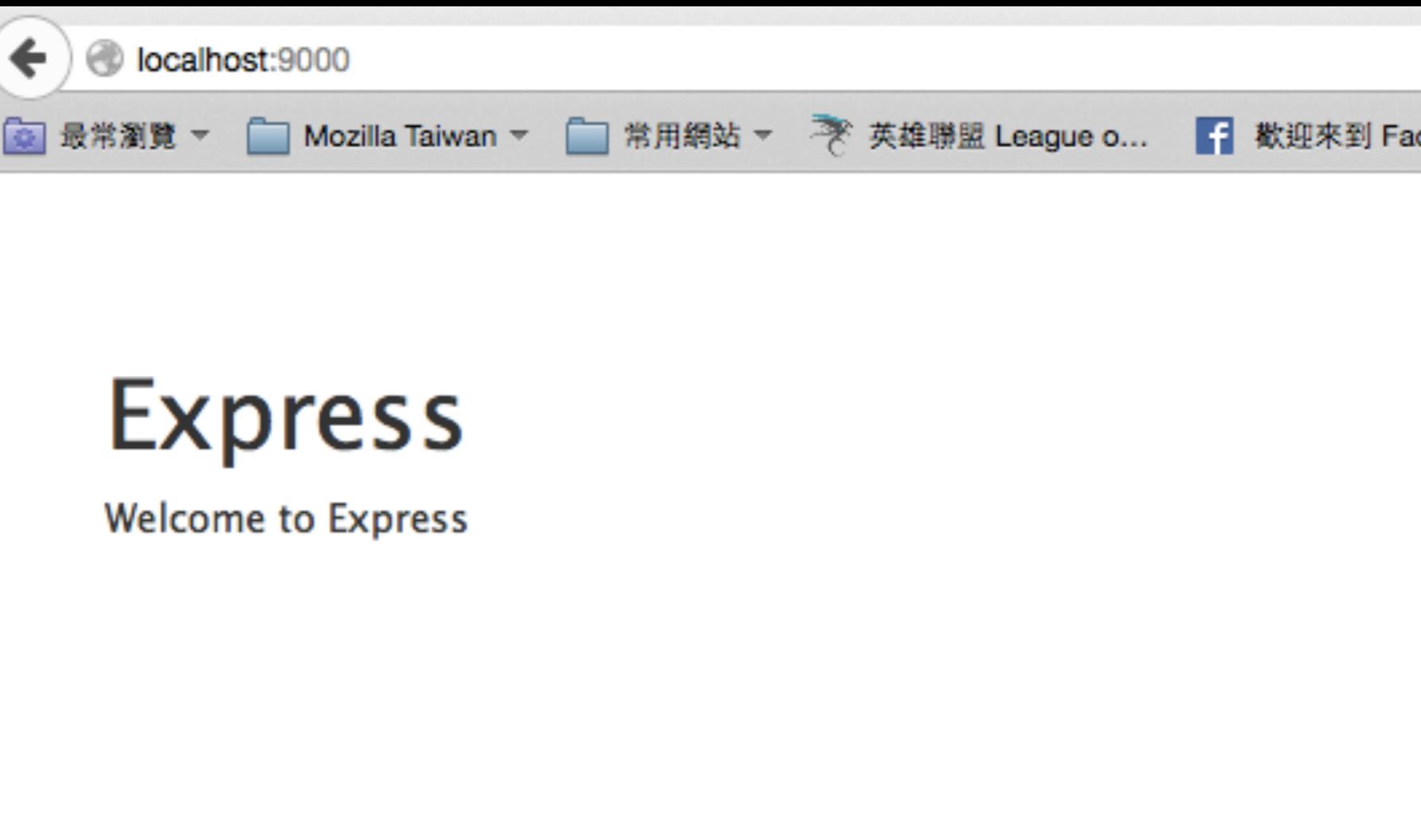
```
doctype html
html
head
meta(charset='utf-8')
meta(http-equiv='X-UA-Compatible', content='IE=edge')
meta(name='viewport', content='width=device-width, initial-scale=1')
title Bootstrap 101 Template
link(href='https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css', rel='stylesheet')
body
h1 Hello, world!
script(src='https://code.jquery.com/jquery-1.11.3.min.js')
script(src='https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js')
```

