



# Node.js Day2

大數據 資深工程師 蔡協哲

mmaaaxxx77@gmail.com



BIG DATA  
大數據股份有限公司

網路輿情 · 數據分析 · 群眾智慧

**KEYPO**  
洞見網路大數據的關鍵引擎  
Insight Key Point of Big Data

<http://keypo.tw/>

# DAY 2 大綱

- › Node.js 開發環境安裝
- › 回顧DAY 1重要內容
- › 前端AJAX與後端串接
- › Database與SQL簡介
- › Node.js + MySQL CRUD
- › Express + DB
- › Web系統開發常見問題
- › 網路上的免費資源

# 開發環境安裝

# 1. Node.js

- › 下載並安裝10.14 64bit版本  
<https://nodejs.org/en/download/>
- › 如何確認是否安裝成功
  - › 開啟cmd
  - › Node.js查版本指令 `node -v`
  - › Npm查版本指令 `npm -v`

## 2. Sublime Text3

- › 下載並安裝

<https://www.sublimetext.com/>

### 3. 範例專案

#### 1. 下載專案

[https://github.com/mmmaaaxxx77/  
NodeJS\\_YuanZe\\_Workshop](https://github.com/mmmaaaxxx77/NodeJS_YuanZe_Workshop)

2. 或 `git clone https://github.com/mmmaaaxxx77/  
NodeJS_YuanZe_Workshop.git`

# DAY 1 重點回顧



# NPM 套件管理

## 1. 如何建立專案package.json

1. npm init

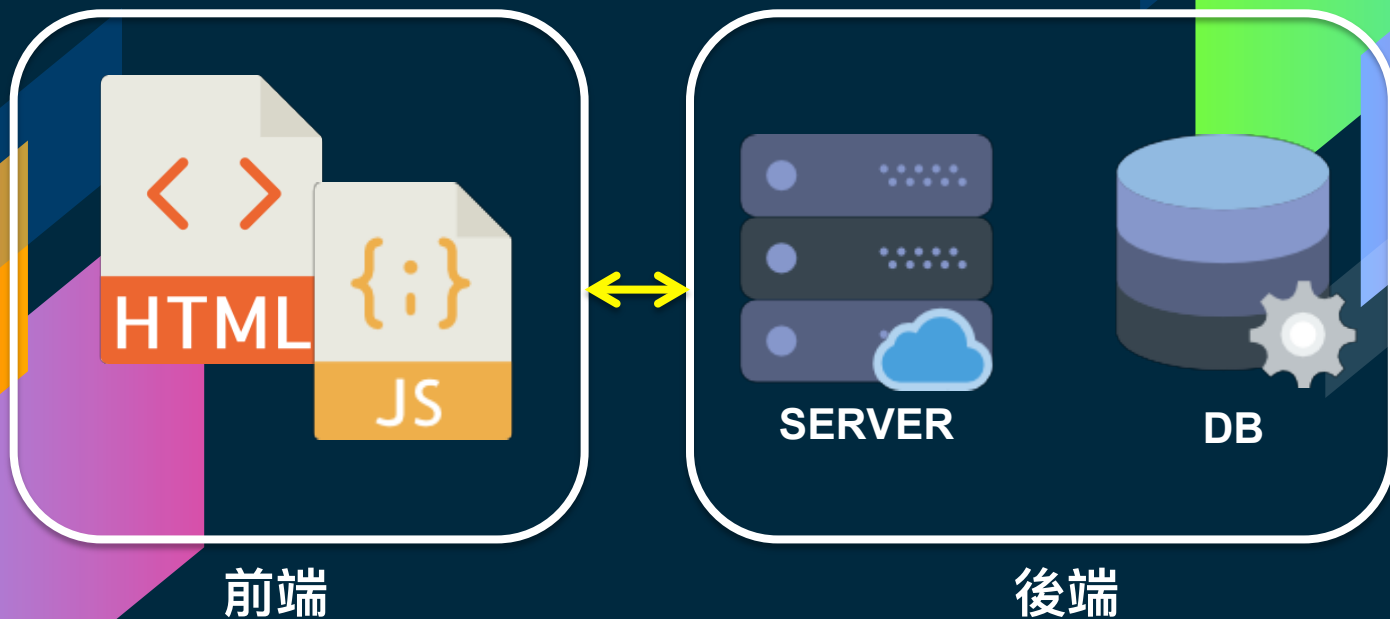
## 2. 如何安裝第三方套件

1. npm install {套件名稱}

## 3. 如何加入第三方套件與自訂套件

```
/* *****  
 * import express lib  
 * *****/  
const express = require('express');  
const bodyParser = require('body-parser');  
  
/* *****  
 * import other lib  
 * *****/  
const path = require('path');  
const User = require('./User');  
const Message = require('./Message');
```

