



# Node.js Day1

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/>

- › 大數據處理
- › 分散式運算
- › AI模型設計開發
- › BOT對話機器人設計開發
- › 數據分析平台開發
- › TEXT MINING

# Node.js Workshop 11/26、12/06

## DAY1

1. Node.js基本開發
2. Node.js與常見問題
3. Web Server業界常聽見的架構術語
4. Web框架 - Express

## DAY2

1. DB與程式的結合應用
2. 介紹業界常用的架構
3. 如何將Node.js結合MySQL
4. CRUD開發
5. 簡介如何使用Docker部署至雲端Server



# Node.js Day 1

大數據 資深工程師  
Johnny 蔡協哲

# DAY 1 大綱

- › Node.js 開發環境安裝
- › Node.js 介紹
- › Node.js 基本開發與常見問題
- › Npm套件管理
- › 非同步問題與處理 (Promise、await、async)
- › Web - MVC、Web Service
- › Web 框架 - Express

# 開發環境安裝

# 1. GIT

- › 下載並安裝

<https://git-scm.com/download/win>



## 2. 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`

### 3. Sublime Text3

- › 下載並安裝

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

# Node.js 簡介

# Node.js

- › Javascript編程
- › 直譯式程式語言
- › 跨平台
- › 事件驅動、非阻塞和非同步輸入輸出
- › 單執行緒

# Node.js 基本開發

# 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`

The background is a dark navy blue. In the top-left and bottom-left corners, there are overlapping, semi-transparent geometric shapes in shades of green, blue, orange, and pink. In the top-right and bottom-right corners, there are similar overlapping shapes in shades of green, blue, purple, and orange. The text is centered in the middle of the slide.

# DEMO

## cmd 編程



# DEMO

## .js 檔案開發

執行指令：node 檔案名稱.js





# DEMO 常見問題範例

範例檔案：  
範例專案/demo/0\_basic/

# DEMO

## Npm套件管理

常用指令：  
npm init  
npm install



# DEMO

## 引入第三方套件

範例檔案：

範例專案/demo/1\_modules/moment.js



# DEMO

## 引入其他開發模組

範例檔案：

範例專案/demo/1\_modules/index.js

範例專案/demo/1\_modules/User.js

# DEMO

## 同步與非同步問題

範例檔案：

`範例專案/demo/2_await_async/await_async.js`

`範例專案/demo/2_await_async/promise.js`

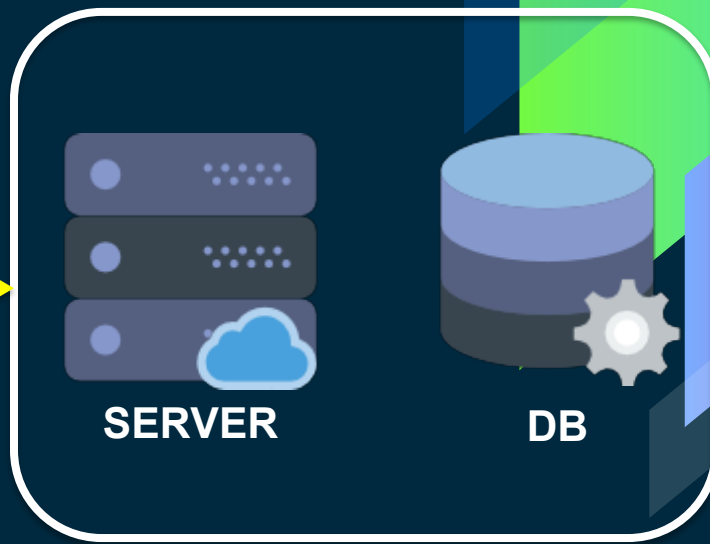
The background is a dark navy blue. It features abstract, overlapping geometric shapes in various colors: bright green, cyan, magenta, orange, and pink. These shapes are arranged in a way that creates a sense of depth and movement, with some shapes appearing to be layered on top of others. The overall aesthetic is modern and digital.

# WEB





前端



後端



# 前後端的溝通方式

1. 由後端直接將資料動態寫在前端網頁
2. 由前端JS的AJAX方法要求後端資料, 以JSON格式
3. FORM向Server動作要求, Server傳遞網頁或JSON回去前端

# HTTP PROTOCOL – REQUEST METHODS

1. GET - 常見於要求網頁或REST要求資料
2. POST - FORM傳遞表單至後端

...