打開我們 express 的 view
資料夾底下的 layout.jade

```
doctype html
html
  head
    meta(charset='utf-8')
    meta(http-equiv='X-UA-Compatible', content='IE=edge')
    meta(name='viewport', content='width=device-width, initial-scale=1')
    title= title
    link(href='https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css', rel='stylesheet')
    link(rel='stylesheet', href='/stylesheets/style.css')
  body
    block content
      script(src='https://code.jquery.com/jquery-1.11.3.min.js')
      script(src='https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js')
```

將 layout.jade 修改成這樣

其實就是引入 js 跟 css 而已啦！
然後順便加個 meta

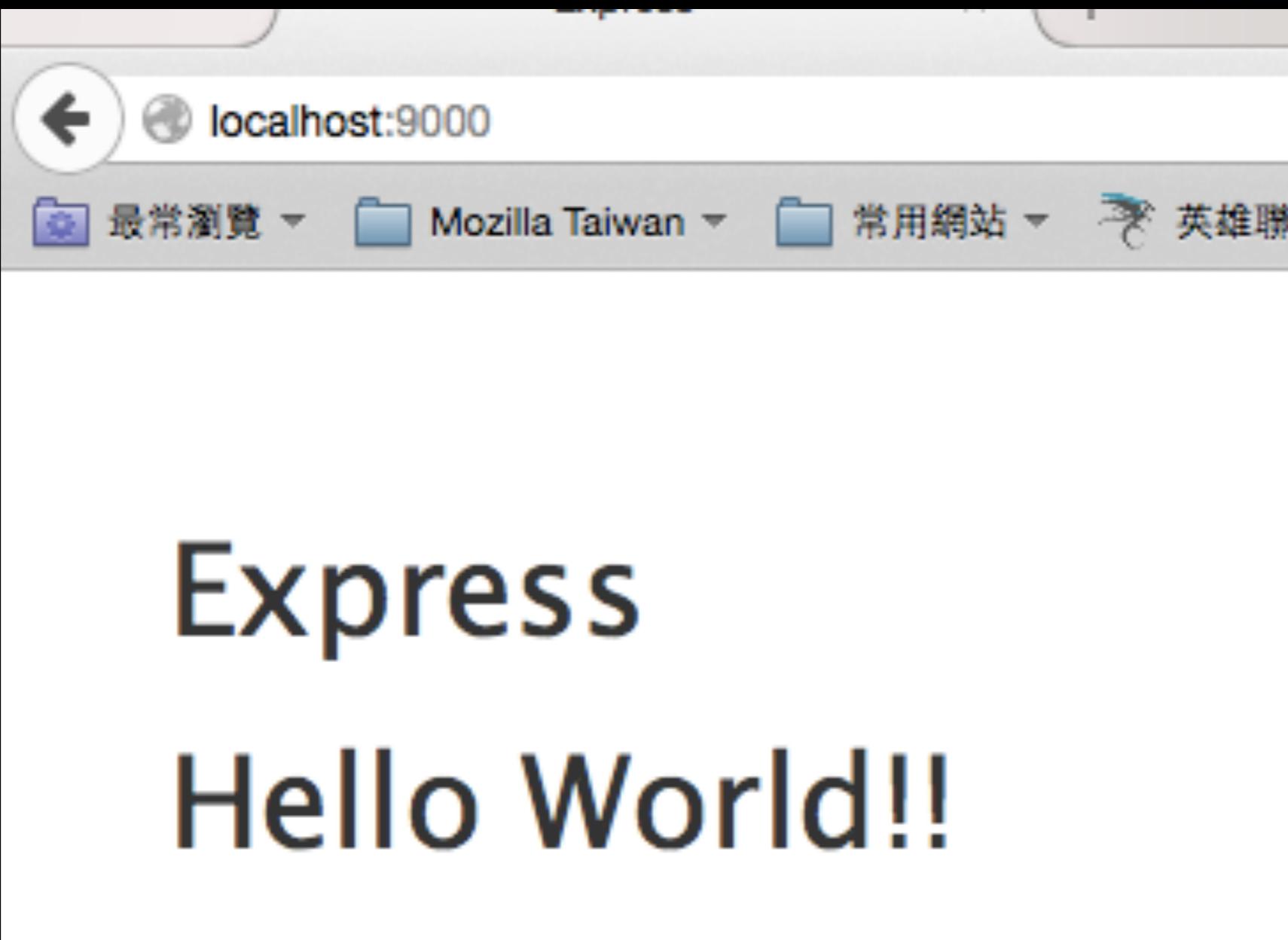


是不是有點不一樣了！

```
1  extends layout
2
3  block content
4    h1= title
5    h1 Hello World!!
6  |
```

然後試著修改 index.jade

將 p 改成 h1 Hello World!!