# Web系統開發中的前端與後端

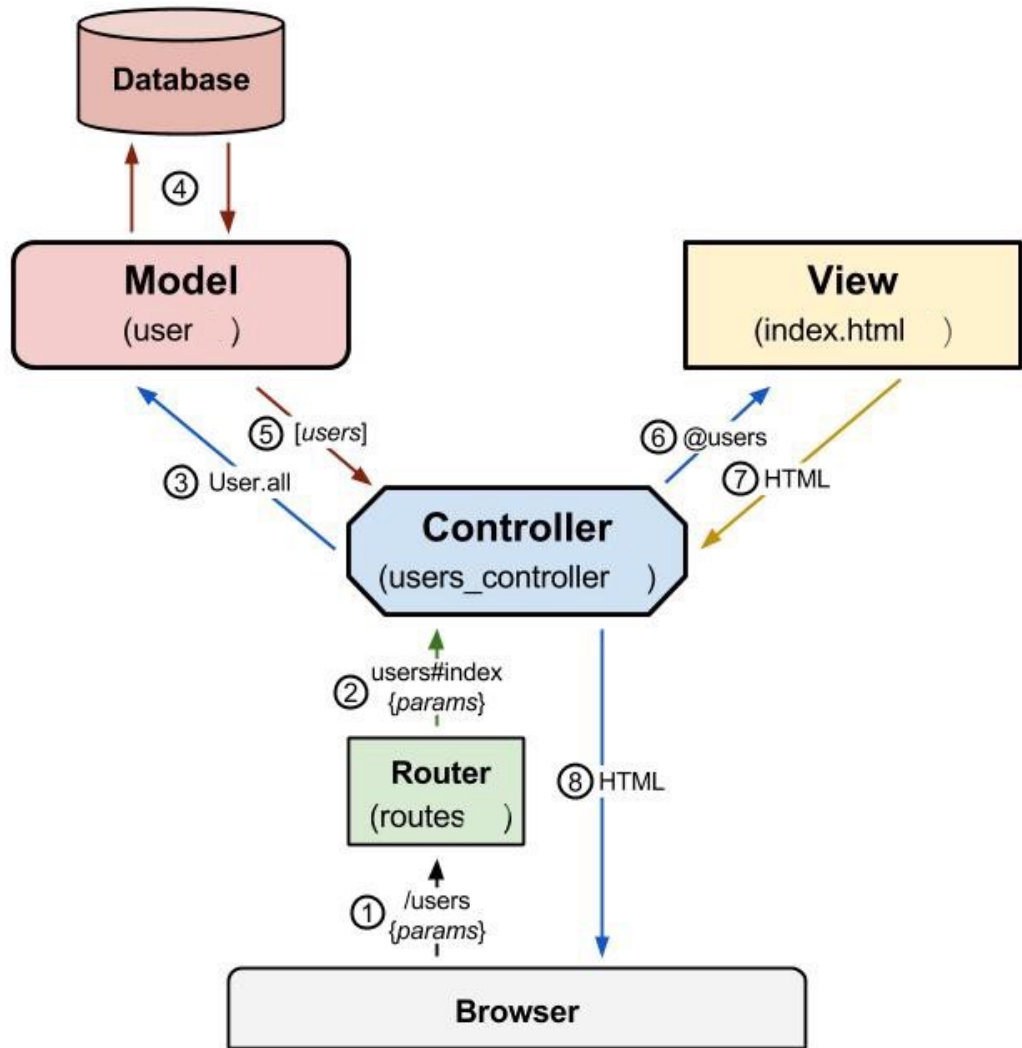


# WEB SERVICE & REST

1. Web Service意指透過http request method溝通進行Server端資料操作
2. REST是一種網路架構風格, 但並不是一種標準
3. 範例
  1. GET `http://demo.api.com/user` 取得所有使用者
  2. GET `http://demo.api.com/user/:id` 取得某人的資料
  3. DELETE `http://demo.api.com/user/:id` 刪除某人資料
  4. PUT `http://demo.api.com/user/:id` 新增某人資料
  5. POST `http://demo.api.com/user/:id` 修改某人資料

# WEB程式開發

## MVC PATTERN



# Express 檔案架構

```
▼ 3_express
  ▼ server 後端程式
    /* app.js MVC - controller
    /* index.js
  ▼ web 前端程式 MVC - view
  ▼ static
    ▼ css
      /* demo.css
    <> form.html
    <> index.html
  /* package.json Node.js 套件管理
```



# DEMO

## AJAX前後端串接

範例檔案：  
[範例專案/demo/4\\_express\\_ajax/](#)

# Databsse

- › 關聯式資料庫

- › MySQL、MS SQL、MariaDB、Oracle DB

# Databsse

- › 非關聯式資料庫 NoSQL
  - › Key-Value DB
    - › Hbase 、 Cassandra
  - › Memory DB
    - › Redis
  - › Document DB
    - › Mongo DB
  - › Graph DB
    - › Neo4j 、 Arango DB





# DEMO

## MySQL & phpmyadmin

<http://demodb.johnnyplanet.com:8012/>

帳號：yzdemo

密碼：yzdemo\_password

# 認識SQL CRUD - Create

- › INSERT INTO “表格名” (“欄位1”, “欄位2”, ...)  