其他參考：

<https://developer.mozilla.org/zh-TW/docs/Web/HTTP/Methods>

# WEB SERVICE

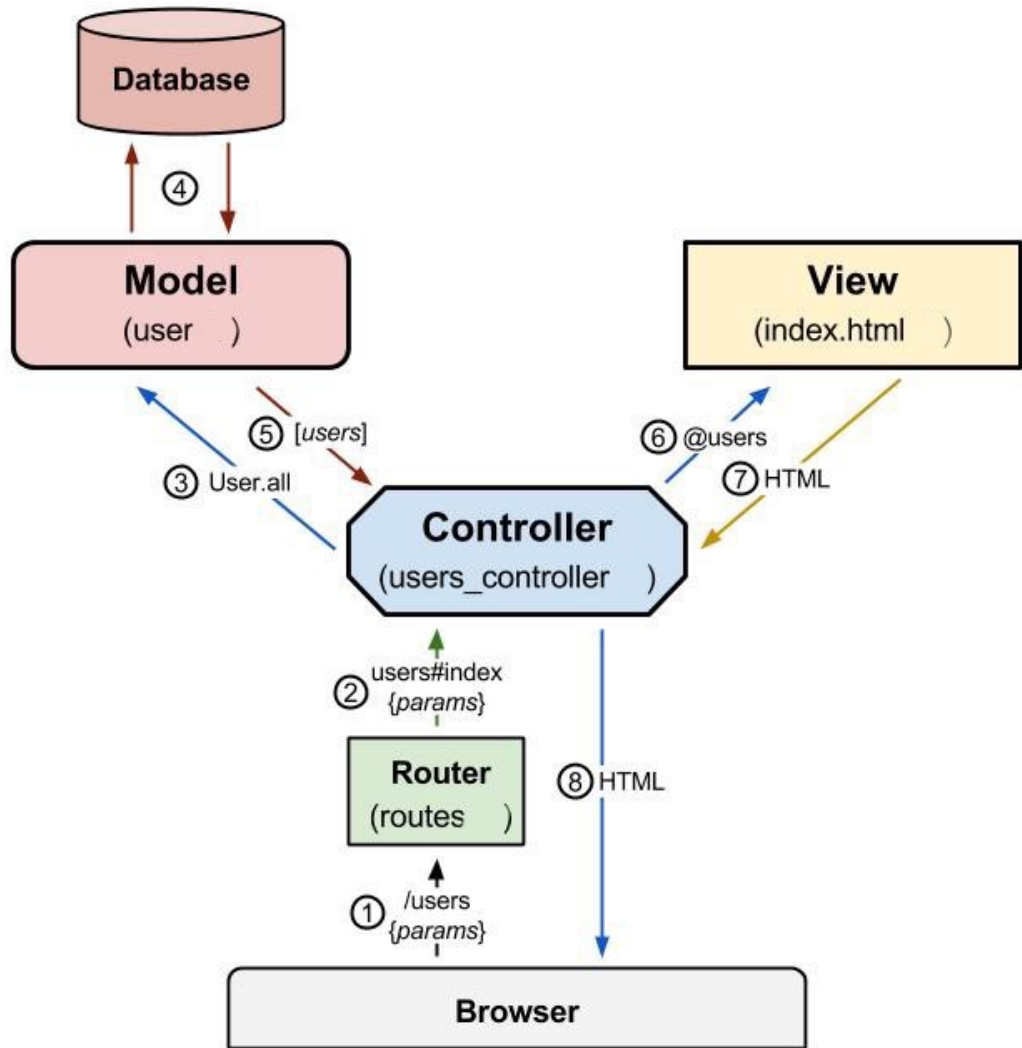
1. 透過http request method溝通進行Server端資料操作
2. 資料格式多為JSON
3. 範例
  1. GET `http://demo.api.com/user` 取得所有使用者
  2. GET `http://demo.api.com/user/johnny` 取得johnny的資料
  3. DELETE `http://demo.api.com/user/johnny` 刪除johnny
  4. PUT `http://demo.api.com/user/johnny` 新增johnny資料

# REST (Representational State Transfer)

1. 一種網路架構風格, 但並不是一種標準
2. 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





# DEMO WEB 框架 - express

範例檔案：  
[範例專案/demo/3\\_express/](#)

```
▼ 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
```



# THANKS!

Any questions?