搞定！

EXAMPLE

Default

Primary

Success

Info

Warning

Danger

Link

```
<!-- Standard button -->
<button type="button" class="btn btn-default">Default</button>

<!-- Provides extra visual weight and identifies the primary action in a set of buttons -->
<button type="button" class="btn btn-primary">Primary</button>

<!-- Indicates a successful or positive action -->
<button type="button" class="btn btn-success">Success</button>

<!-- Contextual button for informational alert messages -->
<button type="button" class="btn btn-info">Info</button>

<!-- Indicates caution should be taken with this action -->
<button type="button" class="btn btn-warning">Warning</button>

<!-- Indicates a dangerous or potentially negative action -->
<button type="button" class="btn btn-danger">Danger</button>

<!-- Deemphasize a button by making it look like a link while maintaining button behavior -->
<button type="button" class="btn btn-link">Link</button>
```

Copy

顏色的規則

讓我們拿官網上 btn 的範例來看

btn

- btn-primary => 深藍色
- btn-success => 綠色
- btn-info => 淺藍色
- btn-warning => 黃色
- btn-danger => 紅色

EXAMPLE

Nullam id dolor id nibh ultricies vehicula ut id elit.

Duis mollis, est non commodo luctus, nisi erat porttitor ligula.

Maecenas sed diam eget risus varius blandit sit amet non magna.

Etiam porta sem malesuada magna mollis euismod.

Donec ullamcorper nulla non metus auctor fringilla.

```
<p class="bg-primary">...</p>
<p class="bg-success">...</p>
<p class="bg-info">...</p>
<p class="bg-warning">...</p>
<p class="bg-danger">...</p>
```

Copy

顏色的規則

接著是 p 標籤的背景色範例

EXAMPLE

Fusce dapibus, tellus ac cursus commodo, tortor mauris nibh.

Nullam id dolor id nibh ultricies vehicula ut id elit.

Duis mollis, est non commodo luctus, nisi erat porttitor ligula.

Maecenas sed diam eget risus varius blandit sit amet non magna.

Etiam porta sem malesuada magna mollis euismod.

Donec ullamcorper nulla non metus auctor fringilla.

[Copy](#)

```
<p class="text-muted">...</p>
<p class="text-primary">...</p>
<p class="text-success">...</p>
<p class="text-info">...</p>
<p class="text-warning">...</p>
<p class="text-danger">...</p>
```

顏色的規則

p 標籤的文字顏色範例

(°Д°)

_| ɔ/(_____ 是不是突破了些什麼 ?
/_\-(_____ /

接著讓我們開始打造頁面吧

首先我需要 header, body, footer...

請在 view 裡面新增兩個檔案

header.jade

footer.jade

```
1 doctype html
2 html
3   head
4     meta(charset='utf-8')
5     meta(http-equiv='X-UA-Compatible', content='IE=edge')
6     meta(name='viewport', content='width=device-width, initial-scale=1')
7     title= title
8     link(href='https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css', rel='stylesheet')
9     link(rel='stylesheet', href='/stylesheets/style.css')
10    body
11      .include header
12
13      .block content
14
15      .include footer
16
17      script(src='https://code.jquery.com/jquery-1.11.3.min.js')
18      script(src='https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js')
19
```

修改 layout.jade

加入 include header 跟 include footer

Fixed to top

Add `.navbar-fixed-top` and include a `.container` or `.container-fluid` to center and pad navbar content.