VALUES (“值1”, “值2”, ...);

# 認識SQL CRUD - Read

- › SELECT “欄位名” FROM “表格名” WHERE “條件”;

# 認識SQL CRUD - Update

- › UPDATE "表格名" SET "欄位1" = [新值] WHERE "條件";

# 認識SQL CRUD - Delete

› DELETE FROM "表格名" WHERE "條件";

# DEMO

# MySQL CRUD

範例檔案：  
[範例專案/demo/5\\_mysql/](#)



# DEMO

## Express + MySQL

### 使用者列表

範例檔案：  
[範例專案/demo/express\\_project/](#)



# DEMO

## Express + MySQL

### 聊天室

範例檔案：  
`範例專案/demo/express_project/`



# 即時更新的幾種方式

- › Polling
- › Comet
- › Long Polling
- › WebSocket

參考

<https://blog.niclin.tw/2017/10/28/%E7%8D%B2%E5%BE%97%E5%AF%A6%E6%99%82%E6%9B%B4%E6%96%B0%E7%9A%84%E6%96%B9%E6%B3%95polling-comet-long-polling-websocket/>

# Web系統開發常見問題

- › 表單驗證
  - › 前後端都需確認, 可採用第三方套件
- › SQL Injection
  - › ORM (Object Relational Mapping)
- › 字串編碼
  - › DB編碼、程式轉碼、Host編碼
- › Try-Catch
  - › 做好Error Handling, 確保系統不會crash
- › 系統測試
  - › Unit Test、Functional Test...

# 免費的線上資源

- › 免費程式部署上線 Heroku
  - › <https://www.heroku.com/>
  - › <https://dwatow.github.io/2018/01-13-heroku-node-mvp/>
- › 免費MySQL DB
  - › <https://www.db4free.net>



# THANKS!

Any questions?