Brand Home Link Link

```
<nav class="navbar navbar-default navbar-fixed-top">
  <div class="container">
    ...
  </div>
</nav>
```

Copy

先來做一個置頂的header

請打開 header.jade

```
1 nav.navbar.navbar-default.navbar-fixed-top
2   .container
3     .navbar-header
4       button.navbar-toggle.collapsed(type='button', data-toggle='collapse', data-target='#navbar')
5         span.icon-bar
6         span.icon-bar
7         span.icon-bar
8         a.navbar-brand(href='#') WEBBER
9       #navbar.collapse.navbar-collapse
10      ul.nav.navbar-nav
11        li.active
12          a(href='#home') Home
13        li
14          a(href='#profile') Profile
15        li
16          a(href='#skills') Skills
17        li
18          a(href='#contact') Contact
```

將上面的 jade 寫入

a.navbar-brand 的文字請輸入自己的名字喔

WEBBER

Home

Profile

Skills

Contact

Express

Hello World!!

接著打開 footer.jade

```
1 .footer
2   .container
3     hr
4     p.text-center © 2015. Bootstrap web page by WEBBER.
```

修改 footer.jade

bootstrap 有提供一些輔助的 class,
像是 text-center 就可以幫你自動對文字進行置中

WEBBER

Home

Profile

Skills

Contact

Express

Hello World!!

© 2015. Bootstrap web page by WEBBER.

再來剩下 Body 了

開啟我們的 index.jade 吧

```
1  extends layout
2
3  block content
4  #home.section
5  .container
6  .row
7  h1 Home
8  #profile.section
9  .container
10 .row
11 h1 Profile
12 #skills.section
13 .container
14 .row
15 h1 Skills
16 #contact.section
17 .container
18 .row
19 h1 Contact
20
```

修改 index.jade

先將區塊分出來

Home

Profile

Skills

Contact

```
1 body {  
2   padding: 50px;  
3   font: 14px "Lucida Grande", Helvetica, Arial, sans-serif;  
4 }  
5  
6 a {  
7   color: #00B7FF;  
8 }  
9  
10 .section{  
11   padding-top: 110px;  
12   padding-bottom: 110px;  
13 }
```

接著修改 public/stylesheets/style.css

讓我們的 .section 看起來霸氣一點

Home

Profile

讓我們加上 bootstrap
的 scrollspy 吧

首先在 public/javascript 加入 main.js 的檔案吧

Via data attributes

To easily add scrollspy behavior to your topbar navigation, add `data-spy="scroll"` to the element you want to spy on (most typically this would be the `<body>`). Then add the `data-target` attribute with the ID or class of the parent element of any Bootstrap `.nav` component.

```
body {  
  position: relative;  
}
```

Copy

```
<body data-spy="scroll" data-target="#navbar-example">  
  ...  
  <div id="navbar-example">  
    <ul class="nav nav-tabs" role="tablist">  
      ...  
    </ul>  
  </div>  
  ...  
</body>
```

Copy

Via JavaScript

After adding `position: relative;` in your CSS, call the scrollspy via JavaScript:

```
$( 'body' ).scrollspy({ target: '#navbar-example' })
```

Copy

看起來要設置三個地方
body, nav 的外層要有id, 要加入js code

打開 layout.jade

```
9      link(rel='stylesheet', href='/stylesheets/style.css')
10     body(data-spy="scroll" data-target="#navbar")
11       include header
12       block content
13       include footer
14
15     script(src='https://code.jquery.com/jquery-1.11.3.min.js')
16     script(src='https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js')
17     script(src="/javascripts/main.js")
18
```

在 layout 上面加入這兩行

data-target指的是你nav外層的div id

```
1 $(function(){
2     $('body').scrollspy({ target: '#navbar' })
3 });
```

修改 main.js

試試看有沒有
scrollspy 效果

我們再回到 main.js

```
1 $(function(){
2   $('body').scrollspy({ target: '#navbar' });
3   $('#navbar a').on('click', function(){
4     $('html, body').animate({
5       scrollTop: $($this).attr('href')).offset().top
6     }, 500);
7     return false;
8   });
9 });
```

```
$('#navbar a').on('click', function(){
  $('html, body').animate({
    scrollTop: $($this).attr('href')).offset().top
  }, 500);
  return false;
});
```

加上簡單的滑動效果

接著打開 index.jade

```
1  extends layout
2
3  block content
4    #home.section
5      .container
6        .row.text-center
7          h1
8            | 我是 WEBBER
9            br |
10           | 而我也是一個 前端網頁 的 Develop
11           a.btn.btn-default.btn-lg(href="#profile") 瞭解更多
```

#home 的部分
在 .row 的地方加入 .text-center
並加入 h1 及 a.btn.btn-default.btn-lg
字可以隨便打沒關係的喔@@

我是 WEBBER
而我也是一個 前端網頁 的 Develop

瞭解更多

```
3  block content
4    #home.section
5      .container
6        .row.text-center
7          h1
8            | 我是 WEBBER
9            br
10           | 而我也是一個 前端網頁 的 Develop
11           a.btn.btn-default.btn-lg(href="#profile") 瞭解更多
12 #profile.section.profile-top
13   .container
14     .row.text-center
15       h1 我很高興你的拜訪
16       h3 我設計了這個網站，你可以從這裡了解我，所以慢慢享受吧～
17       img.img-circle(src="http://graph.facebook.com/bird23074035/picture?width=180")
```

#profile 的 top 部分

在 #profile 加上 .profile-top, 這是用來辨別上跟下
加入 h1, h3, img.img-circle 這三個部分

<http://graph.facebook.com/bird23074035/picture?width=180>
這是 FB 抓大頭照的 API, 將藍色的部份換成你們的id就可以了

```
1 body {
2     padding: 50px;
3     font: 14px "Lucida Grande", Helvetica, Arial, sans-serif;
4 }
5
6 a {
7     color: #00B7FF;
8 }
9
10 .section{
11     padding-top: 110px;
12     padding-bottom: 110px;
13 }
14
15 .profile-top img{
16     position: absolute;
17     margin-left: -100px;
18     left: 50%;
19 }
20
```

然後幫我在 style.css 上面加上 .profile-top img 的這些 css

我很高興你的拜訪

我設計了這個網站，你可以從這裡了解我，所以慢慢享受吧～



Skills

```
12#profile.section.profile-top.bg-info
13  .container
14  .row.text-center
15    h1 我很高興你的拜訪
16    h3 我設計了這個網站，你可以從這裡了解我，所以慢慢享受吧～
17      img.img-circle(src="http://graph.facebook.com/bird23074035/picture?width=
18.section.profile-bottom
19  .container
20  .row.text-center
21  .col-md-4
22    spanglyphicon.glyphicon-heart.img-circle
23    h3 為什麼我喜歡程式
24    h4.text-muted 我喜歡用程式來消磨我的時間，我很享受結果出來的那一個瞬間。
25  .col-md-4
26    spanglyphicon.glyphicon-music.img-circle
27    h3 為什麼我愛音樂
28    h4.text-muted 他能讓我更加放鬆，聽著這些音符發出來的聲音能讓我沈澱一些事情。
29  .col-md-4
30    spanglyphicon.glyphicon-apple.img-circle
31    h3 蘋果是一個好東西
32    h4.text-muted 恩..我指的是”一天一蘋果，醫生遠離我“這句話。
```

#profile bottom 的部份
.section 加上 .profile-bpttom
.row 裡面加入3個 .col-md-4,
並在裡面加入
spanglyphicon.glyphicon-** 跟 h3, h4.text-muted

```
19  }
20
21 .profile-bottom .col-md-4 .glyphicon{
22   font-size: 60px;
23   padding: 25px;
24   border: 5px solid #fff;
25 }
26 .profile-bottom .col-md-4:hover .glyphicon{
27   border: 5px solid #000;
28 }
29 |
```

在 style.css 上面加上 .profile-bottom img 的
這些 CSS

```
.profile-bottom .col-md-4:hover .glyphicon{border: 5px solid #000;}
.profile-bottom .col-md-4 .glyphicon{font-size:60px;padding: 25px; border:5px solid #000}
```

我很高興你的拜訪

我設計了這個網站，你可以從這裡了解我，所以慢慢享受吧～



為什麼我喜歡程式

我喜歡用程式來消磨我的時間，我很享受結果出來的那一個瞬間。

為什麼我愛音樂

他能讓我更加放鬆，聽著這些音符發出來的聲音能讓我沈澱一些事情。

蘋果是一個好東西

恩..我指的是”一天一蘋果，醫生遠離我“這句話。

```
33▼ #skills.section
34▼   .container
35▼     .row
36       h1 我的技能_
37▼     .col-md-6
38       h3.text-muted
39         |我的興趣源自於學校 homepage 的課程，讓我覺得原來網路世界是如此的有趣，慢慢琢磨到至今。
40▼     .col-md-6
41▼       .progress
42         .progress-bar.progress-bar-success(role="progressbar" style="width: 70%;")
43           | HTML 7/10
44▼       .progress
45         .progress-bar.progress-bar-success(role="progressbar" style="width: 70%;")
46           | CSS 7/10
47▼       .progress
48         .progress-bar.progress-bar-success(role="progressbar" style="width: 70%;")
49           | JS 7/10
```

#skills 的部分

加上 h1, 並加上兩個 .col-md-6

第一個 .col-md-6 裡面加入 h3.text-muted

第二個 .col-md-6 加入 .progress

裡面再加入 .progress-bar.progress-bar-success

```
{ 29  .progress{  
30    height: 30px;  
}31 }
```

修改 style.css

讓長條圖看起來飽滿一點

我的技能_

我的興趣源自於學校 homepage 的課程，讓我覺得原來網路世界是如此的有趣，慢慢琢磨到至今。

HTML 7/10

CSS 7/10

JS 7/10

```
50 #contact.section
51   .container
52     .row.text-center
53       h1 打招呼
54       h4.text-muted 如果你有任何問題，我會回答你
55       form.form-horizontal(action="/contact" method="post" role="form")
56         .col-md-1
57         .col-md-3
58           .form-group
59             .col-md-12
60               input.form-control(name="name" type="text" placeholder="Name")
61             .form-group
62               .col-md-12
63                 input.form-control(name="email" type="email" placeholder="Email")
64             .col-md-6
65             .form-group
66               .col-md-12
67                 textarea.form-control(name="messqge" rows="4" placeholder="Message" required)
68             .col-md-10
69               input.btn.btn-success.pull-right(type="submit" value="送出")
70
```

#contact部分

加上 h1, h4.text-muted, form.form-horizontal
form 裡面加入 col-md-1, col-md-3, col-md-6, col-md-10
每個 form 元素在加入 form-group及col-md-12

打招呼

如果你有任何問題，我會回答你

送出

現在你的網站完成了！

不過.. 再加一些小修改吧

```
12 #profile.section.profile-top.bg-info
13   .container
14     .row.text-center
15       h1 我很高興你的拜訪
16       h3 我設計了這個網站，你可以從這裡了解我，所以慢慢享受吧～
17       img.img-circle(src="http://graph.facebook.com/bird23074035/picture?width=180")
18 .section.profile-bottom=
33 #skills.section.bg-info
34   .container
35     .row
36       h1 我的技能_
37       .col-md-6
38         h3.text-muted
39           |我的興趣源自於學校 homepage 的課程，讓我覺得原來網路世界是如此的有趣，慢慢琢磨到至今。
40       .col-md-6
41         .progress
42           .progress-bar.progress-bar-success(role="progressbar" style="width: 70%;")
43             | HTML 7/10
44           progress
```

在 #profile 以及 #skills 上面加上 .bg-info 的 class
將背景加上一點顏色

```
1 body {
2   padding-top: 50px;
3   font: 14px "Lucida Grande", Helvetica, Arial, sans-serif;
4 }
5
6 a {
7   color: #00B7FF;
8 }
9
10 .section{
11   width: 100%
```

style.css 内的 body padding 改成 padding-top

恭喜！你的網頁完成搜

如果還有時間...

來做一個簡單的 mail server 吧

```
$ npm install nodemailer --save
nodemailer@1.3.4 node_modules/nodemailer
├── nodemailer-smtp-transport@1.0.3 (nodemailer-wellknown@0.1.5, clone@1.0.2, smtp-connection@1.2.0)
└── buildmail@1.2.4 (libbase64@0.1.0, libqp@1.0.0, addressparser@0.3.2)
  ├── nodemailer-direct-transport@1.0.2 (smtp-connection@1.2.0)
  └── libmime@1.0.0 (libbase64@0.1.0, libqp@1.0.0, iconv-lite@0.4.8)
    └── hyperquest@1.2.0 (through2@0.6.5, duplexer2@0.0.2)
```

npm install nodemailer --save

忘記 npm 的趕快複習！！

打開 app.js

讓我們做一些小修改

```
8 var routes = require('./routes/index');
9 var contact = require('./routes/contact');
10
11 var app = express();
12
13 // view engine setup
14 app.set('views', path.join(__dirname, 'views'));
15 app.set('view engine', 'jade');
16
17 // uncomment after placing your favicon in /public
18 //app.use(favicon(__dirname + '/public/favicon.ico'));
19 app.use(logger('dev'));
20 app.use(bodyParser.json());
21 app.use(bodyParser.urlencoded({ extended: false }));
22 app.use(cookieParser());
23 app.use(express.static(path.join(__dirname, 'public')));
24
25 app.use('/', routes);
26 app.use('/contact', contact);
27
```

請把 user 的部分改成 contact

順便幫我把 routes/user.js 改成 routes/contact.js

接著打開 routes/contact.js

```
1 var express = require('express');
2 var router = express.Router();
3
4 var nodemailer = require('nodemailer');
5 var transporter = nodemailer.createTransport({
6   service: 'Gmail',           // mail 的伺服器，這邊我用 Gmail
7   auth:{                      // 當成 server 的 Gmail 帳號
8     user:'nodejsdemo@gmail.com',
9     pass:'20150516'
10 }
11 });
12
```

先將 nodemailer require 進來
再來建立傳輸的 email

我這裡是用 Gmail, 帳號可以用我的
{user: 'nodejsdemo@gmail.com', pass: '20150516'}

```
12  ...
13▼ router.post('/', function(req, res) {
14  GuestName = req.body.name || 'NO_NAME';
15  GuestEmail = req.body.email || 'NO_EMAIL';
16  GuestMessage = req.body.message || 'NO_MESSAGE';
17▼ });
18 |
```

設定從 router 進來後的動作
req.body 是 post 過來的參數
並將每個值傳進我設定的參數裡(Guest...)
不知道 req.body 的我等等 log 出來給各位看看！

```
12
13▼ router.post('/', function(req, res) {
14  GuestName = req.body.name || 'NO_NAME';
15  GuestEmail = req.body.email || 'NO_EMAIL';
16  GuestMessage = req.body.message || 'NO_MESSAGE';
17
18▼   var mailOptions = {
19    ...   from: 'nodejsdemo@gmail.com',
20    ...   to: GuestEmail,
21    ...   subject: 'Hello~',
22    ...   text: "Hi~" + GuestName + "，很高興收到您的來信！"
23  }
24
```

設定 mailOptions 參數
from 是寄件者的 email
to 是客人的 email
subject 是 email 的主題
text 是 email 的內容

```
18▼ var mailOptions = {  
19    from: 'nodejsdemo@gmail.com',  
20    to: GuestEmail,  
21    subject: 'Hello~',  
22    text: "Hi~" + GuestName + "，很高興收到您的來信！"  
23  }  
24  
25▼ transporter.sendMail(mailOptions, function(err, info){  
26  if(err){  
27    console.error(err);  
28▼ }else{  
29    console.log(info);  
30    res.redirect('/');  
31  }  
32});  
33});  
34
```

接著就讓他 sendMail 吧
錯誤的話讓它秀出錯誤訊息
成功就導回 '/' 的頁面 (res.redirect('/'))
不要忘記33行的那個 }); 喔

```
1 var express = require('express');
2 var router = express.Router();
3
4 var nodemailer = require('nodemailer');
5 var transporter = nodemailer.createTransport({
6   service: 'Gmail', // mail 的伺服器，這邊我用 Gmail
7   auth:{ // 當成 server 的 Gmail 帳號
8     user:'nodejsdemo@gmail.com',
9     pass:'20150516'
10  }
11 });
12
13 router.post('/', function(req, res) {
14   GuestName = req.body.name || 'NO_NAME';
15   GuestEmail = req.body.email || 'NO_EMAIL';
16   GuestMessage = req.body.message || 'NO_MESSAGE';
17
18   var mailOptions = {
19     from: 'nodejsdemo@gmail.com',
20     to: GuestEmail,
21     subject: 'Hello~',
22     text: "Hi~" + GuestName + "，很高興收到您的來信！"
23   }
24
25   transporter.sendMail(mailOptions, function(err, info){
26     if(err){
27       console.error(err);
28     }else{
29       console.log(info);
30       res.redirect('/');
31     }
32   });
33 });
34
35 module.exports = router;
```

完整的 Code ! !

讓我們重新啟動我們的 web 吧
記得 `ctrl+C` 中斷喔！

恭喜你們成功了～
打完收工！



我隨風而來 隨風而去

Q & A